

John R. Birge
François Louveaux

Introduction to Stochastic Programming

Second Edition

Springer Series in Operations Research and Financial Engineering

Series Editors:

Thomas V. Mikosch
University of Copenhagen
Laboratory of Actuarial Mathematics
DK-1017 Copenhagen
Denmark
mikosh@act.ku.dk

Sidney I. Resnick
Cornell University
School of Operations Research and
Industrial Engineering
Ithaca, NY 14853
U.S.A.
sirl@cornell.edu

Stephen M. Robinson
University of Wisconsin-Madison
Department of Industrial Engineering
Madison, WI 53706
U.S.A.
smrobins@wisc.edu

For further volumes:
<http://www.springer.com/series/3182>

John R. Birge • François Louveaux

Introduction to Stochastic Programming

Second Edition



John R. Birge
Booth School of Business
University of Chicago
5807 South Woodlawn Avenue
Chicago, Illinois 60637
USA
john.birge@chicagobooth.edu

François Louveaux
Department of Business Administration
University of Namur
Rempart de la Vierge 8
B-5000, Namur
Belgium
francois.louveaux@fundp.ac.be

ISSN 1431-8598
ISBN 978-1-4614-0236-7 e-ISBN 978-1-4614-0237-4
DOI 10.1007/978-1-4614-0237-4
Springer New York Dordrecht Heidelberg London

Library of Congress Control Number: 2011929942

Mathematics Subject Classification (2010): 37N40, 46N10, 49L20, 49Mxx (all), 49N30, 49N15, 90-01, 90B50, 90C05, 90C06, 90C15, 90C39

© Springer Science+Business Media, LLC 2011

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.
The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

*To Richard and Joelle,
Sebastien, Jérôme, Quentin, and
Géraldine.*

Preface

Since the publication of the first edition of this book, we have been encouraged by the growing interest in stochastic programming and its application in a variety of areas, including routine use in many industries from transportation and logistics to finance and energy. We have also been heartened by the many new methodological and theoretical advances within the field. In this second edition, we have attempted to capture aspects of both recent applications and models as well as new practically relevant methods and theory. As in the first edition, our primary goal is to provide students and other readers with an appreciation of how to build uncertainty into an optimization model, what differences in decisions might result from recognizing the presence of uncertainty, and how and what kinds of models are amenable to solution. We have focused the second edition on satisfying these main objectives while also uncovering basic research questions to give beginning researchers a foundation upon which to build more in-depth knowledge.

To help make the relevant issues in modeling, solving, and analyzing stochastic programs more evident, we have incorporated more examples than in the first edition so that each of the main modeling, solution, and analysis processes are illustrated with a detailed example. We have also added many exercises whose solutions provide additional insights into stochastic programming concepts and tools. Many of these exercises assume the availability of software to solve basic linear and nonlinear optimization models and to construct algorithmic procedures involving matrix operations. Since we view completing these exercises as a key part of understanding the material, instructors should ensure that students have adequate programming skills to implement the methods described in the book.

Besides additional examples and exercises throughout the book, we have reorganized the material to improve the logical flow and to eliminate unnecessary or complicating issues before explaining the most practically relevant material. Specific changes in the second edition include the following:

- a new section (Section 1.5) and routing example in Chapter 1;
- a worked-out modeling exercise (Section 2.8) and a section on risk modeling and robust formulation (Section 2.9 in Chapter 2);

- re-arrangement and simplification of the material in Chapter 3 to emphasize basic model characteristics and illustrate them with examples;
- complete re-organization and combination of Chapters 5 and 6 into a new Chapter 5 that unifies the treatment of cutting-plane methods and again provides additional examples;
- an additional section on Lagrangian multistage methods in Chapter 6 (formerly Chapter 7);
- a completely re-organized version of Chapter 7 (formerly Chapter 8) including new methods and review material on combinatorial optimization;
- additional examples in Chapter 8 (formerly Chapter 9) including bounds on loss probabilities in loan portfolios;
- re-organization of Chapter 9 (formerly Chapter 10) to place practical methods earlier and to include a new section on Monte Carlo methods for probabilistic constraints;
- re-organization of Chapter 10 (formerly Chapter 11) to include new sections on scenario generation, multistage sampling methods, and approximate dynamic programming methods;
- removal of the short chapter (formerly Chapter 12) on a capacity expansion case study.

We anticipate that classes would follow much of the same sequence as we suggested for the first edition, but, with the increased availability of software to implement methods, we recommend that instructors include more computational exercises and additional modeling projects to fit students' interests. Any course should again start with the first two chapters to provide the application and modeling context. Depending on student interest, a typical class would generally include Chapters 3, 4, and Sections 5.1, 5.2, and 5.5 to present the most typical types of methods. For basic approximations, a modeling-focused class could focus on the main techniques in Chapters 8, 9, and 10 (for dynamic models), while a theoretically-oriented class might emphasize the analytical results in those chapters. A more computationally focussed class might emphasize the remainder of Chapter 5 plus Chapters 6 and 7.

We wish to thank the many people who sent us comments and suggestions about the first edition of the book and the numerous students we have worked with and all those who have helped us see stochastic programming from a fresh perspective every time we encounter something new. Among the many who have contributed, we thank Michael Dempster, Michel Gendreau, Maarten van der Vlerk, and Bill Ziemba. Thanks are also due to Martine Van Caeneghem for her patient typing of the modifications in Namur. We also again thank Fonds National de la Recherche Scientifique, the National Science Foundation, as well as the U.S. Department of Energy, and the University of Chicago Booth School of Business for their financial support.

In our first edition, we finished the preface with special thanks to our wives, Pierrette and Marie, to whom our book was dedicated. These thanks are more than ever very much present in our hearts. Now, we also want to express our proudness and joy of having such great children. We have thus decided to dedicate this second edition to them. We may thus expect that the third edition will be dedicated to our

grandchildren, although the timing of this edition and the number of lines needed for this future dedication remain unknown.

Chicago, Illinois, USA
Namur, Belgium

John R. Birge
François Louveaux

Preface to the First Edition

According to a French saying “Gérer, c'est prévoir,” which we may translate as “(The art of) Managing is (in) foreseeing.” Now, probability and statistics have long since taught us that the future cannot be perfectly forecast but instead should be considered random or uncertain. The aim of stochastic programming is precisely to find an optimal decision in problems involving uncertain data. In this terminology, *stochastic* is opposed to *deterministic* and means that some data are random, whereas programming refers to the fact that various parts of the problem can be modeled as linear or nonlinear mathematical programs. The field, also known as *optimization under uncertainty*, is developing rapidly with contributions from many disciplines such as operations research, economics, mathematics, probability, and statistics. The objective of this book is to provide a wide overview of stochastic programming, without requiring more than a basic background in these various disciplines.

Introduction to Stochastic Programming is intended as a first course for beginning graduate students or advanced undergraduate students in such fields as operations research, industrial engineering, business administration (in particular, finance or management science), and mathematics. Students should have some basic knowledge of linear programming, elementary analysis, and probability as given, for example, in an introductory book on operations research or management science or in a combination of an introduction to linear programming (optimization) and an introduction to probability theory.

Instructors may need to add some material on convex analysis depending on the choice of sections covered. We chose not to include such introductory material because students' backgrounds may vary widely and other texts include these concepts in detail. We did, however, include an introduction to random variables while modeling stochastic programs in Section 2.1 and short reviews of linear programming, duality, and nonlinear programming at the end of Chapter 2. This material is given as an indication of the prerequisites in the book to help instructors provide any missing background. In the Subject Index, the first reference to a concept is where it is defined or, for concepts specific to a single section, where a source is provided.

In our view, the objective of a first course based on this book is to help students build an intuition on how to model uncertainty into mathematical programs, which changes uncertainty brings into the decision process, what difficulties uncertainty may bring, and what problems are solvable. To begin this development, the first section in Chapter 1 provides a worked example of modeling a stochastic program. It introduces the basic concepts, without using any new or specific techniques. This first example can be complemented by any one of the other proposed cases of Chapter 1, in finance, in multistage capacity expansion, and in manufacturing. Based again on examples, Chapter 2 describes how a stochastic model is formally built. It also stresses the fact that several different models can be built, depending on the type of uncertainty and the time when decisions must be taken. This chapter links the various concepts to alternative fields of planning under uncertainty.

Any course should begin with the study of those two chapters. The sequel would then depend on the students' interests and backgrounds. A typical course would consist of elements of Chapter 3, Sections 4.1 to 4.5, Sections 5.1 to 5.3 and 5.7, and one or two more advanced sections of the instructor's choice. The final case study may serve as a conclusion. A class emphasizing modeling might focus on basic approximations in Chapter 9 and sampling in Chapter 10. A computational class would stress methods from Chapters 6 to 8. A more theoretical class might concentrate more deeply on Chapter 3 and the results from Chapters 9 to 11.

The book can also be used as an introduction for graduate students interested in stochastic programming as a research area. They will find a broad coverage of mathematical properties, models, and solution algorithms. Broad coverage cannot mean an in-depth study of all existing research. The reader will thus be referred to the original papers for details. Advanced sections may require multivariate calculus, probability measure theory, or an introduction to nonlinear or integer programming. Here again, the stress is clearly in building knowledge and intuition in the field. Mathematical results are given so long as they are either basic properties or helpful in developing efficient solution procedures. The importance of the various sections clearly reflects our own interests, which focus on results that may lead to practical applications of stochastic programming.

To conclude, we may use the following little story. An elderly person, celebrating her one hundredth birthday, was asked how she succeeded in reaching that age. She answered, "It's very simple. You just have to wait."

In comparison, stochastic programming may well look like a field of young impatient people who not only do not want to wait and see but who consider waiting to be suboptimal. We realize how much patience was needed from our friends and colleagues who encouraged us to write this book, which took us much longer than expected. To all of them, we are extremely thankful for their support. The authors also wish to thank the Fonds National de la Recherche Scientifique and the National Science Foundation for their financial support. Both authors are deeply grateful to the people who introduced us to the field, George Dantzig, Roger Wets, Jacques

Drèze, and Guy de Ghellinck. Our special thanks go to our wives, Pierrette and Marie, to whom we dedicate this book.

Ann Arbor, Michigan
Namur, Belgium

John R. Birge
François Louveaux

Contents

Part I Models

1	Introduction and Examples	3
1.1	A Farming Example and the News Vendor Problem	4
a.	The farmer's problem	4
b.	A scenario representation	6
c.	General model formulation	10
d.	Continuous random variables	11
e.	The news vendor problem	15
1.2	Financial Planning and Control	20
1.3	Capacity Expansion	28
1.4	Design for Manufacturing Quality	35
1.5	A Routing Example	40
a.	Presentation	40
b.	Wait-and-see solutions	42
c.	Expected value solution	43
d.	Recourse solution	44
e.	Other random variables	46
f.	Chance-constraints	47
1.6	Other Applications	48
2	Uncertainty and Modeling Issues	55
2.1	Probability Spaces and Random Variables	55
2.2	Deterministic Linear Programs	57
2.3	Decisions and Stages	57
2.4	Two-Stage Program with Fixed Recourse	59
a.	Fixed distribution pattern, fixed demand, r_i, v_j, t_{ij} stochastic	62
b.	Fixed distribution pattern, uncertain demand	63
c.	Uncertain demand, variable distribution pattern	64
d.	Stages versus periods; Two-stage versus multistage	65

2.5	Random Variables and Risk Aversion	66
2.6	Implicit Representation of the Second Stage	68
a.	A closed form expression is available for $\mathcal{Q}(x)$	69
b.	For a given x , $\mathcal{Q}(x)$ is computable	70
2.7	Probabilistic Programming	71
a.	Deterministic linear equivalent: a direct case	71
b.	Deterministic linear equivalent: an indirect case	72
c.	Deterministic nonlinear equivalent: the case of random constraint coefficients	73
2.8	Modeling Exercise	74
a.	Presentation	74
b.	Discussion of solutions	76
2.9	Alternative Characterizations and Robust Formulations	84
2.10	Relationship to Other Decision-Making Models	87
a.	Statistical decision theory and decision analysis	87
b.	Dynamic programming and Markov decision processes	89
c.	Machine learning and online optimization	90
d.	Optimal stochastic control	91
e.	Summary	93
2.11	Short Reviews	94
a.	Linear programming	94
b.	Duality for linear programs	96
c.	Nonlinear programming and convex analysis	97

Part II Basic Properties

3	Basic Properties and Theory	103
3.1	Two-Stage Stochastic Linear Programs with Fixed Recourse	103
a.	Formulation	103
b.	Discrete random variables	105
c.	General cases	109
d.	Special cases: relatively complete, complete, and simple recourse	113
e.	Optimality conditions and duality	115
f.	Stability and nonanticipativity	118
3.2	Probabilistic or Chance Constraints	124
a.	General case	124
b.	Probabilistic constraints with discrete random variables	130
3.3	Stochastic Integer Programs	135
a.	Recourse problems	135
b.	Simple integer recourse	140
c.	Probabilistic constraints	146
3.4	Multistage Stochastic Programs with Recourse	149
3.5	Stochastic Nonlinear Programs with Recourse	156

4 The Value of Information and the Stochastic Solution	163
4.1 The Expected Value of Perfect Information	163
4.2 The Value of the Stochastic Solution	165
4.3 Basic Inequalities	166
4.4 The Relationship between <i>EVPI</i> and <i>VSS</i>	167
a. <i>EVPI</i> = 0 and <i>VSS</i> ≠ 0	168
b. <i>VSS</i> = 0 and <i>EVPI</i> ≠ 0	169
4.5 Examples	170
4.6 Bounds on <i>EVPI</i> and <i>VSS</i>	171

Part III Solution Methods

5 Two-Stage Recourse Problems	181
5.1 The <i>L</i> -Shaped Method	182
a. Optimality cuts	184
b. Feasibility cuts	191
c. Proof of convergence	196
d. The multicut version	198
5.2 Regularized Decomposition	202
5.3 The Piecewise Quadratic Form of the <i>L</i> -shaped Methods	210
5.4 Bunching and Other Efficiencies	217
a. Full decomposability	218
b. Bunching	219
5.5 Basis Factorization and Interior Point Methods	222
5.6 Inner Linearization Methods and Special Structures	237
5.7 Simple and Network Recourse Problems	242
5.8 Methods Based on the Stochastic Program Lagrangian	253
5.9 Additional Methods and Complexity Results	262
6 Multistage Stochastic Programs	265
6.1 Nested Decomposition Procedures	266
6.2 Quadratic Nested Decomposition	276
6.3 Block Separability and Special Structure	282
6.4 Lagrangian-Based Methods for Multiple Stages	284
7 Stochastic Integer Programs	289
7.1 Stochastic Integer Programs and LP-Relaxation	289
7.2 First-stage Binary Variables	291
a. Improved optimality cuts	294
b. Example with continuous random variables	299
7.3 Second-stage Integer Variables	302
a. Looking in the space of tenders	303
b. Discontinuity points	305
c. Algorithm	306
7.4 Reformulation	312
a. Difficulties of reformulation in stochastic integer programs	312

b.	Disjunctive cuts	314
c.	First-stage dependence	316
d.	An algorithm	317
7.5	Simple Integer Recourse	319
a.	χ restricted to be integer	322
b.	The case where $S = 1$, χ not integral	325
7.6	Cuts Based on Branching in the Second Stage	326
a.	Feasibility cuts	326
b.	Optimality cuts	329
7.7	Extensive Forms and Decomposition	331
7.8	Short Reviews	334
a.	Branch-and-bound	334
b.	A simple example of valid inequalities	335
c.	Disjunctive cuts	336

Part IV Approximation and Sampling Methods

8	Evaluating and Approximating Expectations	341
8.1	Direct Solutions with Multiple Integration	342
8.2	Discrete Bounding Approximations	346
8.3	Using Bounds in Algorithms	352
8.4	Bounds in Chance-Constrained Problems	357
8.5	Generalized Bounds	363
a.	Extensions of basic bounds	363
b.	Bounds based on separable functions	367
c.	General-moment bounds	372
8.6	General Convergence Properties	381
9	Monte Carlo Methods	389
9.1	Sample Average Approximation and Importance Sampling in the L -Shaped Method	390
9.2	Stochastic Decomposition	395
9.3	Stochastic Quasi-Gradient Methods	399
9.4	Sampling Methods for Probabilistic Constraints and Quantiles	404
9.5	General Results for Sample Average Approximation and Sequential Sampling	409
10	Multistage Approximations	417
10.1	Extensions of the Jensen and Edmundson-Madansky Inequalities ..	418
10.2	Bounds Based on Aggregation	422
10.3	Scenario Generation and Distribution Fitting	426
10.4	Multistage Sampling and Decomposition Methods	432
10.5	Approximate Dynamic Programming and Special Cases	436
a.	Network revenue management	438
b.	Vehicle allocation problems	439
c.	Piecewise-linear separable bounds	441

Contents	xix
d. Nonlinear bounds and a production planning example	444
e. Extensions	446
Sample Distribution Functions	449
A.1 Discrete Random Variables	449
A.2 Continuous Random Variables	450
References	451
Author Index	471
Subject Index	477

Notation

The following describes the major symbols and notations used in the text. To the greatest extent possible, we have attempted to keep unique meanings for each item. In those cases where an item has additional uses, they should be clear from context. We include here only notation used in more than one section. Additional notation may be needed within specific sections and is explained when used.

In general, vectors are assumed to be columns with transposes to indicate row vectors. This yields $c^T x$ to denote the inner product of two n -vectors, c and x . We reserve prime ($'$) for first derivatives with respect to time (e.g., $f' = df/dt$).

Vectors in primal programs are represented by lowercase Latin letters while matrices are uppercase. Dual variables and certain scalars are generally Greek letters. Superscripts indicate a stage while subscripts indicate components followed by realization index. Boldface indicates a random quantity. Expectations of random variables are indicated by a bar ($\bar{\xi}$), μ , or ($E(\xi)$). We also use the bar notation to denote sample means in Chapter 9.

Equations are numbered consecutively in the text by section and number within the section (e.g., (1.2) for Section 1, Equation 2). For references to chapters other than the current one, we use three indices: chapter, section, and equation, (e.g., (3.1.2) for Chapter 3, Section 1, Equation 2). Exercises are given at the end of sections (or subsections in the cases of Sections 3.2 and 5.1) and are referenced in the same manner as equations. All other items (figures, tables, declarations, examples) are labeled consecutively through the entire chapter with a single reference (e.g., Figure 1) if within the current chapter and chapter and number if in a different chapter (e.g., Figure 3.1 for Chapter 3, Figure 1).

Symbol	Definition
$+$	Superscript indicates the positive part of a real (i.e., $a^+ = \max(a, 0)$) or unrestricted variable (e.g., $y = y^+ - y^-$, $y^+ \geq 0, y^- \geq 0$) and its objective coefficients (e.g., q^+), subscript as non-negative values in a set (e.g., \mathfrak{R}_+) or the right-limit ($F^+(t) = \lim_{s \downarrow t} F(s)$)
$-$	Superscript indicates the negative part of a real (i.e., $a^- = \max(-a, 0)$) or unrestricted variable (e.g., $y = y^+ - y^-$, $y^+ \geq 0, y^- \geq 0$) and its objective coefficients (e.g., q^-) or the left-limit ($F^-(t) = \lim_{s \uparrow t} F(s)$)
$*$	Indicates an optimal value or solution (e.g., x^*)
$0 \wedge / \sim$	Indicate given nonoptimal values or solutions (e.g., $x^0, \hat{x}, x', \tilde{x}$)
0	Zero matrix (subscripts denote dimension when present)
$\mathbf{1}_X$	Indicator function of set X
a	Ancestor scenario, real value or vector
A	First-stage matrix (e.g., $Ax = b$), also used to indicate an event or subset, $A \in \mathcal{A} \subset \Omega$
\mathcal{A}	Collection of subsets
b	First-stage right-hand side (e.g., $Ax = b$)
B	Matrix, basis submatrix, Borel sets, or index set of a basis
\mathcal{B}	Collection of subsets (notably Borel sets)
c	First-stage objective ($c^T x$), t -th stage objective ($(c^t(\omega))^T x^t$) or real vectors
C	Matrix or index set of continuous variables
d	Right-hand side of a feasibility cut in the L-shaped method, a demand, or real vector
D	Left-hand side vector of a feasibility cut in the L-shaped method, a matrix, a set, or an index set of discrete variables
\mathcal{D}	Set of descendant scenarios
e	Exponential, right-hand side of an optimality cut in the L-shaped method, an extreme point, or the unit vector ($e^T = (1, \dots, 1)$)
E	Mathematical expectation operator, left-hand side vector of an optimality cut in the L-shaped method, or an event
f	Function (usually in an objective ($f(x)$ or $f_i(x)$) or a density
F	Cumulative probability distribution

Symbol	Definition
g	Function (usually in constraints ($g(x)$ or $g_j(x)$))
h	Right-hand side in second-stage ($Wy = h - Tx$), also $h'(\omega)$ in multistage problems
H	Number of stages (horizon) in multistage problems
i	Subscript index of functions (f_i) or vector elements (x_i , x_{ij})
I	Identity matrix or index set ($i \in I$)
j	Subscript index of functions (g_j) or vector elements (y_j , y_{ij})
J	Matrix or index set
k	Index of a realization of a random vector ($k = 1, \dots, K$)
K	Feasibility sets (K_1, K_2) or total number of realizations of a discrete random vector
\mathcal{K}	Number of realizations or sample paths in a scenario tree with \mathcal{K}^t nodes at stage t
l	Index, lower bound on a variable, or Lagrangian function
L	The L-shaped method, objective value lower bound, or real value
m	Number of constraints (m_1, m_2) or number of elements ($i = 1, \dots, m$)
n	Number of variables (n_1, n_2) or number of elements ($i = 1, \dots, n$)
N	Set, normal cone, normal distribution, or number of random elements
p	Probability of a random element (e.g., p_k $= P(\xi = \xi_k)$) or matrix of probabilities
P	Probability of events (e.g., $P(\xi \leq 0)$)
q	Second-stage objective vector ($q^T y$)
Q	Second-stage (multistage) value function with random argument ($Q(x, \xi)$ or $Q^t(x^t, \xi^t)$)
\mathcal{Q}	Second-stage (multistage) expected value value (recourse) function ($\mathcal{Q}(x)$ or $\mathcal{Q}^t(x^t)$)
r	Revenue or return in examples, real vector, or index
\mathfrak{R}	Real numbers
R	Matrix or set
s	Scenario or index

Symbol	Definition
S	Set or matrix
t	Superscript stage or period index for multistage programs ($t = 1, \dots, H$), a real-valued parameter, or an index
T	Technology matrix ($Wy = h - Tx$ or $T^{t-1}(\omega)(x)$); as a superscript, the transpose of a matrix or vector
u	General vector, upper-bound vector, or expected shortage
U	Objective value upper bound
v	Variable vector or expected surplus
V	Set, matrix or an operator
w	Second-stage decision vector in some examples
W	Recourse matrix ($Wy = h - Tx$)
x	First-stage decision vector or multistage decision vector (x^t)
X	First-stage feasible set ($x \in X$) or t th stage feasible set (X^t)
y	Second-stage decision vector
Y	Second-stage feasible set ($y \in Y$)
z	Objective value ($\min z = c^T x + \dots$)
Z	Integers
α	Real value, vector, or probability level with probabilistic constraints
β	Real value or vector
γ	Real value or function
δ	Real value or function
ε	Real value
ζ	Random variable
η	Real value or random variable
θ	Lower bound on $\mathcal{Q}(x)$ in the L-shaped method
κ	Index
λ	Dual multiplier, parameter in a convex combination, or measure
μ	Expectation (used mostly in examples of densities) or a parameter for non-negative multiples
v	Algorithm iteration index (sometimes also the number of samples in Monte Carlo sampling algorithms)
ξ	Random vector (often indexed by time, ξ^t) with realizations as ξ (without boldface)
Ξ	Support of the random vector ξ
π	Dual multiplier

Symbol	Definition
Π	Product, projection operator, or aggregated problem dual multiplier
ρ	Dual multiplier or discount factor
σ	Dual multiplier, standard deviation, or σ -field
Σ	Summation or covariance matrix
τ	Possible right-hand side in bundles or index of time
ϕ	Function in computing the value of the stochastic solution or a measure
Φ	Function, cumulative distribution of standard normal
\emptyset	Empty set
χ	Tender or offer from first to second period ($\chi = Tx$)
ψ	Second stage value function defined on tenders and with random argument, $\psi(\chi, \xi(\omega))$
Ψ	Expected second stage value function defined on tenders, $\Psi(\chi)$
ω	Random event ($\omega \in \Omega$)
Ω	Set of all random events

Part I

Models

Chapter 1

Introduction and Examples

This chapter presents stochastic programming examples from a variety of areas with wide application. These examples are intended to help the reader build intuition on how to model uncertainty. They also reflect different structural aspects of the problems. In particular, we show the variety of stochastic programming models in terms of the objectives of the decision process, the constraints on those decisions, and their relationships to the random elements.

In each example, we investigate the value of the stochastic programming model over a similar deterministic problem. We show that even simple models can lead to significant savings. These results provide the motivation to lead us into the following chapters on stochastic programs, solution properties, and techniques.

In the first section, we consider a farmer who must decide on the amounts of various crops to plant. The yields of the crops vary according to the weather. From this example, we illustrate the basic foundation of stochastic programming and the advantage of the stochastic programming solution over deterministic approaches. We also introduce the classical news vendor (or newsboy) problem and give the fundamental properties of these problems' general class, called *two-stage stochastic linear programs with recourse*.

The second section contains an example in planning finances for a child's education. This example fits the situation in many discrete time control problems. Decisions occur at different points in time so that the problem can be viewed as having multiple stages of observations and actions.

The third section considers power system capacity expansion. Here, decisions are taken dynamically about additional capacity and about the allocation of capacity to meet demand. The resulting problem has multiple decision stages and a valuable property known as *block separable recourse* that allows efficient solution. The problem also provides a natural example of constraints on reliability within the area called *probabilistic or chance-constrained programming*.

The fourth example concerns the design of a simple axle. It includes market reaction to the design and performance characteristics of products made by a manufacturing system with variable performance. The essential characteristics of the

maximum performance of the product illustrate a problem with fundamental nonlinearities incorporated directly into the stochastic program.

The fifth section presents a simple routing problem. It illustrates models where some decisions (traveling on an arc or not) are represented by integer decision variables. As this example is easily illustrated and does not require any solver, it may also be used as a preliminary example.

The final section of this chapter briefly describes several other major application areas of stochastic programs. The exercises at the end of the chapter develop modeling techniques. This chapter illustrates some of the range of stochastic programming applications but is not meant to be exhaustive. Applications in location and distribution, for example, are discussed in Chapter 2.

1.1 A Farming Example and the News Vendor Problem

a. *The farmer's problem*

Consider a European farmer who specializes in raising wheat, corn, and sugar beets on his 500 acres of land. During the winter, he wants to decide how much land to devote to each crop. (We refer to the farmer as "he" for convenience and not to imply anything about the gender of European farmers.)

The farmer knows that at least 200 tons (T) of wheat and 240 T of corn are needed for cattle feed. These amounts can be raised on the farm or bought from a wholesaler. Any production in excess of the feeding requirement would be sold. Over the last decade, mean selling prices have been \$170 and \$150 per ton of wheat and corn, respectively. The purchase prices are 40% more than this due to the wholesaler's margin and transportation costs.

Another profitable crop is sugar beet, which he expects to sell at \$36/T; however, the European Commission imposes a quota on sugar beet production. Any amount in excess of the quota can be sold only at \$10/T. The farmer's quota for next year is 6000 T. 这个地方有点绕

Based on past experience, the farmer knows that the mean yield on his land is roughly 2.5 T, 3 T, and 20 T per acre for wheat, corn, and sugar beets, respectively. Table 1 summarizes these data and the planting costs for these crops.

To help the farmer make up his mind, we can set up the following model. Let

- x_1 = acres of land devoted to wheat,
- x_2 = acres of land devoted to corn,
- x_3 = acres of land devoted to sugar beets,
- w_1 = tons of wheat sold,
- y_1 = tons of wheat purchased,
- w_2 = tons of corn sold,
- y_2 = tons of corn purchased,
- w_3 = tons of sugar beets sold at the favorable price,

Table 1 Data for farmer's problem.

	Wheat	Corn	Sugar Beets
Yield (T/acre)	2.5	3	20
Planting cost (\$/acre)	150	230	260
Selling price (\$/T)	170	150	36 under 6000 T 10 above 6000 T
Purchase price (\$/T)	238	210	—
Minimum requirement (T)	200	240	—
Total available land: 500 acres			

w_4 = tons of sugar beets sold at the lower price.

The problem reads as follows:

$$\begin{aligned}
 \min \quad & 150x_1 + 230x_2 + 260x_3 + 238y_1 - 170w_1 \\
 & + 210y_2 - 150w_2 - 36w_3 - 10w_4 \\
 \text{s. t.} \quad & x_1 + x_2 + x_3 \leq 500, \quad 2.5x_1 + y_1 - w_1 \geq 200, \\
 & 3x_2 + y_2 - w_2 \geq 240, \quad w_3 + w_4 \leq 20x_3, \quad w_3 \leq 6000, \\
 & x_1, x_2, x_3, y_1, y_2, w_1, w_2, w_3, w_4 \geq 0.
 \end{aligned} \tag{1.1}$$

After solving (1.1) with his favorite linear program solver, the farmer obtains an optimal solution, as in Table 2.

Table 2 Optimal solution based on expected yields.

Culture	Wheat	Corn	Sugar Beets
Surface (acres)	120	80	300
Yield (T)	300	240	6000
Sales (T)	100	—	6000
Purchase (T)	—	—	—
Overall profit: \$118,600			

This optimal solution is easy to understand. The farmer devotes enough land to sugar beets to reach the quota of 6000 T. He then devotes enough land to wheat and corn production to meet the feeding requirement. The rest of the land is devoted to wheat production. Some wheat can be sold.

To an extent, the optimal solution follows a very simple heuristic rule: to allocate land in order of decreasing profit per acre. In this example, the order is sugar beets at a favorable price, wheat, corn, and sugar beets at the lower price. This simple

heuristic would, however, no longer be valid if other constraints, such as labor requirements or crop rotation, would be included.

After thinking about this solution, the farmer becomes worried. He has indeed experienced quite different yields for the same crop over different years mainly because of changing weather conditions. Most crops need rain during the few weeks after seeding or planting, then sunshine is welcome for the rest of the growing period. Sunshine should, however, not turn into drought, which causes severe yield reductions. Dry weather is again beneficial during harvest. From all these factors, yields varying 20 to 25% above or below the mean yield are not unusual.

In the next sections, we study two possible representations of these variable yields. One approach using discrete, correlated random variables is described in Sections 1.1b, and 1.1c. Another, using continuous uncorrelated random variables, is described in Section 1.1d.

The influence of price fluctuations, illustrated by the dramatic price increases in 2007, is discussed in Exercise 8.

b. A scenario representation

A first possibility is to assume some correlation among the yields of the different crops. A very simplified representation of this would be to assume that years are good, fair, or bad for all crops, resulting in above average, average, or below average yields for all crops. To fix these ideas, “above” and “below” average indicate a yield 20% above or below the mean yield given in Table 1. For simplicity, we assume that weather conditions and yields for the farmer do not have a significant impact on prices.

The farmer wishes to know whether the optimal solution is sensitive to variations in yields. He decides to run two more optimizations based on above average and below average yields. Tables 3 and 4 give the optimal solutions he obtains in these cases.

Again, the solutions in Tables 3 and 4 seem quite natural. When yields are high, smaller surfaces are needed to raise the minimum requirements in wheat and corn and the sugar beet quota. The remaining land is devoted to wheat, whose extra production is sold. When yields are low, larger surfaces are needed to raise the minimum requirements and the sugar beet quota. In fact, corn requirements cannot be satisfied with the production, and some corn must be bought.

The optimal solution is very sensitive to changes in yields. The optimal surfaces devoted to wheat range from 100 acres to 183.33 acres. Those devoted to corn range from 25 acres to 80 acres and those devoted to sugar beets from 250 acres to 375 acres. The overall profit ranges from \$59,950 to \$167,667.

Long-term weather forecasts would be very helpful here. Unfortunately, as even meteorologists agree, weather conditions cannot be accurately predicted six months ahead. The farmer must make up his mind without perfect information on yields.

Table 3 Optimal solution based on above average yields (+ 20%).

Culture	Wheat	Corn	Sugar Beets
Surface (acres)	183.33	66.67	250
Yield (T)	550	240	6000
Sales (T)	350	—	6000
Purchase (T)	—	—	—
Overall profit: \$167,667			

Table 4 Optimal solution based on below average yields (−20%).

Culture	Wheat	Corn	Sugar Beets
Surface (acres)	100	25	375
Yield (T)	200	60	6000
Sales (T)	—	—	6000
Purchase (T)	—	180	—
Overall profit: \$59,950			

The main issue here is clearly on sugar beet production. Planting large surfaces would make it certain to produce and sell the quota, but would also make it likely to sell some sugar beets at the unfavorable price. Planting small surfaces would make it likely to miss the opportunity to sell the full quota at the favorable price.

The farmer now realizes that he is unable to make a perfect decision that would be best in all circumstances. He would, therefore, want to assess the benefits and losses of each decision in each situation. Decisions on land assignment (x_1, x_2, x_3) have to be taken now, but sales and purchases ($w_i, i = 1, \dots, 4, y_j, j = 1, 2$) depend on the yields. It is useful to index those decisions by a scenario index $s = 1, 2, 3$ corresponding to above average, average, or below average yields, respectively. This creates a new set of variables of the form $w_{is}, i = 1, 2, 3, 4, s = 1, 2, 3$ and $y_{js}, j = 1, 2, s = 1, 2, 3$. As an example, w_{32} represents the amount of sugar beets sold at the favorable price if yields are average.

Assuming the farmer wants to maximize long-run profit, it is reasonable for him to seek a solution that maximizes his expected profit. (This assumption means that the farmer is neutral about risk. For a discussion of risk aversion and alternative utilities, see Chapter 2.) If the three scenarios have an equal probability of $1/3$, the farmer's problem reads as follows:

$$\begin{aligned}
 & \min 150x_1 + 230x_2 + 260x_3 \quad \text{所有情景求期望, 两阶段 (一个阶段确定, 另一个阶段情景)} \\
 & \quad - \frac{1}{3}(170w_{11} - 238y_{11} + 150w_{21} - 210y_{21} + 36w_{31} + 10w_{41}) \\
 & \quad - \frac{1}{3}(170w_{12} - 238y_{12} + 150w_{22} - 210y_{22} + 36w_{32} + 10w_{42}) \\
 & \quad - \frac{1}{3}(170w_{13} - 238y_{13} + 150w_{23} - 210y_{23} + 36w_{33} + 10w_{43}) \\
 & \text{s.t. } x_1 + x_2 + x_3 \leq 500, \quad 3x_1 + y_{11} - w_{11} \geq 200, \quad (1.2) \\
 & \quad 3.6x_2 + y_{21} - w_{21} \geq 240, \quad w_{31} + w_{41} \leq 24x_3, \quad w_{31} \leq 6000, \\
 & \quad 2.5x_1 + y_{12} - w_{12} \geq 200, \quad 3x_2 + y_{22} - w_{22} \geq 240, \\
 & \quad w_{32} + w_{42} \leq 20x_3, \quad w_{32} \leq 6000, \quad 2x_1 + y_{13} - w_{13} \geq 200, \\
 & \quad 2.4x_2 + y_{23} - w_{23} \geq 240, \quad w_{33} + w_{43} \leq 16x_3, \\
 & \quad w_{33} \leq 6000, \quad x, y, w \geq 0.
 \end{aligned}$$

Such a model of a stochastic decision program is known as the *extensive form* of the stochastic program because it explicitly describes the second-stage decision variables for all scenarios. The optimal solution of (1.2) is given in Table 5. The top line gives the planting areas, which must be determined before realizing the weather and crop yields. This decision is called the *first stage*. The other lines describe the yields, sales, and purchases in the three scenarios. They are called the *second stage*. The bottom line shows the overall expected profit.

Table 5 Optimal solution based on the stochastic model (1.2).

		Wheat	Corn	Sugar Beets
First Stage	Area (acres)	170	80	250
$s = 1$ Above	Yield (T)	510	288	6000
	Sales (T)	310	48	6000 (favor. price)
	Purchase (T)	—	—	—
$s = 2$ Average	Yield (T)	425	240	5000
	Sales (T)	225	—	5000 (favor. price)
	Purchase (T)	—	—	—
$s = 3$ Below	Yield (T)	340	192	4000
	Sales (T)	140	—	4000 (favor. price)
	Purchase (T)	—	48	—
Overall profit: \$108,390				

The optimal solution can be understood as follows. The most profitable decision for sugar beet land allocation is the one that always avoids sales at the unfavorable price even if this implies that some portion of the quota is unused when yields are average or below average.

The area devoted to corn is such that it meets the feeding requirement when yields are average. This implies sales are possible when yields are above average

and purchases are needed when yields are below average. Finally, the rest of the land is devoted to wheat. This area is large enough to cover the minimum requirement. Sales then always occur.

This solution illustrates that it is impossible, under uncertainty, to find a solution that is ideal under all circumstances. Selling some sugar beets at the unfavorable price or having some unused quota is a decision that would never take place with a perfect forecast. Such decisions can appear in a stochastic model because decisions have to be balanced or hedged against the various scenarios.

The hedging effect has an important impact on the expected optimal profit. Suppose yields vary over years but are cyclical. A year with above average yields is always followed by a year with average yields and then a year with below average yields. The farmer would then take optimal solutions as given in Table 3, then Table 2, then Table 4, respectively. This would leave him with a profit of \$167,667 the first year, \$118,600 the second year, and \$59,950 the third year. The mean profit over the three years (and in the long run) would be the mean of the three figures, namely \$115,406 per year.

Now, assume again that yields vary over years, but on a random basis. If the farmer gets the information on the yields before planting, he will again choose the areas on the basis of the solution in Table 2, 3, or 4, depending on the information received. In the long run, if each yield is realized one third of the years, the farmer will get again an expected profit of \$115,406 per year. This is the situation under perfect information.

As we know, the farmer unfortunately does not get prior information on the yields. So, the best he can do in the long run is to take the solution as given by Table 5. This leaves the farmer with an expected profit of \$108,390. The difference between this figure and the value, \$115,406, in the case of perfect information, namely \$7016, represents what is called *the expected value of perfect information (EVPI)*. This concept, along with others, will be studied in Chapter 4. At this introductory level, we may just say that it represents the loss of profit due to the presence of uncertainty.

Another approach the farmer may have is to assume expected yields and always to allocate the optimal planting surface according to these yields, as in Table 2. This approach represents the *expected value solution*. It is common in optimization but can have unfavorable consequences. Here, as shown in Exercise 1, using the expected value solution every year results in a long run annual profit of \$107,240. The loss by not considering the random variations is the difference between this and the stochastic model profit from Table 5. This value, $\$108,390 - \$107,240 = \$1,150$, is the *value of the stochastic solution (VSS)*, the possible gain from solving the stochastic model. Note that it is not equal to the expected value of perfect information, and, as we shall see in later models, may in fact be larger than the *EVPI*.

These two quantities give the motivation for stochastic programming in general and remain a key focus throughout this book. *EVPI* measures the value of knowing the future with certainty while *VSS* assesses the value of knowing and using distributions on future outcomes. Our emphasis will be on problems where no further information about the future is available so the *VSS* becomes more practically

relevant. In some situations, however, more information might be available through more extensive forecasting, sampling, or exploration. In these cases, *EVPI* would be useful for deciding whether to undertake additional efforts.

c. General model formulation

We may also use this example to illustrate the general formulation of a stochastic problem. We have a set of decisions to be taken without full information on some random events. These decisions are called *first-stage decisions* and are usually represented by a vector x . In the farmer example, they are the decisions on how many acres to devote to each crop. Later, full information is received on the realization of some random vector ξ . Then, second-stage or corrective actions y are taken. We use boldface notation here and throughout the book to denote that these vectors are random and to differentiate them from their realizations. We also sometimes use a functional form, such as $\xi(\omega)$ or $y(s)$, to show explicit dependence on an underlying element, ω or s .

In the farmer example, the random vector is the set of yields and the corrective actions are purchases and sales of products. In mathematical programming terms, this defines the so-called two-stage stochastic program with recourse of the form

$$\begin{aligned} & \min c^T x + E_{\xi} Q(x, \xi) \\ \text{s. t. } & Ax = b, \\ & x \geq 0, \end{aligned} \tag{1.3}$$

where $Q(x, \xi) = \min\{\mathbf{q}^T \mathbf{y} \mid W\mathbf{y} = \mathbf{h} - \mathbf{T}x, y \geq 0\}$, ξ is the vector formed by the components of \mathbf{q}^T , \mathbf{h}^T , and \mathbf{T} , and E_{ξ} denote mathematical expectation with respect to ξ . We assume here that W is fixed (*fixed recourse*). Reasons for this restriction are explained in Section 3.1.

In the farmer example, the random vector is a discrete variable with only three different values. Only the T matrix is random. A second-stage problem for one particular scenario s can thus be written as

$$\begin{aligned} Q(x, s) = & \min \{238y_1 - 170w_1 + 210y_2 - 150w_2 - 36w_3 - 10w_4\} \\ \text{s. t. } & t_1(s)x_1 + y_1 - w_1 \geq 200, \\ & t_2(s)x_2 + y_2 - w_2 \geq 240, \\ & w_3 + w_4 \leq t_3(s)x_3, \\ & w_3 \leq 6000, \\ & y, w \geq 0, \end{aligned} \tag{1.4}$$

where $t_i(s)$ represents the yield of crop i under scenario s (or state of nature s). To illustrate the link between the general formulation (1.3) and the example (1.4), observe that in (1.4) we may say that the random vector $\xi = (\mathbf{t}_1, \mathbf{t}_2, \mathbf{t}_3)$ is formed by

the three yields and that ξ can take on three different values, say ξ_1 , ξ_2 , and ξ_3 , which represent $(t_1(1), t_2(1), t_3(1))$, $(t_1(2), t_2(2), t_3(2))$, and $(t_1(3), t_2(3), t_3(3))$, respectively.

An alternative interpretation would be to say that the random vector $\xi(s)$ in fact depends on the scenario s , which takes on three different values¹.

In this section, we have illustrated two possible representations of a stochastic program. The form (1.2) given earlier for the farmer's example is known as the extensive form. It is obtained by associating one decision vector in the second-stage to each possible realization of the random vector. The second form (1.3) or (1.4) is called the implicit representation of the stochastic program. A more condensed implicit representation is obtained by defining $\mathcal{Q}(x) = E_{\xi} Q(x, \xi)$ as the *value function* or *recourse function* so that (1.3) can be written as

$$\begin{aligned} & \min c^T x + \mathcal{Q}(x) \\ & \text{s. t. } Ax = b, \\ & \quad x \geq 0. \end{aligned} \tag{1.5}$$

d. Continuous random variables

Contrary to the assumption made in Section 1.1b., we may also assume that yields for the different crops are independent. In that case, we may as well consider a continuous random vector for the yields. To illustrate this, let us assume that the yield for each crop i can be appropriately described by a uniform random variable, inside some range $[l_i, u_i]$ (see Appendix A.2). For the sake of comparison, we may take l_i to be 80% of the mean yield and u_i to be 120% of the mean yield so that the expectations for the yields will be the same as in Section 1.1b. Again, the decisions on land allocation are first-stage decisions because they are taken before knowledge of the yields. Second-stage decisions are purchases and sales after the growing period. The second-stage formulation can again be described as $\mathcal{Q}(x) = E_{\xi} Q(x, \xi)$, where $Q(x, \xi)$ is the value of the second stage for a given realization of the random vector.

Now, in this particular example, the computation of $Q(x, \xi)$ can be separated among the three crops due to independence of the random vector. (Note that this separability property also holds in the discrete representation of Section 1.1b.) We can then write:

$$E_{\xi} Q(x, \xi) = \sum_{i=1}^3 E_{\xi} Q_i(x_i, \xi) = \sum_{i=1}^3 \mathcal{Q}_i(x_i), \tag{1.6}$$

where $Q_i(x_i, \xi)$ is the optimal second-stage value of purchases and sales of crop i .

We are in fact in position to give an exact analytical expression for the second-stage value functions $\mathcal{Q}_i(x_i)$, $i = 1, \dots, 3$. We first consider sugar beet sales. For

¹ Note that the decisions y_1 , y_2 , w_1 , w_2 , w_3 , and w_4 also depend on the scenario. This dependence is not always made explicit. It appears explicitly in (1.7) but not in (1.4).

a given value $t_3(\xi)$ of the sugar beet yield, one obtains the following second-stage problem:

$$\begin{aligned} Q_3(x_3, \xi) &= \min -36w_3(\xi) - 10w_4(\xi) \\ \text{s. t. } w_3(\xi) + w_4(\xi) &\leq t_3(\xi)x_3, \\ w_3(\xi) &\leq 6000, \\ w_3(\xi), w_4(\xi) &\geq 0. \end{aligned} \quad (1.7)$$

The optimal decisions for this problem are clearly to sell as many sugar beets as possible at the favorable price, and to sell the possible remaining production at the unfavorable price, namely

$$\begin{aligned} w_3(\xi) &= \min[6000, t_3(\xi)x_3], \\ w_4(\xi) &= \max[t_3(\xi)x_3 - 6000, 0]. \end{aligned} \quad (1.8)$$

This results in a second-stage value of

$$Q_3(x_3, \xi) = -36 \min[6000, t_3(\xi)x_3] - 10 \max[t_3(\xi)x_3 - 6000, 0].$$

We first assume that the surface x_3 devoted to sugar beets will not be so large that the quota would be exceeded for any possible yield or so small that production would always be less than the quota for any possible yield. In other words, we assume that the following relation holds:

$$l_3x_3 \leq 6000 \leq u_3x_3, \quad (1.9)$$

where, as already defined, l_3 and u_3 are the bounds on the possible values of $t_3(\xi)$. Under this assumption, the expected value of the second stage for sugar beet sales is

$$\begin{aligned} \mathcal{Q}_3(x_3) &= E_{\xi} Q_3(x_3, \xi) \\ &= - \int_{l_3}^{6000/x_3} 36tx_3f(t)dt \\ &\quad - \int_{6000/x_3}^{u_3} (216000 + 10tx_3 - 60000)f(t)dt, \end{aligned}$$

where $f(t)$ denotes the density of the random yield $t_3(\xi)$. Given the assumption that this density is uniform over the interval $[l_3, u_3]$, one obtains, after some computation, the following analytical expression

$$\mathcal{Q}_3(x_3) = -18 \frac{(u_3^2 - l_3^2)x_3}{u_3 - l_3} + \frac{13(u_3x_3 - 6000)^2}{x_3(u_3 - l_3)},$$

which can also be expressed as

$$\mathcal{Q}_3(x_3) = -36\bar{t}_3x_3 + \frac{13(u_3x_3 - 6000)^2}{x_3(u_3 - l_3)}, \quad (1.10)$$

where \bar{t}_3 denotes the expected yield for sugar beet production, which is $\frac{u_3+l_3}{2}$ for a uniform density.

Note that assumption (1.9) is not really limiting. We can still compute the analytical expression of $\mathcal{Q}_3(x_3)$ for the other situations.

For example, if the surface x_3 is such that the production exceeds the quota for any possible yield ($l_3x_3 > 6000$), then the optimal second-stage decisions are simply

$$\begin{aligned} w_3(\xi) &= 6000, \\ w_4(\xi) &= t_3(\xi)x_3 - 6000, \text{ for all } \xi. \end{aligned}$$

The second-stage value for a given ξ is now

$$Q_3(x_3, \xi) = -216000 - 10(t_3(\xi)x_3 - 6000) = -156000 - 10t_3(\xi)x_3,$$

and the expected value is simply

$$\mathcal{Q}_3(x_3) = -156000 - 10\bar{t}_3x_3. \quad (1.11)$$

Similarly, if the surface devoted to sugar beets is so small that for any yield the production is lower than the quota, the second-stage value function is

$$\mathcal{Q}_3(x_3) = -36\bar{t}_3x_3. \quad (1.12)$$

We may therefore draw the graph of the function $\mathcal{Q}_3(x_3)$ for all possible values of x_3 as in Figure 1. Note that with our assumption of $\bar{t}_3 = 20$, we would then have the limits on x_3 in (1.9) as $250 \leq x_3 \leq 375$.

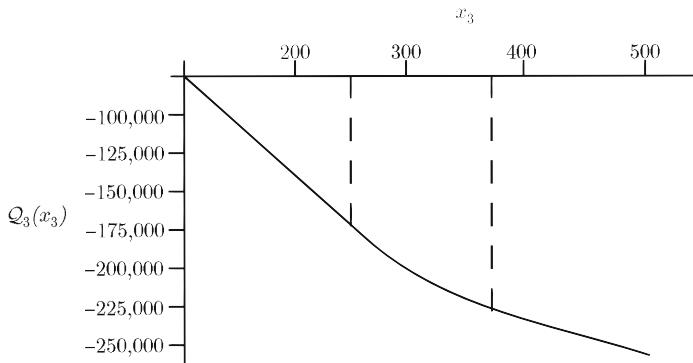


Fig. 1 The expected recourse value for sugar beets as a function of acres planted.

We immediately see that the function has three different pieces. Two of these pieces are linear and one is nonlinear, but the function $\mathcal{Q}_3(x_3)$ is continuous and convex. This property will be proved when we consider the generalization of this problem,

known as the *news vendor, newsboy, or Christmas tree problem*. In fact, this property holds for a large class of second-stage problems, as will be seen in Chapter 3.

Similar computations can be done for the other two crops. For wheat, we obtain

$$\mathcal{Q}_1(x_1) = \begin{cases} 47600 - 595x_1 & \text{for } x_1 \leq 200/3, \\ 119\frac{(200-2x_1)^2}{x_1} - 85\frac{(200-3x_1)^2}{x_1} & \text{for } \frac{200}{3} \leq x_1 \leq 100, \\ 34000 - 425x_1 & \text{for } x_1 \geq 100, \end{cases}$$

and, for corn, we obtain

$$\mathcal{Q}_2(x_2) = \begin{cases} 50400 - 630x_2 & \text{for } x_2 \leq 200/3, \\ 87.5\frac{(240-2.4x_2)^2}{x_2} - 62.5\frac{(240-3.6x_2)^2}{x_2} & \text{for } 200/3 \leq x_2 \leq 100, \\ 36000 - 450x_2 & \text{for } x_2 \geq 100. \end{cases}$$

The global problem is therefore

$$\begin{aligned} \min \quad & 150x_1 + 230x_2 + 260x_3 + \mathcal{Q}_1(x_1) + \mathcal{Q}_2(x_2) + \mathcal{Q}_3(x_3) \\ \text{s. t. } & x_1 + x_2 + x_3 \leq 500, \\ & x_1, x_2, x_3 \geq 0. \end{aligned}$$

Given that the three functions $\mathcal{Q}_i(x_i)$ are convex, continuous, and differentiable functions and the first-stage objective is linear, this problem is a convex program for which Karush-Kuhn-Tucker (K-K-T) conditions are necessary and sufficient for a global optimum. (This result is from nonlinear programming. For more on this result about optimality, see Section 2.11.) Denoting by λ the multiplier of the surface constraint and as before by c_i the first-stage objective coefficient of crop i , the K-K-T conditions require

$$\begin{aligned} x_i \left[c_i + \frac{\partial \mathcal{Q}_i(x_i)}{\partial x_i} + \lambda \right] &= 0, \quad c_i + \frac{\partial \mathcal{Q}_i(x_i)}{\partial x_i} + \lambda \geq 0, \quad x_i \geq 0, \quad i = 1, 2, 3; \\ \lambda [x_1 + x_2 + x_3 - 500] &= 0, \quad x_1 + x_2 + x_3 \leq 500, \quad \lambda \geq 0. \end{aligned}$$

Assume the optimal solution is such that $100 \leq x_1$, $\frac{200}{3} \leq x_2 \leq 100$, and $250 \leq x_3 \leq 375$ with $\lambda \neq 0$. Then the conditions read

$$\begin{cases} -275 + \lambda = 0, \\ -76 - \frac{1.44 \cdot 10^6}{x_2^2} + \lambda = 0, \\ 476 - \frac{5.85 \cdot 10^7}{x_3^2} + \lambda = 0, \\ x_1 + x_2 + x_3 = 500. \end{cases}$$

Solving this system of equations gives $\lambda = 275.00$, $x_1 = 135.83$, $x_2 = 85.07$, $x_3 = 279.10$, which satisfies all the required conditions and is therefore optimal. We observe that this solution is similar to the one obtained by using the scenario approach, although more surface is devoted to sugar beet and less to wheat than before. This similarity represents a characteristic robustness of a well-formed stochastic programming formulation. We shall consider it in more detail in our discussion of approximations in Chapter 8.

e. The news vendor problem

The previous section illustrates an example of a famous and basic problem in stochastic optimization, *the news vendor problem*. In this problem, a news vendor goes to the publisher every morning and buys x newspapers at a price of c per paper. This number is usually bounded above by some limit u , representing either the news vendor's purchase power or a limit set by the publisher to each vendor. The vendor then walks along the streets to sell as many newspapers as possible at the selling price q . Any unsold newspaper can be returned to the publisher at a return price r , with $r < c$.

We are asked to help the news vendor decide how many newspapers to buy every morning. Demand for newspapers varies over days and is described by a random variable ξ .

It is assumed here that the news vendor cannot return to the publisher during the day to buy more newspapers. Other news vendors would have taken the remaining newspapers. Readers also only want the last edition.

To describe the news vendor's profit, we define y as the effective sales and w as the number of newspapers returned to the publisher at the end of the day. We may then formulate the problem as

$$\begin{aligned} \min \quad & cx + \mathcal{D}(x) \\ \text{subject to} \quad & 0 \leq x \leq u, \end{aligned}$$

where

$$\mathcal{D}(x) = E_\xi Q(x, \xi)$$

and

$$\begin{aligned} Q(x, \xi) &= \min -qy(\xi) - rw(\xi) \\ \text{s. t. } & y(\xi) \leq \xi, \\ & y(\xi) + w(\xi) \leq x, \\ & y(\xi), w(\xi) \geq 0, \end{aligned}$$

where again E_ξ denotes the mathematical expectation with respect to ξ .

In this notation, $-\mathcal{Q}(x)$ is the expected profit on sales and returns, while $-Q(x, \xi)$ is the profit on sales and returns if the demand is at level ξ . The model illustrates the two-stage aspect of the news vendor problem. The buying decision has to be taken before any information is given on the demand. When demand is known in the so-called second stage, which represents the end of the sales period of a given edition, the profit can be computed. This is done using the following simple rule:

$$\begin{aligned} y^*(\xi) &= \min(\xi, x), \\ w^*(\xi) &= \max(x - \xi, 0). \end{aligned}$$

Sales can never exceed the number of available newspapers or the demand. Returns occur only when demand is less than the number of newspapers available. The second-stage expected value function is simply

$$\mathcal{Q}(x) = E_\xi[-q \min(\xi, x) - r \max(x - \xi, 0)].$$

As we will learn later, this function is convex and continuous. It is also differentiable when ξ is a continuous random vector. In that case, the optimal solution of the news vendor's problem is simply:

$$\begin{cases} x = 0 & \text{if } c + \mathcal{Q}'(0) > 0, \\ x = u & \text{if } c + \mathcal{Q}'(u) < 0, \\ \text{a solution of } c + \mathcal{Q}'(x) = 0 & \text{otherwise,} \end{cases}$$

where $\mathcal{Q}'(x)$ denotes the first order derivative of $\mathcal{Q}(x)$ evaluated at x .

By construction, $\mathcal{Q}(x)$ can be computed as

$$\begin{aligned} \mathcal{Q}(x) &= \int_{-\infty}^x (-q\xi - r(x - \xi)) dF(\xi) + \int_x^\infty -qx dF(\xi) \\ &= -(q - r) \int_{-\infty}^x \xi dF(\xi) - rx F(x) - qx(1 - F(x)), \end{aligned}$$

where $F(\xi)$ represents the cumulative probability distribution of ξ (see Section 2.1).

Integrating by parts, we observe that

$$\int_{-\infty}^x \xi dF(\xi) = xF(x) - \int_{-\infty}^x F(\xi) d\xi$$

under mild conditions on the distribution function $F(\xi)$. It follows that

$$\mathcal{Q}(x) = -qx + (q-r) \int_{-\infty}^x F(\xi) d\xi .$$

We may thus conclude that

$$\mathcal{Q}'(x) = -q + (q-r)F(x)$$

and therefore that the optimal solution is

$$\begin{cases} x^* = 0 & \text{if } \frac{q-c}{q-r} < F(0) , \\ x^* = u & \text{if } \frac{q-c}{q-r} > F(u) , \\ x^* = F^{-1}\left(\frac{q-c}{q-r}\right) & \text{otherwise,} \end{cases}$$

where $F^{-1}(\alpha)$ is the α -quantile of F (see Section 2.1). If F is continuous, $x = F^{-1}(\alpha)$ means $\alpha = F(x)$. Any reasonable representation of the demand would imply $F(0) = 0$ so that the solution is never $x^* = 0$.

As we shall see in Chapter 3, this problem is an example of a basic type of stochastic program called the *stochastic program with simple recourse*. The ideas of this section can be generalized to larger problems in this class of examples. Also observe that, as such, we only come to a partial answer, under the form of an expression for x^* . The vendor may still need to consult a statistician, who would provide an accurate cumulative distribution $F(\cdot)$. Only then will a precise figure be available for x^* .

Exercises

1. Value of the stochastic solution

Assume the farmer allocates his land according to the solution of Table 2, i.e., 120 acres for wheat, 80 acres for corn, and 300 acres for sugar beets. Show that if yields are random (20% below average, average, and 20% above average for all crops with equal probability one third), his expected annual profit is \$107,240. To do this observe that planting costs are certain but sales and purchases depend on the yield. In other words, fill in a table such as Table 5 but with the first-stage decisions given here.

2. Price effect

When yields are good for the farmer, they are usually also good for many other farmers. The supply is thus increasing, which will lower the prices. As an example, we may consider prices going down by 10% for corn and wheat when yields are above average and going up by 10% when yields are below average. Formulate the model where these changes in prices affect both sales and purchases of corn and wheat. Assume sugar beet prices are not affected by yields.

3. Binary first stage

Consider the case where the farmer possesses four fields of sizes 185, 145, 105, and 65 acres, respectively. Observe that the total of 500 acres is unchanged. Now, the fields are unfortunately located in different parts of the village. For reasons of efficiency the farmer wants to raise only one type of crop on each field. Formulate this model as a two-stage stochastic program with a first-stage program with binary variables.

4. Integer second stage

Consider the case where sales and purchases of corn and wheat can only be obtained through contracts involving multiples of hundred tons. Formulate the model as a stochastic program with a mixed-integer second stage.

5. Consider any one of Exercises 2 to 4. Using standard mixed integer programming software, obtain an optimal solution of the extensive form of the stochastic program. Compute the expected value of perfect information and the value of the stochastic solution.

6. Multistage program

It is typical in farming to implement crop rotation in order to maintain good soil quality. Sugar beets would, for example, appear in triennial crop rotation, which means they are planted on a given field only one out of three years. Formulate a multistage program to describe this situation. To keep things simple, describe the case when sugar beets cannot be planted two successive years on the same field, and assume no such rule applies for wheat and corn.

(On a two-year basis, this exercise consists purely of formulation: with the basic data of the example, the solution is clearly to repeat the optimal solution in Table 5, i.e., to plant 170 acres of wheat, 80 acres of corn, and 250 acres of sugar beets. The problem becomes more relevant on a three-year basis. It is also relevant on a two-year basis with fields of the sizes given in Exercise 1.

In terms of formulation, it is sufficient to consider a three-stage model. The first stage consists of first-year planting. The second stage consists of first-year purchases and sales and second-year planting. The third-stage consists of second-year purchases and sales. Alternatively, a four-stage model can be built, separating first-year purchases and sales from second-year planting. Also discuss the question of discounting the revenues and expenses of the various stages.)

7. Risk aversion

Economic theory tells us that, like many other people, the farmer would normally act as a risk-averse person. There are various ways to model risk aversion. One simple way is to plan for the worst case. More precisely, it consists of maximizing the profit under the worst situation. Note that for some models, it is not known in advance which scenario will turn out to induce the lowest profit.

In our example, the worst situation corresponds to Scenario 3 (below average yields). Planning for the worst case implies the solution of Table 4 is optimal.

- (a) Compute the loss in expected profit if that solution is taken.
- (b) A median situation would be to require a reasonable profit under the worst case. Find the solution that maximizes the expected profit under the constraint that in the worst case the profit does not fall below \$58,000. What is now the loss in expected profit?
- (c) Repeat part (b) with other values of minimal profit: \$56,000, \$54,000, \$52,000, \$50,000, and \$48,000. Graph the curve of expected profit loss. Also compare the associated optimal decisions.

8. Data fluctuations

Table 1 contains mean data over a relatively long period, from the late nineties till 2006. Yield fluctuations have been treated through random yields. What about other data's fluctuations? Planting costs in euros have not changed so much over time. (The story is different when expressed in dollars. However, the farmer's decisions are unaffected by currency modifications as they simply shift the objective function. The only element which could be affected by currency rates is the world price of sugar beets, but it has stayed low enough to play no significant role for the farmer.) Starting from the deterministic model (1.1), sensitivity analysis tells us that the optimal solution remains valid if wheat and corn selling prices remain below 220 and 168.333, respectively, and if sugar beet's favorable price remains over 26.75. This implies the solution of model (1.1) remains stable even if relatively large changes in prices occur (with the provision that the results of linear programming sensitivity analysis are guaranteed to hold when only one price is changing at a time). For joint modifications of prices, it is interesting to look at the returns of each crop. Then, one can see that profound changes in solutions only occur if the sales of a given crop provide a higher return than sugar beets at the favorable price. This happened in 2007, with wheat's price more than doubling in a 12-month period. At the moment of this writing, the current costs and prices are as follows (rounded figures):

	Wheat	Corn	Sugar Beets
Yield (T/acre)	2.5	3	20
Planting cost (\$/acre)	180	280	310
Selling price (\$/T)	300	170	41 under 6000 T 11 above 6000 T

The increase in wheat's selling price is due to a strong demand and low yields in Asia. These conditions may not prevail next year. Consider a model with a random selling price of wheat being 300 or 220 with equal probability. Purchase prices are as before 40% higher than selling prices. Compare the optimal solution with that of Table 5. How much would a farmer be willing to pay for a perfect forecast on the selling price of wheat?

9. If prices are also random variables, the news vendor's problem becomes more complicated. However, if prices and demands are independent random variables, show that the solution of the news vendor's problem is the one obtained before, where q and r are replaced by their expected values. Indicate under which conditions the same proposition is true for the farmer's problem.
10. In the news vendor's problem, we have assumed for simplicity that the random variable takes value from $-\infty$ to $+\infty$. Show that the optimal decisions are insensitive to this assumption, so that if the random variables have a nonzero density on a limited interval then the optimal solutions are obtained by the same analytical expression.
11. Suppose $c = 10$, $q = 25$, $r = 5$, and demand is uniform on $[50, 150]$. Find the optimal solution of the news vendor problem. Also, find the optimal solution of the deterministic model obtained by assuming a demand of 100. What is the value of the stochastic solution?

1.2 Financial Planning and Control

Financial decision-making problems can often be modeled as stochastic programs. In fact, the essence of financial planning is the incorporation of risk into investment decisions. The area represents one of the largest application areas of stochastic programming. Many references can be found in, for example, Mulvey and Vladimirov [1989, 1991b, 1992], Ziemba and Vickson [1975], and Zenios [1993].

We consider a simple example that illustrates additional stochastic programming properties. As in the farming example of Section 1.1, this example involves randomness in the constraint matrix instead of the right-hand side elements. These random variables reflect uncertain investment yields.

This section's example also has the characteristic that decisions are highly dependent on past outcomes. In the following capacity expansion problem of Section 1.3, this is not the case. In Chapter 3, we define this difference by a block separable recourse property that is present in some capacity expansion and similar problems.

For the current problem, suppose we wish to provide for a child's college education Y years from now. We currently have \$ b to invest in any of I investments. After Y years, we will have a wealth that we would like to have exceed a tuition goal of \$ G . We suppose that we can change investments every v years, so we have $H = Y/v$ investment periods. For our purposes here, we ignore transaction costs and taxes on income although these considerations would be important in reality. We also assume that all figures are in constant dollars.

In formulating the problem, we must first describe our objective in mathematical terms. We suppose that exceeding \$ G after Y years would be equivalent to our having an income of $q\%$ of the excess while not meeting the goal would lead to borrowing for a cost $r\%$ of the amount short. This gives us the concave utility

function in Figure 2. Many other forms of nonlinear utility functions are, of course, possible. See Kallberg and Ziemba [1983] for a description of their relevance in financial planning.

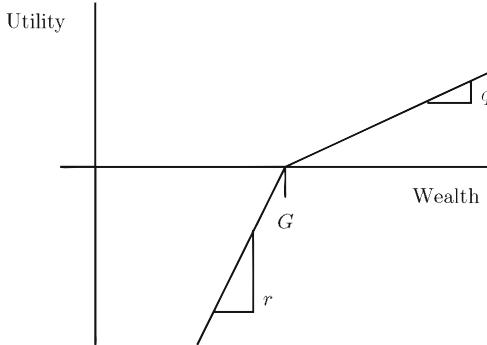


Fig. 2 Utility function of wealth at year Y for a goal G .

The major uncertainty in this model is the return on each investment i within each period t . We describe this random variable as $\xi(i, t) = \xi(i, t, \omega)$ where ω is some underlying random element. The decisions on investments will also be random. We describe these decisions as $\mathbf{x}(i, t) = x(i, t, \omega)$. From the randomness of the returns and investment decisions, our final wealth will also be a random variable.

A key point about this investment model is that we cannot completely observe the random element ω when we make all our decisions $x(i, t, \omega)$. We can only observe the returns that have already taken place. In stochastic programming, we say that we cannot *anticipate* every possible outcome so our decisions are *nonanticipative* of future outcomes. Before the first period, this restriction corresponds to saying that we must make fixed investments, $x(i, 1)$, for all $\omega \in \Omega$, the space of all random elements or, more specifically, returns that could possibly occur.

To illustrate the effects of including stochastic outcomes as well as modeling effects from choosing the time horizon Y and the coarseness of the period approximations H , we use a simple example with two possible investment types, stocks ($i = 1$) and government securities (bonds) ($i = 2$). We begin by setting Y at 15 years and allow investment changes every five years so that $H = 3$.

We assume that, over the three decision periods, eight possible scenarios may occur. The scenarios correspond to independent and equal likelihoods of having (inflation-adjusted) returns of 1.25 for stocks and 1.14 for bonds or 1.06 for stocks and 1.12 for bonds over the five-year period. We indicate the scenarios by an index $s = 1, \dots, 8$, which represents a collection of the outcomes ω that have common characteristics (such as returns) in a specific model. When we wish to allow more general interpretations of the outcomes, we use the base element ω . With the scenarios defined here, we assign probabilities for each s , $p(s) = 0.125$. The returns are $\xi(1, t, s) = 1.25$, $\xi(2, t, s) = 1.14$ for $t = 1, s = 1, \dots, 4$, for $t = 2$,

$s = 1, 2, 5, 6$, and for $t = 3$, $s = 1, 3, 5, 7$. In the other cases, $\xi(1, t, s) = 1.06$, $\xi(2, t, s) = 1.12$.

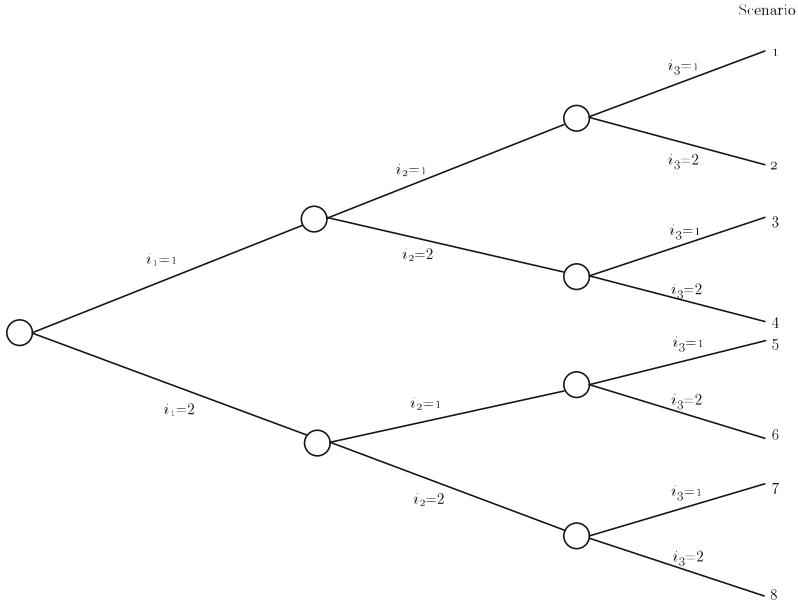


Fig. 3 Tree of scenarios for three periods.

The eight scenarios are represented by the tree in Figure 3. The scenario tree divides into branches corresponding to different realizations of the random returns. Because Scenarios 1 to 4, for example, have the same return for $t = 1$, they all follow the same first branch. Scenarios 1 and 2 then have the same second branch and finally divide completely in the last period. To show this more explicitly, we may refer to each scenario by the history of returns indexed by s_t for periods $t = 1, 2, 3$ as indicated on the tree in Figure 3. In this way, Scenario 1 may also be represented as $(s_1, s_2, s_3) = (1, 1, 1)$.

With the tree representation, we need only have a decision vector for each node of the tree. The decisions at $t = 1$ are just $x(1, 1)$ and $x(2, 1)$ for the amounts invested in stocks (1) and bonds (2) at the outset. For $t = 2$, we would have $x(i, 2, s_1)$ where $i = 1, 2$ for the type of investment and $s_1 = 1, 2$ for the first-period return outcome. Similarly, the decisions at $t = 3$ are $x(i, 3, s_1, s_2)$.

With these decision variables defined, we can formulate a mathematical program to maximize expected utility. Because the concave utility function in Figure 1 is piecewise linear, we just need to define deficit or shortage and excess or surplus variables, $w(i_1, i_2, i_3)$ and $y(i_1, i_2, i_3)$, and we can maintain a linear model. The objective is simply a probability- and penalty-weighted sum of these terms, which, in general, becomes:

$$\sum_{s_H} \cdots \sum_{s_1} p(s_1, \dots, s_H) (-rw(s_1, \dots, s_H) + qy(s_1, \dots, s_H)) .$$

The first-period constraint is simply to invest the initial wealth:

$$\sum_i x(i, 1) = b .$$

The constraints for periods $t = 2, \dots, H$ are, for each s_1, \dots, s_{t-1} :

$$\begin{aligned} \sum_i -\xi(i, t-1, s_1, \dots, s_{t-1}) x(i, t-1, s_1, \dots, s_{t-2}) \\ + \sum_i x(i, t, s_1, \dots, s_{t-1}) = 0 , \end{aligned}$$

while the constraints for period H are:

$$\begin{aligned} \sum_i \xi(i, H, s_1, \dots, s_H) x(i, H, s_1, \dots, s_{H-1}) \\ - y(s_1, \dots, s_H) + w(s_1, \dots, s_H) = G . \end{aligned}$$

Other constraints restrict the variables to be non-negative.

To specify the model in this example, we use initial wealth, $b = 55,000$; target value, $G = 80,000$; surplus reward, $q = 1$; and shortage penalty, $r = 4$. The result is a stochastic program in the following form where the units are thousands of dollars:

$$\begin{aligned} \max z = & \sum_{s_1=1}^2 \sum_{s_2=1}^2 \sum_{s_3=1}^2 0.125(y(s_1, s_2, s_3) - 4w(s_1, s_2, s_3)) & (2.1) \\ \text{s. t.} & x(1, 1) + x(2, 1) = 55 , \\ & -1.25x(1, 1) - 1.14x(2, 1) + x(1, 2, 1) + x(2, 2, 1) = 0 , \\ & -1.06x(1, 1) - 1.12x(2, 1) + x(1, 2, 2) + x(2, 2, 2) = 0 , \\ & -1.25x(1, 2, 1) - 1.14x(2, 2, 1) + x(1, 3, 1, 1) + x(2, 3, 1, 1) = 0 , \\ & -1.06x(1, 2, 1) - 1.12x(2, 2, 1) + x(1, 3, 1, 2) + x(2, 3, 1, 2) = 0 , \\ & -1.25x(1, 2, 2) - 1.14x(2, 2, 2) + x(1, 3, 2, 1) + x(2, 3, 2, 1) = 0 , \\ & -1.06x(1, 2, 2) - 1.12x(2, 2, 2) + x(1, 3, 2, 2) + x(2, 3, 2, 2) = 0 , \\ & 1.25x(1, 3, 1, 1) + 1.14x(2, 3, 1, 1) - y(1, 1, 1) + w(1, 1, 1) = 80 , \\ & 1.06x(1, 3, 1, 1) + 1.12x(2, 3, 1, 1) - y(1, 1, 2) + w(1, 1, 2) = 80 , \\ & 1.25x(1, 3, 1, 2) + 1.14x(2, 3, 1, 2) - y(1, 2, 1) + w(1, 2, 1) = 80 , \\ & 1.06x(1, 3, 1, 2) + 1.12x(2, 3, 1, 2) - y(1, 2, 2) + w(1, 2, 2) = 80 , \\ & 1.25x(1, 3, 2, 1) + 1.14x(2, 3, 2, 1) - y(2, 1, 1) + w(2, 1, 1) = 80 , \\ & 1.06x(1, 3, 2, 1) + 1.12x(2, 3, 2, 1) - y(2, 1, 2) + w(2, 1, 2) = 80 , \\ & 1.25x(1, 3, 2, 2) + 1.14x(2, 3, 2, 2) - y(2, 2, 1) + w(2, 2, 1) = 80 , \\ & 1.06x(1, 3, 2, 2) + 1.12x(2, 3, 2, 2) - y(2, 2, 2) + w(2, 2, 2) = 80 , \\ & x(i, t, s_1, \dots, s_{t-1}) \geq 0 , y(s_1, s_2, s_3) \geq 0 , w(s_1, s_2, s_3) \geq 0 , \\ & \text{for all } i, t, s_1, s_2, s_3 . \end{aligned}$$

Solving the problem in (2.1) yields an optimal expected utility value of -1.514 . We call this value, RP , for the expected *recourse problem* solution value. The optimal solution (in thousands of dollars) appears in Table 6.

Table 6 Optimal solution with three-period stochastic program.

Period, Scenario	Stock	Bonds
1,1-8	41.5	13.5
2,1-4	65.1	2.17
2,5-8	36.7	22.4
3,1-2	83.8	0.00
3,3-4	0.00	71.4
3,5-6	0.00	71.4
3,7-8	64.0	0.00
Scenario	Above G	Below G
1	24.8	0.00
2	8.87	0.00
3	1.43	0.00
4	0.00	0.00
5	1.43	0.00
6	0.00	0.00
7	0.00	0.00
8	0.00	12.2

In this solution, the initial investment is heavily in stock (\$41,500) with only \$13,500 in bonds. Notice the reaction to first-period outcomes, however. In the case of Scenarios 1 to 4, stocks are even more prominent, while Scenarios 5 to 8 reflect a more conservative government security portfolio. In the last period, notice how the investments are either completely in stocks or completely in bonds. This is a general trait of one-period decisions. It occurs here because in Scenarios 1 and 2, there is no risk of missing the target. In Scenarios 3 to 6, stock investments may cause one to miss the target, so they are avoided. In Scenarios 7 and 8, the only hope of reaching the target is through stocks.

We compare the results in Table 6 to a deterministic model in which all random returns are replaced by their expectation. For that model, because the expected return on stock is 1.155 in each period, while the expected return on bonds is only 1.13 in each period, the optimal investment plan places all funds in stocks in each period. If we implement this policy each period, but instead observed the random returns, we would have an expected utility called the *expected value* solution, or EV . In this case, we would realize an expected utility of $EV = -3.788$, while the stochastic program value is again $RP = -1.514$. The difference between these quantities is the value of the stochastic solution:

$$VSS = RP - EV = -1.514 - (-3.788) = 2.274 .$$

This comparison gives us a measure of the utility value in using a decision from a stochastic program compared to a decision from a deterministic program. Another comparison of models is in terms of the probability of reaching the goal. Models with these types of objectives are called *chance-constrained programs or programs with probabilistic constraints* (see Charnes and Cooper [1959] and Prékopa [1973]). Notice that the stochastic program solution reaches the goal 87.5% of the time. The expected value deterministic model solution only reaches the goal 50% of the time. In this case, the value of the stochastic solution may be even more significant.

The formulation we gave in (2.1) can become quite cumbersome as the time horizon, H , increases and the decision tree of Figure 3 grows quite bushy. Another modeling approach to this type of multistage problem is to consider the full horizon scenarios, s , directly, without specifying the history of the process. We then substitute a scenario set S for the random elements Ω . Probabilities, $p(s)$, returns, $\xi(i, t, s)$, and investments, $x(i, t, s)$, become functions of the H -period scenarios and not just the history until period t .

The difficulty is that, when we have split up the scenarios, we may have lost nonanticipativity of the decisions because they would now include knowledge of the outcomes up to the end of the horizon. To enforce nonanticipativity, we add constraints explicitly in the formulation. First, the scenarios that correspond to the same set of past outcomes at each period form groups, $S'_{s_1, \dots, s_{t-1}}$, for scenarios at time t . Now, all actions up to time t must be the same within a group. We do this through an explicit constraint. The new general formulation of (2.1) becomes:

$$\begin{aligned} \max z = & \sum_s p(s)(qy(s) - rw(s)) \\ \text{s. t. } & \sum_{i=1}^I x(i, 1, s) = b, \quad \forall s \in S, \\ & \sum_{i=1}^I \xi(i, t, s)x(i, t-1, s) - \sum_{i=1}^I x(i, t, s) = 0, \quad \forall s \in S, \\ & \quad t = 2, \dots, H, \\ & \sum_{i=1}^I \xi(i, H, s)x(i, H, s) - y(s) + w(s) = G, \\ & \left(\sum_{s' \in S'_{J(s,t)}} p(s')x(i, t, s') \right) - \left(\sum_{s' \in S'_{J(s,t)}} p(s') \right)x(i, t, s) = 0, \\ & \quad \forall 1 \leq i \leq I, \quad \forall 1 \leq t \leq H, \quad \forall s \in S, \\ & x(i, t, s) \geq 0, \quad y(s) \geq 0, \quad w(s) \geq 0, \\ & \quad \forall 1 \leq i \leq I, \quad \forall 1 \leq t \leq H, \quad \forall s \in S, \end{aligned} \tag{2.2}$$

where $J(s, t) = \{s_1, \dots, s_{t-1}\}$ such that $s \in S'_{s_1, \dots, s_{t-1}}$. Note that the last equality constraint indeed forces all decisions within the same group at time t to be the same. Formulation (2.2) has a special advantage for the problem here because these

nonanticipativity constraints are the only constraints linking the separate scenarios. Without them, the problem would decompose into a separate problem for each s , maintaining the structure of that problem.

In modeling terms, this simple additional constraint makes it relatively easy to move from a deterministic model to a stochastic model of the same problem. This ease of conversion can be especially useful in modeling languages. For example, Figure 4 gives a complete AMPL (Fourer, Gay, and Kernighan [1993]) model of the problem in (2.2). In this language, *set*, *param*, and *var* are keywords for sets, parameters, and variables. The addition of the scenario indicators and nonanticipativity constraints (*nonanticip*) are the only additions to a deterministic model.

```
# This problem describes a simple financial planning problem
# for financing college education
set investments; # different investment options
param initwealth; # initial holdings
param H; # number of periods
param scenarios; # number of scenarios (total S)
# The following 0-1 array shows which scenarios are combined at period H
param scen_links { 1..scenarios,1..scenarios,1..H } ;
param target; # target value G at time H
param invest; # value of investing beyond target value
param penalty; # penalty for not meeting target
param return { investments,1..scenarios,1..H } ; # return on each inv
param prob { 1..scenarios } ; # probability of each scenario
# variables
var amtinvest { investments,1..scenarios,1..H } _i=0; #actual amounts inv'd
var above_target { 1..scenarios } _i=0; # amt above final target
var below_target { 1..scenarios } _i=0; # amt below final target
# objective
maximize exp_value : sum { i in 1..scenarios } prob[i]*(invest*above_target[i]
- penalty*below_target[i]);
# constraints
subject to budget { i in 1..scenarios } :
sum { k in investments } (amtinvest[k,i,1]) = initwealth;#invest initial wealth
subject to nonanticip { k in investments,j in 1..scenarios,i in 1..H } :
(sum { i in 1..scenarios } scen_links[j,i,t]*prob[i]*amtinvest[k,i,t]) -
(sum { i in 1..scenarios } scen_links[j,i,t]*prob[i])*_
amtinvest[k,j,H] = 0; # makes all investments nonanticipative
subject to balance { j in 1..scenarios, t in 1..H-1 } :
(sum { k in investments } return[k,j,t]*amtinvest[k,j,t]) - sum { k in investments } amtinvest[k,j,t+1] = 0; # reinvest each time period
subject to scenario_value { j in 1..scenarios } : (sum { k in investments } return[k,j,H]*amtinvest[k,j,H]) - above_target[j] +
below_target[j] = target; # amounts not meeting target
```

Fig. 4 AMPL format of financial planning model.

Given the ease of this modeling effort, standard optimization procedures can be simply applied to this problem. However, as we noted earlier, the number of scenarios can become extremely large. Standard methods may not be able to solve the problem in any reasonable amount of time, necessitating other techniques. The remaining chapters in this book focus on these other methods and on procedures for creating models that are amenable to those specialized techniques.

In financial problems, it is particularly worthwhile to try to exploit the underlying structure of the problem without the nonanticipativity constraints. This relaxed

problem is in fact a *generalized network* that allows the use of efficient network optimization methods that cannot apply to the full problem in (2.2). We discuss this option more thoroughly in Chapter 5.

With either formulation (2.1) or (2.2), in completing the model, some decisions must be made about the possible set of outcomes or scenarios and the coarseness of the period structure, i.e., the number of periods H allowed for investments. We must also find probabilities to attach to outcomes within each of these periods. These probabilities are often approximations that can, as we shall see in Chapter 8, provide bounds on true values or on uncertain outcomes with incompletely known distributions. A key observation is that the important step is to include stochastic elements at least approximately and that deterministic solutions most often give misleading results.

In closing this section, note that the mathematical form of this problem actually represents a broad class of control problems (see, for example, Varaiya and Wets [1989]). In fact, it is basically equivalent to any control problem governed by a linear system of differential equations. We have merely taken a discrete time approach to this problem. This approach can be applied to the control of a wide variety of electrical, mechanical, chemical, and economic systems. We merely redefine state variables (now, wealth) in each time period and controls (investment levels). The random gain or loss is reflected in the return coefficients. Typically, these types of control problems would have nonlinear (e.g., *quadratic*) costs associated with the control in each time period. This presents no complication for our purposes, so we may include any of these problems as potential applications. In Section 1.4, we will look at a fundamentally nonlinear problem in more detail.

Exercises

1. Suppose you consider just a five-year planning horizon. Choose an appropriate target and solve over this horizon with a single first-period decision.
2. Suppose you implement a buy-and-hold strategy and make a single investment decision without any additional trading until the end of the time horizon. Formulate and solve this problem to determine an optimal allocation.
3. Suppose that goal G is also a random parameter and could be \$75,000 or \$85,000 with equal probabilities. Formulate and solve this problem. Compare this solution to the solution for the problem with a known target.
4. Suppose that every trade (purchase or sale) of an asset involves a transaction cost that is equal to 1% of the amount traded. Re-formulate the problem with this transaction cost and solve for the optimal solution.

1.3 Capacity Expansion

Capacity expansion models optimal choices of the timing and levels of investments to meet future demands of a given product. This problem has many applications. Here we illustrate the case of power plant expansion for electricity generation: we want to find optimal levels of investment in various types of power plants to meet future electricity demand.

We first present a *static deterministic analysis* of the electricity generation problem. *Static* means that decisions are taken only once. *Deterministic* means that the future is supposed to be fully and perfectly known.

Three properties of a given power plant i can be singled out in a static analysis: the investment cost r_i , the operating cost q_i , and the availability factor a_i , which indicates the percent of time the power plant can effectively be operated. Demand for electricity can be considered a single product, but the level of demand varies over time. Analysts usually represent the demand in terms of a so-called *load duration curve* that describes the demand over time in decreasing order of demand level (Figure 5). The curve gives the time, τ , that each demand level, D , is reached. Because here we are concerned with investments over the long run, the load duration curve we consider is taken over the life cycle of the plants.

The load duration curve can be approximated by a piecewise constant curve (Figure 6) with m segments. Let $d_1 = D_1$, $d_j = D_j - D_{j-1}$, $j = 2, \dots, m$ represent the additional power demanded in the so-called *mode* j for a duration τ_j . To obtain a good approximation of the load curve, it is necessary to consider large values of m . In the static situation, the problem consists of finding the optimal investment for each mode j , i.e., to find the particular type of power plant i , $i = 1, \dots, n$, that minimizes the total cost of effectively producing 1 MW (megawatt) of electricity during the time τ_j . It is given by

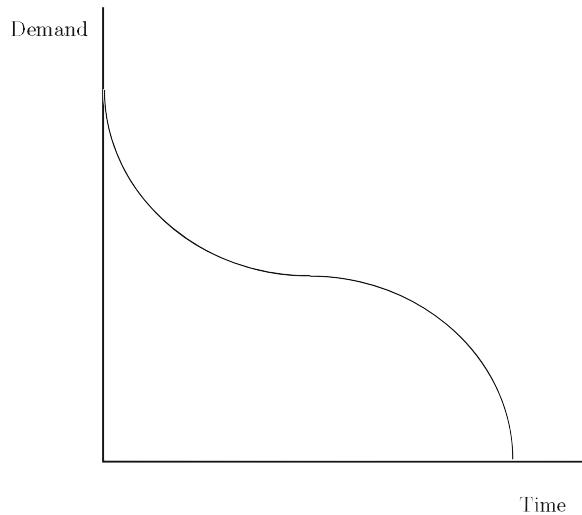
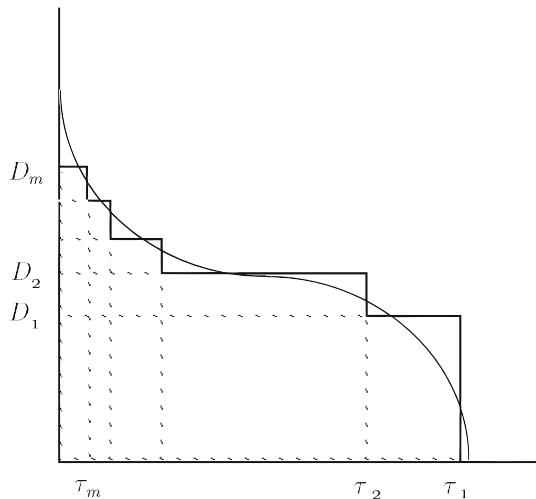
$$i(j) = \operatorname{argmin}_{i=1, \dots, n} \left\{ \frac{r_i + q_i \tau_j}{a_i} \right\}, \quad (3.1)$$

where n is the number of available technologies and argmin represents the index i for which the minimum is achieved.

The static model (3.1) captures one essential feature of the problem, namely, that base load demand (associated with large values of τ_j , i.e., small indices j) is covered by equipment with low operating costs (scaled by availability factor), while peak-load demand (associated with small values of τ_j , i.e., large indices j) is covered by equipment with low investment costs (also scaled by their availability factor). For the sake of completeness, peak-load equipment should also offer operational flexibility.

At least four elements justify considering a *dynamic* or *multistage model* for the electricity generation investment problem:

- the long-term evolution of equipment costs;
- the long-term evolution of the load curve;

**Fig. 5** The load duration curve.**Fig. 6** A piecewise constant approximation of the load duration curve.

- the appearance of new technologies;
- the obsolescence of currently available equipment.

The equipment costs are influenced by technological progress but also (and, for some, drastically) by the evolution of fuel costs.

Of significant importance in the evolution of demand is both the total energy demanded (the area under the load curve) and the peak-level D_m , which determines the total capacity that should be available to cover demand. The evolution of the load curve is determined by several factors, including the level of activity in industry, energy savings in general, and the electricity producers' rate policy.

The appearance of new technologies depends on the technical and commercial success of research and development while obsolescence of available equipment depends on past decisions and the technical lifetime of equipment. All the elements together imply that it is no longer optimal to invest only in view of the short-term ordering of equipment given by (3.1) but that a long-term optimal policy should be found.

The following multistage model can be proposed. Let

- $t = 1, \dots, H$ index the periods or stages;
- $i = 1, \dots, n$ index the available technologies;
- $j = 1, \dots, m$ index the operating modes in the load duration curve.

Also define the following:

- a_i = availability factor of i ;
- L_i = lifetime of i ;
- g_i^t = existing capacity of i at time t , decided before $t = 1$;
- r_i^t = unit investment cost for i at time t (assuming a fixed plant life cycle for each type i of plant);
- q_i^t = unit production cost for i at time t ;
- d_j^t = maximal power demanded in mode j at time t ;
- τ_j^t = duration of mode j at time t .

Consider, finally, the set of decisions

- x_i^t = new capacity made available for technology i at time t ;
- w_i^t = total capacity of i available at time t ;
- y_{ij}^t = capacity of i effectively used at time t in mode j .

The electricity generation H-stage problem can be defined as

$$\min_{x,y,w} \sum_{t=1}^H \left(\sum_{i=1}^n r_i^t \cdot w_i^t + \sum_{i=1}^n \sum_{j=1}^m q_i^t \cdot \tau_j^t \cdot y_{ij}^t \right) \quad (3.2)$$

$$\text{s. t. } w_i^t = w_i^{t-1} + x_i^t - x_i^{t-L_i}, \quad i = 1, \dots, n, \quad t = 1, \dots, H, \quad (3.3)$$

$$\sum_{i=1}^n y_{ij}^t = d_j^t, \quad j = 1, \dots, m, \quad t = 1, \dots, H, \quad (3.4)$$

$$\sum_{j=1}^m y_{ij}^t \leq a_i(g_i^t + w_i^t), \quad i = 1, \dots, n, \quad t = 1, \dots, H, \quad (3.5)$$

$x, y, w \geq 0.$

Decisions in each period t involve new capacities x_i^t made available in each technology and capacities y_{ij}^t operated in each mode for each technology.

Newly decided capacities increase the total capacity w_i^t made available, as given by (3.3), where the equipment's becoming obsolete after its lifetime is also considered. We assume $x_i^t = 0$ if $t \leq 0$, so equation (3.3) only involves newly decided capacities.

By (3.4), the optimal operation of equipment must be chosen to meet demand in all modes using available capacities, which by (3.5) depend on capacities g_i^t decided before $t = 1$, newly decided capacities x_i^t , and the availability factor.

The objective function (3.2) is the sum of the investment plus maintenance costs and operating costs. Compared to (3.1), availability factors enter constraints (3.5) and do not need to appear in the objective function. The operating costs are exactly the same and are based on operating decisions y_{ij}^t , while the investment annuities and maintenance costs r_i^t apply on the cumulative capacity w_i^t . Placing annuities on the cumulative capacity, instead of charging the full investment cost to the decision x_i^t , simplifies the treatment of end of horizon effects and is currently used in many power generation models. It is a special case of the salvage value approach and other period aggregations discussed in Section 10.2.

The same reasons that plead for the use of a multistage model motivate resorting to a *stochastic model*. The evolution of equipment costs, particularly fuel costs, the evolution of total demand, the date of appearance of new technologies, even the lifetime of existing equipment, can all be considered truly random. The main difference between the stochastic model and its deterministic counterpart is in the definition of the variables x_i^t and w_i^t . In particular, x_i^t now represents the new capacity of i decided at time t , which becomes available at time $x_i^{t+\Delta_i}$, where Δ_i is the construction delay for equipment i . In other words, to have extra capacity available at time t , it is necessary to decide at $t - \Delta_i$, when less information is available on the evolution of demand and equipment costs. This is especially important because it would be preferable to be able to wait until the last moment to take decisions that would have immediate impact.

Assume that each decision is now a random variable. Instead of writing an explicit dependence on the random element, ω , we again use boldface notation to denote random variables. We then have:

- \mathbf{x}_i^t = new capacity decided at time t for equipment i , $i = 1, \dots, n$;
- \mathbf{w}_i^t = total capacity of i available and in order at time t ;
- $\boldsymbol{\xi}$ = the vector of random parameters at time t ;

and all other variables as before. The stochastic model is then

$$\min E_{\xi} \sum_{t=1}^H \left(\sum_{i=1}^n \mathbf{r}_i^t \mathbf{w}_i^t + \sum_{i=1}^n \sum_{j=1}^m \mathbf{q}_i^t \tau_j^t \mathbf{y}_{ij}^t \right) \quad (3.6)$$

$$\text{s. t. } \mathbf{w}_i^t = \mathbf{w}_i^{t-1} + \mathbf{x}_i^t - \mathbf{x}_i^{t-L_i}, \quad i = 1, \dots, n, t = 1, \dots, H, \quad (3.7)$$

$$\sum_{i=1}^n \mathbf{y}_{ij}^t = \mathbf{d}_j^t, \quad j = 1, \dots, m, t = 1, \dots, H, \quad (3.8)$$

$$\sum_{j=1}^m \mathbf{y}_{ij}^t \leq a_i(g_i^t + \mathbf{w}_i^{t-\Delta_i}), \quad i = 1, \dots, n, t = 1, \dots, H, \quad (3.9)$$

$$\mathbf{w}, \mathbf{x}, \mathbf{y} \geq 0,$$

where the expectation is taken with respect to the random vector $\xi = (\xi^2, \dots, \xi^H)$. Here, the elements forming ξ^t are the demands, $(\mathbf{d}_1^t, \dots, \mathbf{d}_k^t)$, and the cost vectors, $(\mathbf{r}^t, \mathbf{q}^t)$. In some cases, ξ^t can also contain the lifetimes L_i , the delay factors Δ_i , and the availability factors a_i , depending on the elements deemed uncertain in the future.

Formulation (3.6)–(3.9) is a convenient representation of the stochastic program. At some point, however, this representation might seem a little confusing. For example, it seems that the expectation is taken only on the objective function, while the constraints contain random coefficients (such as \mathbf{d}_j^t in the right-hand side of (3.8)).

Another important aspect is the fact that decisions taken at time t , $(\mathbf{w}^t, \mathbf{y}^t)$, are dependent on the particular realization of the random vector, ξ^t , but cannot depend on future realizations of the random vector. This is clearly a desirable feature for a truly stochastic decision process. If demands in several periods are high, one would expect investors to increase capacity much more than if, for example, demands remain low.

Formally, if the decision variables $(\mathbf{w}^t, \mathbf{y}^t)$ were not dependent on ξ^t , the objective function in (3.6) could be replaced by

$$\sum_t \sum_i \left(E_{\xi} \mathbf{r}_i^t w_i^t + \sum_j E_{\xi} \mathbf{q}_i^t \tau_j^t y_{ij}^t \right) = \sum_t \sum_i \left(\bar{\mathbf{r}}_i^t \cdot w_i^t + \sum_j (\bar{q}_i \tau_j) y_{ij}^t \right), \quad (3.10)$$

where $\bar{\mathbf{r}}_i^t = E_{\xi} \mathbf{r}_i^t$ and $\bar{q}_i \tau_j = E_{\xi} (\mathbf{q}_i^t \tau_j^t)$, making problem (3.6) to (3.9) deterministic. In the next section, we will make the dependence of the decision variables on the random vector explicit.

The formulation given earlier is convenient in its allowing for both continuous and discrete random variables. Theoretical properties such as continuity and convexity can be derived for both types of variables. Solution procedures, on the other hand, strongly differ.

Problem (3.6) to (3.9) is a multistage stochastic linear program with several random variables that actually has an additional property, called *block separable recourse*. This property stems from a separation that can be made between the aggregate-level decisions, $(\mathbf{x}^t, \mathbf{w}^t)$, and the detailed-level decisions, \mathbf{y}^t .

We will formally define block separability in Chapter 3, but we can make an observation about its effect here. Suppose future demands are always independent of the past. In this case, the decision on capacity to install in the future at some t only depends on available capacity and does not depend on the outcomes up to time t . The same \mathbf{x}^t must then be optimal for any realization of ξ . The only remaining stochastic decision is in the operation-level vector, \mathbf{y}^t , which now depends separately on each period's capacity. The overall result is that a multiperiod problem now becomes a much less complex two-period problem.

As a simple example, consider the following problem that appears in Louveaux and Smeers [1988]. In this case, the resulting two period model has three operating modes, $n = 4$ technologies, $\Delta_i = 1$ period of construction delay, full availabilities, $a \equiv 1$, and no existing equipment, $g \equiv 0$. The only random variable is $\mathbf{d}_1 = \xi$. The other demands are $d_2 = 3$ and $d_3 = 2$. The investment costs are $r^1 = (10, 7, 16, 6)^T$ with production costs $q^2 = (4, 4.5, 3.2, 5.5)^T$ and load durations $\tau^2 = (10, 6, 1)^T$. We also add a budget constraint to keep all investment below 120. The resulting two-period stochastic program is:

$$\begin{aligned} \min & 10x_1^1 + 7x_2^1 + 16x_3^1 + 6x_4^1 + E_{\xi} \left[\sum_{j=1}^3 \tau_j^2 (4y_{1j}^2 + 4.5y_{2j}^2 \right. \\ & \quad \left. + 3.2y_{3j}^2 + 5.5y_{4j}^2) \right] \\ \text{s. t. } & 10x_1^1 + 7x_2^1 + 16x_3^1 + 6x_4^1 \leq 120, \\ & -x_i^1 + \sum_{j=1}^3 y_{ij}^2 \leq 0, \quad i = 1, \dots, 4, \\ & \sum_{i=1}^y y_{i1}^2 = \xi, \\ & \sum_{i=1}^y y_{ij}^2 = d_j^2, \quad j = 2, 3, \\ & x_1^1 \geq 0, \quad x_2^1 \geq 0, \quad x_3^1 \geq 0, \quad x_4^1 \geq 0, \\ & y_{ij}^2 \geq 0, \quad i = 1, \dots, 4, \quad j = 1, 2, 3. \end{aligned} \tag{3.11}$$

Assuming that ξ takes on the values 3, 5, and 7 with probabilities 0.3, 0.4, and 0.3, respectively, an optimal stochastic programming solution to (3.11) includes $x^{1*} = (2.67, 4.00, 3.33, 2.00)^T$ with an optimal objective value of 381.85. We can again consider the expected value solution, which would substitute $\xi \equiv 5$ in (3.11). An optimal solution here (again not unique) is $\bar{x}^1 = (0.00, 3.00, 5.00, 2.00)^T$. The objective value, if this single event occurs, is 365. However, if we use this solution in the stochastic problem, then with probability 0.3, demand cannot be met. This would yield an infinite value of the stochastic solution.

Infinite values probably do not make sense in practice because an action can be taken somehow to avoid total system collapse. The power company could buy from neighboring utilities, for example, but the cost would be much higher than

any company operating cost. An alternative technology (internal or external to the company) that is always available at high cost is called a *backstop* technology. If we assume, for example, in problem (3.11) that some other technology is always available, without any required investment costs at a unit operating cost of 100, then the expected value solution would be feasible and have an expected stochastic program value of 427.82. In this case, the value of the stochastic solution becomes $427.82 - 381.85 = 45.97$.

In many power problems, focus is on the reliability of the system or the system's ability to meet demand. This reliability is often described as expressing a minimum probability for meeting demand using the non-backstop technologies. If these technologies are $1, \dots, n-1$, then the reliability restriction (in the two-period situation where capacity decisions need not be random) is:

$$P\left[\sum_{i=1}^{n-1} a_i(g_i^t + w_i^t) \geq \sum_{j=1}^m \mathbf{d}_j^t\right] \geq \alpha, \quad \forall t, \quad (3.12)$$

where $0 < \alpha \leq 1$. Inequality (3.12) is called a *chance* or *probabilistic constraint* in stochastic programming. In production problems, these constraints are often called *fill rate* or *service rate constraints*. They place restrictions on decisions so that constraint violations are not too frequent. Hence, we would often have α quite close to 1.

If the only probabilistic constraints are of the form in (3.12), then we simply want the cumulative available capacity at time t to be at least the α quantile of the cumulative demand in all modes at time t . We then obtain a *deterministic equivalent* constraint to (3.12) of the following form:

$$\sum_{i=1}^{n-1} a_i(g_i^t + w_i^t) \geq (F^t)^{-1}(\alpha), \quad \forall t, \quad (3.13)$$

where F^t is the (assumed continuous) distribution function of $\sum_{j=1}^m \mathbf{d}_j^t$ and $F^{-1}(\alpha)$ is the α -quantile of F . Constraints of the form in (3.13) can then be added to (3.6) to (3.9) or, indeed, to the deterministic problem in (3.2) to (3.5), where expected values replace the random variables.

By adding these chance constraint equivalents, many of the problems of deterministic formulations can be avoided. For example, if we choose $\alpha = 0.7$ for the problem in (3.11), then adding a constraint of the form in (3.13) would not change the deterministic expected value solution. However, we would get a different result if we set $\alpha = 1.0$. In this case, constraint (3.13) for the given data becomes simply:

$$\sum_{i=1}^4 w_i^1 \geq 12. \quad (3.14)$$

Adding (3.14) to the expected value problem results in an optimal solution with $w^{1*} = (0.833, 3.00, 4.17, 4.00)^T$. The expected value of using this solution in the stochastic program is 383.99, or only 2.14 more than the optimal value in (3.11).

In general, probabilistic constraints are represented by deterministic equivalents and are often included in stochastic programs. We discuss some of the theory of these constraints in Chapter 3. Our emphasis in this book is, however, on optimizing the expected value of continuous utility functions, such as the costs in this capacity expansion problem. We, therefore, concentrate on recourse problems and assume that probabilistic constraints are represented by deterministic equivalents within our formulations.

This problem illustrates a multistage decision problem and the addition of probabilistic constraints. The structure of the problem, however, allows for a two-stage equivalent problem. In this way, the capacity expansion problem provides a bridge between the two-stage example of Section 1.1 and the multistage problem of Section 1.2.

This problem also has a natural interpretation with discrete decision variables. For most producing units, only a limited number of possible sizes exists. Typical sizes for high-temperature nuclear reactors would be 1000 MW and 1300 MW, so that capacity decisions could only be taken as integer multiples of these values.

Exercises

1. The detailed-level decisions can be found quite easily according to an *order of merit rule*. In this case, one begins with Mode 1 and uses the least expensive equipment until its capacity is exhausted or demand is satisfied. One continues to exhaust capacity or satisfy demand in order of increasing unit operating cost and mode. Show that this procedure is indeed optimal for determining the y_{ij}^t values.
2. Prove that, in the case of no serial correlation (ξ^t and ξ^{t+1} stochastically independent), an optimal solution has the same value for w^t and x^t for all ξ . Give an example where this does not occur with serial correlation.
3. For the example in (3.11), suppose we add a reliability constraint of the form in (3.14) to the expected value problem, but we use a right-hand side of 11 instead of 12. What is the stochastic program expected value of this solution?

1.4 Design for Manufacturing Quality

This section illustrates a common engineering problem that we model as a stochastic program. The problem demonstrates nonlinear functions in stochastic programming and provides further evidence of the importance of the stochastic solution.

Consider a designer deciding various product specifications to achieve some measure of product cost and performance. The specifications may not, however, completely determine the characteristics of each manufactured product. Key characteristics of the product are often random. For example, every item includes variations

due to machining or other processing. Each consumer also does not use the product in the same way. Cost and performance characteristics thus become random variables.

Deterministic methods may yield costly results that are only discovered after production has begun. From this experience, designing for quality and consideration of variable outcomes has become an increasingly important aspect of modern manufacturing (see, for example, Taguchi et al. [1989]). In industry, the methods of Taguchi have been widely used (see also Taguchi [1986]). Taguchi methods can, in fact, be seen as examples of stochastic programming, although they are often not described this way.

In this section, we wish to give a small example of the uses of stochastic programming in manufacturing design and to show how the general stochastic programming approach can be applied. We note that we base our analysis on actual performance measures, whereas the Taguchi methods generally attach surrogate costs to deviations from nominal parameter values.

We consider the design of a simple axle assembly for a bicycle cart. The axle has the general appearance in Figure 7.

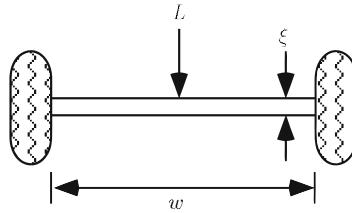


Fig. 7 An axle of length w and diameter ξ with a central load L .

The designer must determine the specified length w and design diameter ξ of the axle. We use inches to measure these quantities and assume that other dimensions are fixed. Together, these quantities determine the performance characteristics of the product. The goal is to determine a combination that gives the greatest expected profit.

The initial costs are for manufacturing the components. We assume that a single process is used for the two components. No alternative technologies are available, although, in practice, several processes might be available. When the axle is produced, the actual dimensions are not exactly those that are specified. For this example, we suppose that the length w can be produced exactly but that the diameter ξ is a random variable, $\xi(x)$, that depends on a specified mean value, x , that represents, for example, the setting on a machine. We assume a triangular distribution for $\xi(x)$ on $[0.9x, 1.1x]$. This distribution has a density,

$$f_x(\xi) = \begin{cases} (100/x^2)(\xi - 0.9x) & \text{if } 0.9x \leq \xi < x, \\ (100/x^2)(1.1x - \xi) & \text{if } x \leq \xi \leq 1.1x, \\ 0 & \text{otherwise.} \end{cases} \quad (4.1)$$

The decision is then to determine w and x , subject to certain limits, $w \leq w^{\max}$ and $x \leq x^{\max}$, in order to maximize expected profits. For revenues, we assume that if the product is profitable, we sell as many as we can produce. This amount is fixed by labor and equipment regardless of the size of the axle. We, therefore, only wish to determine the maximum selling price that generates enough demand for all production. From marketing studies, we determine that this maximum selling price depends on the length and is expressed as

$$r(1 - e^{-0.1w}), \quad (4.2)$$

where r is the maximum possible for any such product.

Our production costs for labor and equipment are assumed fixed, so only material cost is variable. This cost is proportional to the mean values of the specified dimensions because material is acquired before the actual machining process. Suppose c is the cost of a single axle material unit. The total manufacturing cost for an item is then

$$c \left(\frac{w\pi x^2}{4} \right). \quad (4.3)$$

In this simplified model, we assume that no quantity discounts apply in the production process.

Other costs are incurred after the product is made due to warranty claims and potential future sales losses from product defects. These costs are often called *quality losses*. In stochastic programming terms, these are the recourse costs. Here, the product may perform poorly if the axle becomes bent or broken due to excess stress or deflection. The stress limit, assuming a steel axle and 100-pound maximum central load, is

$$\frac{w}{\xi^3} \leq 39.27. \quad (4.4)$$

For deflection, we use a maximum 2000-rpm speed (equivalent to a speed of 60 km/hour for a typical 15-centimeter wheel) to obtain:

$$\frac{w^3}{\xi^4} \leq 63,169. \quad (4.5)$$

When either of these constraints is violated, the axle deforms. The expected cost for not meeting these constraints is assumed proportional to the square of the violation. We express it as

$$Q(w, x, \xi) = \min_y \{ qy^2 \text{ s. t. } \frac{w}{\xi^3} - y \leq 39.27, \frac{w^3}{\xi^4} - 300y \leq 63,169 \}, \quad (4.6)$$

where y is, therefore, the maximum of stress violation and (to maintain similar units) $\frac{1}{300}$ of the deflection violation.

The expected cost, given w and x , is

$$\mathcal{Q}(w,x) = \int_{\xi} Q(w,x,\xi) f_x(\xi) d\xi , \quad (4.7)$$

which can be written as:

$$\begin{aligned} \mathcal{Q}(w,x) &= q \int_{.9x}^{1.1x} (100/x^2) \min\{\xi - .9x, 1.1x - \xi\} \\ &\quad [\max\{0, \left(\frac{w}{\xi^3}\right) - 39.27, \left(\frac{w^3}{300\xi^4}\right) - 210.56\}]^2 d\xi . \end{aligned} \quad (4.8)$$

The overall problem is to find:

$$\begin{aligned} \max & \text{ (total revenue per item} - \text{manufacturing cost per item} \\ & - \text{expected future cost per item}). \end{aligned} \quad (4.9)$$

Mathematically, we write this as:

$$\begin{aligned} \max z(w,x) &= r(1 - e^{-0.1w}) - c \left(\frac{w\pi x^2}{4} \right) - \mathcal{Q}(w,x) \\ \text{s. t. } & 0 \leq w \leq w^{\max}, 0 \leq x \leq x^{\max}. \end{aligned} \quad (4.10)$$

In stochastic programming terms, this formulation gives the deterministic equivalent problem to the stochastic program for minimizing the current value for the design decision plus future reactions to deviations in the axle diameter. Standard optimization procedures can be used to solve this problem. Assuming maximum values of $w^{\max} = 36$, $x^{\max} = 1.25$, a maximum sales price of \$10 ($r = 10$), a material cost of \$0.025 per cubic inch ($c = .025$), and a unit penalty $q = 1$, an optimal solution is found at $w^* = 33.6$, $x^* = 1.038$, and $z^* = z(w^*, x^*) = 8.94$. The graphs of z as a function of w for $x = x^*$ and as a function of x for $w = w^*$ appear in Figures 8 and 9. In this solution, the stress constraint is only violated when $.9x = 0.934 \leq \xi \leq 0.949 = (w/39.27)^{1/3}$.

We again consider the expected value problem where random variables are replaced with their means to obtain a deterministic problem. For this problem, we would obtain:

$$\begin{aligned} \max z(w,x,\bar{\xi}) &= r(1 - e^{-0.1w}) - c \left(\frac{w\pi x^2}{4} \right) \\ &\quad - q[\max\{0, \left(\frac{w}{\bar{x}^3}\right) - 39.27, \left(\frac{w^3}{300\bar{x}^4}\right) - 210.56\}]^2 \\ \text{s. t. } & 0 \leq w \leq w^{\max}, 0 \leq x \leq x^{\max}. \end{aligned} \quad (4.11)$$

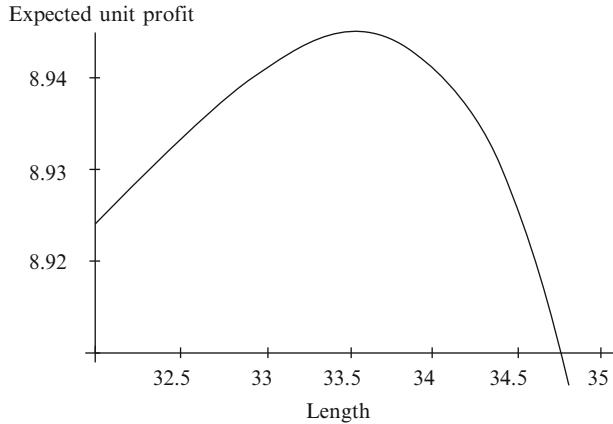


Fig. 8 The expected unit profit as a function of length with a diameter of 1.038 inches.

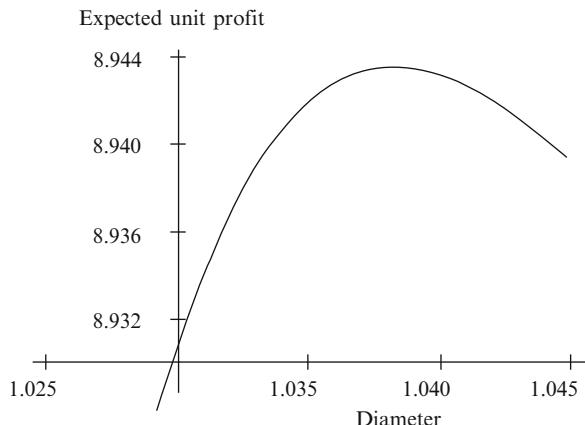


Fig. 9 The expected unit profit as a function of diameter with a length of 33.6 inches.

Using the same data as earlier, an optimal solution to (4.11) is $\bar{w}(\bar{\xi}) = 35.0719$, $\bar{x}(\bar{\xi}) = 0.963$, and $z(\bar{w}, \bar{x}, \bar{\xi}) = 9.07$.

At first glance, it appears that this solution obtains a better expected profit than the stochastic problem solution. However, as we shall see in Chapter 8 on approximations, this deterministic problem paints an overly optimistic picture of the actual situation. The deterministic objective is (in the case of concave maximization) always an *overestimate* of the actual expected profit. In this case, the true *expected* value of the deterministic solution is $z(\bar{w}, \bar{x}) = -26.8$. This problem then has a value of the stochastic solution equal to the difference between the expected value of the stochastic solution and the expected value of the deterministic solution,

$z^* - z(\bar{w}, \bar{x}) = 35.7$. In other words, solving the stochastic program results in a significant profit compared to a considerable loss associated with solving the deterministic problem.

This problem is another example of how stochastic programming can be used. The problem has nonlinear functions and a simple recourse structure. We will discuss further computational methods for problems of this type in Chapter 5. In other problems, decisions may also be taken after the observation of the outcome. For example, we could inspect and then decide whether to sell the product (Exercise 3). This often leads to tolerance settings and is the focus of much of quality control.

The general stochastic program provides a framework for uniting design and quality control. Many loss functions can be used to measure performance degradation to help improve designs in their initial stages. These functions may include the stress and performance penalties described earlier, the Taguchi-type quadratic loss, or methods based on reliability characterizations.

Most traditional approaches assume some form for the distribution as we have done here. This situation rarely matches practice, however. Approximations can nevertheless be used that obtain bounds on the actual solution value so that robust decisions may be made without complete distributional information. This topic will be discussed further in Chapter 8.

Exercises

1. For the example given, what is the probability of exceeding the stress constraint for an axle designed according to the stochastic program optimal specifications?
2. Again, for the example given, what is the probability of exceeding the stress constraint for an axle designed according to the deterministic program's (4.11) optimal specifications?
3. Suppose that every axle can be tested before being shipped at a cost of s per test. The test completely determines the dimensions of the product and thus informs the producer of the risk of failure. Formulate the new problem with testing.

1.5 A Routing Example

a. Presentation

Consider the following simplified vehicle routing problem. A vehicle has to visit four clients (A, B, C, D) in a route starting and ending at a depot (or at the “home sweet home” of the traveling salesperson). One single vehicle of capacity 10 is available. There is no limit on the travel time, so that the vehicle can make consecutive legs if needed.

It is easy to represent a routing problem on a graph (see Figure 10.). A graph $G = (V, E)$ consists of a set V of vertices (or nodes) and a set E of edges (or arcs). Here, the nodes correspond to the set of clients plus the depot $V = \{0, A, B, C, D\}$ where 0 is the depot. Arc (i, j) corresponds to traveling from node i to node j . Arcs may be traveled in either direction. We assume that the vehicle can travel from any point (client or depot) to another. This is equivalent to saying that the graph is complete.

The demands of clients A , B and D are known and equal to 2. Demand of client C is random. To put things to the extreme, assume that the demand of C is either 1 or 7 with equal probability $\frac{1}{2}$. (As we will see later, the example also works with less extreme situations, like a demand of 3 and 5 with equal probability. Direct calculation of all cases is easier here as there are more infeasible cases). All demands must be served. To make things clear, we assume in the sequel that demand is collected at the client. All results and terminologies are easily adapted if demand is delivered. The case of simultaneous pick-ups and deliveries is more involved.

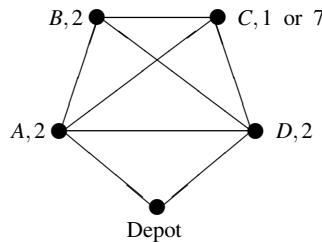


Fig. 10 Graph representation of the vehicle routing problem.

The distances between any two points are given under the form of a symmetrical matrix $C = (c_{ij})$, where c_{ij} is the distance between i and j . Data are in Table 7.

Table 7 Distance matrix.

	0	A	B	C	D
0	-	2	4	4	1
A	2	-	3	4	2
B	4	3	-	1	3
C	4	4	1	-	3
D	1	2	3	3	-

The distance matrix is symmetrical, which means that the distance between two points is the same when traveling in either direction. Distance matrices usually

satisfy the so-called *triangle inequality*:

$$c_{ij} \leq c_{ik} + c_{kj} \quad \forall i, j, k. \quad (5.1)$$

The triangle inequality simply means that it is shorter (or at least not longer) to go directly from i to j than through an intermediate node k . The distance matrix in Table 7 satisfies the triangle inequality, but not always strictly. As an example, the distance between A and C is equal to the distance between A and B plus that between B and C . This is due to using small integer data.

The problem of finding the shortest route to visit all clients starting and ending at the depot is known as the TSP (traveling salesperson problem). The optimal TSP route is $(0, A, B, C, D, 0)$ of length 10.

This is checked by using a TSP solver. This can also be checked by brute force calculation of all routes. For a problem with n clients, there are $n!$ routes. Indeed, starting from the depot, there are n possible clients to be visited first. When the first client is fixed, there remain $(n - 1)$ clients to be visited next and so on. By symmetry, only half of the $n!$ routes have to be checked. As an example, $(0, D, C, B, A, 0)$ has the same length as $(0, A, B, C, D, 0)$. Here, 12 routes have to be checked. Alternatively, you may trust the authors.

Finding the shortest distance or TSP route is not enough here: the vehicle has a limited capacity of 10 and the demand at C is random. The treatment of the uncertainty depends on the moment when the information becomes available.

b. Wait-and-see solutions

A first case is when the level of the demand is known before starting the route. This could be the case, for instance, if the delivered product is part of a just-in-time production process. If the process works in batches, the number of batches required in C may be 1 or 7, depending on the production process. But the number of batches may then be adequately forecasted.

Alternatively, the products may be wastes generated during the production process. The amount to be collected can be known if an agreement exists with the client or if the client is a subsidiary.

This is known as a situation of *a priori information*. The decision process corresponds to the *wait-and-see* approach. It consists of making the choice of the route after getting the information on the demand level.

The optimal solution in the wait-and-see situation is illustrated in Figure 11.

- Whenever client C requires a single unit to be collected, the vehicle's capacity is large enough to accommodate the demand of the four clients. It is optimal to follow the TSP route of length 10.
- Whenever client C requires 7 units, the total demand of 13 exceeds the vehicle's capacity. The vehicle must travel two successive routes. The combination of

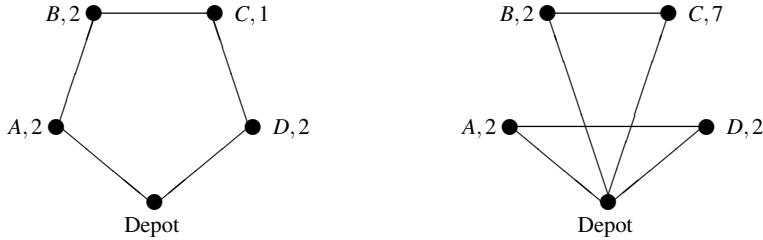


Fig. 11 Wait-and-see solutions (when demand in C is 1 or 7).

two routes with smallest distance is the sequence $(0, A, D, 0, B, C, 0)$ of total distance 14 .

This can be checked as follows. As the demand of C is 7 and the vehicle capacity is 10 , the part of the route that visits C can either visit C alone or C with one other client.

There are three possibilities in the first case depending on the order of visit of A , B and D , the best one being $(0, A, B, D)$. There are also three possibilities for the second case, depending on the client which belongs to the route visiting C .

As both situations occur half of the time, optimal routes of length 10 and 14 are traveled half time each. It follows that the mean (or expected) distance traveled under the wait-and-see approach is

$$WS = \frac{1}{2} 10 + \frac{1}{2} 14 = 12 .$$

c. Expected value solution

If the demand is not known in advance, it is discovered when arriving at client C . One first attitude is to forget uncertainty. The route is planned in view of the expected demand. As the expected demand of client C is 4 , the vehicle's capacity is large enough to accommodate the demand of the four clients (in fact, the expected demand of C and the known demand of the other clients). It is optimal to follow the TSP route $(0, A, B, C, D, 0)$ of length 10 .

Planning for the expected case is in fact “forgetting” uncertainty. It does not mean uncertainty is absent. To say it in other words, “even if you forget uncertainty, uncertainty will not forget you”.

Demand in C is revealed when arriving in C . It is 1 half of the time and 7 the other half of the time, but in a random fashion. Figure 12 shows what really happens.

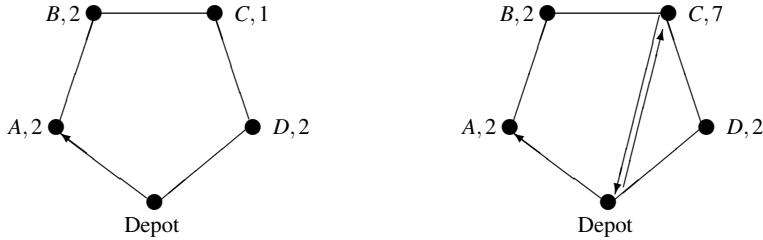


Fig. 12 Effective travel (when demand in C is 1 or 7) if TSP route is planned.

- When the vehicle arrives in C and demand is 1 , it simply proceeds with the planned route. The total demand is 7 and is less than the capacity. The traveled distance is 10. Everything goes well in a beautiful world.
- When the vehicle arrives in C , its load is already 4 . If the demand in C is 7 , the vehicle is unable to collect the total demand. Assuming the goods are divisible, it collects 6 units, then returns to the depot to unload, goes back to C to take the last unit and resumes its trip. The vehicle travels $(0,A,B,C,0,C,D,0)$ for a total length of 18 . In the routing literature, the situation when a vehicle is unable to load a client's demand is known as a *failure*. The extra distance traveled due to this failure is a *return trip* to the depot. The length of 18 is equal to the planned distance 10 of the TSP tour plus the distance 8 of the return trip from C . You may also observe that the same solution is obtained if goods are not divisible.

As both situations occur half of the time, the true cost under uncertainty of the expected value solution is the so-called *expectation of the expected value problem* or

$$EEV = \frac{1}{2} 10 + \frac{1}{2} 18 = 14 .$$

d. Recourse solution

Let us now improve the route choice, in view of the uncertainty at C .

First, observe that it is possible to travel the TSP route $(0,A,B,C,D,0)$ in the opposite direction. The situation is represented on Figure 13. Travelling $(0,D,C,B,A,0)$ implies that

- when the vehicle arrives in C and demand is 1 , it simply proceeds with the planned route. The traveled distance is 10 , as before.
- when the vehicle arrives in C and demand is 7 , the vehicle is able to collect the demand in C . It will not be able to collect the total demand. After collecting demand in C , it returns to the depot, unloads, and then goes to B and A . This

situation is known as a *preventive return*. (It is already known in C that the load in B cannot be collected. It is thus better to return to the depot and resume the tour in B , instead of going to B and making a return trip to the depot.) The vehicle travels $(0, D, C, 0, B, A, 0)$ for a total length of 17.

The true cost under uncertainty of traveling $(0, D, C, B, A, 0)$ is $\frac{1}{2} 10 + \frac{1}{2} 17 = 13.5$.

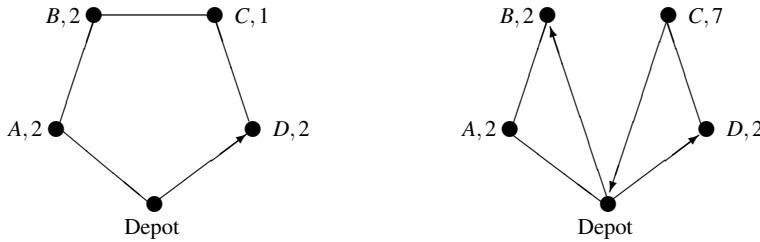


Fig. 13 Effective travel (when demand in C is 1 or 7) if TSP route is planned counterclockwise.

Thus, we have seen that the uncertainty implies that there is a difference between a planned route and the route that is effectively traveled. In the stochastic terminology, deciding on the planned route (or a priori route) is a *first-stage decision*, taken before the random parameters are known. When the uncertainty is revealed, additional or second stage actions are possible. They are called *recourse actions*. In the present example, we have two possible such actions: a return trip to the depot or a preventive return.

After some calculations, it turns out that the optimal solution is to select $(0, C, B, A, D, 0)$ as the planned route. If demand in C is 1, the route is followed with length 11. Otherwise, a preventive return occurs in B . The traveled route is $(0, C, B, 0, A, D, 0)$ with length 14. The optimal solution is represented in Figure 14. The expected length under the optimal recourse policy is

$$RP = \frac{1}{2} 11 + \frac{1}{2} 14 = 12.5.$$

This example illustrates three important aspects of stochastic programming:

- when dealing with uncertainty, it is important to consider what happens before (first-stage) and after (second-stage) the uncertainty is revealed. It is also important to consider a wider variety of decisions (reversing the travel direction in the first-stage, or doing return trips or preventive returns in the second-stage in this example).
- due to uncertainty, a worse solution is often chosen in the favorable case. This happens here. When demand is low, the vehicle travels the planned route $(0, C, B, A, D, 0)$, which is longer than the TSP tour. This may seem stupid: “why

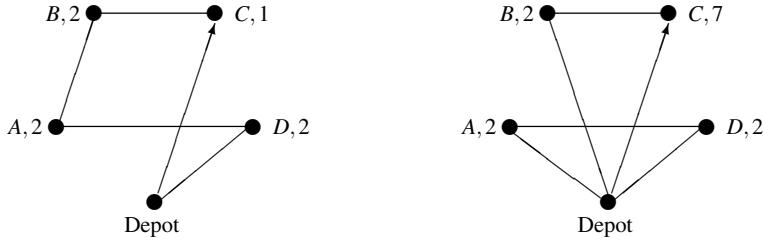


Fig. 14 Effective travel (when demand in C is 1 or 7) if optimal recourse route is planned.

didn't you simply pick up the shortest route?" or lead to some "regret." The reason is simple. By visiting C first, the demand becomes known early in the route and an efficient recourse action (preventive return after B) can be taken when the demand in C is high. This implies indeed some extra cost when the demand in C is low.

- the following relations hold :

$$WS \leq RP \leq EEV .$$

The first relation $WS \leq RP$ simply says that it is always better to get the information in advance. The difference $RP - WS$ is known as the *EVPI*, expected value of perfect information. Here, $EVPI = 0.5$. This is the maximal amount the planner would be ready to pay client C to get the information in advance. The second relation says that it is better to solve the stochastic program than to pretend uncertainty does not exist. The difference $EEV - RP$ is known as the *VSS*, value of stochastic solution. Here, $VSS = 1.5$. It tells says that dealing with uncertainty really matters.

e. Other random variables

The present example may seem a bit extreme, with a demand being either 1 or 7. In fact, it extends to more general random variables. Let ξ denote the random demand in C . We assume ξ has an expectation of 4 (as above). We also assume that the probability of a negative demand is negligible and, similarly, that the probability of ξ exceeding 8 is negligible.

Denote by $p_f = P(\xi > 4)$, where the index f is a mnemonic to recall that a failure will occur if the expected value solution is chosen. Then the following relations hold:

$$WS = (1 - p_f)10 + p_f14 ,$$

$$\begin{aligned}EEV &= (1 - p_f)10 + p_f 18, \\RP &= (1 - p_f)11 + p_f 14.\end{aligned}$$

In the wait-and-see case, the TSP route of length 10 is optimal when demand is less than or equal to 4 and the sequence $(0, A, D, 0, B, C, 0)$ with length 14 otherwise. In the EEV , a distinction is made between no failure (length 10) or a failure with a return trip (length 18). Finally, in the RP , the route is either $(0, C, B, A, D, 0)$ with length 11 when demand is less or equal to 4 or $(0, C, B, 0, A, D, 0)$ with length 14 otherwise.

Now, consider that demand in C follows a normal distribution with expectation 4 and a variance such that $P(\xi < 0) \cong 0$. Symmetry implies $P(\xi > 8) \cong 0$. Symmetry also implies $p_f = \frac{1}{2}$. Thus, all results obtained in the above discrete case are also obtained in the same manner for a normal distribution. The same is true for any continuous uniform distribution of the type $\xi \sim U[4 - a, 4 + a]$, with $0 < a \leq 4$.

The table of the Poisson(4) distribution shows that $p_f = 0.371$. However, there exists a nonzero probability of the demand exceeding 8. We may denote this probability as $p_e = P(\xi > 8) = 0.0214$. If demand exceeds 8, the recourse solution must be adapted as traveling $(0, B, C, 0)$ becomes infeasible. A possible solution for the recourse case is to travel $(0, C, B, D, A, 0)$ with length 11 when demand is less or equal to 4, travel $(0, C, B, 0, A, D, 0)$ with length 14 when demand is between 5 and 8 and, finally, travel $(0, C, 0, A, B, D, 0)$ with length 17 otherwise. The corresponding expected cost is:

$$\text{Expected cost} = (1 - p_f)11 + (p_f - p_e)14 + p_e 17.$$

f. Chance-constraints

The chance-constraint approach consists of finding the smallest distance feasible route or sequence of routes. A route or sequence of routes is feasible if the vehicle can collect the total demand with a large probability. A typical large probability is, as usual, 90 or 95%. To make things concrete, we take a 95% requirement. This corresponds to a 5% probability of failure.

In the initial example, demand is 1 or 7 with probability $\frac{1}{2}$. Feasibility with a 95% confidence level implies demand of 7 must always be collected. If not, the confidence of the solution would only be 50%. The chance-constraint solution is the sequence $(0, A, D, 0, B, C, 0)$ of total distance 14, much worse than the recourse solution.

In line with the previous subsection, we now show how to deal with other random variables.

Let ξ be the random variable representing the demand in C . Any route that does not return to the depot has a capacity of 10. The probability that it can cover the demand is equal to $P(6 + \xi \leq 10) = P(\xi \leq 4) = 1 - p_f$. Any route that returns once to the depot consists of two legs, each having a capacity of 10. Feasibility depends

on the leg that visits C (as the other leg has a known demand less than the vehicle capacity).

We can summarize all cases as follows:

- visiting C with the three other clients is feasible with probability $P(\xi \leq 4) = 1 - p_f$. The best such route is the TSP tour $(0, A, B, C, D, 0)$ of length 10.
- visiting C with two other clients is feasible with probability $P(4 + \xi \leq 10) = P(\xi \leq 6)$. The smallest distance corresponding route is the sequence $(0, D, 0, A, B, C, 0)$ of total distance 12.
- if C is visited with one other client, the route is feasible with probability $P(\xi \leq 8)$. The corresponding route with smallest distance is the sequence $(0, A, D, 0, B, C, 0)$ of total distance 14.
- if the leg that visits C does not visit any other client, it is feasible with probability $P(\xi \leq 10)$. The best corresponding route is $(0, C, 0, A, B, D, 0)$ of length 17.

The various solutions have increased lengths but also increased probabilities of being feasible. To find the chance-constraint solution, it suffices to consider each case in turn. The first that has a probability larger than the requested 95% is the chance-constraint solution.

For a Poisson random variable with expectation 4, $p_f = 0.371$ and thus any route that does not return to the depot is infeasible. A route that returns once to the depot and visits C with two other clients has a probability $P(\xi \leq 6) = 0.8893$ to cover the demand and is thus infeasible. A route that returns once to the depot and visits C with at most one other client has a probability $P(\xi \leq 8) = 0.9786$ to cover the demand. The route $(0, A, D, 0, B, C, 0)$ is, as before, the optimal solution for a 95% chance-constraint.

Exercises

1. Consider a continuous uniform distribution of the type $\xi \sim U[4-a, 4+a]$, with $0 < a \leq 4$. Obtain the optimal chance constraint solution as a function of a .
2. Consider the case where the demand in C follows a Normal distribution with expectation 4 and a variance such that $P(\xi < 0) \cong 0$. Obtain the optimal chance constraint solution as a function of σ .

1.6 Other Applications

In this chapter, we discussed a few examples of stochastic programming applications. The examples were chosen because of their frequency in stochastic programming application as well as to illustrate various aspects of stochastic programming models in terms of number of stages, continuous or discrete variables, separable or

nonseparable recourse, probabilistic constraints, and linear or nonlinear constraint and objective functions.

Several other application areas deserve some recognition but were not discussed yet. A particular example is in airline planning. One of the first applications of stochastic programming was a decision on the allocation of aircraft to routes (*fleet assignment*) by Ferguson and Dantzig [1956]. In this problem, penalties were incurred for lost passengers. The problem becomes a simple recourse problem in stochastic programming terms that they solved using a variant of the standard transportation simplex method (see Section 5.7).

Production planning is another major area that was not in our examples. This area also has been the subject of stochastic programming models for many years. The original chance-constrained stochastic programming model of Charnes, Cooper, and Symonds [1958], for example, considered the production of heating oil with constraints on meeting sales and not exceeding capacity. Other examples include the study by Escudero et al. [1993] for IBM procurement policies.

Water resource modeling has also received widespread application. A good example of this area is the paper by Prékopa and Szántai [1976], where they discuss regulation of Lake Balaton's water level and show how stochastic programming could have avoided floods that occurred before such planning methods were available. Approaches to pollution and the environmental area of water resource planning are also common. An example discussion appears in Somlyódy and Wets [1988].

Energy planning has been the focus of many stochastic programming studies. We note in particular Manne's [1974] analysis of the U.S. decision on whether to invest in breeder reactors. The more recent work of Manne and Richels [1992] on buying insurance against the greenhouse effect is also an excellent example of how stochastic programming can model uncertain future situations so that informed public policy decisions may be made.

Stochastic programming has been applied in many other areas. Of particular note is the forestry planning model in Gassmann ([1989]) and the hospital staffing problem in Kao and Queyranne ([1985]). We also include two exercises in stochastic programming in sports. Many other references appear in King's survey (King [1988b]), the volume by Ermoliev and Wets [1988], and the collection edited by Wallace and Ziemba [2005]. Many more applications are open to stochastic programming, especially with the powerful techniques now available. In the remainder of this book, we will explore those methods, their properties, and the general classes of problems they solve.

Exercises

These exercises all contain a stochastic programming problem that can be solved using standard linear, nonlinear and integer programming software. For each problem, you should develop the model, solve the stochastic program, solve the expected value problem, and find the value of the stochastic solution.

1. Northam Airlines is trying to decide how to partition a new plane for its Chicago–Detroit route. The plane can seat 200 economy class passengers. A section can be partitioned off for first class seats but each of these seats takes the space of 2 economy class seats. A business class section can also be included, but each of these seats takes as much space as 1.5 economy class seats. The profit on a first class ticket is, however, three times the profit of an economy ticket. A business class ticket has a profit of two times an economy ticket's profit. Once the plane is partitioned into these seating classes, it cannot be changed. Northam knows, however, that the plane will not always be full in each section. They have decided that three scenarios will occur with about the same frequency: (1) weekday morning and evening traffic, (2) weekend traffic, and (3) weekday midday traffic. Under Scenario 1, they think they can sell as many as 20 first class tickets, 50 business class tickets, and 200 economy tickets. Under Scenario 2, these figures are 10, 25, and 175. Under Scenario 3, they are 5, 10, and 150. You can assume they cannot sell more tickets than seats in each of the sections. (In reality, the company may allow overbooking, but then it faces the problem of passengers with reservations who do not appear for the flight (*no-shows*). The problem of determining how many passengers to accept is part of the field called *yield management* or *revenue management*. For one approach to this problem, see Brumelle and McGill [1993]. This subject is explored further in Exercise 1 of Section 2.7.)
2. Tomatoes Inc. (TI) produces tomato paste, ketchup, and salsa from four resources: labor, tomatoes, sugar, and spices. Each box of the tomato paste requires 0.5 labor hours, 1.0 crate of tomatoes, no sugar, and 0.25 can of spice. A ketchup box requires 0.8 labor hours, 0.5 crate of tomatoes, 0.5 sacks of sugar, and 1.0 can of spice. A salsa box requires 1.0 labor hour, 0.5 crate of tomatoes, 1.0 sack of sugar, and 3.0 cans of spice.

The company is deciding production for the next three periods. It is restricted to using 200 hours of labor, 250 crates of tomatoes, 300 sacks of sugar, and 100 cans of spices in each period at regular rates. The company can, however, pay for additional resources at a cost of 2.0 per labor hour, 0.5 per tomato crate, 1.0 per sugar sack, and 1.0 per spice can. The regular production costs for each product are 1.0 for tomato paste, 1.5 for ketchup, and 2.5 for salsa.

Demand is not known with certainty until after the products are made in each period. TI forecasts that in each period two possibilities are equally likely, corresponding to a good or bad economy. In the good case, 200 boxes of tomato paste, 40 boxes of ketchup, and 20 boxes of salsa can be sold. In the bad case, these values are reduced to 100, 30, and 5, respectively. Any surplus production is stored at costs of 0.5, 0.25, and 0.2 per box for tomato paste, ketchup, and salsa, respectively. TI also considers unmet demand important and assigns costs of 2.0, 3.0, and 6.0 per box for tomato paste, ketchup, and salsa, respectively, for any demand that is not met in each period.

3. The Clear Lake Dam controls the water level in Clear Lake, a well-known resort in Dreamland. The Dam Commission is trying to decide how much water to release in each of the next four months. The Lake is currently 150 mm below flood

stage. The dam is capable of lowering the water level 200 mm each month, but additional precipitation and evaporation affect the dam. The weather near Clear Lake is highly variable. The Dam Commission has divided the months into two two-month blocks of similar weather. The months within each block have the same probabilities for weather, which are assumed independent of one another. In each month of the first block, they assign a probability of 1/2 to having a natural 100-mm increase in water levels and probabilities of 1/4 to having a 50-mm decrease or a 250-mm increase in water levels. All these figures correspond to natural changes in water level without dam releases. In each month of the second block, they assign a probability of 1/2 to having a natural 150-mm increase in water levels and probabilities of 1/4 to having a 50-mm increase or a 350-mm increase in water levels. If a flood occurs, then damage is assessed at \$10,000 per mm above flood level. A water level too low leads to costly importation of water. These costs are \$5000 per mm less than 250 mm below flood stage. The commission first considers an overall goal of minimizing expected costs. They also consider minimizing the probability of violating the maximum and minimum water levels. (This makes the problem a special form of chance-constrained model.) Consider both objectives.

4. The Energy Ministry of a medium-size country is trying to decide on expenditures for new resources that can be used to meet energy demand in the next decade. There are currently two major resources to meet energy demand. These resources are, however, exhaustible. Resource 1 has a cost of 5 per unit of demand met and a total current availability equal to 25 cumulative units of demand. Resource 2 has a cost of 10 per unit of demand met and a total current availability of 10 demand units. An additional resource from outside the country is always available at a cost of 16.7 per unit of demand met.

Some investment is considered in each of Resources 1 and 2 to discover new supplies and build capital. Resource 1 is, however, elusive. A unit of investment in new sources of Resource 1 yields only 0.1 demand unit of Resource 1 with probability 0.5 and yields 1 demand unit with probability 0.5 . For Resource 2, investment is well known. Each unit of investment yields a demand unit equivalent of Resource 2. Cumulative demand in the current decade is projected to be 10 , while demand in the next decade will be 25 .

The ministry wants to minimize expected costs of meeting demands in the current and following decade assuming that the results of Resource 1 investment will only be known when the current decade ends. Next-decade costs are discounted to 60% of their future real values (which should not change).

5. Pacific Pulp and Paper is deciding how to manage their main forest. They have trees at a variety of ages, which we will break into Classes 1 to 4 . Currently, they have 8000 acres in Class 1 , 10,000 acres in Class 2 , 20,000 in Class 3, and 60,000 in Class 4 . Each class corresponds to about 25 years of growth. The company would like to determine how to harvest in each of the next four 25-year periods to maximize expected revenue from the forest. They also foresee the company's continuing after a century, so they place a constraint of having 40,000 acres in Class 4 at the end of the planning horizon.

Each class of timber has a different yield. Class 1 has no yield, Class 2 yields 250 cubic feet per acre, Class 3 yields 510 cubic feet per acre, and Class 4 yields 700 cubic feet per acre. Without fires, the number of acres in Class i (for $i = 2, 3$) in one period is equal to the amount in Class $i - 1$ from the previous period minus the amount harvested from Class $i - 1$ in the previous period. Class 1 at period t consists of the total amount harvested in the previous period $t - 1$, while Class 4 includes all remaining Class 4 land plus the increment from Class 3.

While weather effects do not vary greatly over 25-year periods, fire damage can be quite variable. Assume that in each 25-year block, the probability is $1/3$ that 15% of all timber stands are destroyed and that the probability is $2/3$ that 5% is lost. Suppose that discount rates are completely overcome by increasing timber value so that all harvests in the 100-year period have the same current value. Revenue is then proportional to the total wood yield.

6. A hospital emergency room is trying to plan holiday weekend staffing for a Saturday, Sunday, and Monday. Regular-time nurses can work any two days of the weekend at a rate of \$300 per day. In general, a nurse can handle 10 patients during a shift. The demand is not known, however. If more patients arrive than the capacity of the regular-time nurses, they must work overtime at an average cost of \$50 per patient overload. The Saturday demand also gives a good indicator of Sunday–Monday demand. More nurses can be called in for Sunday–Monday duty after Saturday demand is observed. The cost is \$400 per day, however, in this case. The hospital would like to minimize the expected cost of meeting demand.

Suppose that the following scenarios of 3-day demand are all equally likely: $(100, 90, 20)$, $(100, 110, 120)$, $(100, 100, 110)$, $(90, 100, 110)$, $(90, 80, 110)$, $(90, 90, 100)$, $(80, 90, 100)$, $(80, 70, 100)$, and $(80, 80, 90)$.

7. After winning the pole at Monza, you are trying to determine the quickest way to get through the first right-hand turn, which begins 200 meters from the start and is 30 meters wide. You are through the turn at 100 meters past the beginning of the next stretch (see Figure 15). As in the figure, you will attempt to stay 10 meters inside the barrier on the starting stretch (maintaining this distance from each barrier as accelerate as fast as possible until point d_1). At this distance, you will start braking as hard as possible and take the turn at the current velocity reached at some point d_2 . (Assume a circular turn with radius equal to the square of velocity divided by maximum lateral acceleration.) Obviously, you do not want to go off the course.

The problem is that you can never be exactly sure of the car and track speed until you start braking at point d_1 . At that point, you can tell whether the track is fast, medium, or slow, and you can then determine the point d_2 where you enter the turn. You suppose that the three kinds of track/car combinations are equally likely. If fast, you accelerate at 27 m/sec^2 , decelerate at 45 m/sec^2 , and have a maximum lateral acceleration of 1.8 g ($= 17.5 \text{ m/sec}^2$). For medium, these values are 24, 42, and 16; for slow, the values are 20, 35, and 14. You want to minimize the expected time through this section. You also assume that

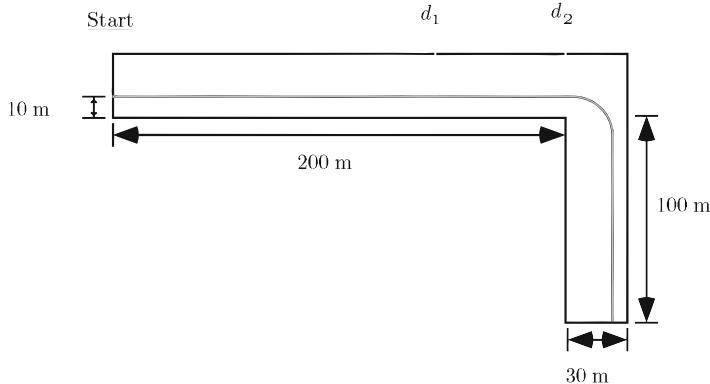


Fig. 15 Opening straight and turn for Problem 7.

if you follow an optimal strategy, other competitors will not throw you out of the race (although you may not be sure of that). After finding the optimal strategy for any feasible position on the second straight-away, find an optimal strategy with a constraint to remain no more than 10 meters from the inside wall after completing the turn and compare the results.

8. In training for the Olympic decathlon, you are trying to choose your takeoff point for the long jump to maximize your expected official jump. Unfortunately, when you aim at a certain spot, you have a 50/50 chance of actually taking off 10 cm beyond that point. If that violates the official takeoff line, you foul and lose that jump opportunity. Assume that you have three chances and that your longest jump counts as your official finish.

You then want to determine your aiming strategy for each jump. Assume that your actual takeoff is independent from jump to jump. Initially you are equally likely to hit a 7.4- or 7.6-meter jump from your actual takeoff point. If you hit a long first jump, then you have a 2/3 chance of another 7.6-meter jump and 1/3 chance of jumping 7.4 meters. The probabilities are reversed if you jumped 7.4 meters the first time. You always seem to hit the third jump the same as the second.

First, find a strategy to maximize the expected official jump. Then, maximize decathlon points from the following Table 8.

Table 8 Decathlon Points for Problem 8.

Distance	Points	Distance	Points
7.30	886	7.46	925
7.31	888	7.47	927
7.32	891	7.48	930
7.33	893	7.49	932
7.34	896	7.50	935
7.35	898	7.51	937
7.36	900	7.52	940
7.37	903	7.53	942
7.38	905	7.54	945
7.39	908	7.55	947
7.40	910	7.56	950
7.41	913	7.57	952
7.42	915	7.58	955
7.43	918	7.59	957
7.44	920	7.60	960
7.45	922	7.61	962

Chapter 2

Uncertainty and Modeling Issues

In the previous chapter, we gave several examples of stochastic programming models. These formulations fit into different categories of stochastic programs in terms of the characteristics of the model. This chapter presents those basic characteristics by describing the fundamentals of any modeling effort and some of the standard forms detailed in later chapters.

Before beginning general model descriptions, however, we first describe the probability concepts that we will assume in the rest of the book. Familiarity with these concepts is essential in understanding the structure of a stochastic program. This presentation is made simple enough to be understood by readers unfamiliar with the field and, thus, leaves aside some questions related to measure theory. Sections 2.2 through 2.7 build on these fundamentals and give the general forms in various categories. Section 2.8 provides a detailed discussion of a modeling exercise. Sections 2.9 and 2.10 give alternative characterizations of stochastic optimization problems and some background on the relationship of stochastic programming to other areas of decision making under uncertainty. Section 2.11 briefly reviews the main optimization concepts used in the book.

2.1 Probability Spaces and Random Variables

Several parameters of a problem can be considered uncertain and are thus represented as random variables. Production and distribution costs typically depend on fuel costs, which are random. Future demands depend on uncertain market conditions. Crop returns depend on uncertain weather conditions.

Uncertainty is represented in terms of random experiments with outcomes denoted by ω . The set of all outcomes is represented by Ω . In a transport and distribution problem, the outcomes range from political conditions in the Middle East to general trade situations, while the random variable of interest may be the fuel cost. The relevant set of outcomes is clearly problem-dependent. Also, it is usually not

very important to be able to define those outcomes accurately because the focus is mainly on their impact on some (random) variables.

The outcomes may be combined into subsets of Ω called *events*. We denote by \mathcal{A} a collection of random events. As an example, if Ω contains the six possible results of the throw of a die, \mathcal{A} also contains combined outcomes such as an odd number, a result smaller than or equal to four, etc. If Ω contains weather conditions for a single day, \mathcal{A} also contains combined events such as “a day without rain,” which might be the union of a sunny day, a partly cloudy day, a cloudy day without showers, etc.

Finally, to each event $A \in \mathcal{A}$ is associated a value $P(A)$, called a *probability*, such that $0 \leq P(A) \leq 1$, $P(\emptyset) = 0$, $P(\Omega) = 1$ and $P(A_1 \cup A_2) = P(A_1) + P(A_2)$ if $A_1 \cap A_2 = \emptyset$. The triplet (Ω, \mathcal{A}, P) is called a *probability space* that must satisfy a number of conditions (see, e.g., Chung [1974]). It is possible to define several random variables associated with a probability space, namely, all variables that are influenced by the random events in \mathcal{A} . If one takes as elements of Ω events ranging from the political situation in the Middle East to the general trade situations, they allow us to describe random variables such as the fuel costs and the interest rates and inflation rates in some Western countries. If the elements of Ω are the weather conditions from April to September, they influence random variables such as the production of corn, the sales of umbrellas and ice cream, or even the exam results of undergraduate students.

In terms of stochastic programming, there exists one situation where the description of random variables is closely related to Ω : in some cases indeed, the elements $\omega \in \Omega$ are used to describe a few *states of the world* or *scenarios*. All random elements then jointly depend on these finitely many scenarios. Such a situation frequently occurs in strategic models where the knowledge of the possible outcomes in the future is obtained through experts’ judgments and only a few scenarios are considered in detail. In many situations, however, it is extremely difficult and pointless to construct Ω and \mathcal{A} ; the knowledge of the random variables is sufficient.

For a particular random variable ξ , we define its cumulative distribution $F_\xi(x) = P(\xi \leq x)$, or more precisely $F_\xi(x) = P(\{\omega | \xi \leq x\})$. Two major cases are then considered. A discrete random variable takes a finite or countable number of different values. It is best described by its probability distribution, which is the list of possible values, ξ^k , $k \in K$, with associated probabilities,

$$f(\xi^k) = P(\xi = \xi^k) \quad \text{s. t. } \sum_{k \in K} f(\xi^k) = 1.$$

Continuous random variables can often be described through a so-called *density* function $f(\xi)$. The probability of ξ being in an interval $[a, b]$ is obtained as

$$P(a \leq \xi \leq b) = \int_a^b f(\xi) d\xi,$$

or equivalently

$$P(a \leq \xi \leq b) = \int_a^b dF(\xi) ,$$

where $F(\cdot)$ is the cumulative distribution as earlier. Contrary to the discrete case, the probability of a single value $P(\xi = a)$ is always zero for a continuous random variable. The distribution $F(\cdot)$ must be such that $\int_{-\infty}^{\infty} dF(\xi) = 1$.

The *expectation* of a random variable is computed as $\mu = \sum_{k \in K} \xi^k f(\xi^k)$ or $\mu = \int_{-\infty}^{\infty} \xi dF(\xi)$ in the discrete and continuous cases, respectively. The *variance* of a random variable is $E[(\xi - \mu)^2]$. The expectation of ξ^r is called the *r th moment* of ξ and is denoted $\bar{\xi}^{(r)} = E[\xi^r]$. A point η is called the α -quantile of ξ if and only if for $0 < \alpha < 1$, $\eta = \min\{x \mid F(x) \geq \alpha\}$. 分位数

The appendix lists the distributions used in the textbook and their expectations and variances. The concepts of probability distribution, density, and expectation easily extend to the case of multiple random variables. Some of the sections in the book use probability measure theory which generalizes these concepts. These sections contain a warning to readers unfamiliar with this field.

2.2 Deterministic Linear Programs

A deterministic linear program consists of finding a solution to

$$\begin{aligned} \min z &= c^T x \\ \text{s. t. } Ax &= b , \\ x &\geq 0 , \end{aligned}$$

where x is an $(n \times 1)$ vector of decisions and c , A and b are known data of sizes $(n \times 1)$, $(m \times n)$, and $(m \times 1)$, respectively. The value $z = c^T x$ corresponds to the objective function, while $\{x \mid Ax = b, x \geq 0\}$ defines the set of feasible solutions. An optimum x^* is a feasible solution such that $c^T x \geq c^T x^*$ for any feasible x . Linear programs typically search for a minimal-cost solution under some requirements (demand) to be met or for a maximum profit solution under limited resources. There exists a wide variety of applications, routinely solved in the industry. As introductory references, we cite Chvátal [1980], Dantzig [1963], and Murty [1983]. We assume the reader is familiar with linear programming and has some knowledge of basic duality theory as in these textbooks. A short review is given in Section 2.11.

2.3 Decisions and Stages

Stochastic linear programs are linear programs in which some problem data may be considered uncertain. *Recourse programs* are those in which some decisions or recourse actions can be taken after uncertainty is disclosed. To be more precise,

追索规划

data uncertainty means that some of the problem data can be represented as random variables. An accurate probabilistic description of the random variables is assumed available, under the form of the probability distributions, densities or, more generally, probability measures. As usual, the particular values the various random variables will take are only known after the random experiment, i.e., the vector $\xi = \xi(\omega)$ is only known after the experiment.

The set of decisions is then divided into two groups:

- A number of decisions have to be taken before the experiment. All these decisions are called *first-stage decisions* and the period when these decisions are taken is called the *first stage*.
- A number of decisions can be taken after the experiment. They are called *second-stage decisions*. The corresponding period is called the *second stage*.

两阶段的决策变量

First-stage decisions are represented by the vector x , while second-stage decisions are represented by the vector y or $y(\omega)$ or even $y(\omega, x)$ if one wishes to stress that second-stage decisions differ as functions of the outcome of the random experiment and of the first-stage decision. The sequence of events and decisions is thus summarized as

$$x \rightarrow \xi(\omega) \rightarrow y(\omega, x).$$

Observe here that the definitions of first and second stages are only related to before and after the random experiment and may in fact contain sequences of decisions and events. In the farming example of Section 1.1, the first stage corresponds to 第一阶段决定种多少, 第二阶段决定卖多少 planting and occurs during the whole spring. Second-stage decisions consist of sales and purchases. Selling extra corn would probably occur very soon after the harvest while buying missing corn will take place as late as possible.

A more extreme example is the following. A traveling salesperson receives one item every day. She visits clients hoping to sell that item. She returns home when a buyer is found or when all clients are visited. Clients buy or do not buy in a random fashion. The decision is not influenced by the previous days' decisions. The salesperson wishes to determine the order in which to visit clients, in such a way as to be at home as early as possible (seems reasonable, does it not?). Time spent involves the traveling time plus some service time at each visited client.

To make things simple, once the sequence of clients to be visited is fixed, it is not changed. Clearly the first stage consists of fixing the sequence and traveling to the first client. The second stage is of variable duration depending on the successive clients buying the item or not. Now, consider the following example. There are two clients with probability of buying 0.3 and 0.8, respectively and traveling times (including service) as in the graph of Figure 1.

Assume the day starts at 8 A.M. If the sequence is (1, 2), the first stage goes from 8 to 9:30. The second stage starts at 9:30 and finishes either at 11 A.M. if Client 1 buys or 4:30 P.M. otherwise. If the sequence is (2, 1), the first stage goes from 8 to 12:00, the second stage starts at 12:00 and finishes either at 4:00 P.M. or at 4:30 P.M. Thus, the first stage if sequence (2, 1) is chosen may sometimes end after the second stage is finished when (1, 2) is chosen if Client 1 buys the item.

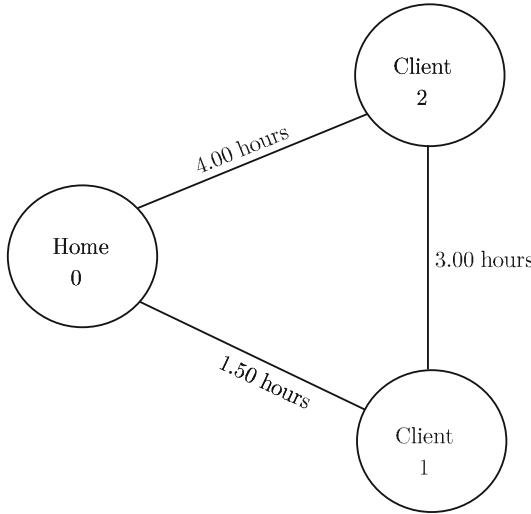


Fig. 1 Traveling salesperson example.

2.4 Two-Stage Program with Fixed Recourse

The classical two-stage stochastic linear program with fixed recourse (originated by Dantzig [1955] and Beale [1955]) is the problem of finding

$$\min z = c^T x + \mathbf{E}_{\xi} [\min q(\omega)^T y(\omega)] \quad (4.1)$$

$$\text{s. t.} \quad Ax = b, \quad (4.2)$$

$$T(\omega)x + Wy(\omega) = h(\omega), \quad (4.3)$$

$$x \geq 0, y(\omega) \geq 0. \quad (4.4)$$

As in the previous section, a distinction is made between the first stage and the second stage. The first-stage decisions are represented by the $n_1 \times 1$ vector x . Corresponding to x are the first-stage vectors and matrices c , b , and A , of sizes $n_1 \times 1$, $m_1 \times 1$, and $m_1 \times n_1$, respectively. In the second stage, a number of random events $\omega \in \Omega$ may realize. For a given realization ω , the second-stage problem data $q(\omega)$, $h(\omega)$ and $T(\omega)$ become known, where $q(\omega)$ is $n_2 \times 1$, $h(\omega)$ is $m_2 \times 1$, and $T(\omega)$ is $m_2 \times n_1$.

y 是第二阶段
决策变量

Each component of q , T , and h is thus a possible random variable. Let $T_{i\cdot}(\omega)$ be the i th row of $T(\omega)$. Piecing together the stochastic components of the second-stage data, we obtain a vector $\xi^T(\omega) = (q(\omega)^T, h(\omega)^T, T_{1\cdot}(\omega), \dots, T_{m_2\cdot}(\omega))$, with potentially up to $N = n_2 + m_2 + (m_2 \times n_1)$ components. As indicated before, a single random event ω (or state of the world) influences several random variables, here, all components of ξ .

Let also $\Xi \subset \Re^N$ be the *support* of ξ , that is, the smallest closed subset in \Re^N such that $P(\Xi) = 1$. As just said, when the random event ω is realized, the second-stage problem data, q , h , and T , become known. Then, the second-stage decision $y(\omega)$ or $(y(\omega), x)$ must be taken. The dependence of y on ω is of a completely different nature from the dependence of q or other parameters on ω . It is not functional but simply indicates that the decisions y are typically not the same under different realizations of ω . They are chosen so that the constraints (4.3) and (4.4) hold *almost surely* (denoted *a.s.*), i.e., for all $\omega \in \Omega$ except perhaps for sets with zero probability. We assume random constraints to hold in this way throughout this book unless a specific probability is given for satisfying constraints.

The objective function of (4.1) contains a deterministic term $c^T x$ and the expectation of the second-stage objective $q(\omega)^T y(\omega)$ taken over all realizations of the random event ω . This second-stage term is the more difficult one because, for each ω , the value $y(\omega)$ is the solution of a linear program. To stress this fact, one sometimes uses the notion of a deterministic equivalent program. For a given realization ω , let

$$Q(x, \xi(\omega)) = \min_y \{q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x, y \geq 0\} \quad (4.5)$$

be the second-stage value function. Then, define the expected second-stage value function

$$\mathcal{Q}(x) = E_{\xi} Q(x, \xi(\omega)) \quad (4.6)$$

and the *deterministic equivalent program* (DEP)

$$\min z = c^T x + \mathcal{Q}(x) \quad (4.7)$$

$$\text{s. t. } Ax = b, \quad (4.8)$$

$$x \geq 0.$$

This representation of a stochastic program clearly illustrates that the major difference from a deterministic formulation is in the second-stage value function. If that function is given, then a stochastic program is just an ordinary nonlinear program.

Formulation (4.1)–(4.4) is the simplest form of a stochastic two-stage program. Extensions are easily modeled. For example, if first-stage or second-stage decisions are to be integers, constraint (4.4) can be replaced by a more general form:

$$x \in X, \quad y(\omega) \in Y,$$

where $X = Z_+^{n_1}$ and $Y = Z_+^{n_2}$. Similarly, nonlinear first-stage and second-stage objectives or constraints can easily be incorporated.

Examples of recourse formulation and interpretations

The definition of first stage versus second stage is not only problem dependent but also context dependent. We illustrate different examples of recourse formulations for one class of problems: *the location problem*.

Let $i = 1, \dots, m$ index clients having demand d_i for a given commodity. The firm can open a facility (such as a plant or a warehouse) in potential sites $j = 1, \dots, n$. Each client can be supplied from an open facility where the commodity is made available (i.e., produced or stored). The problem of the firm is to choose the number of facilities to open, their locations, and market areas to maximize profit or minimize costs.

Let us first present the deterministic version of the so-called simple plant location or uncapacitated facility location problem. Let x_j be a binary variable equal to one if facility j is open and zero otherwise. Let c_j be the fixed cost for opening and operating facility j and let v_j be the variable operating cost of facility j . Let y_{ij} be the fraction of the demand of client i served from facility j and t_{ij} be the unit transportation cost from j to i .

All costs and profits should be taken in conformable units, typically on a yearly equivalent basis. Let r_i denote the unit price charged to client i and $q_{ij} = (r_i - v_j - t_{ij})d_i$ be the total revenue obtained when all of client i 's demand is satisfied from facility j . Then the simple plant location problem or uncapacitated facility location problem (UFLP) reads as follows:

$$\text{UFLP: } \max_{x,y} z(x,y) = - \sum_{j=1}^n c_j x_j + \sum_{i=1}^m \sum_{j=1}^n q_{ij} y_{ij} \quad (4.9)$$

$$\text{s. t. } \sum_{j=1}^n y_{ij} \leq 1, \quad i = 1, \dots, m, \quad (4.10)$$

$$0 \leq y_{ij} \leq x_j, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \quad (4.11)$$

$$x_j \in \{0, 1\}, \quad j = 1, \dots, n. \quad (4.12)$$

Constraints (4.10) ensure that the sum of fractions of clients i 's demand served cannot exceed one. Constraints (4.11) ensure that clients are served only through open plants.

It is customary to present the uncapacitated facility location in a different canonical form that minimizes the sum of the fixed costs of opening facilities and of the transportation costs plus possibly the variable operating costs. (There are several ways to arrive at this canonical representation. One is to assume that unit prices are much larger than unit costs in such a way that demand is always fully satisfied.) This presentation more clearly stresses the link between the deterministic and stochastic cases.

In the UFLP, a trade-off is sought between opening more plants, which results in higher fixed costs and lower transportation costs and opening fewer plants with the opposite effect. Whenever the optimal solution is known, the size of an open

facility is computed as the sum of demands it serves. (In the deterministic case, it is always optimal to have each y_{ij} equal to either zero or one.) The market areas of each facility are then well-defined.

The notation x_j for the location variables and y_{ij} for the distribution variables is common in location theory and is thus not meant here as first stage and second stage, respectively, although in some of the models it is indeed the case.

Several parameters of the problem may be uncertain and may thus have to be represented by random variables. Production and distribution costs may vary over time. Future demands for the product may be uncertain.

As indicated in the introduction of the section, we will now discuss various situations of recourse. It is customary to consider that the location decisions x_j are first-stage decisions because it takes some time to implement decisions such as moving or building a plant or warehouse. The main modeling issue is on the distribution decisions. The firm may have full control on the distribution, for example, when the clients are shops owned by the firm. It may then choose the distribution pattern after conducting some random experiments. In other cases, the firm may have contracts that fix which plants serve which clients, or the firm may wish fixed distribution patterns in view of improved efficiency because drivers would have better knowledge of the regions traveled.

a. Fixed distribution pattern, fixed demand, r_i, v_j, t_{ij} stochastic

Assume the only uncertainties are in production and distribution costs and prices charged to the client. Assume also that the distribution pattern is fixed in advance, x 与 y 都事先确定了 i.e., is considered first stage. The second stage then just serves as a measure of the cost of distribution. We now show that the problem is in fact a deterministic problem in which the total revenue $q_{ij} = (r_i - v_j - t_{ij})d_i$ can be replaced by its expectation. To do this, we formally introduce extra second-stage variables w_{ij} , with the constraint $w_{ij}(\omega) = y_{ij}$ for all ω . We obtain

$$\max - \sum_{j=1}^n c_j x_j + E_\xi \sum_{i=1}^m \sum_{j=1}^n q_{ij}(\omega) w_{ij}(\omega)$$

s.t. (4.10), (4.11), (4.12), and

$$w_{ij}(\omega) = y_{ij}, \quad i = 1, \dots, m, \quad j = 1, \dots, n \quad \forall \omega. \quad (4.13)$$

By (4.13), the second-stage objective function can be replaced by

$$E_\xi \sum_{i=1}^m \sum_{j=1}^n q_{ij}(\omega) y_{ij}$$

or

$$\sum_{i=1}^n \sum_{j=1}^n \mathbb{E}_\xi q_{ij}(\omega) y_{ij},$$

because y_{ij} is fixed and summations and expectation can be interchanged. The problem is thus the deterministic problem

$$\max - \sum_{j=1}^n c_j x_j + \sum_{i=1}^m \sum_{j=1}^n (\mathbb{E}_\xi q_{ij}(\omega)) y_{ij}$$

s.t. (4.10), (4.11), (4.12).

Although there exists uncertainty about the distribution costs and revenues, the only possible action is to plan in view of the expected costs.

b. Fixed distribution pattern, uncertain demand

Assume now that demand is uncertain, but, for some of the reasons cited earlier, the distribution pattern is fixed in the first stage. Depending on the context, the distribution costs and revenues (v_j, t_{ij}, r_i) may or may not be uncertain.

We define y_{ij} = quantity transported from j to i , a quantity no longer defined as a function of the demand d_i , because demand is now stochastic. For simplicity, we assume that a penalty q_i^+ is paid per unit of demand d_i which cannot be satisfied from all quantities transported to i (they might have to be obtained from other sources) and a penalty q_i^- is paid per unit on the products delivered to i in excess of d_i (the cost of inventory, for example). We thus introduce second-stage variables: $w_i^-(\omega)$ = amount of extra products delivered to i in state ω ; $w_i^+(\omega)$ = amount of unsatisfied demand to i in state ω .

The formulation becomes

$$\begin{aligned} \max & - \sum_{j=1}^n c_j x_j + \sum_{i=1}^m \sum_{j=1}^n (\mathbb{E}_\xi (-v_j - t_{ij})) y_{ij} + \mathbb{E}_\xi [- \sum_{i=1}^m q_i^+ w_i^+(\omega) \\ & - \sum_{i=1}^m q_i^- w_i^-(\omega)] + \mathbb{E}_\xi \sum_{i=1}^m r_i d_i(\omega) \end{aligned} \quad (4.14)$$

$$\text{s. t. } \sum_{i=1}^m y_{ij} \leq M x_j, \quad j = 1, \dots, n, \quad (4.15)$$

$$w_i^+(\omega) - w_i^-(\omega) = d_i(\omega) - \sum_{j=1}^n y_{ij}, \quad i = 1, \dots, m, \quad (4.16)$$

$$\begin{aligned} x_j & \in \{0, 1\}, \quad 0 \leq y_{ij}, \quad w_i^+(\omega) \geq 0, \quad w_i^-(\omega) \geq 0, \\ & i = 1, \dots, m, \quad j = 1, \dots, n. \end{aligned} \quad (4.17)$$

This model is a location extension of the transportation model of Williams [1963]. The objective function contains the investment costs for opening plants, the expected

production and distribution costs, the expected penalties for extra or insufficient demands, and the expected revenue. This last term is constant because it is assumed that all demands must be satisfied by either direct delivery or some other means reflected in the penalty for unmet demand. The problem only makes sense if q_i^+ is large enough, for example, larger than $E_\xi(v_j + t_{ij})$ for all j , although weaker conditions may sometimes suffice. Constraint (4.15) guarantees that distribution only occurs from open plants, i.e., plants such that $x_j = 1$. The constant M represents the maximum possible size of a plant.

Observe that here the variables y_{ij} are first-stage variables. Also observe that in the second stage, the constraints (4.16), (4.17) have a very simple form, as $w_i^+(\omega) = \mathbf{d}_i - \sum_{j=1}^n y_{ij}$ if this quantity is non-negative and $w_i^-(\omega) = \sum_{j=1}^n y_{ij} - \mathbf{d}_i$ otherwise. This is an example of a *second stage with simple recourse*.

Also note that in Cases a and b, the size or capacity of plant j is simply obtained as the sum of the quantity transported from j , namely, $\sum_{i=1}^m d_i y_{ij}$ in Case a and $\sum_{i=1}^m y_{ij}$ in Case b.

c. Uncertain demand, variable distribution pattern

We now consider the case where the distribution pattern can be adjusted to the realization of the random event. This might be the case when uncertainty corresponds to long-term scenarios, of which only one is realized. Then the distribution pattern can be adapted to this particular realization. This also implies that the sizes of the plants cannot be defined as the sum of the quantity distributed, because those quantities depend on the random event. We thus define as before:

$$x_j = \begin{cases} 1 & \text{if plant } j \text{ is open,} \\ 0 & \text{otherwise.} \end{cases}$$

We now let y_{ij} depend on ω with $y_{ij}(\omega)$ = fraction of demand $d_i(\omega)$ served from j and define new variables w_j = size (capacity) of plant j , with unit investment cost g_j .

The model now reads

$$\max - \sum_{j=1}^n c_j x_j - \sum_{j=1}^n g_j w_j + E_\xi \max \sum_{i=1}^m \sum_{j=1}^n q_{ij}(\omega) y_{ij}(\omega) \quad (4.18)$$

$$\text{s. t. } x_j \in \{0, 1\}, w_j \geq 0, \quad j = 1, \dots, n, \quad (4.19)$$

$$\sum_{j=1}^n y_{ij}(\omega) \leq 1, \quad i = 1, \dots, m, \quad (4.20)$$

$$\sum_{i=1}^m d_i(\omega) y_{ij}(\omega) \leq w_j, \quad j = 1, \dots, n, \quad (4.21)$$

$$0 \leq y_{ij}(\omega) \leq x_j, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \quad (4.22)$$

where $q_{ij}(\omega) = (r_i - v_j - t_{ij})d_i(\omega)$ now includes the demand $d_i(\omega)$.

Constraint (4.20) indicates that no more than 100% of i 's demand can be served, but that the possibility exists that not all demand is served. Constraint (4.21) imposes that the quantity distributed from plant j does not exceed the capacity w_j decided in the first stage. For the sake of clarity, one could impose a constraint $w_j \leq Mx_j$, but this is implied by (4.21) and (4.22). For a discussion of algorithmic solutions of this problem, see Louveaux and Peeters [1992].

d. Stages versus periods; Two-stage versus multistage

In this section, we highlight again the difference in a stochastic program between *stages* and *periods* of times. Consider the case of a distribution firm that makes its plans for the next 36 months. It may formulate a model such as (4.18)–(4.22). The location of warehouses would be first-stage decisions, while the distribution problem would be second-stage decisions. The duration of the first stage would be something like six months (depending on the type of warehouse) and the second stage would run over the 30 remaining months. Although we may think of a problem over 36 periods, a two-stage model is totally relevant. In this case, the only moment where the number of periods is important is when the precise values of the objective coefficients are computed.

In this example, a multistage model becomes necessary if the distribution firm foresees additional periods where it is ready to change the location of the warehouses. In this example, suppose the firm decides that the opening of new warehouses can be decided after one year. A three-stage model can be constructed. The first stage would consist of decisions on warehouses to be built now. The second stage would consist of the distribution patterns between months 7 and 18 as well as new openings decided in month 12. The third stage would consist of distribution patterns between months 19 and 36.

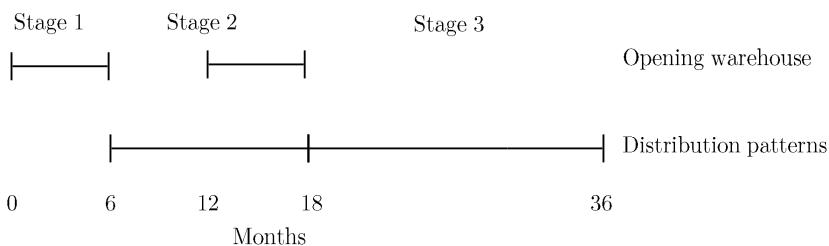


Fig. 2 Three-stage model decisions and times.

Let x^1 and $x^2(\omega_2)$ be the binary vectors representing opening warehouses in stages 1 and 2, respectively. Let $y^2(\omega_2)$ and $y^3(\omega_3)$ be the vectors representing the distribution decisions in stages 2 and 3, respectively, where ω_2 and ω_3 are the states of the world in stages 2 and 3. Assuming each warehouse can only have a fixed size M , the following model can be built:

$$\begin{aligned}
 \max \quad & - \sum_{j=1}^n c_j x_j^1 + E_{\xi_2} \max \left\{ \sum_{i=1}^m \sum_{j=1}^n q_{ij}^2(\omega_2) y_{ij}^2(\omega_2) - \sum_{j=1}^n c_j^2(\omega_2) x_j^2(\omega_2) \right. \\
 & \quad \left. + E_{\xi_3|\xi_2} \max \left[\sum_{i=1}^m \sum_{j=1}^n q_{ij}^3(\omega_3) y_{ij}^3(\omega_3) \right] \right\} \\
 \text{s. t.} \quad & \sum_{j=1}^n y_{ij}^2(\omega_2) \leq 1, \quad i = 1, \dots, m, \\
 & \sum_{i=1}^m d_i(\omega_2) y_{ij}^2(\omega_2) \leq M x_j^1, \quad j = 1, \dots, n, \\
 & \sum_{j=1}^n y_{ij}^3(\omega_3) \leq 1, \quad i = 1, \dots, m, \\
 & \sum_{i=1}^m d_i(\omega_3) y_{ij}^3(\omega_3) \leq M(x_j^1 + x_j^2(\omega_2)), \quad j = 1, \dots, n, \\
 & x_j^1 + x_j^2(\omega_2) \leq 1, \quad j = 1, \dots, n, \\
 & x_j^1, x_j^2(\omega_2) \in \{0, 1\}, \quad j = 1, \dots, n, \\
 & y_{ij}^2(\omega_2), y_{ij}^3(\omega_3) \geq 0, \quad i = 1, \dots, m, \quad j = 1, \dots, n.
 \end{aligned}$$

Multistage programs will be further studied in Section 3.4.

2.5 Random Variables and Risk Aversion

In our view, one can often classify random events and random variables in two major categories. In the first category, we would place uncertainties that recur frequently on a short-term basis. As an example, uncertainty may correspond to daily or weekly demands. This normally leads to a model similar to the one in Section 2.4, Case b (4.b), where allocation cannot be adjusted every time period. It follows that the expectation in the second stage somehow represents a mean over possible values of the random variables, of which many will occur. Thus, the expectation takes into account realizations that might not occur and many realizations that will occur. To fix ideas here, if in Model 4.b the units in the objective function are in a yearly basis and the randomness involves daily or weekly demands, one may expect that the value of the objective of stochastic model will closely match the realized total yearly revenue.

As one interesting example of a real-world application of a location model of this first category, we may recommend the paper by Psarafitis, Tharakkan, and Ceder [1986]. It deals with the optimal location and size of equipment to fight oil spills. Occurrence and sizes of spills are random. The sizes of the spills are represented by a discrete random variable taking three possible values, corresponding to small, medium, or large spills. Sadly enough, spills are sufficiently frequent that the expectation may be considered close enough to the mean cost, as just described. Occurrence of spills at a given site is also random. It is described by a Poisson process. By making the assumption of non-concomitant occurrence of spills, all equipment is made available for each spill, which simplifies the second-stage descriptions compared to (4.14)–(4.17).

As a common example, consider revenue management decisions such as those considered in Problem 1.1 for an airline that must determine reservation controls for hundreds of daily flights. This area has become one of the most widespread applications of analytical methods to determining optimal choices under uncertain conditions (see Talluri and van Ryzin [2005]). Airlines routinely solve thousands of these stochastic programs each month and can reasonably expect to receive close to the expected revenue from their decisions each month (if not each day). Risk aversion has little affect in that case.

In the second category, we would place uncertainties that can be represented as scenarios, of which basically only one or a small number are realized. An example in a similar situation to the airline might be the problem of the organizers of the World Cup championship soccer game, which only occurs once every four years, to choose prices and seat allocations to maximize revenues but also to protect against possible losses. This consideration would also be the case in long-term models where scenarios represent the general trend or path of the variables. As already indicated, this is the spirit in which Model 4.c is built. In the second stage, among all scenarios over which expectation is taken, only one is realized. The objective function with only expected values may then be considered a poor representation of risk aversion, which is typically assumed in decision making (if we exclude gambling).

Starting from the von Neumann and Morgenstern [1944] theory of utility, this field of modeling preferences has been developed by economics. Models such as the mean-variance approach of Markowitz [1959] have been widely used. Other methods have been proposed based on mixes of mean-variance and other approaches (see, e.g, Ben-Tal and Teboulle [1986]). From a theoretical point of view, considering a nonlinear utility function transforms the problems into stochastic nonlinear programs, which can require more computational effort than linear versions. In practice, risk aversion is often captured with a piecewise-linear representation, as in the financial planning example in Section 1.2, to maintain a linear problem structure.

One interesting alternative to nonlinear utility models is to include risk aversion in a linear utility model under the form of a linear constraint, called *downside risk* (Eppen, Martin, and Schrage [1989]). The problem there is to determine the type and level of production capacity at each of several locations. Plants produce various types of cars and may be open, closed, or retooled. The demand for each type of car

in the medium term is random. The decisions about the locations and configurations of plants have to be made before the actual demands are known.

Scenarios are based on pessimistic, neutral, or optimistic realizations of demands. A scenario consists of a sequence of realizations for the next five years. The stochastic model maximizes the present value of expected discounted cash flows. The linear constraint on risk is as follows: the downside risk of a given scenario is the amount by which profit falls below some given target value. It is thus zero for larger profits. The expected downside risk is simply the expectation of the downside risk over all scenarios. The constraint is thus that the expected downside risk must fall below some level.

To give an idea of how this works, consider a two-stage model similar to (4.1)–(4.4) but in terms of profit maximization, by

$$\max z = c^T x + E_{\xi} [\max q^T(\omega) y(\omega)]$$

s.t. (4.2)–(4.4).

Then define the target level g on profit. The downside risk $u(\xi)$ is thus defined by two constraints:

负面风险

$$u(\xi(\omega)) \geq g - q^T(\omega) y(\omega) \quad (5.1)$$

$$u(\xi(\omega)) \geq 0. \quad (5.2)$$

The constraint on expected downside risk is

$$E_{\xi} u(\xi) \leq l, \quad (5.3)$$

where l is some given level. For a problem with a discrete random vector ξ , constraint (5.3) is linear. **Observe that (5.3) is in fact a first-stage constraint as it runs over all scenarios.** It can be used directly in the extensive form. It can also be used indirectly in a sequential manner, by imposing such a constraint only when needed. This can be done in a way similar to the induced constraints for feasibility that we will study in Chapter 5.

2.6 Implicit Representation of the Second Stage

This book is mainly concerned with stochastic programs of the form (4.1)–(4.4), assuming that an adequate and computationally tractable representation of the recourse problem exists. This is not always the case. Two possibilities then exist that still permit some treatment of the problem:

- A closed form expression is available for the expected value function $\mathcal{Q}(x)$.
- For a given first-stage decision x , the expected value function $\mathcal{Q}(x)$ is computable.

情景可以按照风险
悲观、中性、乐观
来划分

These possibilities are described in the following sections.

a. A closed form expression is available for $\mathcal{Q}(x)$

We may illustrate this case by the *stochastic queue median* model (SQM) first proposed by Berman, Larson, and Chiu [1985] from which we take the following in a simplified form. The problem consists of locating an emergency unit (such as an ambulance). When a call arrives, there is a certain probability that the ambulance is already busy handling an earlier demand for ambulance service. In that event, the new service demand is either referred to a backup ambulance service or entered into a queue of other waiting “customers.” Here, the first-stage decision consists of finding a location for the ambulance. The second stage consists of the day-to-day response of the system to the random demands. Assuming a first-in, first-out decision rule, decisions in the second stage are somehow automatic. On the other hand, the quality of response, measured, e.g., by the expected service time, depends on the first-stage decision. Indeed, when responding to a call, an ambulance typically goes to the scene and returns to the home location before responding to the next call. The time when it is unavailable for another call is clearly a function of the home location.

Let λ be the total demand rate, $\lambda \geq 0$. Let p_i be the probability that a demand originates from demand region i , with $\sum_{i=1}^m p_i = 1$. Let also $t(i,x)$ denote the travel time between location x and call i . On-scene service time is omitted for simplicity. Given facility location x , the expected response time is the sum of the mean-in-queue delay $w(x)$ and the expected travel time $\bar{t}(x)$,

$$\mathcal{Q}(x) = w(x) + \bar{t}(x), \quad (6.1)$$

where

$$w(x) = \begin{cases} \frac{\lambda \bar{t}^{(2)}(x)}{2(1-\lambda \bar{t}(x))} & \text{if } \lambda \bar{t}(x) < 1, \\ 0 & \text{otherwise,} \end{cases} \quad (6.2)$$

$$\bar{t}(x) = \sum_{i=1}^m p_i t(i,x), \quad (6.3)$$

and

$$\bar{t}^{(2)}(x) = \sum_{i=1}^m p_i t^2(i,x). \quad (6.4)$$

The global problem is then of the form:

$$\min_{x \in X} \mathcal{Q}(x), \quad (6.5)$$

where the first-stage objective function is usually taken equal to zero and X represents the set of possible locations, which typically consists of a network.

It should be clear that no possibility exists to adequately describe the exact sequence of decisions and events in the so-called second stage and that the expected recourse $\mathcal{Q}(x)$ represents the result of a computation assuming the system is in steady state.

b. For a given x , $\mathcal{Q}(x)$ is computable

The deterministic traveling salesperson problem (TSP) consists of finding a Hamiltonian tour of least cost or distance. Following a Hamiltonian tour means that the traveling salesperson starts from her home location, visits all customers, (say $i = 1, \dots, m$) exactly, and returns to the home location.

Now, assume each customer has a probability p_i of being present. A full optimization that would allow the salesperson to decide the next customer to visit at each step would be a difficult multistage stochastic program. A simpler two-stage model, known as *a priori optimization* is as follows: in the first-stage, an *a priori* Hamiltonian tour is designed. In the second stage, the *a priori* tour is followed by skipping the absent customers. The problem is to find the tour with minimal expected cost (Jaillet [1988]).

The exact representation of such a second-stage recourse problem as a mathematical program with binary decision variables might be possible in theory but would be so cumbersome that it would be of no practical value. On the other hand, the expected length of the tour (and thus $\mathcal{Q}(x)$) is easily computed when the tour (x) is given.

Let c_{ij} be the distance between i and j . Assume for simplicity of notation that the given tour is $\{0, 1, 2, \dots, n, 0\}$ where 0 is the depot.

Define $t(k)$ as the expected length from k till the depot if k is present. Thus we search for $\mathcal{Q}(x) = t(0)$.

Start with $t(n+1) = 0$ and $t(n) = c_{n0}$. Let $p_0 = 1$ and $c_{in+1} = c_{i0}$. Then

$$t(k) = \sum_{r=0}^{n-k} \prod_{j=1}^r (1 - p_{k+j}) p_{k+r+1} (c_{kk+r+1} + t(k+r+1)) \quad \text{for } k = n-1, \dots, 0,$$

where the condensed product is equal to 1 if $r = 0$.

This calculation is a backward recursion: assuming k is present, it considers the next present customer to be $k+r+1$ (and thus $k+1$ to $k+r$ being absent) for all possible successors ($k+1$ to $n+1 := 0$).

2.7 Probabilistic Programming

In probabilistic programming, some of the constraints or the objective are expressed in terms of probabilistic statements about first-stage decisions. The description of second-stage or recourse actions is thus avoided. This is particularly useful when the cost and benefits of second-stage decisions are difficult to assess.

For some probabilistic constraints, it is possible to derive a deterministic linear equivalent. A first example was given in Section 1.3. We now detail two other examples where a deterministic linear equivalent is obtained and one where it is not.

a. Deterministic linear equivalent: a direct case

Consider Exercise 1.6.1. An airline wishes to partition a plane of 200 seats into three categories: first, business, economy. Now, assume the airline wishes a special guarantee for its clients enrolled in its loyalty program. In particular, it wants 98% probability to cover the demand of first-class seats and 95% probability to cover the demand of business class seats (by clients of the loyalty program). First-class passengers are covered if they get a first-class seat. Business class passengers are covered if they get either a business or a first-class seat (upgrade). Assume weekday demands of loyalty-program passengers are normally distributed, say $\xi_F \sim N(16, 16)$ and $\xi_B \sim N(30, 48)$ for first-class and business, respectively. Also assume that the demands for first-class and business class seats are independent.

Let x_1 be the number of first-class seats and x_2 the number of business seats. The probabilistic constraints are simply

$$P(x_1 \geq \xi_F) \geq 0.98, \quad (7.1)$$

$$P(x_1 + x_2 \geq \xi_F + \xi_B) \geq 0.95. \quad (7.2)$$

Given the assumptions on the random variables, these probabilistic constraints can be transformed into a deterministic linear equivalent.

Constraint (7.1) can be written as $F_F(x_1) \geq 0.98$, where $F_F(\cdot)$ denotes the cumulative distribution of ξ_F . Now, the 0.98 quantile of the normal distribution is 2.054. As $\xi_F \sim N(16, 16)$, $F_F(x_1) \geq 0.98$ is the same as $(x_1 - 16)/4 \geq 2.054$ or $x_1 \geq 24.216$. Thus, the probabilistic constraint (7.1) is equivalent to a simple bound.

Similarly, constraint (7.2) can be written as $F_{FB}(x_1 + x_2) \geq 0.95$, where $F_{FB}(\cdot)$ denotes the cumulative distribution of $\xi_F + \xi_B$. By the independence assumption and the properties of the normal distribution, $\xi_F + \xi_B \sim N(46, 64)$. The 0.95 quantile of the standard normal distribution is 1.645. Thus, $F_{FB}(x_1 + x_2) \geq 0.95$ is the same as $(x_1 + x_2 - 46)/8 \geq 1.645$ or $x_1 + x_2 \geq 59.16$.

Thus, the probabilistic constraint (7.2) is equivalent to a linear constraint. We say that (7.2) has a linear deterministic equivalent. This is the desired situation with probabilistic constraints.

b. Deterministic linear equivalent: an indirect case

We now provide an example where finding the deterministic equivalent requires some transformation.

Consider the following covering location problem. Let $j = 1, \dots, n$ be the potential locations with, as usual, $x_j = 1$ if site j is open and 0 otherwise, and c_j the investment cost. Let $i = 1, \dots, m$ be the clients. Client i is served if there exists an open site within distance t_i . The distance between i and j is t_{ij} . Define $N_i = \{j \mid t_{ij} < t_i\}$ as the set of eligible sites for client i . The deterministic covering problem is

$$\min \sum_{j=1}^n c_j x_j \quad (7.3)$$

$$\text{s. t. } \sum_{j \in N_i} x_j \geq 1, \quad i = 1, \dots, m, \quad (7.4)$$

$$x_j \in \{0, 1\}, \quad j = 1, \dots, n. \quad (7.5)$$

Taking again the case of an ambulance service, one site may cover more than one region or demand area. When a call is placed, the emergency units may be busy serving another call. Let q be the probability that no emergency unit is available at site j . For simplicity, assume this probability is the same for every site (see Toregas et al. [1971]). Then, the deterministic covering constraint (7.4) may be replaced by the requirement that P (at least one emergency unit from an open eligible site is available) $\geq \alpha$ where α is some confidence level, typically 90 or 95%. Here, the probability that none of the eligible sites has an available emergency unit is q to the power $\sum_{j \in N_i} x_j$, so that the probabilistic constraint is

$$1 - q^{\sum_{j \in N_i} x_j} \geq \alpha, \quad i = 1, \dots, m \quad (7.6)$$

or

$$q^{\sum_{j \in N_i} x_j} \leq 1 - \alpha.$$

Taking the logarithm on both sides, one obtains

$$\sum_{j \in N_i} x_j \geq b \quad (7.7)$$

with

$$b = \left\lceil \frac{\ln(1 - \alpha)}{\ln q} \right\rceil, \quad (7.8)$$

where $\lceil a \rceil$ denotes the smallest integer greater than or equal to a . Thus, the probabilistic constraint (7.6) has a linear deterministic equivalent (7.7).

c. Deterministic nonlinear equivalent: the case of random constraint coefficients

The diet problem is a classical example of linear programming (discussed in Dantzig [1963] for the case in Stigler [1945]). It consists of selecting a number of foods in order to get the cheapest menus that meet the daily requirements in the main nutrients (energy, protein, vitamins, ...). Consider the data in the introductory example of Chvátal (1980). Polly wants to choose among six foods (oatmeal, chicken, eggs, whole milk, cherry pie and pork with beans). Each food has a given serving size; for instance, a serving of eggs is two large eggs and a serving of pork with beans is 260 grams. Each food has therefore a known content of nutrients. If we take the case of protein, the content is 4, 32, 13, 8, 4 and 14 grams (grams) of proteins, respectively, for the given serving sizes.

Let x_1, \dots, x_6 represent the number of servings of each product per day. As Polly is a girl of 18 years of age, she needs 55 grams of protein per day. The protein constraint reads as follows:

$$4x_1 + 32x_2 + 13x_3 + 8x_4 + 4x_5 + 14x_6 \geq 55.$$

(We omit here the other constraints and the objective function, which are very important to Polly but not central to our discussion.)

The same book later on contains an interesting discussion on the difficulty to get precise reliable RDA (recommended daily allowances) as well as precise nutrient contents per serving (Chvátal [Chapter 11, pp. 182–187]). Let us concentrate on this second aspect. It is indeed very unlikely that every large egg has exactly 6.5 grams of protein, or every serving of 260 grams of pork with beans has exactly 14 grams of protein. This implies that the nutrient content of each serving is in fact a random variable. Let a_1, \dots, a_6 be the random content in proteins for the six products. The probabilistic constraint reads as follows:

$$P(a_1 x_1 + a_2 x_2 + a_3 x_3 + a_4 x_4 + a_5 x_5 + a_6 x_6 \geq 55) \geq \alpha. \quad (7.9)$$

Let us now assume the contents of the products are normally distributed, say $a_i \sim N(\mu_i, \sigma_i^2)$, $i = 1, \dots, 6$. We can clearly assume independence between the six products. Then $a_1 x_1 + a_2 x_2 + a_3 x_3 + a_4 x_4 + a_5 x_5 + a_6 x_6 \sim N(\mu, \sigma^2)$ with $\mu = \mu_1 x_1 + \mu_2 x_2 + \mu_3 x_3 + \mu_4 x_4 + \mu_5 x_5 + \mu_6 x_6$ and $\sigma^2 = \sigma_1^2 x_1^2 + \sigma_2^2 x_2^2 + \sigma_3^2 x_3^2 + \sigma_4^2 x_4^2 + \sigma_5^2 x_5^2 + \sigma_6^2 x_6^2$.

Classical probabilistic analysis of the normal distribution implies that (7.9) is equivalent to

$$(55 - \mu)/\sigma \leq z_{1-\alpha}$$

with $z_{1-\alpha}$ the $(1 - \alpha)$ -quantile of the normal distribution. Taking $\alpha = 0.98$, the constraint reads $(55 - \mu)/\sigma \leq -2.054$ or $\mu \geq 55 + 2.054 \cdot \sigma$. As $\sigma^2 = \sigma_1^2 x_1^2 + \sigma_2^2 x_2^2 + \sigma_3^2 x_3^2 + \sigma_4^2 x_4^2 + \sigma_5^2 x_5^2 + \sigma_6^2 x_6^2$, this constraint is non-linear and convex.

2.8 Modeling Exercise

In this section, we propose a modeling exercise and comment on a number of possible answers.

a. Presentation

Consider a production or assembly problem. It consists of producing two products, say A and B . They are obtained by assembling two components, say $C1$ and $C2$, in fixed quantities. The following table shows the components usage for the two products:

Components usage	A	B
$C1$	6	10
$C2$	8	5

Components are produced within the plant. Material (and / or operating) costs for $C1$ and $C2$ are 0.4 and 1.2, respectively. The level of production, or capacity, is related to the work-force and the equipment. Each unit of capacity costs 150 and 180 and can produce batches of 60 and 90 components, respectively for $C1$ and $C2$. Current capacity level is (40,20) batches and cannot be decreased. The total number of batches must not exceed 120. An integer number of batches is not requested here.

In the deterministic case, the demands and unit selling prices are certain and are as follows:

	A	B
Demand	500	200
Unit selling price	50	60

Unmet demand results in lost sales. This does not imply any additional penalty.

1. Select adequate units for each data. Formulate and solve the deterministic problem.

Then, consider a number of stochastic variants. For the sake of comparison, in all cases, the random variables have expectations which are the corresponding deterministic values.

2. Stochastic prices (known demand).

The selling prices of A and B are described by a random vector, say $\zeta^T = (\zeta_1, \zeta_2)$. The rest of the data is unchanged. Formulate a recourse model in the following cases:

- (a) ζ^T takes on the values $(54, 56)$, $(50, 60)$, and $(46, 64)$ with probability 0.3, 0.4 and 0.3 respectively.
- (b) ζ_1 takes on the values $(46, 50, 54)$ with probability 0.3, 0.4 and 0.3; ζ_2 takes on the values $(56, 60, 64)$ with probability 0.3, 0.4 and 0.3; ζ_1 and ζ_2 are independent.
- (c) ζ_1 has a continuous uniform distribution in the range $[46, 54]$; ζ_2 has a continuous uniform distribution in the range $[56, 64]$; ζ_1 and ζ_2 are independent.
- (d) ζ^T takes on the values $(70, 50)$, $(50, 60)$, $(30, 70)$ with probability 0.3, 0.4 and 0.3.
- (e) ζ_1 takes on the values $(30, 50, 70)$ with probability 0.3, 0.4 and 0.3; ζ_2 takes on the values $(50, 60, 70)$ with probability 0.3, 0.4 and 0.3; ζ_1 and ζ_2 are independent.

3. Stochastic demands (known prices).

The demand levels of A and B are described by a random vector, say $\eta^T = (\eta_1, \eta_2)$. The rest of the data is as in the deterministic model.

- (a) Formulate and solve a recourse model when η^T takes on the values $(400, 100)$, $(500, 200)$, $(600, 300)$ with probability 0.3, 0.4 and 0.3.
- (b) Assume η_1 and η_2 are independent random variables with normal distributions, $\eta_1 \sim N(500, 6000)$ and $\eta_2 \sim N(200, 12000)$. Find the optimal solution of the recourse problem if the production of A and B is decided in the first-stage and there is no restriction at all on the number of batches of $C1$ and $C2$.
- (c) Consider case (b). Add the restriction that the total number of batches must not exceed 120. Also ensure that the probability that the demand of B is covered must be larger than 80%.

4. Stochastic prices and demands.

Demands and prices are described by three scenarios $S1$, $S2$ and $S3$, as follows.

Demand level	$S1$	$S2$	$S3$
A	700	500	300
B	100	200	300
Unit selling price			
A	45	50	55
B	70	60	50

Formulate and solve a recourse model assuming the three scenarios have probability 0.3 , 0.4 and 0.3 respectively.

5. Obtain *EVPI* and *VSS* for some relevant cases among these alternatives.

b. Discussion of solutions

1. Choice of units and deterministic model.

Units are as follows. First, define the unit of time. We may assume here data are given per day for example. Then, demand is the number of units of *A* and *B* per day. Selling prices are given as \$ per unit of *A* and *B*. The level of production is given by the number of batches (of 60 *C*1 and 90 *C*2) per day. Capacity cost must include work-force cost, operating costs, and depreciation per day. Material costs are \$ per component. The distinction among these costs is important for the stochastic model.

There is more than one formulation for the deterministic problem. The following formulation (M1) is useful in view of later stochastic models. Let

- x_1 = number of batches of *C*1 available for production;
- x_2 = number of batches of *C*2 available for production;
- x_3 = number of units of *A* produced and sold per day;
- x_4 = number of units of *B* produced and sold per day.

For batches of *C*1 and *C*2, the objective contains the daily capacity cost. For products *A* and *B*, it contains the selling price minus the material costs. (Each unit of *A*, e.g. has a selling price of \$50. It requests 6 units of *C*1 and 8 units of *C*2 for a total material cost of \$12. The difference is the objective coefficient 38.) The first two constraints state that the usage of components is smaller than the availability. The third constraint is the upper limit on the number of batches. Demand and capacity bounds follow.

$$(M1) \quad z = \max -150x_1 - 180x_2 + 38x_3 + 50x_4 \\ \text{s. t. } 6x_3 + 10x_4 \leq 60x_1, \\ 8x_3 + 5x_4 \leq 90x_2, \\ x_1 + x_2 \leq 120, \\ 40 \leq x_1, 20 \leq x_2, 0 \leq x_3 \leq 500, 0 \leq x_4 \leq 200.$$

The optimal solution of (M1) is $z = 5800$, $x_1 = 220/3$, $x_2 = 140/3$, $x_3 = 400$, $x_4 = 200$. Product *B* is at the maximum corresponding to its demand. All 120 batches of capacity are used. The rest of the solutions follow.

A shorter formulation (M2) is to define two variables:

- x_1 = number of units of *A* produced and sold per day;
- x_2 = number of units of *B* produced and sold per day.

This formulation requires computing the margins of A and B . Each unit of A obtains the selling price of \$50. It requires 6 components $C1$ and 8 components $C2$ for a total material cost of \$12. It also requires 6/60 batches of capacity for $C1$ and 8/90 batches for $C2$ at a cost of \$31. The net margin for A is thus \$7 per unit. Similarly, the net margin for B is \$15 per unit. Note that this calculation of the margins of A and B is only valid if there is no unused capacity or unsold product, which is not always the case in a stochastic model. The first two constraints correspond to maintaining at least the existing capacity levels of 40 and 20 respectively. The third constraint corresponds to a maximal capacity level of 120 (each unit of A requires 6/60 of $C1$ and 8/90 of $C2$, or 17/90 capacity units; each unit of B requires 10/60 of $C1$ and 5/90 of $C2$ or 20/90 capacity units). The model also includes the demand constraints and reads as follows:

$$(M2) \quad z = \max 7x_1 + 15x_2 \\ \text{s. t. } 6x_1 + 10x_2 \geq 2400, \\ 8x_1 + 5x_2 \geq 1800, \\ 17x_1 + 20x_2 \leq 10800, \\ 0 \leq x_1 \leq 500, 0 \leq x_2 \leq 200.$$

This model has the same optimal solution, $z = 5800$, $x_1 = 400$, $x_2 = 200$, as previously. It is clear in (M2) that the margin of B is larger than that of A . Thus, product B is at the maximum corresponding to its demand. Product A is then reduced from the limit of 120 batches of capacity. The number of batches for $C1$ and $C2$ can be computed from the production of A and B , and are equal to 220/3 and 140/3, respectively.

2. Stochastic prices.

The essential modeling question concerns the timing of the decisions. Typically, the capacity decisions are made in the long run. They are first-stage decisions. Sales occur when the price is known. They are always second-stage decisions. Depending on the flexibility of the production process, the decision on the quantity to be produced may be first- or second-stage. We may thus distinguish between two formulations: production is first-stage (M3) or second-stage (M4).

2.1. Production is first-stage.

Let

- x_1 = number of batches of $C1$ available for production;
- x_2 = number of batches of $C2$ available for production;
- x_3 = number of units of A produced per day;
- x_4 = number of units of B produced per day;
- y_1 = number of units of A sold per day;
- y_2 = number of units of B sold per day;

$$\begin{aligned}
z = & \max -150x_1 - 180x_2 - 12x_3 - 10x_4 \\
& + E_{\xi}(q_1(\omega)y_1(\omega) + q_2(\omega)y_2(\omega)) \\
\text{s. t. } & 6x_3 + 10x_4 \leq 60x_1, \\
& 8x_3 + 5x_4 \leq 90x_2, \\
& x_1 + x_2 \leq 120 \\
& y_1(\omega) \leq x_3, y_2(\omega) \leq x_4, \\
& 40 \leq x_1, 20 \leq x_2, 0 \leq x_3, 0 \leq x_4, \\
& 0 \leq y_1(\omega) \leq 500, 0 \leq y_2(\omega) \leq 200,
\end{aligned}$$

where $\xi^T(\omega) = (q_1(\omega), q_2(\omega)) = \zeta^T(\omega)$ corresponds to the selling prices.

In practice, it is customary to use a simplified notation where the dependence of y and ξ on ω is not made explicit. This (abuse of) notation is used here.

$$\begin{aligned}
(M3) \quad z = & \max -150x_1 - 180x_2 - 12x_3 - 10x_4 \\
& + E_{\xi}(q_1 y_1 + q_2 y_2) \\
\text{s. t. } & 6x_3 + 10x_4 \leq 60x_1, \\
& 8x_3 + 5x_4 \leq 90x_2, \\
& x_1 + x_2 \leq 120, \\
& y_1 \leq x_3, y_2 \leq x_4, \\
& 40 \leq x_1, 20 \leq x_2, 0 \leq x_3, 0 \leq x_4, \\
& 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 200,
\end{aligned}$$

where $\xi^T = (q_1, q_2) = \zeta^T$.

We now transform (M3) as in Section 2.4a. Assuming q_1 and q_2 are never negative (a much needed assumption for the producer to survive), we obtain

$$\begin{aligned}
(M3') \quad z = & \max -150x_1 - 180x_2 - 12x_3 - 10x_4 \\
& + E_{\xi}(q_1 \min\{x_3, 500\} + q_2 \min\{x_4, 200\}) \\
\text{s. t. } & 6x_3 + 10x_4 \leq 60x_1, \\
& 8x_3 + 5x_4 \leq 90x_2, \\
& x_1 + x_2 \leq 120, \\
& 40 \leq x_1, 20 \leq x_2, 0 \leq x_3, 0 \leq x_4,
\end{aligned}$$

or

$$\begin{aligned}
(M3'') \quad z = & \max -150x_1 - 180x_2 - 12x_3 - 10x_4 \\
& + \mu_1 \min\{x_3, 500\} + \mu_2 \min\{x_4, 200\} \\
\text{s. t. } & 6x_3 + 10x_4 \leq 60x_1 \\
& 8x_3 + 5x_4 \leq 90x_2 \\
& x_1 + x_2 \leq 120 \\
& 40 \leq x_1, 20 \leq x_2, 0 \leq x_3, 0 \leq x_4
\end{aligned}$$

where (μ_1, μ_2) is the expectation of ξ^T .

As (μ_1, μ_2) is equal to the deterministic selling prices $(50, 60)$, it is easy to show that (M3'') has the same optimal solution as the model (M1). This is true for each of the considered cases (a) to (e). To put it another way, if production is decided in the first-stage, the stochastic model where only the selling prices are

random can be replaced by a deterministic model with the random prices replaced by their expectations.

2.2. Production is second-stage

Let x_1 and x_2 be as in (M3) and

- $y_1 = \text{number of units of } A \text{ produced and sold per day};$
- $y_2 = \text{number of units of } B \text{ produced and sold per day}.$

$$(M4) \quad z = \max -150x_1 - 180x_2 + E_{\xi}(q_1 y_1 + q_2 y_2)$$

s. t. $x_1 + x_2 \leq 120,$
 $6y_1 + 10y_2 \leq 60x_1,$
 $8y_1 + 5y_2 \leq 90x_2,$
 $40 \leq x_1, 20 \leq x_2, 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 200,$

where $\xi^T = (q_1, q_2) = \zeta^T - (12, 10)$ corresponds to selling prices minus material costs.

Before using formulation (M4), consider the deterministic formulation (M2). As long as the margin of B is larger than the margin of A and the margin of A remains positive, it is optimal to produce and sell 400 A and 200 B . If this holds for all realizations of the selling prices, the same optimal solution is obtained for all realizations of ζ . It is thus the optimal solution of the stochastic model. (This will be elaborated in the comments after Proposition 5 of Chapter 4.) The expected margin is simply $E_{\zeta}(400\zeta_1 + 200\zeta_2 - 26, 200)$ where 26,200 is the total of the material and capacity costs for the daily production of 400 A and 200 B . As (ζ_1, ζ_2) has expectation (50,60) as in the deterministic model, the expected margin is again the same as in the deterministic model. This situation occurs in cases (a), (b) and (c) of this exercise: the margin of A is $\zeta_1 - 43$, the margin of B is $\zeta_2 - 45$ and the relation $\zeta_2 - 45 \geq \zeta_1 - 43 \geq 0$ holds.

If at some point, the margin of A becomes negative or exceeds that of B , then (M4) is a truly stochastic model. For cases (d) and (e), there are values of the selling prices where the margin of A exceeds that of B . The stochastic model (M4) has to be solved.

In case (d), ζ^T takes on the values $(70,50)$, $(50,60)$, $(30,70)$ with probability 0.3, 0.4 and 0.3, respectively. First-stage optimal capacity decisions are $(x_1, x_2) = (69.167, 50.833)$. Second-stage optimal production and sale decisions (x_3, x_4) are $(500, 115)$, $(500, 115)$ and $(358.333, 200)$ for the three possible scenarios. The optimal objective value is $z = 5990$.

In case (e), the two random variables ζ_1 and ζ_2 are independent, taking three different values each. Thus, the second-stage must consider 9 realizations. The optimal solution is the same as in the deterministic case: first-stage decisions are $(x_1, x_2) = (73.333, 46.667)$, second-stage decisions are $(x_3, x_4) = (400, 200)$, with objective value $z = 5800$.

3. Stochastic demands.

(a) As in Question 2, the first modeling question is the timing of the decisions. Capacity decisions are made in the long run and are first-stage decisions. Sales occur when price is known and are second-stage. The decisions on the quantities to be produced may be first- or second-stage.

(a.1) Production is first-stage.

If production is first-stage, lost sales occur when demand exceeds production. What happens when production exceeds demand is problem dependent. In some situations, excess production may be held in inventory. This would be the case when the randomness represents day-to-day variations in demand. Then excess production is used later to compensate for possible lost sales. Randomness only results in inventory costs. On the other hand, for products such as perishable goods, production is lost (C_1 and C_2 could be flour and eggs, A and B could be bread and pastry, e.g.) and lost sales cannot be compensated. The same is true when the randomness describes a set of scenarios of which only one is realized. The scenarios could represent the uncertainty about the success of a new product. If a product is not successful, extra production is lost. If it is very successful, sales are lost to competitors if the production level is insufficient. Or, alternative actions are needed such as subcontracting or overtime.

We now present a formulation (M5) corresponding to a scenario situation (excess production is lost, lost sales are not compensated). The decision variables are the same as in (M3).

$$(M5) \quad z = \max -150x_1 - 180x_2 - 12x_3 - 10x_4 + E_{\xi}(50y_1 + 60y_2)$$

$$\text{s. t. } \begin{aligned} 6x_3 + 10x_4 &\leq 60x_1, \\ 8x_3 + 5x_4 &\leq 90x_2, \\ x_1 + x_2 &\leq 120, \\ y_1 &\leq x_3, y_2 \leq x_4, \\ 40 &\leq x_1, 20 \leq x_2, 0 \leq x_3, 0 \leq x_4, \\ 0 &\leq y_1 \leq d_1, 0 \leq y_2 \leq d_2, \end{aligned}$$

where $\xi^T = (d_1, d_2) = \eta^T$ correspond to the demand level.

The first-stage optimal capacity decisions are $(x_1, x_2) = (56.667, 41.111)$. The second-stage optimal production and sale decisions (x_3, x_4) are $(400, 100)$ in the three possible scenarios. The optimal objective value is $z = 4300$. Observe that the production is set to meet the lowest possible demand.

(a.2) Production is second-stage.

If production is second-stage, lost sales occur when the available production capacities are insufficient to cover the demand. Excess production does not occur as the level of production can be adjusted to the downside. The decision variables are the same as in (M4). Formulation (M6) reads as follows:

$$(M6) \quad z = \max -150x_1 - 180x_2 + E_{\xi}(38y_1 + 50y_2)$$

s. t. $x_1 + x_2 \leq 120$,
 $6y_1 + 10y_2 \leq 60x_1$,
 $8y_1 + 5y_2 \leq 90x_2$,
 $40 \leq x_1, 20 \leq x_2, 0 \leq y_1 \leq d_1, 0 \leq y_2 \leq d_2$,

where $\xi^T = (d_1, d_2) = \eta^T$ corresponds to the demand level.

The first-stage optimal capacity decisions are $(x_1, x_2) = (67.083, 41.111)$. The second-stage optimal production and sale decisions (x_3, x_4) are $(400, 100)$, $(337.5, 200)$ and $(337.5, 200)$ for the three possible scenarios. The optimal objective value is $z = 4575$. Observe that the capacity limit of 120 batches is not fully used.

(b) We consider a variant of formulation (M5) where the only constraints on x_1 and x_2 are the components usage:

$$(M7) \quad z = \max -150x_1 - 180x_2 - 12x_3 - 10x_4$$

$$+ E_{\xi}(50 \min\{x_3, d_1\} + 60 \min\{x_4, d_2\})$$

s. t. $6x_3 + 10x_4 \leq 60x_1$,
 $8x_3 + 5x_4 \leq 90x_2$,
 $0 \leq x_1, 0 \leq x_2, 0 \leq x_3, 0 \leq x_4$,

where $\xi^T = (d_1, d_2) = \eta^T$ corresponds to the demand level.

Clearly, the two constraints are always tight. Replacing x_1 by $(6x_3 + 10x_4)/60$ and x_2 by $(8x_3 + 5x_4)/90$, the model becomes

$$z = \max \{-43x_3 - 45x_4 + E_{\xi}(50 \min\{x_3, d_1\} + 60 \min\{x_4, d_2\}) \mid 0 \leq x_3, 0 \leq x_4\},$$

or

$$(M7') \quad z = \max \{-43x_3 + 50E_{\xi_1} \min\{x_3, \xi_1\} - 45x_4 + 60E_{\xi_2} \min\{x_4, \xi_2\} \mid 0 \leq x_3, 0 \leq x_4\}.$$

This optimization is separable in x_3 and x_4 . Both variables will be nonzero. So, we are searching twice for the unconstrained minimum of an expression of the form $-ax + b\mathcal{Q}(x)$, with $\mathcal{Q}(x) = E_{\xi} \min\{x, \xi\}$ and $\xi \sim N(\mu, \sigma^2)$. From Exercise 2.8.2, we obtain that $\mathcal{Q}'(x) = 1 - F(x)$. As $\mathcal{Q}''(x) = -f(x)$, the second-order conditions are satisfied. Thus the unconstrained minimum is obtained for $\mathcal{Q}'(x) = a/b$, i.e. $1 - F(x) = a/b$.

Denote by $F_i(\cdot)$ the cumulative distribution of ξ_i , $i = 1, 2$. For x_3 , the unconstrained optimum satisfies $1 - F_1(x_3) = 43/50$, or $F_1(x_3) = 0.14$. It corresponds to a quantile $q = -1.08$ and a decision $x_3 = 500 - 1.08\sqrt{6000} = 416.34$. For x_4 , we have $1 - F_2(x_4) = 45/60$, or $F_2(x_4) = 0.25$. It corresponds to a

quartile $q = -0.675$ and a decision $x_4 = 200 - 0.675\sqrt{12000} = 126.06$. For the sake of comparison, we may compute $x_1 = (6x_3 + 10x_4)/60 = 62.644$ and $x_2 = (8x_3 + 5x_4)/90 = 44.011$. Also, using the closed form expression of $\mathcal{Q}(x)$, (see again Exercise 2.8.2), one can obtain the optimal value of z .

(c) Requesting that the probability that the demand of B is covered must be larger than 80% is $P(x_4 \geq \xi_2) \geq 0.8$ or $F_2(x_4) \geq 0.8$. The 0.8 quantile is 0.84. Thus, $F_2(x_4) \geq 0.8$ is equivalent to $(x_4 - \mu_2)/\sigma_2 \geq 0.8$, or $x_4 \geq 200 + 0.84\sqrt{12000}$, or $x_4 \geq 292.02$.

The model to solve is:

$$(M8) \quad z = \max \left\{ -43x_3 + 50E_{\xi_1} \min\{x_3, \xi_1\} - 45x_4 + 60E_{\xi_2} \min\{x_4, \xi_2\} \mid 0 \leq x_3, 292.02 \leq x_4, 17x_3 + 20x_4 \leq 10800 \right\},$$

where the constraint on the 120 batches has been transformed as in (M2).

By applying the Karush-Kuhn-Tucker conditions (see Review Section 2.11c.), one can show that $(x_3, x_4) = (291.74, 292.02)$ is the optimal solution.

4. Just as in the previous cases, there are two possible formulations as the production decisions may be first- or second-stage. Model (M9) corresponds to first-stage production while (M10) corresponds to second-stage production.

$$(M9) \quad z = \max -150x_1 - 180x_2 - 12x_3 - 10x_4 + E_{\xi}(q_1 y_1 + q_2 y_2)$$

s. t. $6x_3 + 10x_4 \leq 60x_1$,
 $8x_3 + 5x_4 \leq 90x_2$,
 $x_1 + x_2 \leq 120$,
 $y_1 \leq x_3$, $y_2 \leq x_4$,
 $40 \leq x_1$, $20 \leq x_2$, $0 \leq x_3$, $0 \leq x_4$,
 $0 \leq y_1 \leq d_1$, $0 \leq y_2 \leq d_2$,

where $\xi^T = (q_1, q_2, d_1, d_2)$, with q_1 and q_2 the selling prices and d_1 and d_2 the demands jointly defined in a scenario. Thus $\xi^T = (45, 70, 700, 100)$, $(50, 60, 500, 200)$ and $(55, 50, 300, 300)$ with probability 0.3, 0.4, and 0.3 respectively. The optimal solution is $z = 3600$, $(x_1, x_2) = (46.667, 32.222)$ with corresponding $(x_3, x_4) = (300, 100)$. The second-stage decisions are $(y_1, y_2) = (300, 100)$ in all three scenarios. As the production cannot be adapted to the demand, the optimal solution is to plan for the lowest demand and the expected margin is low.

$$(M10) \quad z = \max -150x_1 - 180x_2 + E_{\xi}(q_1 y_1 + q_2 y_2)$$

s. t. $x_1 + x_2 \leq 120$
 $6y_1 + 10y_2 \leq 60x_1$,
 $8y_1 + 5y_2 \leq 90x_2$,
 $40 \leq x_1$, $20 \leq x_2$, $0 \leq y_1 \leq d_1$, $0 \leq y_2 \leq d_2$,

where $\xi^T = (q_1, q_2, d_1, d_2)$ with q_1 and q_2 the selling prices minus the material costs and d_1 and d_2 the demands. Thus, $\xi^T = (33, 60, 700, 100)$, $(38, 50, 500, 200)$ and $(43, 40, 300, 300)$ with probability 0.3, 0.4, and 0.3. The optimal solution is $z = 4048.75$, $(x_1, x_2) = (73.333, 46.667)$. The second-stage decisions are $(y_1, y_2) = (462.5, 100)$, $(400, 200)$ and $(300, 260)$ in the three scenarios. While

obtaining the optimal solution of (M10) with your favorite LP solver, you may observe that there is a high shadow price for the maximum number of batches.

Exercises

1. Consider Exercise 1 of Section 1.6.
 - (a) Show that this is a two-stage stochastic program with first-stage integer decision variables. Observe that, for a random variable with integer realizations, the second-stage variables can be assumed continuous because the optimal second-stage decisions are automatically integer. Assume that Northam revises its seating policy every year. Is a multistage program needed?
 - (b) Assume that the data in Exercise 1 correspond to the demand for seat reservations. Assume that there is a 50% probability that all clients with a reservation effectively show up and that 10 or 20% no-shows occur with equal probability. Model this situation as a three-stage program, with first-stage decisions as before, second-stage decisions corresponding to the number of accepted reservations, and third-stage decisions corresponding to effective seat occupation. Show that the third stage is a simple recourse program with a reward for each occupied seat and a penalty for each denied reservation.
 - (c) Consider now the situation where the number of seats has been fixed to 12, 24, and 140 for the first class, business class, and economy class, respectively. Assume the top management estimates the reward of an occupied seat to be 4, 2, and 1 in the first class, business class, and economy class, respectively, and the penalty for a denied reservation is 1.5 times the reward. Model the corresponding problem as a recourse program. Find the optimal acceptance policy with the data of Exercise 1 in Section 1.6 and no-shows as in (b) of the current exercise. To simplify, assume that passengers with a denied reservation are not seated in a higher class even if a seat is available there.
2. Let $\mathcal{Q}(x) = E_{\xi} \min\{x, \xi\}$.
 - (a) Obtain a closed form expression for $\mathcal{Q}(x)$ when ξ follows a Poisson distribution.
 - (b) Obtain a closed form expression for $\mathcal{Q}(x)$ when ξ follows a normal distribution. (Hint: for a normal distribution, the relation $\xi f(\xi) = \mu f(\xi) - \sigma^2 f'(\xi)$ holds for any given ξ .)
 - (c) Assume ξ has a continuous distribution. Show that $\mathcal{Q}'(x) = 1 - F(x)$.
3. Consider an airplane with x seats. Assume passengers with reservations show up with probability 0.90, independently of each other.

- (a) Let $x = 40$. If 42 passengers receive a reservation, what is the probability that at least one is denied a seat.
- (b) Let $x = 50$. How many reservations can be accepted under the constraint that the probability of seating all passengers who arrive for the flight is greater than 90%?
4. Consider the design problem in Section 1.4. Suppose the design decision does not completely specify x in (1.4.1), but the designer only knows that if a value \hat{x} is specified then $x \in [.99\hat{x}, 1.01\hat{x}]$. Suppose a uniform distribution for x is assumed initially on this interval. How would the formulation in Section 1.4 be modified to account for information as new parts are produced?
5. Consider the example in Section 2.7a.
- (a) One may feel uncomfortable with the deterministic linear equivalent yielding a non-integer number of seats. Show how to cope with this.
- (b) One may also feel uncomfortable with the demands represented by normal distributions. Show that deterministic linear equivalents are also obtained if $\xi_F \sim P(3)$ and $\xi_B \sim P(4)$ for example.

2.9 Alternative Characterizations and Robust Formulations

While the main focus of this book is on problems that can be represented in the form in (4.1–4.4) as stochastic linear programs, this formulation can still represent a wide range of risk preferences. As observed in Section 2.5, an expected von Neumann-Morgenstern concave utility objective can be represented as a piecewise-linear function. For example, if the utility function is $U(-q(\omega)^T y(\omega) - \gamma)$ where γ is a scaling parameter for fitting the function, then an additional set of variables $y'(\omega)_j$ with bounds u_j and slopes $-q'_j$ such that $0 \leq y'(\omega)_j \leq u_j$, $-q'_j \geq -q'_{j+1}$, and for $j = 0, \dots, J$ can be defined with an additional linear constraint as:

$$-y'_0 + \sum_{j=1}^J y'_j(\omega) - q(\omega)^T y(\omega) = \gamma, \quad (9.1)$$

and with a new recourse function objective to minimize

$$-q'_0 y_0(\omega) + \sum_{j=1}^J q'_j y_j(\omega). \quad (9.2)$$

The parameters γ , q' , and u' can be chosen to fit the utility function U as closely as desired while maintaining the same linear optimization form as in (4.1–4.4).

Other risk-measures may be included in the objective and as fixed or probabilistic constraints. A common use of these constraints in financial applications is to maximize expected return subject to a constraint on *value-at-risk* (*VaR*), the greatest loss in portfolio value that can occur with a given probability α , defined as

$$VaR_\alpha(q(\omega)^T y(\omega)) = \min\{t | P(q(\omega)^T y(\omega) \leq t) \geq \alpha\}. \quad (9.3)$$

A *VaR* constraint to limit losses to be no greater than \bar{t} with probability at most α can then be written as

$$P(q(\omega)^T y(\omega) \leq \bar{t}) \geq \alpha, \quad (9.4)$$

since this ensures that $VaR_\alpha(q(\omega)^T y(\omega)) \leq \bar{t}$.

A criticism of *VaR* as a measure of risk is that it does not have the useful property of subadditivity such that the *VaR* of the sum of two random variables is at most the sum of the *VaR*'s of each individual random variable. The subadditive property is part of the set of axioms that define coherent risk measures (see Artzner, Delbaen, Eber, and Heath [1999]), such that $R(\cdot)$ is a *coherent risk measure* if the following hold:

- Definition 2.1.**
1. *subadditivity*: $R(\xi + \zeta) \leq R(\xi) + R(\zeta)$ for any random variables ξ and ζ ;
 2. *positive homogeneity (of degree one)*: $R(\lambda \xi) = \lambda R(\xi)$ for all $\lambda \geq 0$;
 3. *monotonicity*: $R(\xi) \leq R(\zeta)$ whenever $\xi \preceq \zeta$, where \preceq indicates first-order stochastic dominance, i.e., $P(\xi \leq t) \geq P(\zeta \leq t), \forall t$;
 4. *translation invariance*: $R(\xi + t) = R(\xi) + t$ for any $t \in \mathfrak{R}$.

A related risk measure to *VaR*, called the *conditional value-at-risk* (*CVaR*), can be defined to avoid the potential problems of a non-subadditive risk measure by taking the conditional expectations over losses in excess of *VaR*. For random loss ξ with distribution function P , the α -confidence level is then defined as

$$CVaR_\alpha(\xi) = E_{P_\alpha}[\xi], \quad (9.5)$$

where P_α is the distribution function defined by

$$P_\alpha(t) = \begin{cases} 0 & \text{if } t < VaR_\alpha(\xi); \\ \frac{P(t) - \alpha}{1 - \alpha} & \text{if } t \geq VaR_\alpha(\xi). \end{cases} \quad (9.6)$$

As shown by Rockafellar and Uryasev [2000,2002], *CVaR* satisfies all of the axioms for a coherent risk measure (Exercise 3) and has a convenient representation as the solution to the following optimization problem:

$$CVaR_\alpha(\xi) = \min_t t + \frac{1}{1 - \alpha} E_P[(\xi - t)^+], \quad (9.7)$$

which can also be written as the linear program:

$$\min t + \frac{1}{1 - \alpha} E_P[y(\omega)] \quad (9.8)$$

$$\text{s. t. } \xi(\omega) - y(\omega) \leq t, \text{ a. s.} \quad (9.9)$$

$$y(\omega) \geq 0, \text{ a. s.} \quad (9.10)$$

With the representation in (9.8), a risk constraint to limit $CVaR_\alpha$ to be less than \bar{t} can be constructed similarly to the probabilistic constraint in (9.4) or the downside risk constraint in (5.3) with additional linear constraints and variables $y'(\omega)$ as follows:

$$t + \frac{1}{1-\alpha} E[y'(\omega)] \leq \bar{t} \quad (9.11)$$

$$-t + q(\omega)^T y(\omega) - y'(\omega) \leq 0, \text{ a.s.,} \quad (9.12)$$

$$y'(\omega) \geq 0, \text{ a.s.} \quad (9.13)$$

The use of coherent risk measures has another useful interpretation that R is a coherent risk measure if and only if there is a class of probability measure \mathcal{P} such that $R(\xi)$ equals the highest expectation of ξ with respect to members of this class (see Huber [1981]):

$$R(\xi) = \sup_{P \in \mathcal{P}} E_P[\xi]. \quad (9.14)$$

This representation provides a worst-case view of the risk, which is discussed in more detail in Chapter 8.

One worst-case version of the approach in (9.14) is to let \mathcal{P} correspond to any distribution with support in a given range or uncertainty set. This worst-case type of risk-measure is called *robust* so that optimization models including a robust risk-measure of this form are *robust optimization* models. A robust version of the two-stage stochastic program can then be written as:

$$\begin{aligned} \min_x \max_{\xi \in \Xi} & c^T x + Q(x, \xi) \\ \text{s. t.} & Ax = b, \\ & x \geq 0. \end{aligned} \quad (9.15)$$

Depending on the properties of Ξ , robust optimization models can be tractable linear or conic optimization models. A variety of results in the area appear in Bertsimas and Sim [2006], Ben-Tal and Nemirovski [2002] with multi-period extensions also appearing, for example, in Ben-Tal, Boyd, and Nemirovski [2006] and Bertsimas, Iancu, and Parrilo [2010].

Exercises

1. Give an example of random variables ξ and ζ where $VaR_\alpha(\xi + \zeta) > VaR_\alpha(\xi) + VaR_\alpha(\zeta)$ for some $0 < \alpha < 1$.
2. Show that VaR satisfies the axioms of positive homogeneity, monotonicity, and translation independence.
3. Show that $CVaR$ satisfies all of the axioms for a coherent risk measure.
4. Give a class of probability distribution \mathcal{P} such that $CVaR$ solves (9.14).

5. Find the robust formulation of the two-stage model (9.15) when uncertainty is only in the right-hand side $h \in \Xi = [l, u]$, a rectangular region.
6. Find the robust formulation of the two-stage model (9.15) when uncertainty is only in the right-hand side $h \in \Xi = \{h | (h - \mu)^T V(h - \mu) \leq 1\}$, an ellipsoidal region.

2.10 Relationship to Other Decision-Making Models

The stochastic programming models considered in this section illustrate the general form of a stochastic program. While this form can apply to virtually all decision-making problems with unknown parameters, certain characteristics typify stochastic programs and form the major emphasis of this book. In general, stochastic programs are generalizations of deterministic mathematical programs in which some uncontrollable data are not known with certainty. The key features are typically many decision variables with many potential values, discrete time periods for decisions, the use of expectation functionals for objectives, and known (or partially known) distributions. The relative importance of these features contrasts with similar areas, such as statistical decision theory, decision analysis, dynamic programming, Markov decision processes, and stochastic control. In the following subsections, we consider these other areas of study and highlight the different emphases.

a. Statistical decision theory and decision analysis

Wald [1950] developed much of the foundation of optimal statistical decision theory (see also DeGroot [1970] and Berger [1985]). The basic motivation was to determine best levels of variables that affect the outcome of an experiment. With variables x in some set X , random outcomes, $\omega \in \Omega$, an associated distribution, $F(\omega)$, and a reward or loss associated with the experiment under outcome ω of $r(x, \omega)$, the basic problem is to find $x \in X$ to

$$\max E_{\omega}[r(x, \omega)|F] = \max \int_{\omega} r(x, \omega) dF(\omega). \quad (10.1)$$

The problem in (10.1) is also the fundamental form of stochastic programming. The major differences in emphases between the fields stem from underlying assumptions about the relative importance of different aspects of the problem.

In stochastic programming, one generally assumes that difficulties in finding the form of the function r and changes in the distribution F as a function of actions are small in comparison to finding the expectations with known distributions and an optimal value x with all other information known. The emphasis is on finding a solution after a suitable problem statement in the form (10.1) has been found.

For example, in the simple farming example in Section 1.1, the number of possible planting configurations (even allowing only whole-acre lots) is enormous. Enumerating the possibilities would be hopeless. Stochastic programming avoids such inefficiencies through an optimization process.

We might suppose that the fields or crop varieties are new and that the farmer has little direct information about yields. In this case, the yield distribution would probably start as some prior belief but would be modified as time went on. This modification and possible effects of varying crop rotations to obtain information are the emphases from statistical decision theory. If we assumed that only limited variation in planting size (such as 50-acre blocks) was possible, then the combinatorial nature of the problem would look less severe. Enumeration might then be possible without any particular optimization process. If enumeration were not possible, the farmer might still update the distributions and objectives and use stochastic programming procedures to determine next year's crops based on the updated information.

In terms of (10.1)), statistical decision theory places a heavy emphasis on changes in F to some updated distribution \hat{F}_x that depends on a partial choice of x and some observations of ω . The implied assumption is that this part of the analysis dominates any solution procedure, as when X is a small finite set that can be enumerated easily.

Decision analysis (see, e.g., Raiffa [1968]) can be viewed as a particular part of optimal statistical decision theory. The key emphases are often on acquiring information about possible outcomes, on evaluating the utility associated with various outcomes, and on defining a limited set of possible actions (usually in the form of a decision tree). For example, consider the capacity expansion problem in Section 1.3. We considered a wide number of alternative technology levels and production decisions. In that model, we assumed that demand in each period was independent of the demand in the previous period. This characteristic gave the block separability property that can allow efficient solutions for large problems.

A decision analytic model might apply to the situation where an electric utility's demand depends greatly on whether a given industry locates in the region. The decision problem might then be broken into separate stochastic programs depending on whether the new industry demand materializes and whether the utility starts on new plants before knowing the industry decision. In this framework, the utility first decides whether to start its own projects. The utility then observes whether the new industry expands into the region and faces the stochastic program form from Section 1.4 with four possible input scenarios about the available capacity when the industry's location decision is known (see Figure 3).

The two stochastic programs given each initial decision allow for the evaluation of expected utility given the two possible outcomes and two possible initial decisions. The actual initial decision taken on current capacity expansion would then be made by taking expectations over these two outcomes.

Separation into distinct possible outcomes and decisions and the realization of different distributions depending on the industry decision give this model a decision analysis framework. In general, a decision analytic approach would probably also consider multiple attributes of the capacity decisions (for example, social costs for a

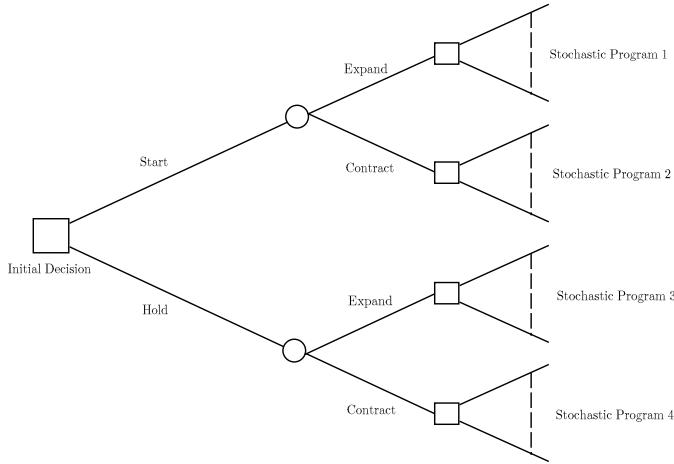


Fig. 3 Decision tree for utility with stochastic programs on leaves.

given location) and would concentrate on the value of risk in the objective. It would probably also entail consideration of methods for obtaining information about the industry's decision and contingent decisions based on the outcomes of these investigations. Of course, these considerations can all be included in a stochastic program, but they are not typically the major components of a stochastic programming analysis.

b. Dynamic programming and Markov decision processes

Much of the literature on stochastic optimization considers dynamic programming and Markov decision processes (see, e.g., Heyman and Sobel [1984], Bellman [1957], Ross [1983], and Kall and Wallace [1994] for a discussion relating to stochastic programming). In these models, one searches for optimal actions to take at generally discrete points in time. The actions are influenced by random outcomes and carry one from some state at some stage t to another state at stage $t + 1$. The emphasis in these models is typically in identifying finite (or, at least, low-dimensional) state and action spaces and in assuming some Markovian structure (so that actions and outcomes only depend on the current state).

With this characterization, the typical approach is to form a backward recursion resulting in an optimal decision associated with each state at each stage. With large state spaces, this approach becomes quite computationally cumbersome although it does form the basis of many stochastic programming computation schemes as given in Chapter 6. Another approach is to consider an infinite horizon and use discounting

to establish a stationary policy (see Howard [1960] and Blackwell [1965]) so that one need only find an optimal decision associated with a state for any stage.

A typical example of this is in investment. Suppose that instead of saving for a specific time period in the example of Section 1.2, you wish to maximize a discounted expected utility of wealth in all future periods. In this case, the state of the system is the amount of wealth. The decision or action is to determine what amount of the wealth to invest in stock and bonds. We could discretize to varying wealth levels and then form a problem as follows:

$$\max \sum_{t=1}^{\infty} \rho^t E[q\mathbf{y}(t) - r\mathbf{w}(t)] \quad (10.2)$$

$$\begin{aligned} \text{s. t.} \quad & x(1,1) + x(2,1) = b, \\ & \xi(1,t)\mathbf{x}(1,t) + \xi(2,t)\mathbf{x}(2,t) - \mathbf{y}(t) + \mathbf{w}(t) = G, \\ & \xi(1,t)\mathbf{x}(1,t) + \xi(2,t)\mathbf{x}(2,t) = \mathbf{x}(1,t+1) + \mathbf{x}(2,t+1), \\ & \mathbf{x}(i,t), \mathbf{y}(t), \mathbf{w}(t) \geq 0, \quad \mathbf{x} \in \mathcal{N}, \end{aligned}$$

where \mathcal{N} is the space of nonanticipative decisions and ρ is some discount factor. This approach could lead to finding a stationary solution to

$$\begin{aligned} z(b) = \max_{x(1)+x(2)=b} & \{E[-q(G - \xi(1)x(1) - \xi(2)x(2))^- \\ & - r(G - \xi(1)x(1) - \xi(2)x(2))^+ + \rho E[z(\xi(1)x(1) + \xi(2)x(2))]\}]. \end{aligned} \quad (10.3)$$

Again, problem (10.2) fits the general stochastic programming form, but particular solutions as in (10.3) are more typical of Markov decision processes. These are not excluded in stochastic programs, but stochastic programs generally do not include the Markovian assumptions necessary to derive (10.3).

c. Machine learning and online optimization

While Markov decision problems have the general character of stochastic programs of including a distribution over some set of uncertain parameters, online optimization problems involve a changing objective (perhaps chosen adversarially) without knowledge of the choice and only considering the history of observations. The objective is then to choose x^1, x^2, \dots sequentially to minimize

$$\sum_{t=1}^H f^t(x^t), \quad (10.4)$$

where H may increase without bound and each x^t is chosen only with knowledge of x^1, \dots, x^{t-1} and $f^1(x^1), \dots, f^{t-1}(x^{t-1})$. Performance is measured in terms of regret, which refers to the difference relative to best possible choices taken *ex post*,

i.e.,

$$\text{regret}_H = \sum_{t=1}^H f^t(x^t) - \min_{x \in X} \sum_{t=1}^H f^t(x), \quad (10.5)$$

where X is some feasible region.

The emphasis in this stream of literature is on algorithms with provable regret bounds. For convex objectives, stochastic search methods (as in Chapter 9) can obtain bounds on regret_H , such as $O(H^{3/4})$, $O(\sqrt{H})$, and $O(\log H)$ depending on properties of f^t and observability of the function (see, respectively, Hazan, Kalai, Kale, and Agarwal [2006], Zinkerich [2003], Flaxman, Kalai, and McMahon [2004]).

d. Optimal stochastic control

Stochastic control models are often similar to stochastic programming models. The differences are mainly due to problem dimension (stochastic programs would generally have higher dimension), emphases on control rules in stochastic control, and more restrictive constraint assumptions in stochastic control. In many cases, the distinction is, however, not at all clear.

As an example, suppose a more general formulation of the financial model in Section 1.2. There, we considered a specific form of the objective function, but we could also use other forms. For example, suppose the objective was generally stated as minimizing some cost $r_t(\mathbf{x}(t), \mathbf{u}(t))$ in each time period t , where $\mathbf{u}(t)$ are the controls $u((i, j), t, s)$ that correspond to actual transactions of exchanging asset i into asset j in period t under scenario s . In this case, problem (1.2.2) becomes:

$$\begin{aligned} \min z &= \sum_s p(s) \left(\sum_{t=1}^H r_t(x(t, s), u(t, s), s) \right) \\ \text{s. t. } & x(0, s) = b, \\ & x(t, s) + \xi(s)^T u(t, s) = x(t+1, s), t = 0, \dots, H, \\ & x(s), u(s) \text{ nonanticipative,} \end{aligned} \quad (10.6)$$

where $\xi(s)$ represents returns on investments minus transaction costs. Additional constraints may be incorporated into the objective of (10.6) through penalty terms.

Problem (10.6) is fairly typical of a discrete time control problem governed by a linear system. The general emphasis in control approaches to such problems is for *linear*, *quadratic*, *Gaussian* (LQG) models (see, for example, Kushner [1971], Fleming and Rishel [1975], and Dempster [1980]), where we have a linear system as earlier, but where the randomness is Gaussian in each period (for example, ξ is known but the state equation for $x(t+1, s)$ includes a Gaussian term), and r_t is quadratic. In these models, one may also have difficulty observing x so that an additional observation variable $y(t)$ may be present.

LQG models can also include forms of risk aversion as, for example, in Whittle [1990]. In this model, instead of an additively time-separable model as generally used here, the objective to minimize becomes:

$$\frac{2}{\theta} \log E[e^{\theta \sum_{t=1}^H (\mathbf{x}^t)^T Q^t \mathbf{x}^t + (\mathbf{u}^t)^T R^t \mathbf{u}^t}], \quad (10.7)$$

where $\mathbf{x}^{t+1} = A^t \mathbf{x}^t + B^t \mathbf{u}^t + \boldsymbol{\varepsilon}^t$. A useful property is that this objective avoids some of the issues with time-additive utility functions that do not appear consistent with preferences (as, for example, discussed in Kreps and Porteus [1979], Epstein and Zinn [1989]). A minimizing solution also has a min-max characterization as in robust optimization models and the max-min utility function proposed in Gilboa and Schmeidler [1989] (see Exercise 3 and Hansen and Sargent [1995]).

The LQG problem leads to Kalman filtering solutions (see, for example, Kalman [1969]). Various extensions of this approach are also possible, but the major emphasis remains on developing controls with specific decision rules to link observations directly into estimations of the state and controls. In stochastic programming models, general constraints (such as non-negative state variables) are emphasized. In this case, most simple decision rules forms (such as when \mathbf{u} is a linear function of state) fail to obtain satisfactory solutions (see, for example, Gartska and Wets [1974]). For this reason, stochastic programming procedures tend to search for more general solution characteristics.

Stochastic control procedures may, of course, apply but stochastic programming tends to consider more general forms of interperiod relationships and state space constraints. Other types of control formulations, such as robust control, may also be considered specific forms of a stochastic program that are amenable to specific techniques to find control policies with given characteristics.

Continuous time stochastic models (see, e.g., Harrison [1985]) are also possible but generally require more simplified models than those considered in stochastic programming. Again, continuous time formulations are consistent with stochastic programs but have not been the main emphasis of research or the examples in this book. In certain examples again, they may be quite relevant (see, for example, Harrison and Wein [1990] for an excellent application in manufacturing) in defining fundamental solution characteristics, such as the optimality of control limit policies.

In all these control problems, the main emphasis is on characterizing solutions of some form of the dynamic programming Bellman-Hamilton-Jacobi equation or application of Pontryagin's maximum principle. Stochastic programs tend to view all decisions from beginning to end as part of the procedure. The dependence of the current decision on future outcomes and the transient nature of solutions are key elements. Section 3.5 provides some further explanation by describing these characteristics in terms of general optimality conditions.

e. Summary

Stochastic programming is simply another name for the study of optimal decision making under uncertainty. The term *stochastic programming* emphasizes a link to mathematical programming and algorithmic optimization procedures. These considerations dominate work in stochastic programming and distinguish stochastic programming from other fields of study. In this book, we follow this paradigm of concentrating on representation and characterizations of optimal decisions and on developing procedures to follow in determining optimal or approximately optimal decisions. This development begins in the next chapter with basic properties of stochastic program solution sets and optimal values.

Exercises

1. Consider the design problem in Section 1.4. Suppose the design decision does not completely specify x in (1.4.1), but the designer only knows that if a value \hat{x} is specified then $x \in [.99\hat{x}, 1.01\hat{x}]$. Suppose a uniform distribution for x is assumed initially on this interval and that the designer can alter the design once after manufacturing and testing N axles out of a total predicted demand of 1,000 axles. The designer assumes that her posterior distribution on the actual mean relative to \hat{x} would not change if she adjusts the target diameter \hat{x} after observing the first N axle diameters. With these assumptions, formulate a Bayesian model to determine an initial specification \hat{x}^1 and N followed by a second specification \hat{x}^2 for the remaining $1000 - N$ axles.
2. From the example in Section 1.2, suppose that a goal in each period is to realize a 16% return in each period with penalties $q = 1$ and $r = 4$ as before. Formulate the problem as in (10.2).
3. Consider the risk-sensitive model in (10.7) given initial state x^1 , $\theta > 0$, $H = 2$, and $\varepsilon^1 \sim N(\mu, \Sigma)$, the multivariate normal distribution with mean μ and variance-covariance matrix, Σ . Show that solving (10.7) is equivalent to solving the min-max problem:

$$\min_{u^1} \max_{\varepsilon^1} \theta [((u^1)^T R^1 u^1 + x^2(x^1, u^1, \varepsilon^1)^T Q^2 x^2(x^1, u^1, \varepsilon^1)^T) + (\varepsilon^1 - \mu)^T \Sigma^{-1} (\varepsilon^1 - \mu)], \quad (10.8)$$

i.e., u^1 optimal in (10.8) is also optimal in (10.7) and vice versa as long as both problems have finite optimal values. To do this, first show that $\int e^{-Q(x,y)} dy = k e^{-\min_y Q(x,y)}$ for some constant k (independent of x) for any positive definite quadratic function $Q(x,y)$.

2.11 Short Reviews

a. Linear programming

Consider a linear program (LP) of the form

$$\max\{c^T x \mid Ax = b, x \geq 0\}, \quad (11.1)$$

where A is an $m \times n$ matrix, x and c are $n \times 1$ vectors, and b is an $m \times 1$ vector. If needed, any inequality constraint can be transformed into an equality by the addition of *slack variables*:

$$a_i \cdot x \leq b_i \quad \text{becomes} \quad a_i \cdot x + s_i = b_i,$$

where s_i is the slack variable of row i and a_i is the i th row of matrix A .

A *solution* to (11.1) is a vector x that satisfies $Ax = b$. A *feasible solution* is a solution x with $x \geq 0$. An *optimal solution* x^* is a feasible solution such that $c^T x^* \geq c^T x$ for all feasible solutions x . A *basis* is a choice of n linearly independent columns of A . Associated with a basis is a submatrix B of the corresponding columns, so that, after a suitable rearrangement, A can be partitioned into $A = [B, N]$. Associated with a basis is a *basic solution*, $x_B = B^{-1}b$, $x_N = 0$, and $z = c_B^T B^{-1}b$, where $[x_B, x_N]$ and $[c_B, c_N]$ are partitions of x and c following the basic and nonbasic columns. We use B^{-1} to denote the inverse of B , which is known to exist because B has linearly independent columns and is square.

In geometric terms, basic solutions correspond to *extreme points* of the polyhedron, $\{x \mid Ax = b, x \geq 0\}$. A basis is feasible (optimal) if its associated basic solution is feasible (optimal). The conditions for feasibility are $B^{-1}b \geq 0$. The conditions for optimality are that in addition to feasibility, the inequalities, $c_N^T - c_B^T B^{-1}N \leq 0$, hold.

Linear programs are routinely solved by widely distributed, easy-to-use LP solvers. Access to such a solver would be useful for some exercises in this book. For a better understanding, some examples and exercises also use manual solutions of linear programs.

Finding an optimal solution is equivalent to finding an optimal *dictionary*, a definition of individual variables in terms of the other variables. In the *simplex algorithm*, starting from a feasible dictionary, the next one is obtained by selecting an *entering variable* (any nonbasic variable whose increase leads to an increase in the objective value), then finding a *leaving variable* (the first to become negative as the entering variable increases), then realizing a *pivot* substituting the entering for the leaving variable in the dictionary. An optimal solution is reached when no entering variable can be found.

A linear program is *unbounded* if an entering variable exists for which no leaving variable can be found. In some cases, a feasible initial dictionary is not available at once. Then, *phase one* of the simplex method consists of finding such an initial dictionary. A number of artificial variables are introduced to make the dictionary

feasible. The phase one procedure minimizes the sum of artificials using the simplex method. If a solution with a sum of artificials equal to zero exists, then the original problem is feasible and *phase two* continues with the true objective function. If the optimal solution of the phase one problem is nonzero, then the original problem is *infeasible*.

As an example, consider the following linear program:

$$\begin{aligned} \max & -x_1 + 3x_2 \\ \text{s. t. } & 2x_1 + x_2 \geq 5, \\ & x_1 + x_2 \leq 3, \\ & x_1, x_2 \geq 0. \end{aligned}$$

Adding slack variables s_1 and s_2 , the two constraints read

$$\begin{aligned} 2x_1 + x_2 - s_1 &= 5, \\ x_1 + x_2 + s_2 &= 3. \end{aligned}$$

The natural choice for the initial basis is (s_1, s_2) . This basis is infeasible as s_1 would obtain the value -5 . An *artificial variable* (a_1) is added to row one to form:

$$2x_1 + x_2 - s_1 + a_1 = 5.$$

The phase-one problem consists of minimizing a_1 , i.e., finding $-\max -a_1$. Let $z = -a_1$ be the phase one objective, which after substituting for a_1 gives the initial dictionary in phase one:

$$\begin{aligned} z &= -5 + 2x_1 + x_2 - s_1, \\ a_1 &= 5 - 2x_1 - x_2 + s_1, \\ s_2 &= 3 - x_1 - x_2, \end{aligned}$$

corresponding to the initial basis (a_1, s_2) . Entering candidates are x_1 and x_2 as they both increase the objective value. Choosing x_1 , the leaving variable is a_1 (because it becomes zero for $x_1 = 2.5$ while s_2 becomes zero only for $x_1 = 3$). Substituting x_1 for a_1 , the second dictionary becomes:

$$\begin{aligned} z &= -a_1, \\ x_1 &= 2.5 - 0.5x_2 + 0.5s_1 - 0.5a_1, \\ s_2 &= 0.5 - 0.5x_2 - 0.5s_1 + 0.5a_1. \end{aligned}$$

This dictionary is an optimal dictionary for phase one. (No nonbasic variable would possibly increase x .) This means the original problem is feasible. (In fact, the basis (x_1, s_2) is feasible with solution $x_1 = 2.5$, $x_2 = 0.0$.)

We now turn to phase two. We replace the phase one objective with the original objective:

$$z = -x_1 + 3x_2 = -2.5 + 3.5x_2 - 0.5s_1.$$

By removing the artificial variable a_1 (as it is not needed anymore), we obtain the following first dictionary in phase two:

$$\begin{aligned} z &= -2.5 + 3.5x_2 - 0.5s_1, \\ x_1 &= 2.5 - 0.5x_2 + 0.5s_1, \\ s_2 &= 0.5 - 0.5x_2 - 0.5s_1. \end{aligned}$$

The next entering variable is x_2 with leaving variable s_2 . After substitution, we obtain the final dictionary:

$$\begin{aligned} z &= 1 - 4s_1 - 7s_2, \\ x_1 &= 2 + s_1 + s_2, \\ x_2 &= 1 - s_1 - 2s_2, \end{aligned}$$

which is optimal because no nonbasic variable is a valid entering variable. The optimal solution is $x^* = (2, 1)^T$ with $z^* = 1$.

b. Duality for linear programs

The *dual* of the so-called primal problem (11.1) is:

$$\min\{\pi^T b \mid \pi^T A \geq c^T, \pi \text{ unrestricted}\}. \quad (11.2)$$

Variables π are called *dual variables*. One such variable is associated with each constraint of the primal. When the primal constraint is an equality, the dual variable is *free* (unrestricted in sign). Dual variables are sometimes called *shadow prices* or *multipliers* (as in nonlinear programming). The dual variable π_i may sometimes be interpreted as the marginal value associated with resource b_i .

If the dual is unbounded, then the primal is infeasible. Similarly, if the primal is unbounded, then the dual is infeasible. Both problems can also be simultaneously infeasible.

If x is primal feasible and π is dual feasible, then $c^T x \leq \pi^T b$. The primal has an optimal solution x^* if and only if the dual has an optimal solution π^* . In that case, $c^T x^* = (\pi^*)^T b$ and the primal and dual solutions satisfy the *complementary slackness conditions*:

$$(a_{i \cdot})x^* = b_i \text{ or } \pi_i^* = 0 \text{ or both, for any } i = 1, \dots, m,$$

$$(\pi^*)^T a_{\cdot j} = c_j \text{ or } x_j^* = 0 \text{ or both, for any } j = 1, \dots, n,$$

where $a_{i \cdot}$ is the j -th column of A and, as before, $a_{\cdot i}$ is the i -th row of A .

An alternative presentation is to say that $s_i^* \pi_i^* = 0$, where s_i is the slack variable of the i -th constraint, i.e., either the slack or the dual variable associated with a constraint is zero, and similarly for the second condition. Thus, the optimal solution of the dual can be recovered from the optimal solution for the primal, and vice versa.

The optimality conditions can also be interpreted to say that either there exists some *improving direction*, w , from a current feasible solution, \hat{x} , so that $c^T w > 0$, $w_j \geq 0$ for all $j \in N$, $N = \{j \mid \hat{x}_j = 0\}$, and $a_i \cdot w = 0$ for all $i \in I$, $I = \{i \mid a_i \cdot \hat{x} = b_i\}$ (hence, for $Ax = b$ in the primal system of (11.1), $I = \{1, \dots, m\}$) or there exists some π such that $\sum_{i \in I} \pi_i a_{ij} \geq c_j$ for all $j \in N$, $\sum_{i \in I} \pi_i a_{ij} = c_j$ for all $j \notin N$, but both cannot occur. This result is equivalent to the *Farkas lemma*, which gives alternative systems with or without solutions.

The *dual simplex method* replicates on the primal solution what the iterations of the simplex method would be on the dual problem: it first finds the leaving variable (one that is strictly negative) then the entering variable (the first one that would become positive in the objective line). The dual simplex is particularly useful when a solution is already available to the original primal problem and some extra constraint or bound is added to the problem. The reader is referred to Chvátal [1980, pp. 152–157] for a detailed presentation.

Other material not covered in this section is meant to be restrictive to a given topic area. The next section discusses more of the mathematical properties of solutions and functions.

c. Nonlinear programming and convex analysis

When objectives and constraints may contain nonlinear functions, the optimization problem becomes a *nonlinear program*. The nonlinear program analogous to (11.1) has the form

$$\min\{f(x) \mid g(x) \leq 0, h(x) = 0\}, \quad (11.3)$$

where $x \in \mathbb{R}^n$, $f : \mathbb{R}^n \rightarrow \mathbb{R}$, $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$, and $h : \mathbb{R}^n \rightarrow \mathbb{R}^l$. We may also assume that the range of f may include ∞ to allow the objective to include constraints directly through an *indicator function*:

$$\delta(x \mid X) = \begin{cases} 0 & \text{if } g(x) \leq 0, h(x) = 0, \\ +\infty & \text{otherwise,} \end{cases}$$

where X is the set of x satisfying the constraints in (11.3), i.e., the *feasible region*.

In this book, the feasible region is usually a *convex set* so that X contains any *convex combination*,

$$\sum_{i=1}^s \lambda^i x^i, \sum_{i=1}^s \lambda^i = 1, \lambda^i \geq 0, i = 1, \dots, s,$$

of points, x^i , $i = 1, \dots, s$, that are in the feasible region. Extreme points of the region are points that cannot be expressed as a convex combination of two distinct points also in the region. The set of all convex combinations of a given set of points is its *convex hull*.

The feasible region is also most generally *closed* so that it contains all limits of infinite sequences of points in the region. The region is also generally *connected*, so that, for any x^1 and x^2 in the region, there exists some path of points in the feasible region connecting x^1 to x^2 by a function, $\eta : [0, 1] \rightarrow \Re^n$ that is continuous with $\eta(0) = x^1$ and $\eta(1) = x^2$. For certain results, we may also assume the region is *bounded* so that a *ball* of radius M , $\{x \mid \|x\| \leq M\}$, contains the entire set of feasible points. Otherwise, the region is *unbounded*. Note that a region may be unbounded while the optimal value in (11.1) or (11.3) is still bounded. In this case, the region often contains a *cone*, i.e., a set S such that if $x \in S$, then $\lambda x \in S$ for all $\lambda \geq 0$. When the region is both closed and bounded, then it is *compact*.

The set of equality constraints, $h(x) = 0$, is often *affine*, i.e., they can be expressed as linear combinations of the components of x and some constant. In this case, each constraint, $h_i(x) = 0$, is a *hyperplane*, $a_i \cdot x - b_i = 0$, as in the linear program constraints. In this case, $h(x) = 0$, defines an *affine space*, a *translation* of the *parallel subspace*, $Ax = 0$. The affine space *dimension* is the same as its parallel subspace, i.e., the maximum number of linearly independent vectors in the subspace.

With nonlinear constraints and inequalities, the region may not be an affine space, but we often consider the lowest-dimension affine space containing them, i.e., the *affine hull* of the region. The affine hull is useful in optimality conditions because it distinguishes *interior* points that can be the center of a ball entirely within the region from the *relative interior* (ri), which can be the center of a ball whose intersection with the affine hull is entirely within the region. When a point is not in a feasible region, we often take its *projection* into the region using an operator, Π . If the region is X , then the projection of x onto X is $\Pi(x) = \operatorname{argmin}\{\|w - x\| \mid w \in X\}$.

In this book, we generally assume that the objective function f is a *convex function*, i.e., such that

$$f(\lambda x^1 + (1 - \lambda)x^2) \leq \lambda f(x^1) + (1 - \lambda)f(x^2),$$

$0 \leq \lambda \leq 1$. If f also is never $-\infty$ and is not $+\infty$ everywhere, then f is a *proper convex function*. The region where f is finite is called the *effective domain* of f ($\operatorname{dom} f$). We can also define convex functions in terms of the *epigraph* of f , $\operatorname{epi}(f) = \{(x, \beta) \mid \beta \geq f(x)\}$. In this case, f is convex if and only if its epigraph is convex. If $-f$ is convex, then f is *concave*.

Often, we assume that f has *directional derivatives*, $f'(x; w)$, that are defined as:

$$f'(x; w) = \lim_{\lambda \downarrow 0} \frac{f(x + \lambda w) - f(x)}{\lambda}.$$

When these limits exist and do not vary in all directions, then f is *differentiable*, i.e., there exists a *gradient*, ∇f , such that

$$\nabla f^T w = f'(x; w)$$

for all directions $w \in \Re^n$. We sometimes distinguish this standard form of differentiability from stricter forms as *Gâteaux* or *G-differentiability*. The stricter forms impose more conditions on the directional derivative such as uniform convergence over compact sets (*Hadamard* derivatives).

We also consider *Lipschitz* continuous or *Lipschitzian* functions such that $|f(x) - f(w)| \leq M\|x - w\|$ for any x and w and some $M < \infty$. If this property holds for all x and w in a set X , then f is *Lipschitzian relative to X* . When this property only holds *locally*, i.e., for $\|w - x\| \leq \varepsilon$ for some $\varepsilon > 0$, then f is *locally Lipschitz* at x .

Among differentiable functions, we often use *quadratic* functions that have a *Hessian* matrix of second derivatives, D , and can be written as

$$f(x) = c^T x + \frac{1}{2} x^T D x .$$

Many functions are not, however, differentiable. In this case, we express optimality in terms of *subgradients* at a point x , or vectors, η , such that

$$f(w) \geq f(x) + \eta^T (w - x)$$

for all w . In this case, $\{(x, \beta) \mid \beta = f(x) + \eta^T (w - x)\}$ is a *supporting hyperplane* of f at x . The set of subgradients at a point x is the *subdifferential* of f at x , written $\partial f(x)$.

Other useful properties include that f is *piecewise linear*, i.e., such that $f(x)$ is linear over regions defined by linear inequalities. When f is *separable* so that $f(x) = \sum_{i=1}^n f_i(x_i)$, then other advantages are possible in computation.

Given f convex and a convex feasible region in (11.3), we can define conditions that an optimal solution x^* and associated multipliers (π^*, ρ^*) must satisfy. In general, these conditions require some form of *regularity* condition. A common form is that there exists some \hat{x} such that $g(\hat{x}) < 0$ and h is affine. This is generally called the *Slater condition*.

Given a regularity condition of this type, if the constraints in (11.3) define a feasible region, then x^* is optimal if and only if the *Karush-Kuhn-Tucker* conditions hold so that $x^* \in X$ and there exists $\pi^* \geq 0, \rho^*$ such that

$$\nabla f(x^*) + (\pi^*)^T \nabla g(x^*) + (\rho^*)^T \nabla h(x^*) = 0, \nabla g(x^*)^T \pi^* = 0 . \quad (11.4)$$

Optimality can also be expressed in terms of the *Lagrangian*:

$$l(x, \pi, \rho) = f(x) + \pi^T g(x) + \rho^T h(x) ,$$

so that sequentially minimizing over x and maximizing over π (in both orders) produces the result in (11.4). This occurs through a *Lagrangian dual problem* to (11.3) as

$$\max_{\pi \geq 0, \rho} \inf_x f(x) + \pi^T g(x) + \rho^T h(x) , \quad (11.5)$$

which is always a lower bound on the objective in (11.3) (*weak duality*), and, under the regularity conditions, yields equal optimal values in (11.3) and (11.4) (*strong duality*). In many cases, the Lagrangian can also be interpreted with the *conjugate* function of f , defined as

$$f^*(\pi) = \sup_x \{\pi^T x - f(x)\},$$

which is also a convex function if f is convex.

Our algorithms often apply to the Lagrangian to obtain *convergence*, i.e., a sequence of solutions, $x^v \rightarrow x^*$. In some cases, we also approximate the function so that $f^v \rightarrow f$ in some way. If this convergence is *pointwise*, then $f^v(x) \rightarrow f(x)$ for each x individually. If the convergence is *uniform* on a set X , then, for any $\varepsilon > 0$, there exists $N(\varepsilon)$ such that for all $v \geq N(\varepsilon)$ and all $x \in X$, $|f^v(x) - f(x)| < \varepsilon$.

Part II

Basic Properties

Chapter 3

Basic Properties and Theory

This chapter considers the basic properties and theory of stochastic programming. As throughout this book, the emphasis is on results that have direct application in the solution of stochastic programs. Proofs are included for those results we consider most central to the overall development.

The main properties we consider are formulations of deterministic equivalent programs to a stochastic program, the forms of the feasible region and objective function, and conditions for optimality and solution stability. Our focus is on stochastic programs with recourse, and, in particular, for stochastic linear programs. The first section describes two-stage versions of these problems in detail. It assumes some knowledge of convex sets and functions.

Sections 3.2 to 3.5 add extensions to the results in Section 3.1 by allowing additional forms of constraints, objectives, and decision variables. Section 3.2 considers problems with probabilistic or chance constraints that occur with some fixed probability. Section 3.4 examines multiple-stage problems, while Section 3.3 considers problems with integer variables. Section 3.5 then extends results to include nonlinear functions.

3.1 Two-Stage Stochastic Linear Programs with Fixed Recourse

a. Formulation

As in Chapter 2, we first form the basic two-stage stochastic linear program with fixed recourse. It is repeated here for clarity.

$$\begin{aligned} \min z &= c^T x + \mathbb{E}_{\xi} [\min q(\omega)^T y(\omega)] \\ \text{s. t.} \quad Ax &= b, \\ T(\omega)x + Wy(\omega) &= h(\omega), \\ x \geq 0, \quad y(\omega) &\geq 0, \end{aligned} \tag{1.1}$$

where c is a known vector in \Re^{n_1} , b a known vector in \Re^{m_1} , A and W are known matrices of size $m_1 \times n_1$ and $m_2 \times n_2$, respectively, and W is called the *recourse matrix*, which we assume here is fixed. This allows us to characterize the feasibility region in a convenient manner for computation. If W is not fixed, we may have difficulties, as shown next.

For each ω , $T(\omega)$ is $m_2 \times n_1$, $q(\omega) \in \Re^{n_2}$ and $h(\omega) \in \Re^{m_2}$. Piecing together the stochastic components of the problem, we obtain a vector $\xi^T(\omega) = (q(\omega)^T, h(\omega)^T, T_1(\omega), \dots, T_{m_2}(\omega))$ with $N = n_2 + m_2 + (m_2 \times n_1)$ components, where $T_i(\omega)$ is the i -th row of the *technology matrix* $T(\omega)$. As before, E_ξ represents the mathematical expectation with respect to ξ . Let also $\Xi \subseteq \Re^N$ be the support of ξ , i.e., the smallest closed subset in \Re^N such that $P\{\xi \in \Xi\} = 1$. As said in Section 2.4, the constraints are assumed to hold almost surely.

Problem (1.1) is equivalent to the so-called *deterministic equivalent program* (DEP):

$$\begin{aligned} \min z &= c^T x + \mathcal{Q}(x) \\ \text{s. t. } Ax &= b, \\ x &\geq 0, \end{aligned} \tag{1.2}$$

where

$$\mathcal{Q}(x) = E_\xi Q(x, \xi(\omega)) \tag{1.3}$$

and

$$Q(x, \xi(\omega)) = \min_y \{q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x, y \geq 0\}. \tag{1.4}$$

Examples of formulations (1.1) and (1.2)–(1.4) have been given in Chapter 1. In the farmer's problem, x represents the surfaces devoted to each crop, ξ represents the yields so that only the technology matrix $T(\omega)$ is stochastic (because prices q and requirements h are fixed), and y represents the sales and purchases of the various crops. Formulations (1.1) and (1.2)–(1.4) apply for both discrete and continuous random variables. Examples with continuous random yields have also been given for the farmer's problem.

This representation clearly illustrates the sequence of events in the recourse problem. First-stage decisions x are taken in the presence of uncertainty about future realizations of ξ . In the second stage, the actual value of ξ becomes known and some corrective actions or recourse decisions y can be taken. First-stage decisions are, however, chosen by taking their future effects into account. These future effects are measured by the value function or recourse function, $\mathcal{Q}(x)$, which computes the expected value of taking decision x .

When T is nonstochastic, the original formulation (1.2)–(1.4) can be replaced by

$$\min z = c^T x + \Psi(\chi)$$

$$\begin{aligned} \text{s. t. } & Ax = b, \\ & Tx - \chi = 0, \\ & x \geq 0, \end{aligned} \tag{1.5}$$

where $\Psi(\chi) = E_{\xi}\psi(\chi, \xi(\omega))$ and $\psi(\chi, \xi(\omega)) = \min\{q(\omega)^T y \mid Wy = h(\omega) - \chi, y \geq 0\}$. This formulation stresses the fact that choosing x corresponds to generating an m_2 -dimensional *tender* $\chi = Tx$ to be bid against the outcomes $h(\omega)$ of the random events.

The difficulty inherent in stochastic programming clearly lies in the computational burden of computing $\mathcal{Q}(x)$ for all x in (1.2)–(1.4), or $\Psi(\chi)$ for all χ in (1.5). It is no surprise therefore that the properties of the deterministic equivalent program in general and of the functions $\mathcal{Q}(x)$ or $\Psi(\chi)$ have been extensively studied. The next sections present some of the known properties.

b. Discrete random variables

We now present some basic properties when ξ is a discrete random variable. This is an important class of random variables. It is widely used in applications, either directly or through sampling of a continuous distribution. The properties presented in this section are used in Section 5.1 for the algorithmic solution of (1.2)–(1.4).

Let $K_1 = \{x \mid Ax = b, x \geq 0\}$ be the set determined by the fixed constraints, namely those that do not depend on the particular realization of the random vector. For any given ξ , we may define a so-called “elementary feasibility set” as

$$K_2(\xi) = \{x \mid y \geq 0 \text{ exists s. t. } W(\omega)y = h(\omega) - T(\omega)x\}.$$

Example 1

Consider the following second-stage program

$$\begin{aligned} & \min 2y_1 + y_2 \\ \text{s. t. } & y_1 + 2y_2 \geq \xi_1 - x_1, \\ & y_1 + y_2 \geq \xi_2 - x_1 - x_2, \\ & 0 \leq y_1 \leq 1, 0 \leq y_2 \leq 1. \end{aligned}$$

Using the upper bounds on y , the first constraint implies $\xi_1 - x_1 \leq 3$ and the second one implies $\xi_2 - x_1 - x_2 \leq 2$. Thus, $K_2(\xi) = \{x \mid x_1 \geq \xi_1 - 3, x_1 + x_2 \geq \xi_2 - 2\}$.

As ξ is discrete, we may easily define the second-stage feasibility set

$$K_2 = \bigcap_{\xi \in \Xi} K_2(\xi).$$

In Example 1, if ξ_1 takes the value 2, 3, 4 and ξ_2 the values 1, 4, 7 with some nonspecified probabilities, independently of each other or not, $K_2 = \{x \mid x_1 \geq 1, x_1 + x_2 \geq 5\}$. In fact, it suffices here to know the componentwise maximum of ξ to obtain K_2 . This set is a polyhedron.

Define $\text{pos}W = \{t \mid Wy = t, y \geq 0\}$. It is called the *positive hull* of W . It represents the set of right-hand sides that can be obtained by a non-negative combination of the columns of W . The positive hull is easily seen to be a convex cone.

Theorem 1.

- a. For a given ξ , the elementary feasibility set $K_2(\xi)$ is a convex polyhedron.
- b. When ξ is a finite discrete random variable, K_2 is a convex polyhedron.

Proof:

a. Consider some x and ξ such that no $y \geq 0$ exists such that $W(\omega)y = h(\omega) - T(\omega)x$. Using the notation $\text{pos}W$, it is the same to say that we consider some x and ξ such that $h(\omega) - T(\omega)x \notin \text{pos}W(\omega)$. Thus, we have a point, $h(\omega) - T(\omega)x$, which does not belong to a convex set, $\text{pos}W(\omega)$. Then, there must exist some hyperplane, say $\{x \mid \sigma^T x = 0\}$, that separates $h(\omega) - T(\omega)x$ from $\text{pos}W(\omega)$. This hyperplane satisfies $\sigma^T t < 0$ for $t \in \text{pos}W(\omega)$ and $\sigma^T(h(\omega) - T(\omega)x) > 0$. For one particular ξ , $W(\omega)$ is fixed and there can be only finitely many different such hyperplanes which completes the proof.

b. The intersection of finitely many convex polyhedra is a convex polyhedron. \square

Efficient ways to obtain the separating hyperplanes (and more generally to obtain K_2) are presented in Chapter 5.

For fixed value of x and ξ , the value $Q(x, \xi)$ of the second-stage program is given by

$$Q(x, \xi) = \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x, y \geq 0\}. \quad (1.6)$$

Difficulties may arise when the mathematical program (1.6) is unbounded below or infeasible. Unboundedness typically results of an ill-defined model and can easily be avoided by adding upper bounds on y . Infeasibility is avoided if we only consider $x \in K_2$. Thus, for $x \in K_2$, $Q(x, \xi)$ is finite for all ξ and we may define

$$\mathcal{Q}(x) = E_\xi Q(x, \xi) = \sum_{k=1}^K p_k Q(x, \xi_k)$$

where $k = 1, \dots, K$ represents the K realizations of ξ . If wanted, the deterministic equivalent program can be rewritten as

$$\min z(x) = c^T x + \mathcal{Q}(x)$$

$$\text{s. t. } x \in K_1 \cap K_2 .$$

We now study the properties of the second-stage value function.

Theorem 2. *For a given ξ , the value function $Q(x, \xi)$ is*

- (a) *a piecewise linear convex function in (h, T) ;*
- (b) *a piecewise linear concave function in q ;*
- (c) *a piecewise linear convex function in x for all $x \in K_2$.*

When ξ is a finite discrete random variable, $Q(x)$ is piecewise linear and convex on K_2 .

Proof: To prove convexity in (a) and (c), we just need to prove that $f(b) = \min\{q^T y \mid Wy = b\}$ is a convex function in b . We consider two different vectors, say b_1 and b_2 , and some convex combination $b_\lambda = \lambda b_1 + (1 - \lambda)b_2$, $\lambda \in (0, 1)$.

Let y_1^* and y_2^* be some optimal solution of $\min\{q^T y \mid Wy = b\}$ for $b = b_1$ and $b = b_2$, respectively. Then, $\lambda y_1^* + (1 - \lambda)y_2^*$ is a feasible solution of $\min\{q^T y \mid Wy = b_\lambda\}$. Now, let y_λ^* be an optimal solution of this last problem. We thus have

$$\begin{aligned} f(b_\lambda) &= q^T y_\lambda^* \leq q^T(\lambda y_1^* + (1 - \lambda)y_2^*) \\ &= \lambda q^T y_1^* + (1 - \lambda)q^T y_2^* = \lambda f(b_1) + (1 - \lambda)f(b_2), \end{aligned}$$

which proves the required proposition. A similar proof can be given to show concavity in q . To prove piecewise linearity, observe that solving (1.6) for given x and ξ amounts to finding some square submatrix $B(\omega)$ of $W(\omega)$, called a basis (see Section 2.11), such that $y_B = B(\omega)^{-1}(h(\omega) - T(\omega)x)$, $y_N = 0$, where y_B is the subvector associated with the columns of B and y_N includes the remaining components of y . A basis is feasible if $y_B \geq 0$ and a feasible basis is optimal if $a_B(\omega)^T B(\omega)^{-1} W(\omega) \leq q(\omega)^T$. As long as these conditions hold, we have

$$Q(x, \xi) = q_B(\omega)^T B(\omega)^{-1}(h(\omega) - T(\omega)x),$$

which is linear in q , h , T and x on a domain defined by the feasibility and optimality conditions. Piecewise linearity follows from the existence of finitely many different optimal bases for the second-stage program. An alternative proof of piecewise linearity of the value function of a linear program can be obtained through the method of projections (see Martin [1999, Corollary 2.49]). \square

Property (c) is important in practice. It is used in Section 5.1 for the algorithmic solution of (1.2)–(1.4).

Example 2

Consider the following second-stage program:

$$\begin{aligned}
& \min 2y_1 + y_2 \\
\text{s. t. } & y_1 + y_2 \geq 1 - x_1, \\
& y_1 \geq \xi - x_1 - x_2, \\
& y_1, y_2 \geq 0.
\end{aligned}$$

To reduce the calculations, assume $0 \leq x_1 \leq 1$, $0 \leq x_2 \leq 1$. The optimal second-stage solutions are as follows:

- i. if $\xi \leq x_1 + x_2 \Rightarrow y_1 = 0$, $y_2 = 1 - x_1$;
- ii. if $\xi > x_1 + x_2 \Rightarrow y_1 = \xi - x_1 - x_2$ and $y_2 = (1 - \xi + x_2)^+$ where $a^+ = \max(a, 0)$.

This results in three situations (as $1 - \xi + x_2$ may be positive or negative). Setting the second-stage decisions into the second-stage objective, one obtains the following three pieces for $Q(x, \xi)$:

$$Q(x, \xi) = \begin{cases} 1 - x_1 & \text{for } 0 \leq \xi < x_1 + x_2, \\ \xi + 1 - 2x_1 - x_2 & \text{for } x_1 + x_2 \leq \xi \leq 1 + x_2, \\ 2(\xi - x_1 - x_2) & \text{for } 1 + x_2 \leq \xi. \end{cases}$$

Thus $Q(x, \xi)$ is clearly piecewise linear in x . The proof of the convexity of this particular $Q(x, \xi)$ is left as part of Exercise 4. In this example, $h(\xi) = \xi$ and $Q(x, \xi)$ is one-dimensional in ξ . Convexity in ξ can be established in any classical way.

Another property is evident from parametric solutions of linear programs when q and T are fixed. Notice that

$$Q(x, [q, \lambda(h') + Tx, T]) = \lambda Q(x, [q, h' + Tx, T]) \quad (1.7)$$

for any $\lambda \geq 0$ because a dual optimal solution for $h = h' + Tx$ is also dual feasible for $h = \lambda(h') + Tx$ and complementary with y^* optimal for $h = h' + Tx$. Because λy^* is also feasible for $h = \lambda(h') + Tx$, λy^* is optimal for $h = \lambda(h') + Tx$, demonstrating (1.7). This says that $Q(x, [q, h' + Tx, T])$ is a *positively homogeneous* function of h' . From the convexity of $Q(x, [q, h' + Tx, T])$ in $h = h' + Tx$, this function is also sublinear (see Theorem 4.7 of Rockafellar [1969]) in h' . This property is central to some bounding procedures described in Chapter 8.

Complete descriptions of $Q(x, \xi)$ are also often useful. Finding the distribution induced on $Q(x, \xi)$ is often the goal of these descriptions. This information can then be used to find \mathcal{Q} or to address other risk criteria that may not be given by the expectation functional (e.g., the probability of losing some percentage of one's wealth). The description of the distribution of $Q(x, \xi)$ is called the *distribution problem*. Its solution is quite difficult although some methods exist (see Wets [1980b] and Bereanu [1980]). Approximations are generally required as in Dempster and Papagaki-Papoulias [1980]; because these results are not central to our solution development, we will not go into further detail.

We now present some of the results when ξ is not a discrete random variable.

c. General cases

For fixed value of x and ξ , the value of the second-stage program is, as before, given by (1.6)

$$Q(x, \xi) = \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x, y \geq 0\}.$$

When the mathematical program (1.6) is unbounded below or infeasible, the value of the second-stage program is defined to be $-\infty$ or $+\infty$, respectively. The expected second-stage value is, as given in (1.3)

$$\mathcal{Q}(x) = E_{\xi} Q(x, \xi).$$

Typically, the definition is made complete by adopting the convention $+\infty + (-\infty) = +\infty$. This corresponds to a conservative attitude, rejecting any first-stage decision that could lead to an undefined recourse action for some realization even if some other realization would induce an infinitely low-cost. It also reflects the fact that second-stage programs can easily be bounded by bounding y , while infeasibilities may be inherent to the problem.

For any given ξ , we may define a so-called “elementary feasibility set” as

$$K_2(\xi) = \{x \mid Q(x, \xi) < \infty\}$$

or, as before,

$$K_2(\xi) = \{x \mid y \geq 0 \text{ exists s. t. } W(\omega)y = h(\omega) - T(\omega)x\}.$$

Both definitions are equivalent for a given ξ and enjoy the properties of Theorem 1. When ξ is not a discrete random variable, we may now define K_2 in two different ways:

$$K_2 = \{x \mid \mathcal{Q}(x) < \infty\}$$

or

$$K_2^P = \bigcap_{\xi \in \Xi} K_2(\xi).$$

The set K_2^P is said to define the *possibility interpretation* of the second-stage feasibility set. A first-stage decision x belongs to K_2^P if, for “all possible” values of the random vector ξ , a feasible second-stage decision can be taken. We now illustrate that the two sets, K_2 and K_2^P , can indeed be different when the random variable is a continuous random variable.

Consider an example where the second stage is defined by

$$Q(x, \xi) = \min_y \{y \mid \xi y = 1 - x, y \geq 0\}$$

where ξ has a triangular distribution on $[0, 1]$, namely, $P(\xi \leq u) = u^2$. Note that here W reduces to a 1×1 matrix and is the only random element.

For all ξ in $(0, 1]$, the optimal y is $\frac{1-x}{\xi}$, so that

$$K_2(\xi) = \{x \mid x \leq 1\}$$

and

$$Q(x, \xi) = \frac{1-x}{\xi}, \quad \text{for } x \leq 1.$$

When $\xi = 0$, no y exists such that $0 \cdot y = 1 - x$, unless $x = 1$, so that

$$K_2(0) = \{x \mid x = 1\}.$$

Now, for $x \neq 1$, $Q(x, 0)$ should normally be $+\infty$. However, because the probability that $\xi = 0$ is zero, the convention is to take $Q(x, 0) = 0$. This corresponds to defining $0 \cdot \infty = 0$.

Hence,

$$K_2^P = \{x \mid x = 1\} \cap \{x \mid x \leq 1\} = \{x \mid x = 1\}$$

while

$$\mathcal{Q}(x) = \int_0^1 \frac{1-x}{\xi} \cdot 2\xi d\xi = 2(1-x) \quad \text{for all } x \leq 1,$$

so that $K_2 = \{x \mid x \leq 1\}$ and K_2^P is strictly contained in K_2 . The difference between the two sets relates to the fact that a point is not in K_2^P as soon as it is infeasible for some ξ value, regardless of the distribution of ξ , while K_2 does not consider infeasibilities occurring with zero probability.

Fortunately, this kind of difficulty rarely occurs for programs with a fixed W matrix. It never occurs when the random vector satisfies some conditions.

Another difficulty that could arise and would cause the sets K_2^P and K_2 to be different, would be to have $Q(x, \xi)$ bounded above with probability one and yet to have $\mathcal{Q}(x)$, the expectation of $Q(x, \xi)$, unbounded.

Proposition 3. *If ξ has finite second moments, then*

$$P(\omega \mid Q(x, \xi) < \infty) = 1 \quad \text{implies } \mathcal{Q}(x) < \infty.$$

To illustrate why this might be true, consider particular x and ξ values. The second-stage program is the linear program

$$Q(x, \xi) = \min\{q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x, y \geq 0\}.$$

As discussed in the proof of Theorem 2, solving this linear program for given x and ξ amounts to finding some optimal basis B for which we have

$$Q(x, \xi) = q_B(\omega)^T B^{-1}(h(\omega) - T(\omega)x).$$

Now, assume $Q(x, \xi)$ is bounded above with probability one and imagine for a while that the same basis B would be optimal for all x and all ξ . Then, ξ having finite second moments is a sufficient condition for $\mathcal{Q}(x)$ to be bounded because it implies $E_\xi(q_B^T B^{-1} h)$ and $E_\xi(q_B^T B^{-1} T x)$ are both bounded above. In general the optimal basis B is different for different x and ξ values so that a more general proof taking care of different submatrices of W is needed. This is done in detail in Walkup and Wets [1967].

Theorem 4. *For a stochastic program with fixed recourse where ξ has finite second moments, the sets K_2 and K_2^P coincide.*

Proof: (Note: This proof uses some concepts from measure theory.) First consider $x \in K_2^P$. This implies $Q(x, \xi) < \infty$ with probability one, so that, by Proposition 3, $\mathcal{Q}(x)$ is bounded above and $x \in K_2$.

Now, consider $x \in K_2$. It follows that $\{\xi \mid Q(x, \xi) < \infty\}$ is a set of measure one. Observe that $Q(x, \xi) < \infty$ is equivalent to $h(\omega) - T(\omega)x \in \text{pos } W$ and that $h(\omega) - T(\omega)x$ is a linear function of ξ , and $\{\xi \in \Sigma \mid Q(x, \xi) < \infty\}$ is a closed subset of Σ of measure one, for any set Σ of measure one. In particular, $\{\xi \in \Xi \mid Q(x, \xi) < \infty\}$ is a closed subset of Ξ having measure one. By definition of Ξ , this set can only be Ξ itself, so that $\{\xi \mid Q(x, \xi) < \infty\} \subseteq \Xi$ and therefore $x \in K_2^P$. \square

Note however that W being fixed and ξ having finite moments are just sufficient conditions for K_2 and K_2^P to coincide. Other, more general, sufficient conditions can be found in Walkup and Wets [1967].

Note also that a third definition of the second-stage feasibility set could be given as $\{x \mid Q(x, \xi) < \infty \text{ with probability one}\}$. For problems with fixed recourse where ξ has finite second moments, this set also coincides with K_2 and K_2^P . In the following, we simply speak of K_2 , the second-stage feasibility set.

Theorem 5. *When W is fixed and ξ has finite second moments:*

- (a) K_2 is closed and convex.
- (b) If T is fixed, K_2 is polyhedral.
- (c) Let Ξ_T be the support of the distribution of T . If $h(\xi)$ and $T(\xi)$ are independent and Ξ_T is polyhedral, then K_2 is polyhedral.

Proof: The proof of (a) is elementary under the possibility representation of K_2 . If T is fixed, $x \in K_2$ if and only if $h(\xi) - Tx \in \text{pos } W$ for all $\xi \in \Xi_h$, where Ξ_h is the support of the distribution of $h(\xi)$.

Consider some x and ξ s.t. $h(\xi) - Tx \notin \text{pos } W$. Then there must exist some hyperplane, say $\{x \mid \sigma^T x = 0\}$ that separates $h(\xi) - Tx$ from $\text{pos } W$. This hyperplane must satisfy $\sigma^T t \leq 0$ for $t \in \text{pos } W$ and $\sigma^T(h(\xi) - Tx) > 0$. Because W is fixed, there need only be finitely many different such hyperplanes, so that

$h(\xi) - Tx \in \text{pos}W$ is equivalent to $W^*(h(\xi) - Tx) \leq 0$ for some matrix W^* . This matrix, called the *polar matrix* of W , is obtained by choosing some minimal set of separating hyperplanes. The set is minimal if removing any hyperplane would no longer guarantee the equivalence between $h(\xi) - Tx \in \text{pos}W$ and $W^*(h(\xi) - Tx) \leq 0$ for all x and ξ in Ξ_h . It follows that $x \in K_2$ if and only if $W^*(h(\xi) - Tx) \leq 0$ for all ξ in Ξ . This can still be an infinite system of linear inequalities due to $h(\xi)$. We may, however, replace this system by

$$(W^*T)_i \cdot x \geq u_i^* = \sup_{h(\xi) \in \Xi_h} W_i^* h(\xi), \quad i = 1, \dots, l, \quad (1.8)$$

where W_i^* is the i -th row of W^* and l is the finite number of rows of W^* . If for some i , u_i^* is unbounded, then the problem is infeasible and the result in (b) is trivially satisfied. If, for all i , $u_i^* < \infty$, then the system (1.8) constitutes a finite system of linear inequalities defining the polyhedron $K_2 = \{x \mid W^*Tx \geq u^*\}$ where u^* is the vector whose i th component is u_i^* . This proves (b). When T is stochastic, a relation similar to (1.8) holds, which, unless Ξ_T is finite, defines an infinite system of inequalities. Whenever Ξ_T is polyhedral, (c) can be proved by working on the extremal elements of Ξ_T . This is done in Wets [1974, Corollary 4.13]. \square

We now turn to the properties of $\mathcal{Q}(x, \xi)$, assuming it is not $-\infty$. First, observe that $\mathcal{Q}(x, \xi)$ enjoys all the properties of Theorem 2.

Theorem 6. *For a stochastic program with fixed recourse where ξ has finite second moments,*

- (a) $\mathcal{Q}(x)$ is a Lipschitzian convex function and is finite on K_2 .
- (b) If $F(\xi)$ is an absolutely continuous distribution, $\mathcal{Q}(x)$ is differentiable on $\text{ri } K_2$.

Proof: Convexity and finiteness in (a) are immediate. A proof of the Lipschitz condition can be found in Wets [1972] or Kall [1976], who also give conditions for $\mathcal{Q}(x)$ to be differentiable. \square

Although many of the proofs of these results become intricate in general, the outcomes are relatively easy to apply.

When the random variables are appropriately described by a finite distribution, the constraint set K_2 is best defined by the possibility interpretation and is easily seen to be polyhedral. The second-stage recourse function $\mathcal{Q}(x)$ is piecewise linear and convex on K_2 . The decomposition techniques of Chapter 5 then apply. This is a category of programs for which computational methods can be made efficient, as we shall see.

When the random variables cannot be described by a finite distribution, they can usually be associated with some probability density. Many common probability densities are absolutely continuous and have finite second moments; so, the constraints set definitions K_2 and K_2^P coincide and the second-stage value function $\mathcal{Q}(x)$ is

differentiable and convex. Classical nonlinear programming techniques could then be applied. A typical example was given in the farmer's problem in Chapter 1. There, a convex differentiable function $\mathcal{Q}(x)$ was constructed analytically. It is easily understood that analytical expressions can reasonably be found only for small second-stage problems or problems with a very specific structure such as separability.

In general, one can only compute $\mathcal{Q}(x)$ by numerical integration of $Q(x, \xi)$, for a given value of x . Most nonlinear techniques would also require the gradients of $\mathcal{Q}(x)$, which in turn require numerical integration. An introduction to numerical integration appears in Chapter 8. From there, we come to the conclusion that numerical integration, as of today, produces an effective computational method only when the random vector is of small dimensionality. As a consequence, the practical solution of stochastic programs having continuous random variables is, in general, a difficult problem. One line of approach is to approximate the random variable by a discrete one and let the discretization be finer and finer, hoping that the solutions of the successive problems with discrete random variables will converge to the optimal solution of the problem with a continuous random variable. This is also discussed in Chapter 8. It is sufficient at this point to observe that approximation is a second reason for constructing efficient methods for stochastic programs with finite random variables.

d. Special cases: relatively complete, complete, and simple recourse

The previous sections presented properties for general problems. In particular instances, the feasible regions and objective values have special properties that are particularly useful in computation. One advantage can be obtained if every solution x that satisfies the first-period constraints, $Ax = b$, also has a feasible completion in the second stage. In other words, $K_1 \subset K_2$. In this case, we say that the stochastic program has *relatively complete recourse*. If, for the example with stochastic W in Section 3.1b., we had the first-period constraints $x \leq 1$, then this problem would have relatively complete recourse.

Although relatively complete recourse is very useful in practice and in many of the theoretical results that follow, it may be difficult to identify because it requires some knowledge of the sets K_1 and K_2 . A special type of relatively complete recourse may, however, often be identified from the structure of W . This form, called *complete recourse*, holds when there exists $y \geq 0$ such that $Wy = t$ for all $t \in \mathbb{R}^{m_2}$.

Complete recourse is also represented by $\text{pos}W = \mathbb{R}^{m_2}$ (the positive cone spanned by the columns of W includes \mathbb{R}^{m_2}), and says that W contains a positive linear basis of \mathbb{R}^{m_2} . Complete recourse is often added to a model to ensure that no outcome can produce infeasible results. With most practical problems, this should be the case. In some instances, complete recourse may not be apparent. An algorithm in Wets and Witzgall [1967] can be used in this situation to determine whether W contains a positive linear basis.

A special type of complete recourse offers additional computational advantages to stochastic programming solutions. This case is the generalization of the news vendor problem introduced in Section 3.1. It is called *simple recourse*. For a simple recourse problem, $W = [I, -I]$, \mathbf{y} is divided correspondingly as $(\mathbf{y}^+, \mathbf{y}^-)$, and $\mathbf{q} = (\mathbf{q}^+, \mathbf{q}^-)$. Note that, in this case, the optimal values of $y_i^+(\omega), y_i^-(\omega)$ are determined purely by the sign of $h_i(\omega) - T_i(\omega)x$ provided that $\mathbf{q}_i^+ + \mathbf{q}_i^- \geq 0$ with probability one. This finiteness result is in the following theorem.

Theorem 7. Suppose the two-stage stochastic program in (1.1) is feasible and has simple recourse and that ξ has finite second moments. Then $\mathcal{Q}(x)$ is finite if and only if $\mathbf{q}_i^+ + \mathbf{q}_i^- \geq 0$ with probability one.

Proof: If $q_i^+(\omega) + q_i^-(\omega) < 0$ for $\omega \in \Omega_1$ where $P(\Omega_1) > 0$, then, for any feasible x in (1.1), for all $\omega \in \Omega_1$ where $h_i(\omega) - T_i(\omega)x > 0$, let $y_i^+(\omega) = h_i(\omega) - T_i(\omega)x + u$, $y_i^-(\omega) = u$. By letting $u \rightarrow \infty$, $Q(x, \omega) \rightarrow -\infty$. A similar argument applies if $h_i(\omega) - T_i(\omega)x \leq 0$, so $\mathcal{Q}(x)$ is not finite.

If $\mathbf{q}_i^+ + \mathbf{q}_i^- \geq 0$ with probability one, then $Q(x, \omega) = \sum_{i=1}^{m_2} (q_i^+(\omega)(h_i(\omega) - T_i(\omega)x)^+ + q_i^-(\omega)(-h_i(\omega) + T_i(\omega)x)^+)$, which is finite for all ω . Using Proposition 2, we obtain the result. \square

We, therefore, assume that $\mathbf{q}_i^+ + \mathbf{q}_i^- \geq 0$ with probability one and can write $\mathcal{Q}(x)$ as $\sum_{i=1}^{m_2} \mathcal{Q}_i(x)$, where $\mathcal{Q}_i(x) = E_\omega[Q_i(x, \xi(\omega))]$, and

$$Q_i(x, \xi(\omega)) = q_i^+(\omega)(h_i(\omega) - T_i(\omega)x)^+ + q_i^-(\omega)(-h_i(\omega) + T_i(\omega)x)^+.$$

When q and T are fixed, this characterization of \mathcal{Q} allows its expression as a separable function in the remaining random components \mathbf{h}_i . Often, in this case, $T_i x$ is substituted with χ_i and Ψ is substituted for \mathcal{Q} so that $\mathcal{Q}(x) = \Psi(\chi)$. We then obtain $\Psi(\chi) = \sum_{i=1}^{m_2} \Psi_i(\chi_i)$ where $\Psi_i(\chi) = E_{\mathbf{h}_i}[\psi_i(\chi_i, \mathbf{h}_i)]$ and $\psi_i(\chi_i, h_i) = q_i^+(h_i - \chi_i)^+ + q_i^-(h_i + \chi_i)^+$. We, however, continue to use $\mathcal{Q}(x)$ to maintain consistency with our previous results.

We can define the objective function even further. In this case, let \mathbf{h}_i have an associated distribution function F_i , mean value \bar{h}_i , and let $q_i = q_i^+ + q_i^-$. We can then write $\mathcal{Q}_i(x)$ as

$$\mathcal{Q}_i(x) = q_i^+ \bar{h}_i - (q_i^+ - q_i F_i(T_i x)) T_i x - q_i \int_{h_i \leq T_i x} h_i dF_i(h_i). \quad (1.9)$$

Of particular importance in optimization is the subdifferential of this function, which has the following simple form:

$$\partial \mathcal{Q}_i(x) = \{\pi(T_i)^T \mid -q_i^+ + q_i F_i^-(T_i x) \leq \pi \leq -q_i^+ + q_i F_i^+(T_i x)\}, \quad (1.10)$$

where $F_i^-(h) = \lim_{t \uparrow h} F_i(t)$ and $F_i^+(h) = \lim_{t \downarrow h} F_i(t) = F_i(h)$. These results can be used to obtain specific optimality conditions. These general conditions are the subject of the next part of this section.

e. Optimality conditions and duality

In this subsection, we consider optimality conditions for stochastic programs. Our goal in describing these conditions is to show the special conditions that can apply to stochastic programs and to show how stochastic programs may differ from other mathematical programs. In particular, we give the additional assumptions that guarantee necessary and sufficient conditions for two-stage stochastic linear programs. The following sections contain generalizations.

The deterministic equivalent problem in (1.2) provides the framework for optimality conditions, but several questions arise.

1. When is a solution to (1.2) attainable?
2. What form do the optimality conditions take and how can they be simplified?
3. What types of dual problems can be formulated to accompany (1.2) and do they obtain bounds on optimal values?
4. How stable is an optimal solution to (1.2) to changes in the parameters and distributions?

This subsection briefly describes answers to these questions. Further details are contained in Kall [1976], Wets [1974, 1990], and Dempster [1980]. Our aim is to give only the basic results that may be useful in formulating, solving, and analyzing practical stochastic programs.

From the previous section, supposing that ξ has finite second moments, we know that \mathcal{Q} is Lipschitzian. We can then apply a direct subgradient result. A question is, however, whether the solution of (1.2) can indeed be obtained, i.e., whether the optimal objective value is finite and attained by some value of x .

To see that this question is indeed relevant, consider the following example. Find

$$\inf\{\mathbb{E}_\xi[y^+(\xi)] \mid y^+(\xi), y^-(\xi) \geq 0, x + y^+(\xi) - y^-(\xi) = \xi, \text{ a.s.}\}, \quad (1.11)$$

where ξ is, for example, negative exponentially distributed on $[0, \infty)$. For any finite value of x , (1.11) has a positive value, but the infimum over x is zero.

The following theorem gives some sufficient conditions to guarantee that a solution to (1.2) exists. In the following, we use rc to denote the *recession cone*, $\{v \mid u + \lambda v \in S, \text{ for all } \lambda \geq 0 \text{ and } u \in S\}$ when applied to a set, S , and the recession value, $\sup_{x \in \text{dom } f} (f(x + v) - f(x))$ when applied to a proper convex function, f .

Theorem 8. *Suppose that the random elements ξ have finite second moments and one of the following:*

- (a) *the feasible region K is bounded; or*
- (b) *the recourse function \mathcal{Q} is eventually linear in all recession directions of K , i.e., $\mathcal{Q}(x + \lambda v) = \mathcal{Q}(x + \tilde{\lambda} v) + (\lambda - \tilde{\lambda})\text{rc } \mathcal{Q}(v)$ for some $\tilde{\lambda} \geq 0$ (dependent on x), all $\lambda \geq \tilde{\lambda}$, and some constant recession value, $\text{rc } \mathcal{Q}(v)$, for all v such that $x + \lambda v \in K$ for all $x \in K$ and $\lambda \geq 0$.*

Then, if problem (1.2) has a finite optimal value, it is attained for some $x \in \mathfrak{R}^n$.

Proof. The proof given (a) follows immediately by noting that the objective is convex and finite on K , which is compact by assumption. The only possibility for not attaining an optimum is, therefore, when the optimal value is only attained asymptotically. By (b), along any recession direction v , we must have $\text{rc } \mathcal{Q}(v) \geq 0$ for a finite value of $\mathcal{Q}(x + \lambda v)$. Hence, the optimal value must be attained. \square

As shown in Wets [1974], if T is fixed and Ξ is compact, the condition in (b) is obtained. In the exercises, we will show that (b) may not hold if either of these conditions is relaxed.

We now assume that an optimal solution can be attained as we would expect in most practical situations. For optimization, we would like to describe the characteristics of such points. The general deterministic equivalent form gives us the following result in terms of Karush-Kuhn-Tucker conditions.

Theorem 9. Suppose (1.2) has a finite optimal value. A solution $x^* \in K_1$, is optimal in (1.2) if and only if there exists some $\lambda^* \in \mathfrak{R}^{m_1}$, $\mu^* \in \mathfrak{R}_+^{n_1}$, $\mu^{*T} x^* = 0$, such that,

$$-c + A^T \lambda^* + \mu^* \in \partial \mathcal{Q}(x^*). \quad (1.12)$$

Proof: From the optimization of a convex function over a convex region (see, for example, Bazaraa and Shetty [1979, Theorem 3.4.3]), we have that $c^T x + \mathcal{Q}(x)$ has a subgradient η at x^* such that $\eta^T(x - x^*) \geq 0$ for all $x \in K_1$ if and only if x^* minimizes $c^T x + \mathcal{Q}(x)$ over K_1 . We can write the set, $\{\eta \mid \eta^T(x - x^*) \geq 0 \text{ for all } x \in K_1\}$, as $\{\eta \mid \eta = A^T \lambda + \mu, \text{ for some } \mu \geq 0, \mu^T x^* = 0\}$. Hence, the general optimality condition states that a nonempty intersection of $\{\eta \mid \eta = A^T \lambda + \mu, \text{ for some } \mu \geq 0, \mu^T x^* = 0\}$ and $\partial(c^T x^* + \mathcal{Q}(x^*)) = c + \partial \mathcal{Q}(x^*)$ is necessary and sufficient for the optimality of x^* . \square

This result can be combined with our previous results on simple recourse functions to obtain specific conditions for that problem as follows.

Corollary 10. Suppose (1.1) has simple recourse and a finite optimal value. Then $x^* \in K_1$ is optimal in (1.2) corresponding to this problem if and only if there exists some $\lambda^* \in \mathfrak{R}^{m_1}$, $\mu^* \in \mathfrak{R}_+^{n_1}$, $\mu^{*T} x^* = 0$, π_i^* such that $-(q_i^+ - q_i F_i^-(T_i x^*)) \leq \pi_i^* \leq -(q_i^+ - q_i F_i^+(T_i x^*))$ and

$$-c + A^T \lambda^* + \mu^* - (\pi^*)^T T = 0. \quad (1.13)$$

Proof: This is a direct application of (1.10) and Theorem 9. \square

Inclusion (1.12) suggests that a subgradient method or other nondifferentiable optimization procedure may be used to solve (1.2). While this is true, we note that finite realizations of the random vector lead to equivalent linear programs (although of large scale), while absolutely continuous distributions lead to a differentiable recourse function \mathcal{Q} .

Obviously if \mathcal{Q} is differentiable, we can replace $\partial\mathcal{Q}(x^*)$ with $\nabla\mathcal{Q}(x^*)$ to obtain:

$$c + \nabla\mathcal{Q}(x^*) = A^T\lambda^* + \mu^* \quad (1.14)$$

in place of (1.12). Possible algorithms based on convex minimization subject to linear constraints are then admissible.

The main practical possibilities for solutions of (1.2) then appear as examples of either large-scale linear programming or smooth nonlinear optimization. The chief difficulty is, however, in characterizing $\partial\mathcal{Q}$ because even evaluating this function is difficult. This evaluation is, however, decomposable into subgradients of the recourse function for each realization of ξ , which form the subdifferential set $\partial Q(x, \xi(\omega))$, where we interpret the subgradient elements as being defined with respect to the decision variables x .

Theorem 11. *If $x \in K$, then*

$$\partial\mathcal{Q}(x) = E_{\omega}\partial Q(x, \xi(\omega)) + N(K_2, x), \quad (1.15)$$

where $N(K_2, x) = \{v \mid v^T y \leq 0, \forall y \text{ such that } x + y \in K_2\}$, the normal cone to K_2 at x .

Proof: From the theory of subdifferentials of random convex functions with finite expectations (see, for example, Wets [1990, Proposition 2.11]),

$$\partial\mathcal{Q}(x) = E_{\omega}\partial Q(x, \xi(\omega)) + rc[\partial\mathcal{Q}(x)], \quad (1.16)$$

where again rc denotes the recession cone, $\{v \mid u + \lambda v \in \partial\mathcal{Q}(x), \text{ for all } \lambda \geq 0 \text{ and } u \in \partial\mathcal{Q}(x)\}$. This set is equivalently $\{v \mid y^T(u + \lambda v) + \mathcal{Q}(x) \leq \mathcal{Q}(x + y) \text{ for all } \lambda \geq 0 \text{ and } y\}$. Hence, $v \in rc[\partial\mathcal{Q}(x)]$ if and only if $y^T v \leq 0$ for all y such that $\mathcal{Q}(x + y) < \infty$. Because $K_2 = \{x \mid \mathcal{Q}(x) < \infty\}$, the result follows. \square

This theorem indeed provides the basis for the results on the differentiability of \mathcal{Q} . In the exercises, we illustrate more of the characteristics of optimal solutions. Also note that if the problem has relatively complete recourse, then, for any y such that $x + y \in K_1$, we must also have $x + y \in K_2$. Hence, $N(K_2, x) \subset N(K_1, x) = \{v \mid v = A^T\lambda + \mu, \mu^T x = 0, \mu \geq 0\}$. This yields the following corollary to Theorems 9 and 11.

Corollary 12. *If (1.2) has relatively complete recourse, a solution x^* is optimal in (1.2) if and only if there exists some $\lambda^* \in \mathbb{R}^{m_1}$, $\mu^* \in \mathbb{R}_+^{n_1}$, $\mu^{*T} x^* = 0$, such that*

$$-c + A^T\lambda^* + \mu^* \in E_{\omega}\partial Q(x, \xi(\omega)). \quad (1.17)$$

Corollary 12 provides the basis for a dual formulation as well. The first step is to identify $\partial Q(x, \xi(\omega))$ (Exercise 10) as follows:

$$E_{\omega}\partial Q(x, \xi(\omega)) = \{-E[\pi T]|\pi^T W \leq \mathbf{q}^T,$$

$$\pi^T(\mathbf{h} - \mathbf{T}x) \geq (\boldsymbol{\pi}')^T(\mathbf{h} - \mathbf{T}x), \forall (\boldsymbol{\pi}')^T W \leq \mathbf{q}^T \text{ a.s.} \}. \quad (1.18)$$

Given this form of the subgradient, an equivalent dual program to (1.2) under the relatively complete recourse assumption can be obtained (Exercise 11) by solving the following maximization problem:

$$\begin{aligned} \max v &= b^T \lambda + E_{\omega}[h(\omega)^T \boldsymbol{\pi}(\omega)] \\ \text{s. t. } &A^T \lambda + E_{\omega}[T(\omega)^T \boldsymbol{\pi}(\omega)] \leq c, \\ &W^T \boldsymbol{\pi}(\omega) \leq q(\omega), \text{ a.s.} \end{aligned} \quad (1.19)$$

f. Stability and nonanticipativity

Another practical concern is whether the optimal solution set is also stable, i.e., whether it changes continuously in some sense when parameters of the problem change continuously. Although this may be of concern when considering changing problem conditions, we do not develop this theory in detail. The main results are that stability is achieved (i.e., some optimal solution of an original problem is close to some optimal solution of a perturbed problem) if problem (1.2) has complete recourse and the set of recourse problem dual solutions, $\{\boldsymbol{\pi} \mid \boldsymbol{\pi}^T W \leq q(\omega)^T\}$, is nonempty with probability one. For further details, we refer to Robinson and Wets [1987] and Römisch and Schultz [1991b].

Another approach to optimality conditions is to consider problem (1.2), in which $y(\omega)$ again becomes an explicit part of the problem and the nonanticipativity constraints also become explicit. The advantage in this representation is that we may obtain information on the value of future information. It also leads naturally to algorithms based on relaxing nonanticipativity.

We discuss the main results in this characterization briefly. The following development assumes some knowledge of measure theory and can be skipped by those unfamiliar with these concepts.

In general, for this approach, we wish to have a different x, y pair for every realization of the random outcomes. We then wish to restrict the x decisions to be the same for almost all outcomes. This says that the decision, $(x(\omega), y(\omega))$, is a function (with suitable properties) on Ω . We restrict this to some space, X , of measurable functions on Ω , for example, the p -integrable functions, $L_p(\Omega, \mathcal{B}, \mu; \mathbb{R}^n)$, for some $1 \leq p \leq \infty$. (For background on these concepts, see, for example, Royden [1968].) The general version of (1.2) (with certain restrictions) then becomes:

$$\inf_{(x(\omega), y(\omega)) \in X} \int_{\Omega} (c^T x(\omega) + q(\omega)^T y(\omega)) \mu(d\omega)$$

$$\begin{aligned}
\text{s. t.} \quad & Ax(\omega) = b, & a.s., \\
& E_{\Omega}(x(\omega)) - x(\omega) = 0, & a.s., \\
& T(\omega)x(\omega) + Wy(\omega) = h(\omega), & a.s., \\
& x(\omega), y(\omega) \geq 0, & a.s.
\end{aligned} \tag{1.20}$$

Problem (1.20) is equivalent to (1.2) if, for example, X is the space of essentially bounded functions on Ω and K is bounded for (1.2). The two formulations are not necessarily the same, however, as in the problem given in Exercise 12.

The condition that the x decision is taken before realizing the random outcomes is reflected in the second set of constraints in (1.20). These constraints are again the nonanticipativity constraints, which imply that almost all $x(\omega)$ values are the same.

The only difference in optimality conditions of (1.20) from those of (1.12) is that we include explicit multipliers for the nonanticipativity constraints. For continuous distributions, these multipliers may, however, have a difficult representation unless (1.20) has relatively complete recourse. The difficulty is that we cannot guarantee boundedness of the multipliers and may not be able to obtain an integrable function to represent them. This difficulty is caused when future constraints restrict the set of feasible solutions at the first stage.

For finite distributions, (1.20) is, however, an implementable problem structure that is used in several algorithms discussed here. In this case, with K possible realizations of ξ with probabilities p^k , $k = 1, \dots, K$, the problem becomes:

$$\begin{aligned}
\inf_{(x^k, y^k), k=1, \dots, K} \quad & \sum_{k=1}^K p^k (c^T x^k + (q^k)^T y^k) \\
\text{s. t.} \quad & Ax^k = b, \quad k = 1, \dots, K, \\
& \sum_{j \neq k} p^j x^j + (p^k - 1)x^k = 0, \quad k = 1, \dots, K, \\
& T^k x^k + Wy^k = h^k, \quad k = 1, \dots, K, \\
& x^k, y^k \geq 0, \quad k = 1, \dots, K.
\end{aligned} \tag{1.21}$$

Notice that (1.21) almost completely decomposes into K separate problems for the K realizations. The only links are in the second set of constraints that impose nonanticipativity. An aim of computation is to take advantage of this structure.

Consider the optimality conditions for (1.19). We wish to illustrate the difficulties that may occur when continuous distributions are allowed. A solution (x^{k*}, y^{k*}) , $k = 1, \dots, K$, is optimal for (1.21) if and only if there exist $(\lambda^{k*}, \rho^{k*}, \pi^{k*})$ such that

$$\begin{aligned}
p^k (c_j - \lambda^{k*T} a_{.j} - \sum_{l \neq k} p^l \rho_j^{l*} - (-1 + p^k) \rho_j^{k*} - \pi^{k*T} T_j^k) & \geq 0, \\
k = 1, \dots, K, \quad j = 1, \dots, n_1,
\end{aligned} \tag{1.22}$$

$$(c_j - \lambda^{k*T} a_{\cdot j} - \sum_{l \neq k} p^l \rho_j^{l*} - (-1 + p^k) \rho_j^{k*} - \pi^{*T} T_{\cdot j}^k) x_j^{k*} = 0, \\ k = 1, \dots, K, \quad j = 1, \dots, n_1, \quad (1.23)$$

$$p^k (q_j^k - \pi^{*T} W_{\cdot j}) \geq 0, \quad k = 1, \dots, K, \quad j = 1, \dots, n_2, \quad (1.24)$$

$$p^k (q_j^k - \pi^{*T} W_{\cdot j}) y_j^{k*} = 0, \quad k = 1, \dots, K, \quad j = 1, \dots, n_2, \quad (1.25)$$

where we have effectively multiplied the constraints in (1.19) by p^k to obtain the form in (1.22)–(1.25). We may also add the condition,

$$\sum_{k=1, \dots, K} p^k \rho^{k*} = 0, \quad (1.26)$$

without changing the feasibility of (1.22)–(1.25). This is true because, if $\sum_{k=1, \dots, K} p^k \rho^{k*} = \kappa$ for some $\kappa \neq 0$ is part of a feasible solution to (1.22)–(1.25), then so is $\rho^{k'} = \rho^{k*} - \kappa$. A problem arises if more realizations are included in the formulation (i.e., K increases) and $\rho^{k'}$ becomes unbounded.

To see how the multipliers may become unbounded, consider the following example (see also Rockafellar and Wets [1976a]). We wish to find $\min_x \{x \mid x \geq 0, x - \mathbf{y} = \boldsymbol{\xi}, a.s., \mathbf{y} \geq 0\}$, where $\boldsymbol{\xi}$ is uniformly distributed on k/K for $k = 0, \dots, K-1$ and $K \geq 2$. In this case, the optimal solution is $x^* = \frac{K-1}{K}$ and $y^{k*} = \frac{K-1-k}{K}$ for $k = 0, \dots, K$. The multipliers satisfying (1.22)–(1.26) are $\rho^{k*} = 1$, $\pi^{k*} = 0$ for $k = 0, \dots, K-2$, and $\rho^{K-1*} = -(K-1)$ and $\pi^{K-1*} = -K+2$. Note that as K increases, ρ^* approaches a distribution with a singular value at one. The difficulty is that ρ^{K-1*} is unbounded so that bounded convergence cannot apply. If relatively complete recourse is assumed, however, then all elements of ρ^* are bounded (see Exercise 13). No singular values are necessary.

In this example, the continuous distribution would tend toward a singular multiplier for some value of ω (i.e., a multiplier with mass one at a single point). If this is the case, we must have that the solution to the dual of the recourse problem is unbounded, or the recourse problem is infeasible for x^* feasible in the first stage. This possibility is eliminated by imposing the relatively complete recourse assumption.

With relatively complete recourse, we can state the following optimality conditions for a solution $(x^*(\omega), y^*(\omega))$ to (1.19). The theorem appears in other ways in Hiriart-Urruty [1978], Rockafellar and Wets [1976a, 1976b], Birge and Qi [1993], and elsewhere. We only note that regularity conditions (other than relatively complete recourse) follow from the linearity of the constraints.

Theorem 13. Assuming that (1.20) with $X = \mathcal{L}_\infty(\Omega, \mathcal{B}, \mu; \mathbb{R}^{n_1+n_2})$ is feasible, has a bounded optimal value, and satisfies relatively complete recourse, a solution $(x^*(\omega), y^*(\omega))$ is optimal in (1.20) if and only if there exist integrable functions on Ω , $(\lambda^*(\omega), \rho^*(\omega), \pi^*(\omega))$, such that

$$c_j - \lambda^*(\omega)A_{j.} - \rho^*(\omega) - \pi^{*T}(\omega)T_{j.}(\omega) \geq 0, \text{ a.s., } j = 1, \dots, n_1, \quad (1.27)$$

$$(c_j - \lambda^*(\omega)A_{j.} - \rho^*(\omega) - \pi^{*T}(\omega)T_{j.}(\omega))x_j^*(\omega) = 0, \\ \text{a.s., } j = 1, \dots, n_1, \quad (1.28)$$

$$q_j(\omega) - \pi^{*T}(\omega)W_{j.} \geq 0, \quad \text{a.s., } j = 1, \dots, n_2, \quad (1.29)$$

$$(q_j(\omega) - \pi^{*T}(\omega)W_{j.})y_j^*(\omega) = 0, \quad \text{a.s., } j = 1, \dots, n_2, \quad (1.30)$$

and

$$\mathbb{E}_\omega[\rho^*(\omega)] = 0. \quad (1.31)$$

Proof: We first show the sufficiency of these conditions directly. If (1.27)–(1.31) are satisfied, then for any $(x(\omega), y(\omega))$ (with expected value (x, y)) such that $(x^*(\omega) + x(\omega), y^*(\omega) + y(\omega))$ is feasible in (1.20), then integrating over ω , summing over j in (1.28), and using (1.29), we obtain that $c^T x - \mathbb{E}_\omega[\pi^{*T}(\omega)T(\omega)]x \geq 0$. We also have that $q(\omega)^T y(\omega) \geq \pi^{*T}(\omega)W y(\omega) = -\pi^{*T}(\omega)T(\omega)x$. Hence, $c^T x + \mathbb{E}_\omega[q(\omega)^T y(\omega)] \geq 0$, giving the optimality of $(x^*(\omega), y^*(\omega))$.

For necessity, we use the equivalence of (1.20) and (1.2), and Corollary 12. In this case, let λ^* from (1.12) replace $\lambda^*(\omega)$ in (1.27). Let $\pi^*(\omega)$ be the optimal dual value in the recourse problem in (1.4). Thus, $\mathbb{E}_\omega[\partial Q(x^*, \xi(\omega))] = \mathbb{E}_\omega[-\pi^{*T}(\omega)T(\omega)]$. Now, if we let $\rho^*(\omega) = \mathbb{E}_\omega[-\pi^{*T}(\omega)T] - \pi^{*T}(\omega)T(\omega)$, we obtain all the conditions in (1.27)–(1.31). \square

The results in this section give conditions that can be useful in algorithms and in checking the optimality of stochastic programming solutions. Dual problems similar to (1.18) can also be formulated based on these conditions either to obtain bounds on optimal solutions by finding corresponding feasible dual solutions or to give an alternative solution procedure that can be used directly or in some combined primal-dual approach (see, for example, Bazaraa and Shetty [1979]). The dual problem directly obtained from (1.27)–(1.31) is to find $(\lambda(\omega), \rho(\omega), \pi(\omega))$ on the dual space to X to maximize

$$\mathbb{E}_\omega[b^T \lambda(\omega) + h(\omega)^T \pi(\omega)] \quad \text{subject to} \quad (1.32)$$

$$A^T \lambda(\omega) + \rho(\omega) + T(\omega)^T \pi(\omega) \leq c, \quad \text{a.s.,} \quad (1.33)$$

$$W^T \pi(\omega) \leq q(\omega), \quad \text{a.s.,} \quad (1.34)$$

and

$$\mathbb{E}_\omega[\rho(\omega)] = 0. \quad (1.35)$$

This fits the general duality framework used by Klein Haneveld [1985] where further details on the properties of these dual problems may be found. Rockafellar and Wets [1976a, 1976b] also discuss this alternative viewpoint with an analysis based on perturbations of both primal and dual forms. Discussion of alternative dual spaces appears in Eisner and Olsen [1975]. In general, Problem (1.20) attains its minimum with a bounded region, and the supremum in (1.32)–(1.35) gives the same value. Relatively complete recourse, or a similar requirement, is necessary to obtain that the dual optimum is also attained. With unbounded regions or without relatively complete recourse, as we have seen, we may have that an optimal solution is not attained for either (1.21) or (1.32)–(1.35). In this case, it is possible that the corresponding dual problem does not have the same optimal value and the two problems exhibit a duality gap. The exercises explore this possibility further.

Exercises

1. Consider Example 1 with a second-stage program defined as

$$\begin{aligned} & \min 2y_1 + y_2 \\ \text{s. t. } & y_1 + 2y_2 \geq \xi_1 - x_1 , \\ & y_1 + y_2 \geq \xi_2 - x_1 - x_2 , \\ & 0 \leq y_1 \leq 1 , \quad 0 \leq y_2 \leq 1 . \end{aligned}$$

We have seen that $K_2(\xi) = \{x \mid x_1 \geq \xi_1 - 3 , x_1 + x_2 \geq \xi_2 - 2\}$. Let ξ_1 and ξ_2 be two independent continuous random variables. Assume they both have uniform density over $[2, 4]$.

- (a) What is K_2^P ?
 - (b) What is K_2 ?
 - (c) Let u_i^* be defined as in (1.7). What are u_1^* and u_2^* in this example?
2. Let the second stage of a stochastic program be

$$\begin{aligned} & \min 2y_1 + y_2 \\ \text{s. t. } & y_1 - y_2 \leq 2 - \xi x_1 , \\ & y_2 \leq x_2 , \\ & 0 \leq y_1, y_2 . \end{aligned}$$

Find $K_2(\xi)$ and K_2 for:

- $\xi \sim U[0, 1]$.
- $\xi \sim \text{Poisson}(\lambda)$, $\lambda > 0$.

What properties do you expect for K_2 ?

3. Consider the following second-stage program:

$$Q(x, \xi) = \min\{y \mid y \geq \xi, y \geq x\}.$$

For simplicity, assume $x \geq 0$.

Let ξ have density

$$f(\xi) = \frac{2}{\xi^3}, \xi \geq 1.$$

Show that $K_2^P = K_2$. Compare this with the statement of Theorem 3.

4. Consider Example 2 where the second-stage program is defined as

$$\begin{aligned} & \min 2y_1 + y_2 \\ \text{s. t. } & y_1 + y_2 \geq 1 - x_1, \\ & y_1 \geq \xi - x_1 - x_2, \\ & y_1, y_2 \geq 0, \end{aligned}$$

where $\Xi \subset \Re^+$.

- (a) Show that this program has complete recourse if ξ has finite expectation.
- (b) Show that $Q(x, \xi)$ is convex in x and convex in ξ .
- (c) Assume $\xi \sim U[0, 2]$. After a tedious integration that probably only the authors of this book will go through, one obtains $\mathcal{Q}(x) = \frac{1}{4}(x_1^2 + 2x_2^2 + 2x_1x_2 - 8x_1 - 6x_2 + 9)$. Check that the relevant properties of Theorem 6 are satisfied.

5. Let a second-stage program be defined as

$$\begin{aligned} & \min \xi y_1 + y_2 \\ \text{s. t. } & y_1 + y_2 \geq 1 - x_1, \\ & y_1 \geq 1 - x_1 - x_2, \\ & y_1, y_2 \geq 0. \end{aligned}$$

Assume $0 \leq x_1, x_2 \leq 1$. Obtain $Q(x, \xi)$ and observe that it is concave in ξ .

- 6. Prove the positive homogeneity property in (1.8).
- 7. Derive the simple recourse results in (1.9) and (1.10).
- 8. Show that the news vendor problem is a special case of a simple recourse problem.
- 9. Consider the following example:

$$\begin{aligned} & \min -x + E_{(t(\omega), h(\omega))}[y^+(\omega) + y^-(\omega)] \\ \text{s. t. } & t(\omega)x + y^+(\omega) - y^-(\omega) = h(\omega), \quad a.s., \\ & x, y^+(\omega), y^-(\omega) \geq 0, \quad a.s., \end{aligned}$$

where \mathbf{h}, \mathbf{t} are uniformly distributed on the unit circle, $h^2 + t^2 \leq 1$. Find $\mathcal{Q}(x)$ and show that it is not eventually linear for $x \rightarrow \infty$ (Wets [1974]).

10. Show that $E_\omega \partial Q(x, \xi(\omega))$ is given by (1.18).
11. Show that an optimal solution $(\lambda^*, \pi^*(\omega))$ to the dual program in (1.19) provides a solution to the optimality conditions in (1.17) using (1.18) and that the optimal objective value v^* is the same as the optimal value z^* in (1.2).
12. Suppose you wish to solve (1.11) in the form of (1.20) over $(x(\omega), y(\omega)) \in \mathcal{L}_\infty(\Omega, \mathcal{B}, \mu : \mathfrak{R}^{n_1+n_2})$. What is the optimal value? How does this differ from using (1.2)?
13. This exercise uses approximation results to give an alternative proof of Theorem 13. As shown in Chapter 8, if a discrete distribution approaches a continuous distribution (in distribution) and problem (1.2) has a bounded optimal solution and the bounded second moment property, then a limiting optimal solution for the discrete distributions is an optimal solution using the continuous distribution. This also implies that recourse solutions, y^* , converge and that the optimality conditions in (1.27)–(1.31) are obtained as long as the ρ^{k*} in the discrete approximations are uniformly bounded. Show that relatively complete recourse implies uniform boundedness of some ρ^{k*} for any discrete approximation approaching a continuous distribution in (1.18). (Hint: Construct a system of equations that must be violated for some iteration v of the discretization and for any bound M on the largest value of ρ^{k*} if the ρ^{k*} are not uniformly bounded. Then show that the complementary system implies no relatively complete recourse.)

3.2 Probabilistic or Chance Constraints

a. General case

As mentioned in Chapter 2, in some models, constraints need not hold *almost surely* as we have assumed to this point. They can instead hold with some probability or reliability level. These *probabilistic*, or *chance*, constraints take the form:

$$P\{A^i(\omega)x \geq h^i(\omega)\} \geq \alpha^i, \quad (2.1)$$

where $0 < \alpha^i < 1$ and $i = 1, \dots, I$ is an index of the constraints that must hold jointly. We can, of course, model these constraints in a general expectational form $E_\omega(f^i(\omega, x(\omega))) \geq \alpha^i$ where f^i is an indicator of $\{\omega \mid A^i(\omega)x \geq h^i(\omega)\}$ but we would then have to deal with a discontinuous function.

In chance-constrained programming (see, e.g., Charnes and Cooper [1963]), the objective is often an expectational functional as we used earlier (the *E-model*), or it may be the variance of some result (the *V-model*) or the probability of some occurrence (such as satisfying the constraints) (the *P-model*).

Another variation includes an objective that is a quantile of a random function (see, e.g., Kibzun and Kurbakovskiy [1991] and Kibzun and Kan [1996]).

The main results with probabilistic constraints refer to forms of deterministic equivalents for constraints of the form in (2.1). Provided the deterministic equivalents of these constraints and objectives have the desired convexity properties, these functions can be added to the recourse problems given earlier (or used as objectives). In this way, all our previous results apply to chance-constrained programming with suitable function characteristics.

The main goal in problems with probabilistic constraints is, therefore, to determine deterministic equivalents and their properties. To maintain consistency with the recourse problem results, we let

$$K_1^i(\alpha^i) = \{x \mid P(A^i(\omega)x \geq h^i(\omega)) \geq \alpha^i\}, \quad (2.2)$$

where $0 < \alpha^i \leq 1$ and $\bigcap_i K_1^i(1) = K_1$ as in Section 3.1. Unfortunately, $K_1^i(\alpha^i)$ need not be convex or even connected. Suppose, for example that $\Omega = \{\omega_1, \omega_2\}$, $P[\omega_1] = P[\omega_2] = \frac{1}{2}$,

$$\begin{aligned} A^i(\omega_1) &= A^i(\omega_2) = \begin{pmatrix} 1 \\ -1 \end{pmatrix} \\ h^i(\omega_1) &= \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\ h^i(\omega_2) &= \begin{pmatrix} 2 \\ -3 \end{pmatrix} \end{aligned} \quad (2.3)$$

for $0 < \alpha^i \leq \frac{1}{2}$, $K_1^i(\alpha^i) = [0, 1] \cup [2, 3]$.

When each i corresponds to a distinct linear constraint and A^i is a fixed row vector, then obtaining a deterministic equivalent of (2.2) is fairly straightforward. In this case, $P(A^i x \geq h^i(\omega)) = F^i(A^i x)$, where F^i is the distribution function of h^i . Hence, $K_1^i(\alpha^i) = \{x \mid F^i(A^i x) \geq \alpha^i\}$, which immediately yields a deterministic equivalent form. In general, however, the constraints must hold jointly so that the set I is a singleton. This situation corresponds to requiring an α -confidence interval that x is feasible. We assume this in the remainder of this section and drop the superscript i indicating the set of joint constraints.

The results to determine the deterministic equivalent often involve manipulations of probability distributions that use measure theory. The remainder of this section is intended for readers familiar with this area. One of the main results in probabilistic constraints is that, in the joint constraint case, a large class of probability measures on $h(\omega)$ (for A fixed) leads to convex and closed $K_1(\alpha)$. A probability measure P is in this class of *quasi-concave* measures if for any convex measurable sets U and V and any $0 \leq \lambda \leq 1$,

$$P((1-\lambda)U + \lambda V) \geq \min\{P(U), P(V)\}. \quad (2.4)$$

The use of this and a special form, called *logarithmically concave measures*, began with Prékopa [1971, 1973]. General discussions also appear in Prékopa [1980, 1995], Kallberg and Ziemba [1983] concerning related utility functions, and the surveys of Wets [1983b, 1990] which include the following theorem.

Theorem 14. *Suppose A is fixed and \mathbf{h} has an associated quasi-concave probability measure P . Then $K_1(\alpha)$ is a closed convex set for $0 \leq \alpha \leq 1$.*

Proof: Let $\mathcal{H}(x) = \{h \mid Ax \geq h\}$. Suppose $x(\lambda) = \lambda x^1 + (1 - \lambda)x^2$ where $x^1, x^2 \in K_1(\alpha)$. Suppose $h^1 \in \mathcal{H}(x^1)$ and $h^2 \in \mathcal{H}(x^2)$. Then $\lambda h^1 + (1 - \lambda)h^2 \leq Ax(\lambda)$, so $\mathcal{H}(x(\lambda)) \supset \lambda \mathcal{H}(x^1) + (1 - \lambda)\mathcal{H}(x^2)$. Hence, $P(\{Ax(\lambda) \geq \mathbf{h}\}) = P(\mathcal{H}(x(\lambda))) \geq P(\lambda \mathcal{H}(x^1) + (1 - \lambda)\mathcal{H}(x^2)) \geq \alpha$. Thus, $K_1(\alpha)$ is convex.

For closure, suppose that $x^v \rightarrow \bar{x}$, where $x^v \in K_1(\alpha)$. Consider $\mathcal{H}(x^v)$. If $h \leq Ax^{v_i}$ for some subsequence $\{v_i\}$ of $\{v\}$, then $h \leq A\bar{x}$. Hence $\limsup_v \mathcal{H}(x^v) \subset \mathcal{H}(\bar{x})$, so $P(\mathcal{H}(\bar{x})) \geq P(\limsup_v \mathcal{H}(x^v)) \geq \limsup_v P(\mathcal{H}(x^v)) \geq \alpha$. \square

The relevance of this result stems from the large class of probability measures which fit these conditions. Some extent of this class is given in the following result of Borell [1975], which we state without proof.

Theorem 15. *If f is the density of a continuous probability distribution in \mathbb{R}^m and $f^{-(\frac{1}{m})}$ is convex on \mathbb{R}^m , then the probability measure*

$$P(B) = \int_B f(x) dx,$$

defined for all Borel sets B in \mathbb{R}^m is quasi-concave. \square

In particular, this result states that any density of the form $f(x) = e^{-l(x)}$ for some convex function l yields a quasi-concave probability measure. These measures include the multivariate normal, beta, and Dirichlet distributions and are logarithmically concave (because, for $0 \leq \lambda \leq 1$, $P((1 - \lambda)U + \lambda V) \geq P(U)^{\lambda}P(V)^{1-\lambda}$ for all Borel sets U and V) as studied by Prékopa. These distributions lead to computable deterministic equivalents as, for example, in the following theorem.

Theorem 16. *Suppose A is fixed and the components $\mathbf{h}_i, i = 1, \dots, m_1$, of \mathbf{h} are stochastically independent random variables with logarithmically concave probability measures, P_i , and distribution functions, F_i , then $K_1(\alpha) = \{x \mid \sum_{i=1}^{m_1} \ln(F_i(A_i.x)) \geq \ln \alpha\}$ and is convex.*

Proof: From the independence assumption, $P[Ax \geq \mathbf{h}] = \prod_{i=1}^{m_1} P_i[A_i.x \geq \mathbf{h}_i] = \prod_{i=1}^{m_1} F_i(A_i.x)$. So, $K_1(\alpha) = \{x \mid \prod_{i=1}^{m_1} F_i(A_i.x) \geq \alpha\}$. Taking logarithms (which is a monotonically increasing function), we obtain $K_1(\alpha) = \{x \mid \sum_{i=1}^{m_1} \ln(F_i(A_i.x)) \geq \ln \alpha\}$. Because

$$\begin{aligned} F_i(A_i.(\lambda x^1 + (1 - \lambda)x^2)) &= P_i(\mathbf{h}_i \leq A_i.(\lambda x^1 + (1 - \lambda)x^2)) \\ &\geq P_i(\lambda \{\mathbf{h}_i \leq A_i.x^1\} + (1 - \lambda)\{\mathbf{h}_i \leq A_i.x^2\}) \end{aligned}$$

$$\begin{aligned} &\geq P_i(\{\mathbf{h}_i \leq A_{i \cdot} x^1\})^\lambda P_i(\{\mathbf{h}_i \leq A_{i \cdot} x^2\})^{1-\lambda} \\ &= F_i(A_{i \cdot} x^1)^\lambda F_i(A_{i \cdot} x^2)^{1-\lambda}, \end{aligned}$$

the logarithm of $F_i(A_{i \cdot} x)$ is a concave function, and $K_1(\alpha)$ is convex. \square

Logarithmically concave distribution functions include the increasing failure rate functions (see Miller and Wagner [1965] and Parikh [1968]) that are common in reliability studies. Other types of quasi-concave measures include the multivariate t and F distributions. Because these distributions include those most commonly used in multivariate analysis, it appears that, with continuous distributions and fixed A , the convexity of the solution set is generally assured.

When A is also random, the convexity of the solution set is, however, not as clear. The following theorem from Prékopa [1974], given without proof, shows this result for normal distributions with fixed covariance structure across columns of A and h .

Theorem 17. *If $\mathbf{A}_1, \dots, \mathbf{A}_{n_1}, \mathbf{h}$ have a joint normal distribution with a common covariance structure, a matrix C , such that $E[(\mathbf{A}_{i \cdot} - E(\mathbf{A}_{i \cdot}))(\mathbf{A}_{j \cdot} - E(\mathbf{A}_{j \cdot}))^T] = r_{ij}C$ for i, j in $1, \dots, n_1$, and*

$$E[(\mathbf{A}_{i \cdot} - E(\mathbf{A}_{i \cdot}))(\mathbf{h} - E(\mathbf{h}))] = s_i C$$

for $i = 1, \dots, n_1$, where r_{ij} and s_i are constants for all i and j , then $K_1(\alpha)$ is convex for $\alpha \geq \frac{1}{2}$. \square

Stronger results than Theorem 17 are difficult to obtain. In general, one must rely on approximations to the deterministic equivalent that maintain convexity although the original solution set may not be convex. We will consider some of these approximations in Chapter 8.

Some other specific examples where A may be random include single constraints (see Exercise 5). In the case of $h \equiv 0$ and normally distributed A , the deterministic equivalent is again readily obtainable as in the following from Parikh [1968].

Theorem 18. *Suppose that $m_1 = 1$, $h_1 = 0$, and $\mathbf{A}_{1 \cdot}$ has mean $\bar{\mathbf{A}}_{1 \cdot}$ and covariance matrix C_1 , then $K_1(\alpha) = \{x \mid \bar{\mathbf{A}}_{1 \cdot} x - \Phi^{-1}(\alpha) \sqrt{x^T C_1 x} \geq 0\}$, where Φ is the standard normal distribution function.*

Proof: Observe that $\mathbf{A}_{1 \cdot} x$ is normally distributed with mean, $\bar{\mathbf{A}}_{1 \cdot} x$, and variance, $x^T C_1 x$. If $x^T C_1 x = 0$, then the result is immediate. If not, then $\frac{\mathbf{A}_{1 \cdot} x - \bar{\mathbf{A}}_{1 \cdot} x}{\sqrt{x^T C_1 x}}$ is a standard normal random variable with cumulative Φ , and

$$\begin{aligned} P(\mathbf{A}_{1 \cdot} x \geq 0) &= P\left(\frac{\mathbf{A}_{1 \cdot} x - \bar{\mathbf{A}}_{1 \cdot} x}{\sqrt{x^T C_1 x}} \geq \frac{-\bar{\mathbf{A}}_{1 \cdot} x}{\sqrt{x^T C_1 x}}\right) \\ &= P\left(\frac{\mathbf{A}_{1 \cdot} x - \bar{\mathbf{A}}_{1 \cdot} x}{\sqrt{x^T C_1 x}} \leq \frac{\bar{\mathbf{A}}_{1 \cdot} x}{\sqrt{x^T C_1 x}}\right) \end{aligned}$$

$$= \Phi\left(\frac{\bar{A}_1 x}{\sqrt{x^T C_1 x}}\right).$$

Substitution in the definition of $K_1(\alpha)$ yields the result. \square

Finally in this chapter, we would like to show some of the similarities between models with probabilistic constraints and problems with recourse. As stated in Chapter 2, models with probabilistic constraints and models with recourse can often lead to the same optimal solutions. Some other aspects of the modeling process may favor one over the other (see, e.g., Hogan, Morris, and Thompson [1981, 1984], Charnes and Cooper [1983]), but, these differences generally just represent decision makers' different attitudes toward risk.

We use an example from Parikh [1968] to relate simple recourse and chance-constrained problems. Consider the following problem with probabilistic constraints:

$$\begin{aligned} \min \quad & c^T x \\ \text{s. t.} \quad & Ax = b, \\ & P_i[T_i x \geq \mathbf{h}_i] \geq \alpha_i, \quad i = 1, \dots, m_2, \\ & x \geq 0, \end{aligned} \tag{2.5}$$

where P_i is the probability measure of \mathbf{h}_i and T_i is the distribution function for \mathbf{h}_i . For the deterministic equivalent to (2.5), we just let $F_i(h_i^*) = \alpha_i$, to obtain:

$$\begin{aligned} \min \quad & c^T x \\ \text{s. t.} \quad & Ax = b, \\ & T_i x \geq h_i^*, \quad i = 1, \dots, m_2, \\ & x \geq 0. \end{aligned} \tag{2.6}$$

Suppose we solve (2.6) and obtain an optimal x^* and optimal dual solution $\{\lambda^*, \pi^*\}$, where $c^T x^* = b^T \lambda^* + h^{*T} \pi^*$. If $\pi_i^* = 0$, let $q_i^+ = 0$ and, if $\pi_i^* > 0$, let $q_i^+ = \frac{\pi_i^*}{1 - \alpha_i}$. An equivalent stochastic program with simple recourse to (2.5) is then:

$$\begin{aligned} \min \quad & c^T x + \mathbb{E}_{\mathbf{h}}[q^+ \mathbf{y}^+] \\ \text{s. t.} \quad & Ax = b, \\ & T_i x + \mathbf{y}_i^+ - \mathbf{y}_i^- = \mathbf{h}_i, \quad i = 1, \dots, m_2, \\ & x, \mathbf{y}^+, \mathbf{y}^- \geq 0. \end{aligned} \tag{2.7}$$

For problems (2.5) and (2.7) to be equivalent, we mean that any x^* optimal in (2.5) corresponds to some (x^*, \mathbf{y}^{*+}) optimal in (2.7) for a suitable definition of q^+ and that any (x^*, \mathbf{y}^{*+}) optimal in (2.7) corresponds to x^* optimal in (2.5) for a suitable definition of α_i . We show the first part of this equivalence in the following theorem.

Theorem 19. For the q_i^+ defined as a function of some optimal π^* for the dual to (2.5), if x^* is optimal in (2.5), there exists $y^{*+} \geq 0$ a.s. such that (x^*, y^{*+}) is optimal in (2.7).

Proof: First, let x^* be optimal in (2.5). It must also be optimal in (2.6) with dual variables, $\{\lambda^*, \pi^*\}$. We must have $\pi^* \geq 0$,

$$\begin{aligned} c^T - \lambda^{*T} A - \pi^{*T} T &\geq 0, \\ Tx^* - h^* &\geq 0, \\ (c^T - \lambda^{*T} A - \pi^{*T} T)x^* &= 0, \end{aligned}$$

and

$$\pi^{*T}(Tx^* - h^*) = 0. \quad (2.8)$$

Now, for x^* to be optimal in (2.7), consider the optimality conditions (1.13) from Corollary 10. These conditions state that if there exists λ^* such that

$$\begin{aligned} c^T - \lambda^{*T} A - \sum_{i=1}^{m_2} T_{i\cdot} (q_i^+ - q_i F_i(T_{i\cdot} x^*)) &\geq 0, \\ (c^T - \lambda^{*T} A - \sum_{i=1}^{m_2} T_{i\cdot} (q_i^+ - q_i F_i(T_{i\cdot} x^*)))x^* &= 0. \end{aligned} \quad (2.9)$$

Substituting for $\pi_i^* = q_i^+(1 - \alpha_i)$ in (2.8) and noting from the complementarity condition that $\alpha_i = F_i(h_i^*) = F_i(T_{i\cdot} x^*)$ if $\pi_i^* > 0$, we obtain

$$\begin{aligned} c^T - \lambda^{*T} A - \pi^{*T} T &= c^T - \lambda^{*T} A - \sum_{i=1}^{m_2} T_{i\cdot} (q_i^+(1 - F_i(T_{i\cdot} x^*))) \\ &= c^T - \lambda^{*T} A - \sum_{i=1}^{m_2} T_{i\cdot} (q_i^+ - q_i F_i(T_{i\cdot} x^*)) \end{aligned} \quad (2.10)$$

from the definitions and noting that $\pi_i^* > 0$ if and only if $q_i^+ > 0$. From (2.10), we can verify the conditions in (2.9) and obtain the optimality of x^* in (2.7). \square

If we assume x^* is optimal in (2.7), we can reverse the argument to show that x^* is also optimal in (2.5) for some value of α_i . This result (from Symonds [1968]) is Exercise 7. Further equivalences are discussed in Gartska [1980]. We note that all of these equivalences are somewhat weak because they require a priori knowledge of the optimal solution to one of the problems (see also the discussion in Gartska and Wets [1974]).

Exercises

1. Suppose a single probabilistic constraint with fixed A and that \mathbf{h} has an exponential distribution with mean λ . What is the resulting deterministic equivalent constraint for $K_1(\alpha)$?
2. For the example in (2.3), what happens for $\frac{1}{2} < \alpha^i \leq 1$?
3. Can you construct an example with continuous random variables where $K_1(\alpha)$ is not connected? (Hint: Try a multimodal distribution such as a random choice of one of two bivariate normal random variables.)
4. Extend Theorem 14 to allow any set of convex constraints, $g_i(x, \xi(\omega)) \leq 0$, $i = 1, \dots, m$.
5. Suppose a single linear constraint in $K_1(\alpha)$ where the components of A and h have a joint normal distribution. Show that $K_1(\alpha)$ is also convex in this case for $\alpha \geq \frac{1}{2}$. (Hint: The random variable, $\mathbf{A}_1 x - \mathbf{h}_1$, is also normally distributed.)
6. Show that $\sqrt{x^T C_1 x}$ is a convex function of x .
7. Prove the converse of Theorem 19 by finding an appropriate α_i so that the x^* that is optimal in (2.7) is also optimal in (2.5).
8. Let $\bar{K}(\alpha) = \{x | P\{A(\omega)x \geq h\} \leq \alpha\}$, where $A(\omega)$ has a joint normal distribution as in Theorem 17 (and h is fixed). Show that, in contrast to the result of Theorem 17, $\bar{K}(\alpha)$ need not be convex for any $0 < \alpha < 1$.¹

b. Probabilistic constraints with discrete random variables

If $\xi(\omega)$ is a discrete random variable, there exists a finite number of *scenarios* which correspond to the realizations of ξ . They are represented as $\xi_1, \xi_2, \dots, \xi_K$. Scenario k has a probability p_k with $\sum_{k=1}^K p_k = 1$.

Scenarios can be obtained through experts' opinions. Another typical way to get scenarios is when the information over the random variables comes from historical data. The distribution of the random vector is then known as the empirical distribution.

Assume we have a constraint of the form

$$P\{g(x, y(\omega), \xi(\omega)) \leq 0\} \geq \alpha. \quad (2.11)$$

It is a joint probabilistic constraint as $g(\cdot) \leq 0$ may contain several constraints under a vector representation. This includes classical cases such as $g(x, y(\omega), \xi(\omega)) = h(\omega) - Ax$. This also includes cases where the probabilistic constraint depends on the recourse actions. Then $g(x, y(\omega), \xi(\omega)) = h(\omega) - T(\omega)x - W(\omega)y(\omega)$.

¹ This exercise was suggested by Yue Rong, University of California at Riverside.

Using the indicator function $\eta(a) = 0$ if $a \leq 0$ and 1 if at least one component of a is strictly positive, the probabilistic constraint is equivalent to

$$\sum_{k=1}^K p_k \eta(g(x, y_k, \xi_k)) \leq 1 - \alpha. \quad (2.12)$$

The left-hand side of (2.12) sums up the probability of the scenarios for which $g(\cdot) \leq 0$ is violated. Assume that for each scenario k , an upper bound vector u_k can be found such that $g(x, y_k, \xi_k) \leq u_k$ for all feasible x, y_k . Then, (2.12) can be transformed into

$$\sum_{k=1}^K p_k w_k \leq 1 - \alpha, \quad (2.13)$$

$$g(x, y_k, \xi_k) \leq u_k w_k, \quad k = 1, \dots, K, \quad (2.14)$$

$$w_k \in \{0, 1\}, \quad k = 1, \dots, K. \quad (2.15)$$

The binary variable w_k plays the role of the indicator function. When $g(x, y_k, \xi_k) \leq 0$, w_k takes the value 0 . When at least one component of $g(x, y_k, \xi_k)$ is strictly positive, then $w_k = 1$ and scenario k contributes p_k to the left-hand side in (2.13).

The joint probabilistic constraint (2.11) with a discrete random variable is transformed into a mixed integer programming (MIP) formulation. When $g(\cdot)$ is linear, the stochastic program with probabilistic constraint is transformed into a mixed integer linear program (MILP) and can be solved using your favorite MILP solver. We now provide two examples of how (2.11) is transformed into (2.13)–(2.15). We then give an introduction to reformulations of (2.13)–(2.15) that allow efficient solutions of large problems.

Example 3

Consider the example from Section 2.7a. We are asked to find the numbers x_1 and x_2 of seats in first and business class for a plane of 200 seats. As in (2.11), assume now a joint probabilistic constraint

$$P(x_1 \geq \xi_F, x_1 + x_2 \geq \xi_F + \xi_B) \geq 0.95, \quad (2.16)$$

where ξ_F and ξ_B represents the weekdays demands in first and business class. This corresponds to the classical case where $g(x, y(\omega), \xi(\omega)) = h(\omega) - Ax$, with $h(\omega)^T = (\xi_F, \xi_F + \xi_B)$ and $A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$.

Assume now the random variables (ξ_F, ξ_B) are given by the empirical data of last year. (These data must correspond to the number of calls and not to the number of passengers, which may depend on the acceptance policy at that time). This creates

an empirical distribution of 260 pairs (ξ_F, ξ_B) for each weekday of last year. Each of the 260 pairs is a scenario of probability $1/260$.

In (2.14), we need an upper bound on $\xi_F - x_1$ and on $\xi_F + \xi_B - x_1 - x_2$ for each k . Here, it suffices to take ξ_F and $\xi_F + \xi_B$, respectively. As an illustration, if scenario k has demands $(14, 32)$ in first and business, then the two corresponding constraints in (2.14) are

$$\begin{aligned} 14 - x_1 &\leq 14w_k, \\ 46 - x_1 - x_2 &\leq 46w_k. \end{aligned}$$

Thus, (2.16) is formulated using 260 binary variables w_k 's, one constraint (2.13) and 520 constraints in (2.14). To put it in more general terms, (2.16) is reformulated using K extra binary variables and $2K+1$ extra constraints.

Example 4

Consider the farmer in Section 1.1. The example was built assuming a discrete random variable with only three scenarios: good, fair, and bad. This number can easily be extended either in a similar manner or by taking past observations of the yields. We now assume K scenarios, each consisting of a vector of three yields.

The farmer finds it inappropriate to purchase large quantities of wheat and/or corn. He considers it excessive to purchase more than a total of 20 T. Owing to the uncertainty of mother nature, he allows for a 20% probability of excessive purchases. Thus, his probabilistic constraint is

$$P(y_1(\omega) + y_2(\omega) \leq 20) \geq 0.80 \quad (2.17)$$

where $y_1(\omega)$ and $y_2(\omega)$ are the purchases of wheat and corn, respectively.

Here is a case where the probabilistic constraint depends on the recourse actions under the general form $g(x, y(\omega), \xi(\omega)) = h(\omega) - T(\omega)x - W(\omega)y(\omega)$.

To obtain (2.14), we start from the representation of the constraint under scenario k as $-20 + y_1^k + y_2^k \leq 0$, where y_1^k and y_2^k represent the purchase of wheat and corn under scenario k . From Table 1 in Section 1.1, the total requirement of wheat and corn is 440. The upper bound to form (2.14) is the value 420, so that a single constraint of the form

$$y_1^k + y_2^k \leq 20 + 420w_k \quad (2.18)$$

is created. (If $y_1^k + y_2^k \leq 20$, then w_k is 0; otherwise, $w_k = 1$ and the constraint imposes no limit on the purchase of wheat and corn as the total cannot exceed 440.) The recourse problem with K scenarios and the extra probabilistic constraint (2.17) is reformulated as an MILP with K extra binary variables and $K+1$ extra constraints.

Improved formulation of a probabilistic constraint with discrete random variables

For large values of K , the MILP may become difficult to solve. This is due to the structure of (2.14). It is indeed a weak constraint on w_k . To see this, consider the example of (2.18).

Suppose that the total purchase under scenario k is 30. Then (2.18) is equivalent to $420w_k \geq 10$, or $w_k \geq 0.0238$. As (2.18) is the only constraint on w_k , integrality can only be recovered through branching. The MILP solver will have to branch on all nonzero binaries, and none of them is likely to be spontaneously 1. Moreover, after some w_k 's are fixed by branching, additional w_k 's may become fractional and require extra branching.

It is classical then to search for efficient *valid inequalities*. A valid inequality is a linear constraint added to the original formulation, which does not eliminate any integer solution but eliminates fractional solutions (see Appendix 2 of Chapter 7 for some examples). A valid inequality provides a reformulation of the problem that contains fewer fractional solutions but the same integer solutions.

To illustrate valid inequalities, we use the example of constraint (2.17) and its reformulation (2.18). As the probabilistic constraint only depends on corn and wheat, we may restrict our attention for this analysis to the first two components of the random vector.

We may say that scenario k *dominates* scenario j if $\xi^k \geq \xi^j$, where the inequality must hold componentwise. In the current farmer example, if scenario k dominates scenario j , the yields of wheat and corn are higher in scenario k . It follows that the purchases of both products can only be smaller under scenario k . Hence, $w_k \leq w_j$. A first set of potential valid inequalities is $w_k \leq w_j$ for all pairs of scenarios such that $\xi^k \geq \xi^j$.

Now, we may define $A_k = \{j \mid \xi^k \geq \xi^j\}$ as the *dominance set* of scenario k . This dominance set includes k and all scenarios dominated by k . By the concept of dominance, if $w_k = 1$, then $w_j = 1$, $\forall j \in A_k$. An immediate consequence is that $w_k = 0$ if $P(A_k) > \alpha$, where $P(A_k) = \sum_{j \in A_k} p_j$.

More generally, if $P(\cup_{k \in C} A_k) > \alpha$, the set C forms a so-called *cover* for which the following constraint is a valid inequality:

$$\sum_{k \in C} w_k \leq |C| - 1,$$

where $|C|$ denotes the cardinality of C . The terminology cover comes from the *knapsack* structure of (2.13), a structure thoroughly studied in integer programming. However, covers are generated here from the probability of the dominance sets A_k instead of simply from the coefficients p_k .

We now illustrate the valid inequalities in the farmer problem with the extra probabilistic constraint (2.17). Imagine the farmer is able to collect 25 scenarios, each having probability 0.04. (He may obtain them in a cooperative fashion with some fellow farmers or get them from an agricultural research institute.)

Assume that the first 9 scenarios (restricted to wheat's and corn's yields) are as follows: $(2.25, 2.4)$, $(2.1, 2.6)$, $(2.4, 2.5)$, $(2.6, 2.3)$, $(2.2, 3)$, $(2, 3.4)$, $(2.5, 2.7)$, $(2.3, 3.6)$, $(2.2, 3.7)$. They are represented in Figure 1. Assume also that, for all other scenarios, $P(A_k) > 0.8$; hence, $w_k = 0$.

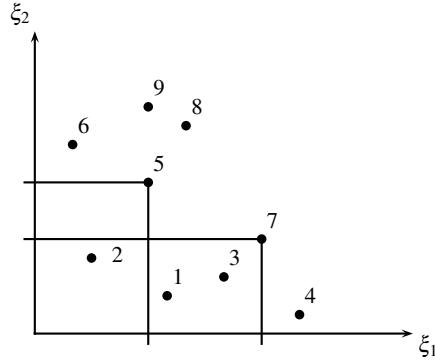


Fig. 1 Wheat and corn's scenarios.

There are several dominance relations: $\xi^3 \geq \xi^1$, $\xi^5 \geq \xi^2$, $\xi^7 \geq \xi^2$, $\xi^7 \geq \xi^3$, $\xi^8 \geq \xi^1$, $\xi^8 \geq \xi^5$, $\xi^8 \geq \xi^6$, $\xi^9 \geq \xi^5$, $\xi^9 \geq \xi^6$, implying valid inequalities $w_3 \leq w_1$, $w_5 \leq w_2$, $w_7 \leq w_2$, $w_7 \leq w_3$, $w_8 \leq w_1$, $w_8 \leq w_5$, $w_8 \leq w_6$, $w_9 \leq w_5$.

Dominance sets A_k can be visualized by drawing a horizontal and a vertical half-line from k . A_5 and A_7 are illustrated in Figure 1. $A_5 = \{2, 5\}$ with $P(A_5) = 0.08$ and $A_7 = \{1, 2, 3, 7\}$ with $P(A_7) = 0.16$. Even if $P(A_5) + P(A_7) > 0.2$, Scenarios 5 and 7 do not constitute a cover as $P(A_5 \cup A_7) = 0.2$. Scenarios 3 and 9 have similar probabilities and constitute a cover: $A_3 = \{1, 3\}$ with $P(A_3) = 0.08$, $A_9 = \{2, 5, 6, 9\}$ with $P(A_9) = 0.16$ and $P(A_3 \cup A_9) = 0.24$. Thus $w_3 + w_9 \leq 1$ is a valid inequality. This example shows that covers based on the dominance sets A_k are difficult to find as probabilities do not sum over sets that may intersect.

Only minimal covers are of interest. As an example, $\{1, 3, 9\}$ is a cover but it is not minimal as removing $\{1\}$ still forms a cover. There are several other minimal covers in this example: $\{1, 4, 9\}$, $\{3, 4, 5, 6\}$, $\{3, 8\}$, $\{4, 5, 7\}$, $\{4, 6, 7\}$, $\{4, 8\}$, $\{7, 8\}$, $\{7, 9\}$, $\{8, 9\}$. In general, the MILP only adds minimal covers if they are violated by the current fractional point. The problem of efficient techniques for finding a violated minimal cover based on dominance sets is studied in Ruszczyński [2002]. This paper also provides a more general treatment on the cases that create what we have called here dominance.

Exercises

9. Consider Example 2 in Section 3.2b. Instead of putting a limit on the total purchase of wheat and corn, the farmer does not want either purchase to be over 10 T. Thus, (2.17) is replaced by $P(y_1(\omega) \leq 10, y_2(\omega) \leq 10) \geq 0.80$. Show how to reformulate the recourse problem with K scenarios and this extra probabilistic constraint as a MILP with K extra binary variables and $2K+1$ extra constraints.
10. Consider Section 3.2b. Restart from the original farming problem of Section 1.1 without a probabilistic constraint on the total purchase of wheat and corn. The farmer now concentrates on sugar beet production. He finds it inappropriate to sell less than 5400 T of sugar beets at the favorable price or more than 300 T of sugar beets at the lower price. If either of these events happen, he considers the sugar beet production planning as unsuccessful. Assume he wants a production planning which maximizes its expected profit, with the constraint that the probability of an unsuccessful sugar beet production planning is no more than 20%.
 - (a) Show how to reformulate the recourse problem with K scenarios and the extra probabilistic constraint on sugar beet production planning as a MILP with K extra binary variables and $2K+1$ extra constraints.
 - (b) Is it still possible to get a dominance result based on the yield of sugar beet production?

3.3 Stochastic Integer Programs

a. Recourse problems

The general formulation of a two-stage integer program resembles that of the general linear case presented in Section 1.1. It simply requires that some variables, in either the first stage or the second stage, are integer. As we have seen in the examples in Chapter 1, in many practical situations the restrictions are, in fact, that the variables must be binary, i.e., they can only take the value zero or one. Formally, we may write

$$\begin{aligned} \min_{x \in X} z &= c^T x + E_{\xi} \min \{ q(\omega)^T y(\omega) \mid Wy(\omega) = h(\omega) - T(\omega)x, y(\omega) \in Y \text{ a. s. } \} \\ \text{s. t. } Ax &= b, \end{aligned}$$

where the definitions of c , b , ξ , A , W , T , and h are as before. However, X and/or Y contains some integrality or binary restrictions on x and/or y . With this definition, we may again define a deterministic equivalent program of the form

$$\min_{x \in X} z = c^T x + \mathcal{Q}(x)$$

$$\text{s. t. } Ax = b$$

with $\mathcal{Q}(x)$ the expected value of the second stage defined as in Section 3.1.

In this section, we are interested in the properties of $\mathcal{Q}(x)$ and $K_2 = \{x \mid \mathcal{Q}(x) < \infty\}$. Clearly, if the only integrality restrictions are in X , the properties of $\mathcal{Q}(x)$ and K_2 are the same as in the continuous case. The main interesting cases are those in which some integrality restrictions are present in the second stage. The properties of $Q(x, \xi)$ for given ξ are those of the value function of an integer program in terms of its right-hand side. This problem has received much attention in the field of integer programming (see, e.g., Blair and Jeroslow [1982] or Nemhauser and Wolsey [1988]). In addition to being *subadditive*, the value function of an integer program can be obtained by starting from a linear function and finitely often repeating the operations of sums, maxima, and non-negative multiples of functions already obtained and rounding up to the nearest integer. Functions so obtained are known as *Gomory functions* (see again Blair and Jeroslow [1982] or Nemhauser and Wolsey [1988]). Clearly, the maximum and rounding up operations imply undesirable properties for $Q(x, \xi)$, $\mathcal{Q}(x)$, and K_2 , as we now illustrate. General proofs can be found in Louveaux and Schultz [2003].

Proposition 20. *The expected recourse function $\mathcal{Q}(x)$ of an integer program is in general, lower semicontinuous, nonconvex and discontinuous.*

Example 5

We illustrate the proposition in the following simple example where the first stage contains a single decision variable $x \geq 0$ and the second-stage recourse function is defined as:

$$Q(x, \xi) = \min\{2y_1 + y_2 \mid y_1 \geq x - \xi, y_2 \geq \xi - x, y \geq 0, \text{ integer}\}. \quad (3.1)$$

Assume ξ can take on the values one and two with equal probability $1/2$. Let $\lceil a \rceil$ denote the smallest integer greater than or equal to a (the rounding up operation) and $\lfloor a \rfloor$ the truncation or rounding down operation ($\lfloor a \rfloor = -\lceil -a \rceil$). Consider $\xi = 1$. For $x \leq 1$, the optimal second-stage solution is $y_1 = 0$, $y_2 = \lceil 1 - x \rceil$. For $x \geq 1$, it is $y_1 = \lceil x - 1 \rceil$, $y_2 = 0$. Hence, $Q(x, 1) = \max\{2(\lceil x - 1 \rceil), \lceil 1 - x \rceil\}$, a typical Gomory function. It is discontinuous at $x = 1$. Nonconvexity can be illustrated by $Q(0.5, 1) > 0.5Q(0, 1) + 0.5Q(1, 1)$. Similarly, $Q(x, 2) = \max\{2(\lceil x - 2 \rceil), \lceil 2 - x \rceil\}$. The three functions, $Q(x, 1)$, $Q(x, 2)$, and $\mathcal{Q}(x)$ are represented in Figure 2.

The recourse function, $\mathcal{Q}(x)$, is clearly discontinuous in all positive integers. Nonconvexity can be illustrated by $\mathcal{Q}(1.5) = 1.5 > 0.5\mathcal{Q}(1) + 0.5\mathcal{Q}(2) = 0.75$. Thus $\mathcal{Q}(x)$ has none of the properties that one may wish for to design an algorithmic

procedure. Note, however, that a convexity-related property exists in the case of simple integer recourse (Proposition 8.4) and that it applies to this example.

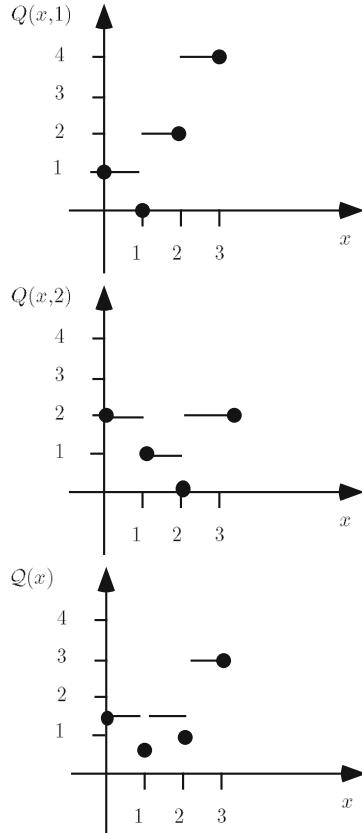


Fig. 2 Example of discontinuity.

Continuity of the recourse function can be regained when the random variable is absolutely continuous (Stougie [1987]).

Proposition 21. *The expected recourse function $\mathcal{Q}(x)$ of an integer program with an absolutely continuous random variable is continuous.*

Note, however, that despite Proposition 21, the recourse function $\mathcal{Q}(x)$ remains, in general, nonconvex.

Example 6

Consider Example 5 but with the (continuous) random variable defined by its cumulative distribution,

$$F(t) = P(\xi \leq t) = 2 - 2/t, 1 \leq t \leq 2.$$

Consider $1 < x < 2$. For $1 \leq \xi < x$, we have $0 < x - \xi < 1$; hence, $y_1 = 1$, $y_2 = 0$, while for $x < \xi \leq 2$, we have $0 < \xi - x \leq 1$; hence, $y_1 = 0$, $y_2 = 1$.

It follows that

$$\begin{aligned} \mathcal{Q}(x) &= \int_1^x 2dF(t) + \int_x^2 1dF(t) = 2F(x) + 1 - F(x) \\ &= F(x) + 1 = 3 - 2/x, \end{aligned}$$

which is easily seen to be nonconvex.

Properties are just as poor in terms of feasibility sets. As in the continuous case, we may define the second-stage feasibility set for a fixed value of ξ as $K_2(\xi(\omega)) = \{x \mid \text{there exists } y \text{ s.t. } Wy = h(\omega) - T(\omega)x, y \in Y\}$ where $\xi(\omega)$ is formed by the stochastic components of $h(\omega)$ and $T(\omega)$.

Proposition 22. *The second-stage feasibility set $K_2(\xi)$ is in general nonconvex.*

Proof: Because $K_2(\xi) = \{x \mid Q(x, \xi) < \infty\}$, nonconvexity of $K_2(\xi)$ immediately follows from nonconvexity of $Q(x, \xi)$. \square

A simple example suffices to illustrate this possibility.

Example 7

Let the second stage of a stochastic program be defined as

$$-y_1 + y_2 \leq \xi - x_1, \quad (3.2)$$

$$y_1 + y_2 \leq 2 - x_2, \quad (3.3)$$

$$y_1, y_2 \geq 0 \text{ and integer.} \quad (3.4)$$

Assume ξ takes on the values 1 and 2 with equal probability 1/2. We then construct $K_2(1)$.

By (3.3), $x_2 \leq 2$ is a necessary condition for second-stage feasibility. For $1 < x_2 \leq 2$, the only feasible integer satisfying (3.3) is $y_1 = y_2 = 0$. This point is also feasible for (3.2) if $\xi - x_1 \geq 0$, i.e., if $x_1 \leq 1$.

For $0 < x_2 \leq 1$, the integer points y satisfying (3.3) are $(0, 0)$, $(0, 1)$, $(1, 0)$. The one yielding the smallest left-hand side (and thus the most likely to yield points

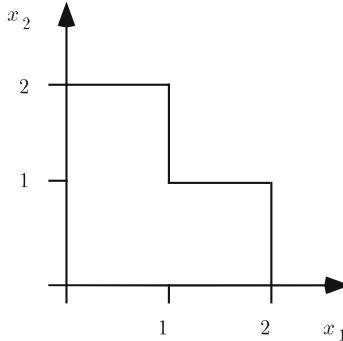


Fig. 3 Feasibility set for Example 7.

in $K_2(1)$) is $(1, 0)$. It requires $\xi - x_1 \geq -1$, i.e., $x_1 \leq 2$. Hence $K_2(1)$ is as in Figure 3 and is clearly nonconvex. It may be represented as $K_2(1) = \{x \mid \min\{x_1 - 1, x_2 - 1\} \leq 0, 0 \leq x_1 \leq 2, 0 \leq x_2 \leq 2\}$ and is again a typical Gomory function due to the minimum operation.

We may then define the second-stage feasibility set K_2 as the intersection of $K_2(\xi)$ over all possible ξ values. This definition poses no difficulty when ξ has a discrete distribution. In Example 7, $K_2 = K_2(1)$ and is thus also nonconvex.

Computationally, it might be very useful to have the constraint matrix of the extensive form *totally unimodular*. (Recall that a matrix is totally unimodular if the determinants of all square submatrices are 0, 1, or -1 .) This would imply that any solution of the associated stochastic continuous program would be integer when right-hand sides of all constraints are also integer. A widely used sufficient condition for total unimodularity is as follows: all coefficients are 0, 1, or -1 ; every variable has at most two nonzero coefficients and constraints can be separated in two groups such that, if a variable has two nonzero coefficients and if they are of the same sign, the two associated rows belong to different sets and if they are of opposite signs they belong to the same set.

To help understand the sufficiency condition, consider the following matrix

$$\begin{pmatrix} 1 & 0 & 1 & -1 \\ 0 & 1 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix}$$

as an example. For this matrix, one set consists of Rows 1 and 3, and the second set contains just Row 2. The constraint matrix of the extensive form of a nontrivial stochastic program cannot satisfy this sufficient condition. For simplicity, consider the case of a fixed T matrix. Assume that any variable that has a nonzero coefficient in T also has a nonzero coefficient in A . Then, if $|\Xi| \geq 2$, the constraint matrix of the extensive form contains a submatrix

$$\begin{pmatrix} A \\ T \\ T \end{pmatrix}$$

that has at least three nonzero coefficients. Thus, only very special cases (a random T matrix with every column having a nonzero element in only one realization, for example) could lead to totally unimodular matrices.

Last but not least, it should be clear that just finding $\mathcal{Q}(x)$ for a given x becomes an extremely difficult task for a general integer second stage. This is especially true because there is no hope to use sensitivity analysis or some sort of bunching procedure (see Section 5.4) to find $Q(x, \xi)$ for neighboring values of ξ . Cases where $\mathcal{Q}(x)$ can be computed or even approximated in a reasonable amount of time should thus be considered exceptions. One such exception is provided in the next section.

b. Simple integer recourse

Let ξ be a random vector with support Ξ in \Re^m , expectation μ , and cumulative distribution F with $F(t) = P\{\xi \leq t\}$, $t \in R^m$. A two-stage stochastic program with simple integer recourse is as follows:

$$\begin{aligned} SIR \quad \min z &= c^T x + E_{\xi} \{ \min(q^+)^T y^+ + (q^-)^T y^- \mid \\ &\quad y^+ \geq \xi - Tx, y^- \geq Tx - \xi, \\ &\quad y^+ \in Z_+^m, y^- \in Z_+^m \text{ a. s. } \} \\ &\text{s. t. } Ax = b, \quad x \in X, \end{aligned} \tag{3.5}$$

where X typically defines either non-negative continuous or non-negative integer decision variables and where we use $\xi = \mathbf{h}$ because both T and q are known and fixed. As in the continuous case, we may replace the second-stage value function $\mathcal{Q}(x)$ by a separable sum over the various coordinates. Let $\chi = Tx$ be a tender to be bid against future outcomes. Then $\mathcal{Q}(x)$ is separable in the components χ_i .

$$\mathcal{Q}(x) = \sum_{i=1}^m \psi_i(\chi_i), \tag{3.6}$$

with

$$\psi_i(\chi_i) = E_{\xi_i} \psi_i(\chi_i, \xi_i) \tag{3.7}$$

and

$$\begin{aligned} \psi_i(\chi_i, \xi_i) &= \min \{ q_i^+ y_i^+ + q_i^- y_i^- \mid y_i^+ \geq \xi_i - \chi_i, \\ &\quad y_i^- \geq \chi_i - \xi_i, y_i^+, y_i^- \in Z_+ \}. \end{aligned} \tag{3.8}$$

As in the continuous case, any error made in bidding χ_i versus ξ_i must be compensated for in the second stage, but this compensation must now be an integer.

Now define the expected shortage as

$$u_i(\chi_i) = E[\xi_i - \chi_i]^+$$

and the expected surplus as

$$v_i(\chi_i) = E[\chi_i - \xi_i]^+,$$

where $[x]^+ = \max\{[x], 0\}$. It follows that $\psi_i(\chi_i)$ is simply

$$\psi_i(\chi_i) = q_i^+ u_i(\chi_i) + q_i^- v_i(\chi_i).$$

As is reasonable from the definition of SIR, we assume $q_i^+ \geq 0, q_i^- \geq 0$.

Studying SIR is thus simply studying the expected shortage and surplus. Unless necessary, we drop the indices in the sequel. Let ξ be some random variable and $x \in \mathfrak{R}$. The expected shortage is

$$u(x) = E[\xi - x]^+ \tag{3.9}$$

and the expected surplus is

$$v(x) = E[x - \xi]^+. \tag{3.10}$$

For easy reference, we also define their continuous counterparts. Let the continuous expected shortage be

$$\hat{u}(x) = E(\xi - x)^+ \tag{3.11}$$

and the continuous expected surplus be

$$\hat{v}(x) = E(x - \xi)^+. \tag{3.12}$$

First observe that Example 5 (and 6) is a case of a stochastic program with simple recourse, from which we know that $u(x) + v(x)$ is in general nonconvex and discontinuous unless ξ has an absolutely continuous probability distribution function. We thus limit our ambitions to study finiteness and computational tractability for $u(\cdot)$ and $v(\cdot)$. The following results appear in Louveaux and van der Vlerk [1993].

Proposition 23. *The expected shortage function is a non-negative non-decreasing extended real-valued function. It is finite for all $x \in \mathfrak{R}$ if and only if $\mu^+ = E \max\{\xi, 0\}$ is finite.*

Proof: We only give the proof for finiteness because the other results are immediate. First, observe that for all t in \mathfrak{R} ,

$$(t - x)^+ \leq [t - x]^+ \leq (t - x + 1)^+ \leq (t - x)^+ + 1.$$

Taking expectation yields

$$\hat{u}(x) \leq u(x) \leq \hat{u}(x-1) \leq \hat{u}(x) + 1 . \quad (3.13)$$

The result follows as $\hat{u}(x)$ is finite if and only if μ^+ is finite. \square

We now provide a computational formula for $u(x)$.

Theorem 24. Let ξ be a random variable with cumulative distribution function F . Then

$$u(x) = \sum_{k=0}^{\infty} (1 - F(x+k)) . \quad (3.14)$$

Proof: Following the previous definitions, we have:

$$\begin{aligned} \sum_{k=0}^{\infty} (1 - F(x+k)) &= \sum_{k=0}^{\infty} P\{\xi - x > k\} \\ &= \sum_{k=0}^{\infty} \sum_{j=k+1}^{\infty} P\{[\xi - x]^+ = j\} \\ &= \sum_{j=1}^{\infty} \sum_{k=0}^{j-1} P\{[\xi - x]^+ = j\} \\ &= \sum_{j=1}^{\infty} jP\{[\xi - x]^+ = j\} = E[\xi - x]^+ = u(x) , \end{aligned}$$

which completes the proof. \square

Similar results hold for $v(x)$.

Theorem 25. Let ξ be a random variable with $\hat{F}(t) = P\{\xi < t\}$ and $\mu^- = E\xi^-$. Then v is a non-negative nondecreasing extended real-valued function, which is finite for all $x \in \mathfrak{R}$ if and only if μ^- is finite. Moreover,

$$v(x) = \sum_{k=0}^{\infty} \hat{F}(x-k) . \quad (3.15)$$

Theorems 24 and 25 provide workable formulas for a number of cases.

Case a. Clearly, if ξ has a finite range, then (3.14) and (3.15) reduce to a finite computation.

Example 8

Let ξ have a uniform density on $[0, a]$ for $a > 0$. Consider $0 \leq x \leq a$. Then

$$\begin{aligned}
u(x) &= \sum_{k=0}^{\infty} (1 - F(x+k)) = \sum_{k=0}^{\lceil a-x \rceil^+ - 1} (1 - F(x+k)) \\
&= \sum_{k=0}^{\lceil a-x \rceil^+ - 1} \left(1 - \frac{x+k}{a}\right) \\
&= \lceil a-x \rceil^+ \left(1 - \frac{x}{a}\right) - \frac{\lceil a-x \rceil^+ (\lceil a-x \rceil^+ - 1)}{2a}.
\end{aligned}$$

Observe that $\lceil a-x \rceil^+$ is piecewise constant. Hence, $u(x)$ is piecewise linear and convex.

Similarly, one computes

$$v(x) = \frac{x(\lfloor x \rfloor + 1)}{a} - \frac{\lfloor x \rfloor(\lfloor x \rfloor + 1)}{2a}.$$

Again, $v(x)$ is piecewise linear and convex. It follows that a simple integer recourse program with uniform densities is a piecewise linear convex program whose second-stage recourse function is easily computable.

Case b. For some continuous random variables, we may obtain analytical expressions for $u(x)$ and $v(x)$.

Example 9

Let ξ follow an exponential distribution with parameter $\lambda > 0$. Then, for $x \geq 0$,

$$u(x) = \sum_{k=0}^{\infty} (1 - F(x+k)) = \sum_{k=0}^{\infty} e^{-\lambda(x+k)} = \frac{e^{-\lambda x}}{1 - e^{-\lambda}},$$

while

$$\begin{aligned}
v(x) &= \sum_{k=0}^{\infty} F(x-k) = \lfloor x \rfloor + 1 - e^{-\lambda(x-\lfloor x \rfloor)} \cdot \sum_{k=0}^{\lfloor x \rfloor} e^{-\lambda k} \\
&= \lfloor x \rfloor + 1 - \left(\frac{e^{-\lambda(x-\lfloor x \rfloor)} - e^{-\lambda(x+1)}}{1 - e^{-\lambda}} \right).
\end{aligned}$$

Observe that $v(x)$ is nonconvex (as it would be $u(x)$ for $x \leq 0$).

Case c. Finite computation can also be obtained when $\Xi \in Z$. From Theorems 24 and 25, we derive the following corollary.

Corollary 26. *For all $n \in \mathbb{Z}_+$, we have*

$$u(x+n) = u(x) - \sum_{k=0}^{n-1} (1 - F(x+k)) \quad (3.16)$$

and

$$v(x+n) = v(x) + \sum_{k=1}^n \hat{F}(x+k) . \quad (3.17)$$

Corollary 27. Let ξ be a discrete random variable with support $\Xi \in Z$. Then

$$u(x) = \begin{cases} \mu^+ - [x] - \sum_{k=[x]}^{-1} F(k) & \text{if } x < 0 , \\ \mu^+ - [x] + \sum_{k=0}^{[x]-1} F(k) & \text{if } x \geq 0 . \end{cases}$$

Proof: Because $\Xi \in Z$, $F(t) = F(\lfloor t \rfloor)$, for all $t \in \mathfrak{R}$. Hence, $u(x) = u(\lfloor x \rfloor)$ for all $x \in R$. Now, $u(0) = \mu^+$. Then apply (3.16) to obtain the result. \square

Corollary 28. Let ξ be a discrete random variable with support $\Xi \in Z$. Then

$$v(x) = \begin{cases} \mu^- - \sum_{k=\lceil x \rceil}^{-1} F(k) & \text{if } x < 0 , \\ \mu^- + \sum_{k=0}^{\lceil x \rceil - 1} F(k) & \text{if } x \geq 0 . \end{cases}$$

Thus, here the finite computation comes from the finiteness of $\lceil x \rceil$.

Case d. Finally, we may have a random variable that does not fall in any of the given categories. We may then resort to approximations.

Theorem 29. Let ξ be a random variable with cumulative distribution function, F . Then

$$\hat{u}(x) \leq u(x) \leq \hat{u}(x) + 1 - F(x) . \quad (3.18)$$

Proof: The first inequality was given in (3.13). Because $1 - F(t)$ is nonincreasing, we have for any $x \in \mathfrak{R}$ and any $k \in \{1, 2, \dots\}$ that

$$1 - F(x+k) \leq 1 - F(t) , \quad t \in [x+k-1, x+k) .$$

Hence,

$$\sum_{k=1}^{\infty} (1 - F(x+k)) \leq \int_x^{\infty} (1 - F(t)) dt .$$

Adding $1 - F(x)$ to both sides gives the desired result. \square

Theorem 30. Let ξ be a random variable with cumulative distribution function F . Let n be some integer, $n \geq 1$. Define

$$u_n(x) = \sum_{k=0}^{n-1} (1 - F(x+k)) + \hat{u}(x+n) . \quad (3.19)$$

Then

$$u_n(x) \leq u(x) \leq u_n(x) + 1 - F(x+n) . \quad (3.20)$$

Proof: The proof follows directly from Theorem 29 and Formula (3.16). \square

To approximate $u(x)$ within an accuracy ε , we have to compute the first n terms in $u(x)$, where n is chosen so that $F(x+n) \geq 1 - \varepsilon$ and $\hat{u}(x+n)$, which involves computing one integral.

Example 10

Let ξ follow a normal distribution with mean μ and variance σ^2 , i.e., $N(\mu, \sigma^2)$, with cumulative distribution function F and probability density function f . Integrating by parts, one obtains:

$$\begin{aligned} u_n(x) &= \sum_{k=0}^{n-1} (1 - F(x+k)) + \int_{x+n}^{\infty} (1 - F(t)) dt \\ &= \sum_{k=0}^{n-1} (1 - F(x+k)) - (x+n)(1 - F(x+n)) + \int_{x+n}^{\infty} tf(t) dt . \end{aligned}$$

Using $tf(t) = \mu f(t) - \sigma^2 f'(t)$, it follows that

$$u_n(x) = \sum_{k=0}^{n-1} (1 - F(x+k)) + (\mu - x - n)(1 - F(x+n)) + \sigma^2 f(x+n) .$$

Similar results apply for $v(x)$.

Theorem 31. Let ξ be a random variable with cumulative distribution function $\hat{F}(t) = P\{\xi < t\}$. Then

$$\hat{v}(x) \leq v(x) \leq \hat{v}(x) + \hat{F}(x) . \quad (3.21)$$

Let n be some integer, $n \geq 1$. Define

$$v_n(x) = \sum_{k=0}^{n-1} \hat{F}(x-k) + \hat{v}(x-n) . \quad (3.22)$$

Then

$$v_n(x) \leq v(x) \leq v_n(x) + \hat{F}(x-n) . \quad (3.23)$$

Example 10 (continued)

Let ξ follow an $N(\mu, \sigma^2)$ distribution, with cumulative distribution function F and probability density function f . Then

$$v_n(x) = \sum_{k=0}^{n-1} F(x-k) + (x-n-\mu)F(x-n) + \sigma^2 f(x-n).$$

As a conclusion, expected shortage, expected surplus, and thus simple integer recourse functions can be computed in finitely many steps either in an exact manner or within a prespecified tolerance ε . Deeper studies of continuity and differentiability properties of the recourse function can be found in Stougie [1987], Louveaux and van der Vlerk [1993], and Schultz [1993].

c. Probabilistic constraints

Probabilistic constraints involving integer decision variables may generally be treated in exactly the same manner as if they involved continuous decision variables. One need only take the intersection of their deterministic equivalents with the integrality requirements. The question is then how to obtain a polyhedral representation of this intersection. This problem sometimes has quite nice solutions.

Example 11: Covering

Consider an example where one can invest in any one of n projects in order to obtain at least b units of a good. Projects could be mines needed to extract at least b tons of ore per year, or buildings to let in order to obtain at least b thousands of rent per year

Let x_i be the binary variable representing the decision to invest ($x_i = 1$) or not ($x_i = 0$) in project i . In a deterministic setting, the yield of a project is the quantity a_i and the requirement of b units is described by the deterministic constraint

$$\sum_{i=1}^n a_i x_i \geq b. \quad (3.24)$$

Now, assume the yields are in fact random. This may come from operational difficulties in a mine or on some floors of the buildings remaining vacant for a period of time. Then, a typical probabilistic constraint would be

$$P\left(\sum_{i=1}^n \xi_i x_i \geq b\right) \geq \alpha, \quad (3.25)$$

where ξ_i is the random yield of project i .

Due to the binary nature of the decision variables x_i , this constraint is equivalent to

$$P \left(\sum_{i \in S} \xi_i \geq b \right) \geq \alpha \quad (3.26)$$

where S is some subset of $\{1, \dots, n\}$ representing the selected projects.

Now, if the random variables ξ_i follow a Poisson distribution with parameter a_i , then $\xi_S = \sum_{i \in S} \xi_i$ follow a Poisson distribution with parameter $\sum_{i \in S} a_i$ and (3.26) is equivalent to

$$P(\xi_S \geq b) \geq \alpha. \quad (3.27)$$

As b and α are given and ξ_S is known to follow a Poisson distribution, this corresponds to finding in the table of the cumulative Poisson distribution the smallest parameter value for which (3.27) holds. Let B be this value. Then (3.27) is equivalent to

$$\sum_{i \in S} a_i \geq B$$

or

$$\sum_{i=1}^S a_i x_i \geq B. \quad (3.28)$$

Thus, the probabilistic constraint (3.25) has the linear equivalent (3.28). This linear equivalent has exactly the same form as (3.24) with b replaced by a larger quantity B .

Example 12

Assume one has five projects with expected yields 2, 2.5, 4, 4.5, and 7. The level $b = 9$ is requested. The deterministic constraint based on expected yields is

$$2x_1 + 2.5x_2 + 4x_3 + 4.5x_4 + 7x_5 \geq 9.$$

The constraint can be satisfied with Project 5 and any other, or by Projects 1, 2 and 4, for instance.

If yields are random and follow a Poisson distribution and if the level of 9 must be obtained with probability 90%, then a value of $B = 13$ is found in the Poisson table and (3.28) gives the linear equivalent:

$$2x_1 + 2.5x_2 + 4x_3 + 4.5x_4 + 7x_5 \geq 13.$$

Example 13: Routing

Let $V = \{v_1, v_2, \dots, v_n\}$ be a set of vertices, typically representing customers. Let v_0 represent the depot and let $V_0 = V \cup \{v_0\}$. A route is an ordered sequence $L = \{i_0 = 0, i_1, i_2, \dots, i_k, i_{k+1} = 0\}$, with $k \leq n$, starting and ending at the depot and visiting each customer at most once. Clearly, if $k < n$, more than one vehicle is needed to visit all customers. Assume a vehicle of given capacity C follows each route, collecting customers' demands d_i . If demands d_i are random, it may turn out that at some point of a given route, the vehicle cannot load a customer's demand. This is clearly an undesirable feature, which is usually referred to as a *failure of the route*. A probabilistic constraint for the capacitated routing requires that only routes with a small probability of failure are considered feasible:

$$P(\text{failure on any route}) \leq \alpha, \quad (3.29)$$

where we note that here and elsewhere we use α as an upper bound on a probability (instead of a lower bound as in (2.1)) in following typical usage in the context. We now show, as in Laporte, Louveaux, and Mercure [1989], that any route that violates (3.29) can be eliminated by a linear inequality. For any route L , let $S = \{i_1, i_2, \dots, i_k\}$ be the index set of visited customers. Violation of (3.29) occurs if

$$P\left(\sum_{i \in S} d_i > C\right) > \alpha. \quad (3.30)$$

Let $V_\alpha(S)$ denote the smallest number of vehicles required to serve S so that the probability of failure in S does not exceed α , i.e., $V_\alpha(S)$ is the smallest integer such that

$$P\left(\sum_{i \in S} d_i > C \cdot V_\alpha(S)\right) \leq \alpha. \quad (3.31)$$

Now, let \bar{S} denote the complement of S versus V_0 , i.e., $\bar{S} = V_0 \setminus S$. Then the following *subtour elimination constraint* imposes, in a linear fashion, that at least $V_\alpha(S)$ vehicles are needed to cover demand in S :

$$\sum_{\substack{i \in S, j \in \bar{S} \text{ or } i \in \bar{S}, j \in S}} x_{ij} \geq 2V_\alpha(S), \quad (3.32)$$

where, as usual, $x_{ij} = 1$ when arc ij is traveled in the solution and $x_{ij} = 0$ otherwise. It follows that routes that violate (3.29) can be eliminated when needed by the linear constraint (3.32). Observe that this result is obtained without any assumption on the random variables. Also observe that (3.32) is not the deterministic equivalent of (3.29). This should be clear from the fact that an analytical expression for (3.29) is difficult to write. Finally, observe that in practice, as for many random variables, the probability distribution of $\sum_{i \in S} d_i$ is easily obtained. The computation of $V_\alpha(S)$ in (3.31) poses no difficulty. Additional results appear in the survey on stochastic vehicle routing by Gendreau, Laporte, and Séguin [1996].

Exercises

1. Consider the following second-stage integer program:

$$Q(x, \xi) = \max\{4y_1 + y_2 \mid y_1 + y_2 \leq \xi x, 0 \leq y_1 \leq 2, 0 \leq y_2 \leq 1, y \text{ integer}\}.$$

- (a) Obtain y_1^* , y_2^* , and $Q(x, \xi)$ as Gomory functions.
 - (b) Consider $\xi = 1$. Observe that $Q(x, 1)$ is piecewise constant on four pieces ($x < 1$, $1 \leq x < 2$, $2 \leq x < 3$, $3 \leq x$).
 - (c) Now assume ξ is uniformly distributed over $[0, 2]$. Obtain $\mathcal{Q}(x)$ on four pieces ($x < 0.5$, $0.5 \leq x < 1$, $1 \leq x < 1.5$, $1.5 \leq x$). Check the nonconcavity of $\mathcal{Q}(x)$. Observe that $\mathcal{Q}(x)$ is concave on each piece separately, but that $\mathcal{Q}(x)$ is not (compare, e.g., $\mathcal{Q}(1)$ to $1/2\mathcal{Q}(3/4) + 1/2\mathcal{Q}(5/4)$).
2. Consider ξ uniformly distributed over $[0, 1]$ and $0 \leq x \leq 1$. Show that $u(x) + v(x) = 1$.
3. Consider ξ uniformly distributed over $[0, 2]$.
- (a) Compute $u(x)$ directly from Definition (3.9) and check with the result in Example 8. Observe that $u(x)$ is piecewise linear, convex, and continuous.
 - (b) Compute $\hat{u}(x)$.
 - (c) Show that $u(x) - \hat{u}(x)$ is decreasing in x .
4. Consider ξ that is Poisson distributed with parameter three. Compute $u(3)$.
5. (a) Let ξ be normally distributed with mean zero and variance one. What is the accuracy level of $u_3(0)$ versus $u(0)$?
- (b) Let ξ be normally distributed with mean μ and variance σ^2 . Show that $u(\mu)$ is independent of μ . Is the accuracy of $u_n(\mu)$, n given, increasing or decreasing with σ^2 ?
6. Consider Example 11. In this example, a probabilistic constraint has a deterministic linear equivalent.
- (a) Does this also hold if x_i are integer variables, instead of binary variables?
 - (b) Does this also hold if the random variables ξ_i are normally distributed with mean a_i and variance σ_i^2 ?

3.4 Multistage Stochastic Programs with Recourse

The previous sections in this chapter concerned stochastic programs with two stages. Most practical decision problems, however, involve a sequence of decisions that react to outcomes that evolve over time. In this section, we will consider the stochastic programming approach to these multistage problems. We present the same basic results as in previous sections. We describe the basic structure of feasible solutions,

objective values, and conditions for optimality. We begin again with the linear, fixed recourse, finite horizon framework because this model has been the most widely implemented. We then continue with more general approaches.

We start with implicit nonanticipativity constraints as in the previous sections. The multistage stochastic linear program with fixed recourse then takes the following form (where we note that transposes are suppressed when they are clear from context to avoid excessive notation):

$$\begin{aligned}
 \min z &= c^1 x^1 + E_{\xi^2} [\min c^2(\omega) x^2(\omega^2) + \dots + E_{\xi^H} [\min c^H(\omega) x^H(\omega^H)] \dots] \\
 \text{s. t.} \quad &W^1 x^1 = h^1, \\
 &T^1(\omega^2) x^1 + W^2 x^2(\omega^2) = h^2(\omega), \\
 &\quad \vdots \\
 &T^{H-1}(\omega^H) x^{H-1}(\omega^{H-1}) + W^H x^H(\omega^H) = h^H(\omega), \\
 &x^1 \geq 0; \quad x^t(\omega^t) \geq 0, \quad t = 2, \dots, H;
 \end{aligned} \tag{4.1}$$

where c^1 is a known vector in \Re^{n_1} , h^1 is a known vector in \Re^{m_1} , $\xi^t(\omega)^T = (c^t(\omega)^T, h^t(\omega)^T, T_1^{t-1}(\omega), \dots, T_{m_t}^{t-1})$ is a random N_t -vector defined on (Ω, Σ^t, P) (where $\Sigma^t \subset \Sigma^{t+1}$) for all $t = 2, \dots, H$, and each W^t is a known $m_t \times n_t$ matrix. The decisions x depend on the history up to time t , which we indicate by ω^t . We also suppose that Ξ^t is the support of ξ^t .

For the financial planning problem in Section 1.2, these parameters are:

$$\begin{aligned}
 c^t(\omega) &= 0, t = 1, \dots, H-1; \\
 c^H(\omega) &= (q, -r); \\
 W^t &= e_I^T, t = 1, \dots, H-1; \\
 W^H &= [1 - 1], t = 1, \dots, H-1; \\
 T^t &= -\xi(\omega^t)^T, t = 1, \dots, H; \\
 h^1 &= b; \\
 h^t &= 0, t = 1, \dots, H-1; \\
 h^H &= -G.
 \end{aligned}$$

We first describe the deterministic equivalent form of this problem in terms of a dynamic program. If the stages are 1 to H , we can define states as $x^t(\omega^t)$. Noting that the only interaction between periods is through this realization, we can define a dynamic programming type of recursion. For terminal conditions, we have:

$$\begin{aligned}
 Q^H(x^{H-1}, \xi^H(\omega)) &= \min c^H(\omega) x^H(\omega) \\
 \text{s. t.} \quad &W^H x^H(\omega) = h^H(\omega) - T^{H-1}(\omega) x^{H-1}, \\
 &x^H(\omega) \geq 0.
 \end{aligned} \tag{4.2}$$

For the financial planning problem in Section 1.2, given x^{H-1} and $\xi^H(\omega)$, (4.2) has an optimal solution given by

$$x^H(\omega) = (y(\omega), w(\omega)) = ((G - \xi^H(\omega)^T x^{H-1}(\omega))^+, (\xi^H(\omega)^T x^{H-1}(\omega) - G)^+).$$

Solutions for other stages can be obtained with a backward recursion, letting $\mathcal{Q}^{t+1}(x^t) = E_{\xi^{t+1}}[Q^{t+1}(x^t, \xi^{t+1}(\omega))]$ for all t to obtain the recursion for $t = 2, \dots, H-1$,

$$\begin{aligned} Q^t(x^{t-1}, \xi^t(\omega)) &= \min c^t(\omega)x^t(\omega) + \mathcal{Q}^{t+1}(x^t) \\ \text{s. t. } W^t x^t(\omega) &= h^t(\omega) - T^{t-1}(\omega)x^{t-1}, \\ x^t(\omega) &\geq 0, \end{aligned} \quad (4.3)$$

where x^t indicates the state of the system. Other state information in terms of the realizations of the random parameters up to time t should be included if the distribution of ξ^t is not independent of the past outcomes. In the financial planning case, the value function, $\mathcal{Q}^{t+1}(x^t)$, represents the expected utility of choosing the asset allocations given by x^t in the t th period and choosing optimal allocations in all subsequent periods.

The value we seek is:

$$\begin{aligned} \min z &= c^1 x^1 + \mathcal{Q}(x^1) \\ \text{s. t. } W^1 x^1 &= h^1, \\ x^1 &\geq 0, \end{aligned} \quad (4.4)$$

which has the same form as the two-stage deterministic equivalent program. The examples of this formulation in Chapter 1 for financial planning and capacity expansion could then be re-cast as two-stage problems if the second-stage value function $\mathcal{Q}(x^1)$ can be found.

We would again like to obtain properties of the problems in (4.2)–(4.4) that allow uses of mathematical programming procedures such as decomposition. We concentrate first on the form of the feasible regions for problems (4.3). Let these be

$$K^t = \{x^t \mid \mathcal{Q}^{t+1}(x^t) < \infty\}.$$

We have the following result which helps in the development of several algorithms for multistage stochastic programs.

Theorem 32. *The sets K^t and functions $\mathcal{Q}^{t+1}(x^t)$ are convex for $t = 1, \dots, H-1$ and, if Ξ^t is finite for $t = 1, \dots, H$, then K^t and $\mathcal{Q}^{t+1}(x^t)$ are polyhedral.*

Proof: Proceed by induction. Because $Q^H(x^{H-1}, \xi^H(\omega))$ is convex for all $\xi^H(\omega)$, so too is $\mathcal{Q}^H(x^{H-1})$. We can then carry this back to each $t < H-1$. The same applies for the polyhedrality property because finite numbers of realizations lead to each $\mathcal{Q}^{t+1}(x^t)$'s being the sum of a finite number of polyhedral functions, which is then polyhedral. \square

We note that we may also describe the feasibility sets K^t in terms of intersections of feasibility sets for each outcome if we have finite second moments for ξ^t in each period. This result is also true when we have a finite number of possible realizations of the future outcomes. In this case, the set of possible future sequences of outcomes are called *scenarios*.

The description of scenarios is often made on a tree such as that in Figure 4. Here, there are seven scenarios that are evident in the last stage ($H = 4$). In previous stages ($t < 4$), we have a more limited number of possible realizations, which we call the *stage t scenarios*. Each of these period t scenarios is said to have a single *ancestor* scenario in stage ($t - 1$) and perhaps several *descendant* scenarios in stage ($t + 1$). We note that different scenarios at stage t may correspond to the same ξ^t realizations and are only distinguished by differences in their ancestors.

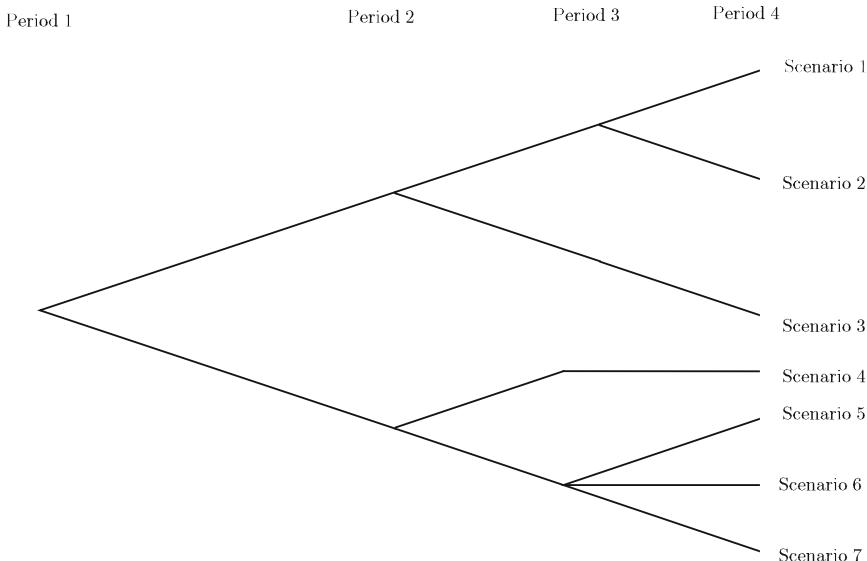


Fig. 4 A tree of seven scenarios over four periods.

The deterministic equivalent program to (4.1) with a finite number of scenarios is still a linear program. It has the structural form indicated in Figure 5, where subscripts indicate different scenario realizations for the T^t matrices. This is often called *arborescent* form and can be exploited in large-scale optimization approaches as in Kallio and Porteus [1977]. A difficulty is still, however, that these problems become extremely large as the number of stages increases, even if only a few realizations are allowed in each stage.

In some problems, however, we can avoid much of this difficulty if the interactions between consecutive stages are sufficiently weak. This is the case in the capacity expansion problem described in Section 1.3. Here, capacity carried over from

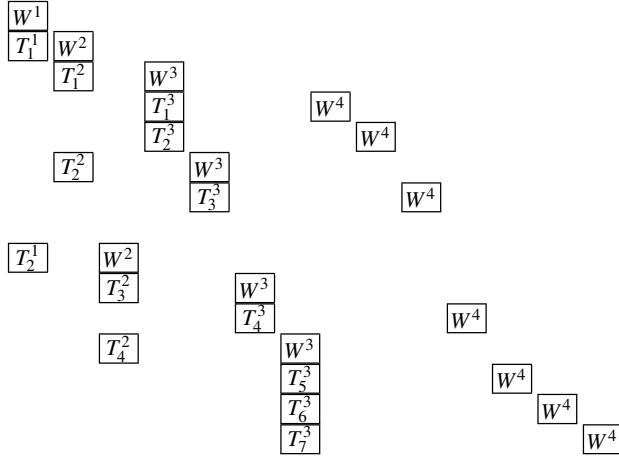


Fig. 5 The deterministic equivalent matrix for a problem with seven scenarios in four periods.

one stage to the next is not affected by the demand in that stage. Decisions about the amount of capacity to install can be made at the beginning and then the future only involves reactions to these outcomes. Problems with this form are called *block separable* as mentioned in Section 1.3. Formally, we have the following definition for block separability (see Louveaux [1986]).

Definition 33. A multistage stochastic linear program (4.1) has block separable recourse if for all periods $t = 1, \dots, H$ and all ω , the decision vectors, $x^t(\omega)$, can be written as $x^t(\omega) = (w^t(\omega), y^t(\omega))$ where w^t represents aggregate level decisions and y^t represents detailed level decisions. The constraints also follow these partitions:

1. The stage t objective contribution is $c^t x^t(\omega) = r^t w^t(\omega) + q^t y^t(\omega)$.
2. The constraint matrix W^t is block diagonal:

$$W^t = \begin{pmatrix} W^t & 0 \\ 0 & T^t \end{pmatrix}. \quad (4.5)$$

3. The other components of the constraints are random but we assume that for each realization of ω , $T^t(\omega)$ and $h^t(\omega)$ can be written:

$$T^t(\omega) = \begin{pmatrix} R^t(\omega) & 0 \\ S^t(\omega) & 0 \end{pmatrix} \quad \text{and} \quad h^t(\omega) = \begin{pmatrix} b^t(\omega) \\ d^t(\omega) \end{pmatrix}, \quad (4.6)$$

where the zero components of T^t correspond to the detailed level variables.

To put the capacity expansion problem in Section 1.3 into this framework, we keep information about the installed capacity from the current and previous

periods as $w^{t,j} = x^{t-j}$ for $j = 0, \dots, L_{\max}$, where $L_{\max} = \max_i L_i$ and x^{t-j} follows the notation in Section 1.3, and re-label the current available capacity at time t as $w^{t,L_{\max}+1}$. With these definitions, we define $A^1 = I_{n(L_{\max}+2) \times n(L_{\max}+2)}$, an $n(L_{\max}+2) \times n(L_{\max}+2)$ identity matrix and let $h^1 = [(x^{-1})^T, (x^{-2})^T, \dots, (x^{-L_{\max}})^T, 0_{1 \times 2n}]^T$, where $0_{1 \times 2n}$ indicates a $1 \times 2n$ matrix of zeroes, as the initial conditions for the problem where x^{-j} is interpreted as capacity installed j periods before the initial period (which then replaces the information in the remaining existing capacity vector g^t used in Section 1.3). We can then define, for $t = 1, \dots, H-1$,

$$R^t = \begin{pmatrix} -I_{nL_{\max} \times nL_{\max}} & 0_{n \times n} & 0_{n \times n} \\ 0_{nL_{\max} \times nL_{\max}} & 0_{n \times n} & -I_{n \times n} \end{pmatrix}; \quad (4.7)$$

$$S^t = \begin{pmatrix} -\sum_{i=1}^n a_i e_i \sum_{j=\Delta_i}^{L_i} e_{n(j-1)+i} & 0_{n \times 2n} \\ 0_{m \times nL_{\max}} & 0_{n \times 2n} \end{pmatrix}; \quad (4.8)$$

and, for $t = 2, \dots, H$,

$$W^t = \begin{pmatrix} 0_{nL_{\max} \times n} & I_{nL_{\max} \times nL_{\max}} & 0_{nL_{\max} \times n} \\ -I_{n \times n} & \sum_{i=1}^n e_i e_{(n-1)L_i+i}^T & I_{n \times n} \end{pmatrix}; \quad (4.9)$$

$$T^t = \begin{pmatrix} \sum_{i=1}^n e_i \sum_{j=1}^m e_{n(j-1)+i} & I_{n \times n} \\ \sum_{j=1}^m e_j \sum_{i=1}^n e_{n(i-1)+j} & 0_{n \times n} \end{pmatrix}; \quad (4.10)$$

$b^t = 0_{n(L_{\max}+1) \times 1}$, and $d^t(\omega) = [\mathbf{d}_1^t, \dots, \mathbf{d}_m^t, 0_{1 \times n}]^T$, where \mathbf{d}^t is defined as in Section 1.3.

Notice that (3) in the definition implies that detailed level variables, corresponding to the capacity usage in each period in the capacity expansion model, have no direct effect on future constraints. This is the fundamental advantage of block separability.

With block separable recourse, we may rewrite $Q^t(x^{t-1}, \xi^t(\omega))$ as the sum of two quantities, $Q_w^t(w^{t-1}, \xi^t(\omega)) + Q_y^t(w^{t-1}, \xi^t(\omega))$, where we need not include the y^{t-1} terms in x^{t-1} ,

$$\begin{aligned} Q_w^t(w^{t-1}, \xi^t(\omega)) &= \min r^t(\omega) w^t(\omega) + \mathcal{Q}^{t+1}(x^t) \\ \text{s. t. } W^t w^t(\omega) &= b^t(\omega) - R^{t-1}(\omega) w^{t-1}, \\ w^t(\omega) &\geq 0, \end{aligned} \quad (4.11)$$

and

$$\begin{aligned} Q_y^t(w^{t-1}, \xi^t(\omega)) &= \min q^t(\omega) y^t(\omega) \\ \text{s. t. } T^t y^t(\omega) &= d^t(\omega) - S^{t-1}(\omega) w^{t-1}, \\ y^t(\omega) &\geq 0. \end{aligned} \quad (4.12)$$

The great advantage of block separability is that we need not consider nesting among the detailed level decisions. In this way, the w variables can all be pulled together into a first stage of aggregate level decisions. The second stage is then composed of

the detailed level decisions. Note that if the b^t and R^t are known, as they are in the model in Section 1.3, then the block separable problem is equivalent to a similarly sized two-stage stochastic linear program.

Separability is indeed a very useful property for stochastic programs. Computational methods should try to exploit it whenever it is inherent in the problem because it may reduce work by orders of magnitude. We will also see in Chapter 10 that separability can be added to a problem (with some error that can be bounded). This approach opens many possible applications with large numbers of random variables.

Another modeling approach that may have some computational advantage appears in Grinold [1976]. This approach extends from analyses of stochastic programs as examples of a Markov decision process. He assumes that ω^t belongs to some finite set $1, \dots, k_t$, that the probabilities are determined by $p_{ij} = P\{\omega^{t+1} = j \mid \omega^t = i\}$ for all t , and that $T^t = T^t(\omega^t, \omega^{t+1})$. In this framework, he can obtain an approximation that again obtains a form of separability of future decisions from previous outcomes. We discuss more approximation approaches for multiple stages in Chapter 10.

Exercises

1. State a set of optimality conditions analogous to those in Theorem 9 for $x^{t*}(\omega)$ to be an optimal solution in (4.3).
2. Assume that the model in (4.1) has relatively complete recourse. In this case, find an expression for $\partial \mathcal{Q}^{t+1}(x^t)$.
3. Give the full set of optimality conditions that are satisfied for an optimal solution $x^{t*}(\omega)$ for $t = 1, \dots, H$ for the financial planning example in Section 1.2 and verify their satisfaction for the solution corresponding to the data in (1.2.1).
4. *Emergency vehicle location:* Suppose a multistage version of the model in Section 2.6, where a city wishes to determine the allocations of V emergency vehicles to each of n stations at times $t = 1, \dots, H$. Each vehicle can serve a single call in any period, where calls are random and can occur at any of m locations according with $d_j^t(\omega)$ corresponding to the random number of calls in location j in period t . The cost of responding to a call at location j with a vehicle from station i is q_{ij}^t and any calls in location j that cannot be served by the city's vehicles are served by an outside vendor at a cost \bar{q}_j^t (regardless of the number of calls). The initial number of vehicles at each station i is given by h_i^1 . Initially and at the end of each period, vehicles may be move from any station i to any other station j at a cost r_{ij}^t .
 - (a) Give a multistage stochastic linear programming formulation for this model (assuming V is sufficiently large that the discrete decision variables may be adequately approximated with a continuous solution).

- (b) Show that this model satisfies the block-separable recourse conditions by giving the corresponding decision vectors $(w^t(\omega), y^t(\omega))$ and constraint parameters, $A^t, B^t, R^t(\omega), S^t(\omega), b^t(\omega)$, and $d^t(\omega)$.

3.5 Stochastic Nonlinear Programs with Recourse

In this section, we generalize the results from the previous sections to problems with nonlinear functions, starting with two-stage problems. The results extend directly so the treatment here will be brief. The basic types of results we would like to obtain concern the structure of the feasible region, the optimal value function, and optimality conditions. As a note of caution, some of the results in this section refer to concepts from measure theory.

We begin with a definition of the two-stage stochastic nonlinear program with recourse. This problem has the form:

$$\begin{aligned} \inf z &= f^1(x) + \mathcal{Q}(x) \\ \text{s. t. } g_i^1(x) &\leq 0, \quad i = 1, \dots, \bar{m}_1, \\ g_i^1(x) &= 0, \quad i = \bar{m}_1 + 1, \dots, m_1, \end{aligned} \tag{5.1}$$

where $\mathcal{Q}(x) = E_\omega[Q(x, \omega)]$ and

$$\begin{aligned} Q(x, \omega) &= \inf f^2(x, y(\omega), \omega) \\ g_i^2(x, y(\omega), \omega) &\leq 0, \quad i = 1, \dots, \bar{m}_2, \\ g_i^2(x, y(\omega), \omega) &= 0, \quad i = \bar{m}_2 + 1, \dots, m_2, \end{aligned} \tag{5.2}$$

where all functions f^1 and g_i^1 for all i are continuous, and $f^2(\cdot, \cdot, \omega)$ and $g_i^2(\cdot, \cdot, \omega)$ are also continuous for any fixed ω and are measurable in ω for any fixed first argument and for any i . Given this assumption, $Q(x, \omega)$ is measurable (Exercise 1) and hence $\mathcal{Q}(x)$ is well-defined.

We make the following definitions consistent with Section 3.1.

$$K_1 \equiv \{x \mid g_i^1(x) \leq 0, \quad i = 1, \dots, \bar{m}_1; \quad g_i^1(x) = 0, \quad i = \bar{m}_1 + 1, \dots, m_1\},$$

$$\begin{aligned} K_2(\omega) &= \{x \mid \exists y(\omega) \mid g_i^2(x, y(\omega), \omega) \leq 0, \quad i = 1, \dots, \bar{m}_2; \\ &\quad g_i^2(x, y(\omega), \omega) = 0, \quad i = \bar{m}_2 + 1, \dots, m_2\}, \end{aligned}$$

and

$$K_2 = \{x \mid \mathcal{Q}(x) < \infty\}.$$

We have not forced fixed recourse in Problem 5.1 because the second-period constraint functions may depend on ω and on $y(\omega)$. For linear programs, we assumed

fixed recourse so we could describe the feasible region in terms of intersections of feasible regions for each random outcome. We could also follow this approach here but the conditions for this result depend directly on the form of the objective and constraint functions. We explore these possibilities in Exercise 1 but we continue here with the more general case.

We make additional assumptions to allow results along the lines of the previous section. These conditions ensure regularity for the application of necessary and sufficient optimality conditions.

1. *Convexity.* The function f^1 is convex on \mathbb{R}^{n_1} , g_i^1 is convex on \mathbb{R}^{n_1} for $i = 1, \dots, \bar{m}_1$, g_i^1 is affine on \mathbb{R}^{n_1} for $i = \bar{m}_1 + 1, \dots, m_1$, $f^2(\cdot, \cdot, \omega)$ is convex and finite on $\mathbb{R}^{n_1+n_2}$ for all $\omega \in \Omega$, $g_i^2(\cdot, \cdot, \omega)$ is convex on $\mathbb{R}^{n_1+n_2}$ for all $i = 1, \dots, \bar{m}_2$ and for all $\omega \in \Omega$, $g_i^2(\cdot, \omega)$ is affine on $\mathbb{R}^{n_1+n_2}$ for $i = \bar{m}_2 + 1, \dots, m_2$ and for all $\omega \in \Omega$.
2. *Slater condition.* If $\mathcal{Q}(x) < \infty$, for almost all $\omega \in \Omega$, there exists some $y(\omega)$ such that $g_i^2(x, y(\omega), \omega) < 0$ for $i = 1, \dots, \bar{m}_2$ and $g_i^2(x, y(\omega), \omega) = 0$ for $i = \bar{m}_2 + 1, \dots, m_2$.

The main purpose of these assumptions is to ensure that the resulting deterministic equivalent nonlinear program is also convex. The following theorem gives conditions for convexity of the recourse function. It follows directly from the definitions.

Theorem 34. *Under Assumptions 1 and 2, the recourse function $Q(x, \omega)$ is a convex function of x for all $\omega \in \Omega$.*

Proof: Let y_1 solve the optimization problem in (5.2) for x_1 and let y_2 solve the corresponding problem for x_2 . Consider $x = \lambda x_1 + (1 - \lambda)x_2$. In this case, $g_i^2(\lambda x_1 + (1 - \lambda)x_2, \lambda y_1 + (1 - \lambda)y_2, \omega) \leq \lambda g_i^2(x_1, y_1, \omega) + (1 - \lambda)g_i^2(x_2, y_2, \omega) \leq 0$ for each $i = 1, \dots, \bar{m}_2$. We also have that $g_i^2(\lambda x_1 + (1 - \lambda)x_2, \lambda y_1 + (1 - \lambda)y_2, \omega) = \lambda \lambda g_i^2(x_1, y_1, \omega) + (1 - \lambda)g_i^2(x_2, y_2, \omega) = 0$ for each $i = \bar{m}_2 + 1, \dots, m_2$. So, $Q(\lambda x_1 + (1 - \lambda)x_2, \omega) \leq f^2(\lambda x_1 + (1 - \lambda)x_2, \lambda y_1 + (1 - \lambda)y_2, \omega) \leq \lambda f^2(x_1, y_1, \omega) + (1 - \lambda)f^2(x_2, y_2, \omega) = \lambda Q(x_1, \omega) + (1 - \lambda)Q(x_2, \omega)$, giving the result. \square

We can also obtain continuity of the recourse function if we assume the recourse feasible region is bounded.

Theorem 35. *If the recourse feasible region is bounded for any $x \in \mathbb{R}^{n_1}$, then the function $Q(x, \omega)$ is lower semicontinuous in x for all $\omega \in \Omega$ (i.e., $Q(x, \omega)$ is a closed convex function).*

Proof: Proving lower semicontinuity is equivalent (see, e.g., Rockafellar [1969]) to showing that

$$\liminf_{x \rightarrow \bar{x}} Q(x, \omega) \geq Q(\bar{x}, \omega)$$

for any $\bar{x} \in \mathbb{R}^{n_1}$, $x \rightarrow \bar{x}$, and $\omega \in \Omega$. Suppose a sequence $x^v \rightarrow \bar{x}$. We can assume that $Q(x^v, \omega) < \infty$ for all v because there is either a subsequence of $\{x^v\}$ that is finite valued in Q or the result holds trivially.

We therefore have $g_i^2(x^v, y^v(\omega), \omega) \leq 0$ for $i = 1, \dots, \bar{m}_2$ and $g_i^2(x^v, y^v(\omega), \omega) = 0$ for $i = \bar{m}_2 + 1, \dots, m_2$ and for some $y^v(\omega)$. Hence, by continuity of each of these functions and the boundedness assumption, the $\{y^v(\omega)\}$ sequence must have some limit point, e.g., $\bar{y}(\omega)$. Thus, $g_i^2(\bar{x}, \bar{y}(\omega), \omega) \leq 0$ for $i = 1, \dots, \bar{m}_2$ and $g_i^2(\bar{x}, \bar{y}(\omega), \omega) = 0$ for $i = \bar{m}_2 + 1, \dots, m_2$. So, \bar{x} is feasible and $Q(\bar{x}, \omega) \leq f^2(\bar{x}, \bar{y}(\omega), \omega) = \lim_{v_k} f^2(x^{v_k}, y^{v_k}(\omega), \omega) = \lim_{v_k} Q(x^v, \omega)$ where v_k is a subsequence such that $y^{v_k}(\omega) \rightarrow \bar{y}(\omega)$. \square

Because integration is a linear operation on the convex function Q , we obtain the following corollaries.

Corollary 36. *The expected recourse function $\mathcal{Q}(x)$ is a convex function in x .*

Corollary 37. *The feasibility set $K_2 = \{x \mid \mathcal{Q}(x) < \infty\}$ is closed and convex.*

Corollary 38. *Under the conditions in Theorem 35, \mathcal{Q} is a lower semi-continuous function on x .*

This corollary then leads directly to the following attainability result.

Theorem 39. *Suppose the conditions in Theorem 35, K_1 is bounded, f^1 continuous, g_i^1 and g_i^2 continuous for each i , and $K_1 \cap K_2 \neq \emptyset$. Then (5.1) has a finite optimal solution and the infimum is attained.*

Proof: From Corollary 38, \mathcal{Q} is continuous on its effective domain. The continuity of g_i^1 also implies that K_1 is closed so the optimization is for a continuous, convex function over the nonempty, compact region $K_1 \cap K_2$. \square

Other results may follow for specific cases from Fenchel's duality theorem (see Rockafellar [1969]). In some cases, it may be difficult to decompose the feasibility set K_2 into $\bigcap_{\omega} K_2(\omega)$. It is possible if f^2 is always dominated by some integrable function in ω for any $y(\omega)$ feasible in the recourse problem for all x . This might be verifiable if, for example, the feasible recourse region is bounded for all $x \in K_1$. Another possibility is for special functions such as the quadratic function in Exercise 2.

We can now proceed to state optimality conditions for (5.1) as in Theorem 9. As a reminder from Section 2.10, in the following, we use ri to indicate *relative interior*.

Theorem 40. *If there exists x such that $x \in \text{ri}(\text{dom}(f^1(x)))$ and $x \in \text{ri}(\text{dom}(\mathcal{Q}(x)))$ and $g_i^1(x) < 0$ for all $i = 1, \dots, \bar{m}_1$ and $g_i^1(x) = 0$ for all $i = \bar{m}_1 + 1, \dots, m_1$, then x^* is optimal in 5.1 if and only if $x^* \in K_1$ and there exists $\lambda_i^* \geq 0$, $i = 1, \dots, \bar{m}_1$, λ_i^* , $i = \bar{m}_1 + 1, \dots, m_1$, such that $\lambda_i^* g_i^1(x^*) = 0$, $i = 1, \dots, \bar{m}_1$, and*

$$0 \in \partial f^1(x^*) + \partial \mathcal{Q}(x^*) + \sum_{i=1}^{m_1} \lambda_i^* \partial g_i^1(x^*). \quad (5.3)$$

Proof: This result is a direct extension of the general optimality conditions in nonlinear programming (see, e.g., Rockafellar [1969, Theorem 28.3]). \square

For most practical purposes, we need to obtain some decomposition of $\partial \mathcal{Q}(x)$ into subgradients of the $Q(x, \omega)$. The same argument as in Theorem 11 applies here so that

$$\partial \mathcal{Q}(x) = E_\omega[\partial Q(x, \omega)] + N(K_2, x) \quad (5.4)$$

for all $x \in K$. Moreover, if we have relatively complete recourse, we can remove the normal cone term in (5.4).

We can also develop optimality conditions that apply to the problem with explicit constraints on nonanticipativity as in Section 3.1. In this case, Problem (5.1) becomes

$$\begin{aligned} & \inf_{(x(\omega), y(\omega)) \in X} \int_{\Omega} (f^1(x(\omega)) + f^2(x(\omega), y(\omega), \omega)) \mu(d\omega) \\ & \text{s. t. } g_i^1(x(\omega)) \leq 0, \quad a.s., i = 1, \dots, \bar{m}_1, \\ & \quad g_i^1(x(\omega)) = 0, \quad a.s., i = \bar{m}_1 + 1, \dots, m_1, \\ & \quad E_{\Omega}(x(\omega)) - x(\omega) = 0, \quad a.s., \\ & \quad g_i^2(x(\omega), y(\omega), \omega) \leq 0, \quad a.s., i = 1, \dots, \bar{m}_2, \\ & \quad g_i^2(x(\omega), y(\omega), \omega) = 0, \quad a.s., i = \bar{m}_2 + 1, \dots, m_2, \\ & \quad x(\omega), y(\omega) \geq 0, \quad a.s. \end{aligned} \quad (5.5)$$

The optimality results appear in the following theorem which is proven similarly to Theorem 13.

Theorem 41. Assume that (5.5) with $X = \mathcal{L}_{\infty}(\Omega, \mathcal{B}, \mu; \mathfrak{R}^{n_1+n_2})$ is feasible, has a bounded optimal value, satisfies relatively complete recourse, and that a feasible solution $(x^*(\omega), y^*(\omega))$ is at a point satisfying the linear independence condition that any vector in $\partial f^2(x^*(\omega), y^*(\omega), \omega)$ cannot be written as a combination of some strict subset of representative vectors from $\partial g_i^2(x^*(\omega), y^*(\omega), \omega)$ for i such that $g_i^2(x^*(\omega), y^*(\omega), \omega) = 0$; then, $(x^*(\omega), y^*(\omega))$ is optimal in (5.5) if and only if there exist integrable functions on Ω , $(\lambda^*(\omega), \rho^*(\omega), \pi^*(\omega))$, $(\eta_0^{x*}(\omega), \eta_0^{y*}(\omega)) \in \partial f^2(x^*(\omega), y^*(\omega), \omega)$, and $(\eta_i^{x*}, \eta_i^{y*}) \in \partial g_i^2(x^*(\omega), y^*(\omega), \omega)$ for $i = 1, \dots, m_2$ such that, for almost all ω ,

$$\rho^*(\omega) \in \partial f^1(x^*(\omega)) + \eta_0^{x*}(\omega) \sum_{i=1}^{m_1} \lambda_i^*(\omega) \partial g_i^1(x^*(\omega)) + \sum_{i=1}^{m_2} \pi_i^*(\omega) \eta_i^{x*}(\omega), \quad (5.6)$$

$$\lambda_i^*(\omega) \geq 0, \quad \lambda_i^*(\omega) g_i^1(x^*(\omega)) = 0, \quad i = 1, \dots, \bar{m}_1, \quad (5.7)$$

$$0 = \eta_0^{y*}(\omega) + \sum_{i=1}^{m_2} \pi_i^*(\omega) \eta_i^{y*}(\omega), \quad (5.8)$$

$$\pi_i^*(\omega) \geq 0, \pi_i^*(\omega) g_i^2(x^*(\omega), y^*(\omega), \omega) = 0, \quad i = 1, \dots, \bar{m}_2, \quad (5.9)$$

and

$$E_\omega[\rho^*(\omega)] = 0. \quad (5.10)$$

Again the ρ functions represent the value of information in each of the scenarios under ω . These results can also be generalized to allow for nonseparability between the first and second stage but for our computational descriptions, this is generally not necessary.

For multiple stages, we can define models analogous to the linear version in Section 5.1. A general representation can be obtained by including the constraint information except for nonanticipativity into the objective so that the objective f^t takes on an infinite value whenever a constraint is violated. To distinguish information from period to period, we associate a filtration with the data process ω as $F := \{\Sigma^t\}_{t=1}^\infty$, where $\Sigma^t := \sigma(\bar{\omega}^t)$ is the σ -field of the *history process* $\bar{\omega}^t := \{\omega^0, \dots, \omega^t\}$, and the Σ^t satisfy $\{0, \Omega\} \subset \Sigma_0 \subset \dots \subset \Sigma$. Nonanticipativity of the decision process at time t implies that decisions must only depend on the data up to time t , i.e., \mathbf{x}^t must be Σ^t -measurable. An alternative characterization of this nonanticipative property is that $\mathbf{x}^t = E\{\mathbf{x}^t | \Sigma^t\}$ a.s., $t = 0, \dots$, where $E\{\cdot | \Sigma^t\}$ is conditional expectation with respect to the σ -field Σ^t . Using the projection operator $\Pi^t : z \rightarrow \Pi^t z := E\{z | \Sigma^t\}$, $t = 0, \dots$, this is equivalent to

$$(I - \Pi_t)x^t = 0, \quad t = 0, \dots \quad (5.11)$$

In this framework, the general multistage stochastic programming model is to find

$$\inf_{\mathbf{x} \in \mathcal{N}} E \sum_{t=0}^H f^t(\omega, x^t(\omega), x^{t+1}(\omega)), \quad (5.12)$$

where “ E ” is expectation with respect to Σ . Using our random variable boldface notation, expression (5.12) then becomes

$$\inf_{\mathbf{x} \in \mathcal{N}} E \sum_{t=0}^H \mathbf{f}^t(\mathbf{x}_t, \mathbf{x}^{t+1}), \quad (5.13)$$

with objective $z(\mathbf{x}) := E \sum_{t=0}^H \mathbf{f}_t(\mathbf{x}^t, \mathbf{x}^{t+1})$.

We can develop optimality conditions for this model that also allow $H \rightarrow \infty$. The conditions are basically the same as in previous sections (in terms of some assumption about relatively complete recourse and some regularity condition), but we need some additional assumptions to control multipliers at $H = \infty$. Detailed descriptions of these conditions and other issues appear in the papers by Rockafellar and Wets [1976a, 1976b], Dempster [1988], Flåm [1985, 1986], and Birge and Dempster [1992].

These basic results for the general model in (5.13) can be extended to results with constraints in the same way as necessary conditions in the previous sections

(Exercise 5). The only requirement is to describe the subdifferentials of f^t in terms of an objective and constraint functions (Exercise 6). The optimality conditions that extend Theorem 41 to multiple stages can then be used to decompose the multistage problem into individual period t problems. In this way, optimization may be applied at each period provided suitable multipliers are available. This property is the basis for the Lagrangian and progressive hedging algorithms described in Chapter 5.

Exercises

1. Show that the assumptions made when defining (5.1) and (5.2) imply that $Q(x, \omega)$ is a measurable function of ω for all x . (Hint: Find $\{\omega \mid Q(x, \omega) \leq \alpha\}$ for any α using a countable covering of \Re^{n_2} .)
2. Suppose f^2 is a convex, quadratic function on \Re^{n_2} for each $\omega \in \Omega$ and the constraints g_i^2 and h_j^2 are affine on \Re^{n_2} for all $i = 1, \dots, \bar{m}_2$ and $j = 1, \dots, m_2 - \bar{m}_2$. What conditions on $\xi(\omega)$ can guarantee that $K_2 = \bigcap_{\omega} K_2(\omega)$?
3. Construct an example in which the recourse function $Q(x, \omega)$ is not lower semi-continuous. (Hint: Try to make the only feasible recourse action tend to ∞ while the first-period action tends to some finite value.)
4. Show that conditions in (5.6)–(5.10) are sufficient to obtain optimality in (5.5).
5. State and prove a set of optimality conditions analogous to Theorem 35 for the multistage model in (5.13).
6. Suppose that constraints are explicitly represented by $\mathbf{g}^t(\mathbf{x}^t, \mathbf{x}^{t+1}) \leq 0$ in (4.11) instead of being incorporated into \mathbf{f}^t . Interpret the optimality conditions from Exercise 5 above in terms of the \mathbf{g}^t functions.

Chapter 4

The Value of Information and the Stochastic Solution

Stochastic programs have the reputation of being computationally difficult to solve. Many people faced with real-world problems are naturally inclined to solve simpler versions. Frequently used simpler versions are, for example, to solve the deterministic program obtained by replacing all random variables by their expected values or to solve several deterministic programs, each corresponding to one particular scenario, and then to combine these different solutions by some heuristic rule.

A natural question is whether these approaches can sometimes be nearly optimal or whether they are totally inaccurate. The theoretical answer to this is given by two concepts: the expected value of perfect information and the value of the stochastic solution. The object of this chapter is to study these two concepts. Section 4.1 introduces the expected value of perfect information. Section 4.2 gives the value of the stochastic solution. Some basic inequalities and the relationships between these quantities are given in Sections 4.3 and 4.4, respectively. Section 4.5 provides some examples of these quantities. Section 4.6 presents additional bounds.

4.1 The Expected Value of Perfect Information

The *expected value of perfect information* (*EVPI*) measures the maximum amount a decision maker would be ready to pay in return for complete (and accurate) information about the future. In the farmer's problem of Chapter 1, we saw that the farmer would greatly benefit from perfect information about future weather conditions, so that he could allocate his land optimally to the various crops.

The concept of *EVPI* was first developed in the context of decision analysis and can be found in a classical reference such as Raiffa and Schlaifer [1961]. In the stochastic programming setting, we may define it as follows. Suppose uncertainty can be modeled through a number of scenarios. Let ξ be the random variable whose realizations correspond to the various scenarios. Define

$$\min z(x, \xi) = c^T x + \min\{q^T y \mid Wy = h - Tx, y \geq 0\}$$

$$\text{s. t. } Ax = b, x \geq 0 , \quad (1.1)$$

as the optimization problem associated with one particular scenario ξ , where, as before, $\xi(\omega)^T = (q(\omega)^T, h(\omega)^T, T_{1\cdot}(\omega), \dots, T_{m_2\cdot}(\omega))$. To make the definition complete, we repeat the notation, $K_1 = \{x \mid Ax = b, x \geq 0\}$ and $K_2(\xi) = \{x \mid \exists y \geq 0 \text{ s.t. } Wy = h - Tx\}$. We define $z(x, \xi) = +\infty$ if $x \notin K_1 \cap K_2(\xi)$ and $z(x, \xi) = -\infty$ if (1.1) is unbounded below. We again use the convention $+\infty + (-\infty) = +\infty$.

We may also reasonably assume that for all $\xi \in \Xi$, there exists at least one $x \in \Re^{n_1}$ such that $z(x, \xi) < \infty$. (Otherwise, there would exist one scenario for which no feasible solution exists at all. No reasonable stochastic model could be constructed in such a situation.) This assumption implies that, for all $\xi \in \Xi$, there exists at least one feasible solution, which in turn implies the existence of at least one optimal solution. Let $\bar{x}(\xi)$ denote some optimal solution to (1.1). As in a scenario approach, we might be interested in finding all solutions $\bar{x}(\xi)$ of problem (1.1) for all scenarios and the related optimal objective values $z(\bar{x}(\xi), \xi)$.

This search is known as the *distribution problem* (as we mentioned in Section 3.1c.) because it looks for the distribution of $\bar{x}(\xi)$ and of $z(\bar{x}(\xi), \xi)$ in terms of ξ . The distribution problem can be seen as a generalization of sensitivity analysis or parametric analysis in linear programming.

Here, we assume we somehow have the ability to find these decisions $\bar{x}(\xi)$ and their objective values $z(\bar{x}(\xi), \xi)$ so that we are in a position to compute the expected value of the optimal solution, known in the literature as the *wait-and-see* solution (WS, see Madansky [1960]) where

$$\begin{aligned} WS &= E_\xi \left[\min_x z(x, \xi) \right] \\ &= E_\xi z(\bar{x}(\xi), \xi) . \end{aligned} \quad (1.2)$$

We may now compare the wait-and-see solution to the so-called *here-and-now* solution corresponding to the recourse problem (RP) defined earlier in Chapter 3 as (1.1), and we may now write that as

$$RP = \min_x E_\xi z(x, \xi) , \quad (1.3)$$

with an optimal solution, x^* .

The expected value of perfect information is, by definition, the difference between the wait-and-see and the here-and-now solution, namely,

$$EVPI = RP - WS . \quad (1.4)$$

An example was given in Chapter 1 in the farmer's problem. The wait-and-see solution value was $-\$115,406$ (when converted to a minimization problem), while the recourse solution value was $-\$108,390$. The expected value of perfect information for the farmer was then $\$7016$.

This is how much the farmer would be ready to pay each year to obtain perfect information on next summer's weather. A meteorologist could reasonably ask him to pay part of this amount to support meteorological research.

4.2 The Value of the Stochastic Solution

For practical purposes, many people would believe that finding the wait-and-see solution or equivalently solving the distribution problem is still too much work (or impossible if perfect information is just not available at any price). This is especially difficult because the wait-and-see approach delivers a set of solutions instead of one solution that would be implementable.

A natural temptation is to solve a much simpler problem: the one obtained by replacing all random variables by their expected values. This is called the *expected value problem* or *mean value problem*, which is simply

$$EV = \min_x z(x, \bar{\xi}) , \quad (2.1)$$

where $\bar{\xi} = E(\xi)$ denotes the expectation of ξ . Let us denote by $\bar{x}(\bar{\xi})$ an optimal solution to (2.1), called the *expected value solution*. Anyone aware of some stochastic programming or realizing that uncertainty is a fact of life would feel at least a little insecure about advising to take decision $\bar{x}(\bar{\xi})$. Indeed, unless $\bar{x}(\bar{\xi})$ is somehow independent of ξ , there is no reason to believe that $\bar{x}(\bar{\xi})$ is in any way near the solution of the recourse problem (1.3).

The value of the stochastic solution (first introduced in Chapter 1) is the concept that precisely measures how good or, more frequently, how bad a decision $\bar{x}(\bar{\xi})$ is in terms of (1.3). We first define the *expected result of using the EV solution* to be

$$EEV = E_{\xi}(z(\bar{x}(\bar{\xi}), \xi)) . \quad (2.2)$$

The quantity, EEV , measures how $\bar{x}(\bar{\xi})$ performs, allowing second-stage decisions to be chosen optimally as functions of $\bar{x}(\bar{\xi})$ and ξ . The value of the stochastic solution is then defined as

$$VSS = EEV - RP . \quad (2.3)$$

Recall, for example, that in Section 1.1 this value was found using $EEV = -\$107,240$ and $RP = -\$108,390$, for $VSS = \$1150$. This quantity is the cost of ignoring uncertainty in choosing a decision.

4.3 Basic Inequalities

The following relations between the defined values have been established by Madansky [1960]. Generalizations to nonlinear functions can be found in Mangasarian and Rosen [1964].

Proposition 1.

$$WS \leq RP \leq EEV . \quad (3.1)$$

Proof: For every realization, ξ , we have the relation

$$z(\bar{x}(\xi), \xi) \leq z(x^*, \xi) ,$$

where, as said before, x^* denotes an optimal solution to the recourse problem (1.3). Taking the expectation of both sides yields the first inequality. x^* being an optimal solution to the recourse problem (1.3) while $\bar{x}(\xi)$ is just one solution to (1.3) yields the second inequality. \square

Proposition 2. *For stochastic programs with fixed objective coefficients and fixed W ,*

$$EV \leq WS . \quad (3.2)$$

Proof: First, note that $z(x, \xi = (h, T)) = c^T x + Q(x, h, T) + \delta(x|Ax = b, x \geq 0)$, where $\delta(x|X)$ is the indicator function of the point x for set X , is jointly convex in x , h , and T . Now, to show that $f(\xi) = \min_x z(x, \xi)$ is convex in ξ , consider ξ_1 and ξ_2 where $z(x_1, \xi_1) = f(\xi_1)$ and $z(x_2, \xi_2) = f(\xi_2)$, then

$$\begin{aligned} \lambda f(\xi_1) + (1 - \lambda)f(\xi_2) &= \lambda z(x_1, \xi_1) + (1 - \lambda)z(x_2, \xi_2) \\ &\geq z(\lambda(x_1, \xi_1) + (1 - \lambda)(x_2, \xi_2)) \\ &\geq \min_x z(x, \lambda\xi_1 + (1 - \lambda)\xi_2) \\ &= f(\lambda\xi_1 + (1 - \lambda)\xi_2), \end{aligned}$$

establishing convexity of $f(\xi)$. Now, Jensen's inequality (Jensen [1906]) states that for any convex function $f(\xi)$ of ξ , $Ef(\xi) \geq f(E\xi)$. \square

Proposition 2 does not hold for general stochastic programs. Indeed, if we consider q only to be stochastic, by Theorem 3.5 the function $z(x, \xi)$ is a concave function of ξ and Jensen's inequality does not apply. An example of a program where $EV > WS$ is given in Exercise 3.

Other bounds can be obtained. We give two more examples of such bounds here.

Proposition 3. *Let x^* represent an optimal solution to the recourse problem (1.3) and let $\bar{x}(\xi)$ be a solution to the expected value problem (1.4). Then*

$$RP \geq EEV + (x^* - \bar{x}(\xi))^T \eta , \quad (3.3)$$

where $\eta \in \partial E_{\xi}z(\bar{x}(\bar{\xi}), \xi)$, the subdifferential set of $E_{\xi}z(x, \xi)$ at $\bar{x}(\bar{\xi})$.

Proof: By convexity of $E_{\xi}z(x, \xi)$, the subgradient inequality applied at point x_1 implies that for any x_2 the relation $E_{\xi}z(x_2, \xi) \geq E_{\xi}z(x_1, \xi) + (x_2 - x_1)^T \eta$ holds. The proposition follows by application of this relation for $x_1 = \bar{x}(\bar{\xi})$ and $x_2 = x^*$, by noting that $RP = E_{\xi}z(x^*, \xi)$ and $EEV = E_{\xi}z(\bar{x}(\bar{\xi}), \xi)$. \square

The last bound is obtained by considering a slightly different version of the recourse problem, defined as follows:

$$\begin{aligned} \min z_u(x, \xi) &= c^T x + \min\{q^T y \mid Wy \geq h(\xi) - Tx, y \geq 0\} \\ \text{s. t. } Ax &= b, \\ x &\geq 0. \end{aligned} \tag{3.4}$$

Problem (3.4) differs from problem (1.1) because in (3.4) only the right-hand side is stochastic and the second-stage constraints are inequalities. It is not difficult to observe that all definitions and relations also apply to z_u . If we further assume that $h(\xi)$ is bounded above, then an additional inequality results.

Proposition 4. Consider problem (3.4) and the related definition

$$RP = \min_x E_{\xi}z_u(x, \xi).$$

Assume further that $h(\xi)$ is bounded above by a fixed quantity h_{\max} . Let x_{\max} be an optimal solution to $z_u(x, h_{\max})$. Then

$$RP \leq z_u(x_{\max}, h_{\max}). \tag{3.5}$$

Proof: For any ξ in Ξ and any $x \in K_1$, a feasible solution to $Wy \geq h_{\max} - Tx, y \geq 0$, is also a feasible solution to $Wy \geq h(\xi) - Tx, y \geq 0$. Hence $z_u(x, h_{\max}) \geq z_u(x, h(\xi))$. Thus $z_u(x, h_{\max}) \geq E_{\xi}z_u(x, h(\xi))$, hence $z_u(x, h_{\max}) \geq \min_x E_{\xi}z_u(x, h(\xi)) = RP$. \square

4.4 The Relationship between *EVPI* and *VSS*

The quantities, *EVPI* and *VSS*, are often different, as our examples have shown. This section describes the relationships that exist between the two measures of uncertainty effects.

From the inequalities in the previous section, the following proposition holds.

Proposition 5.

a. For any stochastic program,

$$0 \leq EVPI, \tag{4.1}$$

$$0 \leq VSS . \quad (4.2)$$

b. For stochastic programs with fixed recourse matrix and fixed objective coefficients,

$$EVPI \leq EEV - EV , \quad (4.3)$$

$$VSS \leq EEV - EV . \quad (4.4)$$

The proposition indicates that the *EVPI* and the *VSS* are (both) nonnegative (anyone would be surprised if this was not true) and are both bounded above by the same quantity $EEV - EV$, which is easily computable. It follows that when $EV = EEV$, both the *EVPI* and *VSS* vanish. A sufficient condition for this to happen is to have $\bar{x}(\xi)$ independent of ξ . This means that optimal solutions are insensitive to the value of the random elements. In such situations, finding the optimal solution for one particular ξ (or for $\bar{\xi}$) would yield the same result, and it is unnecessary to solve a recourse problem. Such extreme situations rarely occur.

From these observations, three lines of research have been addressed. The first one studies relationships between *EVPI* and *VSS*. It is illustrated in the sequel of this paragraph by showing an example where *EVPI* is zero and *VSS* is not and an example of the reverse. The second one studies classes of problems for which one can observe or theorize that the *EVPI* is low. Examples and counterexamples are given in Section 4.5. The third one studies refined bounds on *EVPI* and *VSS*. Results about refined upper and lower bounds on *EVPI* and *VSS* appear in Section 4.6.

We thus end this section by showing examples taken from Birge [1982] that illustrate cases in which one of the two concepts (*EVPI* and *VSS*) is null and the other is positive.

a. $EVPI = 0$ and $VSS \neq 0$

Consider the following problem

$$\begin{aligned} z(x, \xi) &= x_1 + 4x_2 + \min\{y_1 + 10y_2^+ + 10y_2^- \mid \\ &\quad y_1 + y_2^+ - y_2^- = \xi + x_1 - 2x_2, y_1 \leq 2, y \geq 0\} \\ \text{s. t. } &x_1 + x_2 = 1 , \\ &x \geq 0 , \end{aligned} \quad (4.5)$$

where the random variable ξ follows a uniform density over $[1, 3]$. For a given x and ξ , we may conclude that

$$y^*(x, \xi) = \begin{cases} y_1 = \xi + x_1 - 2x_2, y_2 = 0 & \text{if } 0 \leq \xi + x_1 - 2x_2 \leq 2, \\ y_1 = 2, y_2^+ = \xi + x_1 - 2x_2 - 2 & \text{if } \xi + x_1 - 2x_2 > 2, \\ y_2^- = 2x_2 - \xi - x_1 & \text{if } \xi + x_1 - 2x_2 < 0, \end{cases}$$

so that

$$z(x, \xi) = \begin{cases} 2x_1 + 2x_2 + \xi & \text{if } 0 \leq \xi + x_1 - 2x_2 \leq 2, \\ -18 + 11x_1 - 16x_2 + 10\xi & \text{if } \xi + x_1 - 2x_2 > 2, \\ -9x_1 + 24x_2 - 10\xi & \text{if } \xi + x_1 - 2x_2 < 0. \end{cases}$$

Given the first-stage constraint $x_1 + x_2 = 1$, one has $z(x, \xi) = 2 + \xi$ in the first of these three regions. Now, using the first-stage constraint and the definition of the regions, one can easily check that $z(x, \xi) \geq 2 + \xi$ in the other two regions. Hence, any $\bar{x} \in \{(x_1, x_2) \mid x_1 + x_2 = 1, x \geq 0\}$ is an optimal solution of (4.5) for $-x_1 + 2x_2 \leq \xi \leq 2 - x_1 + 2x_2$, or equivalently for $2 - 3x_1 \leq \xi \leq 4 - 3x_1$.

In particular, $(\frac{1}{3}, \frac{2}{3})$ is optimal for all ξ , $(0, 1)$ is optimal for all $\xi \in [2, 3]$, and $(1, 0)$ is optimal for $\xi = \{1\}$.

Taking $\bar{x}(\xi) = (\frac{1}{3}, \frac{2}{3})$ for all ξ leads to the conclusion that $\bar{x}(\xi)$ is identical for all ξ , hence $WS = RP = 4$, so that $EVPI = 0$. On the other hand, solving $z(x, \xi = 2)$ may yield a different solution, for example, $\bar{x}(2) = (0, 1)$, with $EV = 4$.

In that case,

$$EEV = E_{\xi \leq 2}(24 - 10\xi) + E_{\xi \geq 2}(2 + \xi) = \frac{27}{4},$$

so that $VSS = 11/4$.

Because linear programs often include multiple optimal solutions, this type of situation is far from exceptional.

b. $VSS = 0$ and $EVPI \neq 0$

We consider the same function $z(x, \xi)$ with $\xi \in \{0, \frac{3}{2}, 3\}$, with each event occurring with probability $1/3$.

For $\xi = 0$, $\bar{x}(0) = \{x \mid x_1 + x_2 = 1, \frac{2}{3} \leq x_1 \leq 1\}$.

For $\xi = 3/2$, $\bar{x}(3/2) = \{x \mid x_1 + x_2 = 1, 1/6 \leq x_1 \leq 5/6\}$.

For $\xi = 3$, $\bar{x}(3) = \{x \mid x_1 + x_2 = 1, 0 \leq x_1 \leq 1/3\}$.

Let us take $\bar{x}(3/2) = (2/3, 1/3)$. Then $EV = z(\bar{x}, 3/2) = 2 + 3/2 = 7/2$, and $EEV = 2 + \frac{1}{3}(0 + \frac{3}{2} + 12) = 2 + \frac{13}{2} = 13/2$.

No single decision is optimal for the three cases, so we expect *EVPI* to be nonzero. In the wait-and-see solution, it is possible for all three cases to take a different optimal solution, such as $\bar{x}(0) = (1, 0)$, $\bar{x}(3/2) = (1/2, 1/2)$, and $\bar{x}(3) = (0, 1)$, yielding

$$\begin{aligned} WS &= \frac{1}{3}(1+1) + \frac{1}{3}\left(\frac{5}{2}+1\right) + \frac{1}{3}(4+1) \\ &= \frac{2}{3} + \frac{7}{6} + \frac{5}{3} = \frac{21}{6} = \frac{7}{2}. \end{aligned}$$

The recourse solution is obtained by solving the stochastic program $\min E_{\xi}(z(x, \xi))$, which yields $x^* = (2/3, 1/3)$ with the *RP* value equal to the *EEV* value. Hence,

$$EV = WS = 7/2 \leq RP = 13/2 = EEV,$$

which means *EVPI* = 3 while *VSS* = 0.

4.5 Examples

There has always been a strong interest in trying to have a better understanding of when the *EVPI* and *VSS* take large values and when they take low values. A definite answer to this question would greatly simplify the practice of stochastic programming. Only those programs with large *EVPI* or *VSS* would require the solution of a stochastic program. Interested readers may find detailed examples in the field of energy policy and exhaustible resources. Manne [1974] provides an example where *EVPI* is low, while H.P. Chao [1981] elaborates general conditions for *EVPI* to be low on a resource exhaustion model. By introducing other types of uncertainty, Louveaux and Smeers [2011] and Birge [1988a] show related examples where *EVPI* and/or *VSS* is large.

In this section, we provide simple examples to show that no general answer is available. It is usually felt that using stochastic programming is more relevant when there is more randomness in the problem. To translate this feeling into a more precise statement, we would, for example, expect that for a given problem, *EVPI* and *VSS* would increase when the variances of the random variables increase. In the following example, we show that this may or may not be the case.

Example 1

Let ξ be a single random variable taking the two values ξ_1 and ξ_2 , with probability p_1 and p_2 , respectively, where $p_2 = 1 - p_1$. Let $\bar{\xi} = E[\xi] = 1/2$. Let x be a single decision variable. Consider the recourse problem:

$$\begin{aligned} \min & 6x + 10E_{\xi}|x - \xi| \\ \text{s. t. } & x \geq 0. \end{aligned}$$

- (a) Let $\xi_1 = 1/3$, $\xi_2 = 2/3$, $p_1 = p_2 = 1/2$ serve as the reference setting. We compute $EVPI = 2/3$ and $VSS = 1$. We also observe that the variance, $\text{Var}(\xi) = 1/36$.
- (b) Consider the case $\xi_1 = 0$, $\xi_2 = 1$ again with equal probability $1/2$ (and unchanged expectation). The variance $\text{Var}(\xi)$ is now $1/4$, 9 times higher. We now obtain $EVPI = 2$ and $VSS = 3$, showing an example where both values clearly increase with the variance of ξ .
- (c) Consider the case $\xi_1 = 0$, $\xi_2 = 5/8$ with probability $p_1 = 0.2$ and $p_2 = 0.8$, respectively. Again, $\bar{\xi} = 0.5$. Now, $\text{Var}(\xi) = 1/16$, larger than in (a). We obtain $EVPI = 2$, larger than in (a) but $VSS = 0$. Knowing this result in advance would mean that the solution of the deterministic problem with $\bar{\xi} = E\xi$ delivers the optimal solution (although $EVPI$ is three times larger than in (a)).
- (d) Consider the case $\xi_1 = 0.4$, $\xi_2 = 0.8$ with $p_1 = 0.75$ and $p_2 = 0.25$, always with $\bar{\xi} = 0.5$. Now, $\text{Var}(\xi) = 0.03$, slightly larger than in (a). We now observe $EVPI = 0.4$ and $VSS = 1.1$, namely the opposite behavior from (c), a decrease in $EVPI$ and an increase in VSS .
- (e) It is also felt that a more “difficult” stochastic program would induce higher $EVPI$ and VSS . One such case would be to have integer decision variables instead of continuous ones. Exercise 3 of Section 1.1 shows that, with first-stage integer variables for the farming problem, VSS remains almost unchanged while $EVPI$ even decreases. On the other hand, Exercise 4 of that section shows that with second-stage integer variables, both $EVPI$ and VSS strongly increase. It would probably not be difficult to reach different conclusions by suitably changing the data.

We may conclude from these simple examples that a general rule is unlikely to be found. One alternative to such a rule is to consider bounds on the information and solution value quantities that require less than complete solutions. We discuss these bounds in the next section.

4.6 Bounds on *EVPI* and *VSS*

Bounds on *EVPI* and *VSS* rely on constructing intervals for the expected value of solutions of linear programs representing *WS*, *RP*, and *EEV*. The simplest bounds stem from the inequalities in Proposition 5. The *EVPI* bound was suggested in Avriel and Williams [1970] while the *VSS* form appears in Birge [1982]. Many other bounds are possible with different limits on the defining quantities. In the remainder of this section, we consider refined bounds that particularly address the value of the stochastic solution. More general approaches to bound expectations of value functions appear in Chapter 8.

The *VSS* bounds were developed in Birge [1982]. To find them, we consider a simplified version of the stochastic program, where only the right-hand side is stochastic ($\xi = h(\omega)$) and Ξ is finite. Let $\xi^1, \xi^2, \dots, \xi^K$ index the possible

realizations of ξ , and p^k , $k = 1, \dots, K$ be their probabilities. It is customary to refer to each realization ξ^k of ξ as a *scenario* k .

To refine the bounds on VSS, we consider a *reference scenario*, say ξ^r . Two classical reference scenarios are $\bar{\xi}$, the expected value of ξ , or the worst-case scenario (for example, the one with the highest demand level for problems when costs have to be minimized under the restriction that demand must be satisfied). Note that in both situations the reference scenario may not correspond to any of the possible scenarios in Ξ . This is obvious for $\bar{\xi}$. The worst-case scenario is, however, a possible scenario when, for example, ξ is formed by components that are independent random variables. If the random variables are not independent, then a meaningful worst-case scenario may be more difficult to construct. Let $p^r = P(\xi = \xi^r)$ be the reference scenario's probability.

The *pairs subproblem* of ξ^r and ξ^k is defined as

$$\begin{aligned} \min z^P(x, \xi^r, \xi^k) &= c^T x + p^r q^T y(\xi^r) + (1 - p^r) q^T y(\xi^k) \\ \text{s. t.} \quad Ax &= b, \\ Wy(\xi^r) &= \xi^r - Tx, \\ Wy(\xi^k) &= \xi^k - Tx, \\ x, y &\geq 0. \end{aligned}$$

Let $(\bar{x}^k, \bar{y}^k, y(\xi^k))$ denote an optimal solution to the pairs subproblem and z_k the optimal objective value $z^P(\bar{x}^k, \bar{y}^k, y(\xi^k))$. We may see the pairs subproblem as a stochastic programming problem with two possible realizations ξ^r and ξ^k , with probability p^r and $1 - p^r$, respectively.

Two particular cases of the pairs subproblem are of interest. First, observe that $z^P(x, \xi^r, \xi^r)$ is well-defined and is in fact $z(x, \xi^r)$, the deterministic problem for which the only scenario is the reference scenario. Next, observe that if the reference scenario is not a possible scenario, $p^r = P(\xi = \xi^r) = 0$, then $z^P(x, \xi^r, \xi^k)$ becomes simply $z(x, \xi^k)$.

We now show the relations between the pairs subproblems and the recourse problem. To do this, we define the *sum of pairs expected values*, denoted by *SPEV*, to be

$$SPEV = \frac{1}{1 - p^r} \sum_{k=1}^K p^k \min z^P(x, \xi^r, \xi^k).$$

Again, observe that this definition still makes sense when scenario r is not possible. In that case, however, it is not really a new concept.

Proposition 6. *When the reference scenario is not in Ξ , then $SPEV = WS$.*

Proof: As we observed before, when $p^r = 0$, the pairs subproblems $z^P(x, \xi^r, \xi^k)$ coincide with $z(x, \xi^k)$. Hence, $SPEV = \sum_{\substack{k=1 \\ k \neq r}}^K p^k \min z(x, \xi^k)$, which by definition (1.2) is WS . \square

In general, the $SPEV$ is related to WS and RP as follows.

Proposition 7. $WS \leq SPEV \leq RP$.

Proof: Let us first prove the first inequality. By definition,

$$SPEV = \sum_{\substack{k=1 \\ k \neq r}}^K p^k \frac{(c^T \bar{x}^k + p^r q^T \bar{y}^k + (1-p^r)q^T y(\xi^k))}{1-p^r},$$

where $(\bar{x}^k, \bar{y}^k, y(\xi^k))$ is a solution to the pairs subproblem of ξ^r and ξ^k . By the constraint definition in the pairs subproblem, the solution (\bar{x}^k, \bar{y}^k) is feasible for the problem $z(x, \xi^r)$ so that

$$c^T \bar{x}^k + q^T \bar{y}^k \geq \min z(x, \xi^r) = z_r^*.$$

Weighting $c^T x^k$ with a p^r and a $(1-p^r)$ term, we obtain:

$$SPEV = \sum_{\substack{k=1 \\ k \neq r}}^K \frac{p^k [p^r(c^T \bar{x}^k + q^T \bar{y}^k) + (1-p^r)(c^T \bar{x}^k + q^T y(\xi^k))] }{1-p^r},$$

which, by the property just given, is bounded by

$$SPEV \geq \sum_{k \neq r} \frac{p^k \cdot p^r \cdot z_r^*}{1-p^r} + \sum_{k \neq r} p^k (c^T \bar{x}^k + q^T y(\xi^k)).$$

Now, we simplify the first term and bound $c^T \bar{x}^k + q^T y(\xi^k)$ by z_k^* in the second term, because $(\bar{x}, y(\xi^k))$ is feasible for $\min z(x, \xi^k) = z_k^*$. Thus,

$$SPEV \geq p^r z_r^* + \sum_{k \neq r} p^k z_k^* = WS.$$

For the second inequality, let $x^*, y^*(\xi^k), k = 1, \dots, K$, be an optimal solution to the recourse problem. For simplicity, we assume here that $r \in \Xi$. By the constraint definitions, $(x^*, y^*(\xi^r), y^*(\xi^k))$ is feasible for the PAIRS subproblem of ξ^r and ξ^k . This implies

$$c^T \bar{x}^k + p^r q^T \bar{y}^k + (1-p^r)q^T y(\xi^k) \leq c^T x^* + p^r q^T y^*(\xi^r) + (1-p^r)q^T y^*(\xi^k).$$

If we take the weighted sums of these inequalities for all $k \neq r$, with p^k as the weight of the k th inequality, the weighted sum of the left-hand side elements is, by definition, equal to $(1-p^r) \cdot SPEV$ and the weighted sum of the right-hand side elements is

$$\sum_{\substack{k=1 \\ k \neq r}}^K p^k (c^T x^* + p^r q^T y^*(\xi^r) + (1-p^r)q^T y^*(\xi^k))$$

$$\begin{aligned}
&= (1 - p^r) \left[c^T x^* + p^r q^T y^*(\xi^r) + \sum_{k \neq r} p^k q^T y^*(\xi^k) \right] \\
&= (1 - p^r) \left[c^T x^* + \sum_{k=1}^K p^k q^T y^*(\xi^k) \right] = (1 - p^r) RP,
\end{aligned}$$

which proves the desired inequality. \square

To obtain upper bounds on RP that relate to the pairs subproblem, we generalize the VSS definition. Let $z(x, \xi^r)$ be the deterministic problem associated with scenario ξ^r (remember ξ^r need not necessarily be a possible scenario) and \bar{x}^r an optimal solution to $\min_x z(x, \xi^r)$. We may then define the expected value of the reference scenario,

$$EVRS = E_{\xi} z(\bar{x}^r, \xi),$$

and the value of a stochastic solution to be

$$VSS = EVRS - RP.$$

Note that VSS is still nonnegative, because \bar{x}^r is either a feasible solution to the recourse problem and $EVRS \geq RP$ or an infeasible solution so that $EVRS = +\infty$.

Now, as before, let $(\bar{x}^k, \bar{y}^k, y(\xi^k))$ be optimal solutions to the pairs subproblem of ξ^r and ξ^k , $k = 1, \dots, K$. Define the expectations of pairs expected value to be

$$EPEV = \min_{k=1, \dots, K \cup \{r\}} E_{\xi} z(\bar{x}^k, \xi).$$

Proposition 8. $RP \leq EPEV \leq EVRS$.

Proof: The three values are the optimal value of the recourse function $\min_x E_{\xi} z(x, \xi)$ over smaller and smaller feasibility sets: the first one over all feasible x in $K_1 \cap K_2$, the second one over $x \in K_1 \cap K_2 \cap \{\bar{x}^k, k = 1, \dots, K \cup \{r\}\}$, and the third one over $\bar{x}^r \cap K_1 \cap K_2$. \square

Putting these two propositions together, one obtains the following theorem.

Theorem 9. $0 \leq EVRS - EPEV \leq VSS \leq EVRS - SPEV \leq EVRS - WS$.

We apply these concepts in the following example.

Example 2

Consider the problem to find:

$$\min 3x_1 + 2x_2 + E_{\xi} \min(-15y_1 - 12y_2)$$

$$\begin{aligned} \text{s. t. } & 3\mathbf{y}_1 + 2\mathbf{y}_2 \leq x_1, \\ & 2\mathbf{y}_1 + 5\mathbf{y}_2 \leq x_2, \\ & .8\xi_1 \leq \mathbf{y}_1 \leq \xi_1, \\ & .8\xi_2 \leq \mathbf{y}_2 \leq \xi_2, \\ & x, y \geq 0, \end{aligned}$$

where $\xi_1 = 4$ or 6 and $\xi_2 = 4$ or 8 , independently of each other, with probability $1/2$ each.

This example can be seen as an investment decision in two resources x_1 and x_2 , which are needed in the second-stage problem to cover at least 80% of the demand. In this situation, the EEV and WS answers are totally inconclusive.

Table 1 gives the various solutions under the four scenarios, the optimal objective values under these scenarios and the WS value. It also describes the EV value under the expected value scenario $\bar{\xi} = (5, 6)^T$. Note that this scenario is not one of those possible. The optimal solution $\bar{x}(\bar{\xi}) = (24.6, 34)^T$ is infeasible for the stochastic problem so that EEV is set to be $+\infty$.

Table 1 Solutions and optimal values under the four scenarios and the expected value scenario.

Scenario	First-Stage Solution	Second-Stage Solution	Optimal Value $z(\bar{x}(\xi), \xi)$
1. (4,4)	(18.4, 24)	(4, 3.2)	4.8
2. (6,4)	(24.4, 28)	(6, 3.2)	0.8
3. (4,8)	(24.8, 40)	(4, 6.4)	17.6
4. (6,8)	(30.8, 44)	(6, 6.4)	13.6
			$WS = 9.2$
$\bar{\xi} = (5, 6)$	(24.6, 34)	(5, 4.8)	$EV = 9.2$
			$EEV = +\infty$

It follows from Table 1 that $EV = WS = 9.2 \leq RP \leq EEV = +\infty$. This relation is of no help: we can only conclude from it that $EVPI$ is somewhere between 0 and $+\infty$, and so is VSS . These statements could have been made without any computation.

It is in such situations that the pairs subproblems are of great interest. Because the problem under consideration is an investment problem with demand satisfaction constraints, the most logical reference scenario corresponds to the largest demand, $\xi^r = (6, 8)^T$, and not to the mean demand $\bar{\xi}$.

This will force the first-stage decisions to take demand satisfaction under the maximal demand into consideration, so that decisions taken under the pairs subproblem are feasible for the recourse problem. Due to independence, ξ^r is one of the possible realizations of ξ , with $p^r = 1/4$.

The pairs subproblems of ξ^r and ξ^k are

$$\begin{aligned} \min & 3x_1 + 2x_2 - \frac{1}{4}(15y_1^r + 12y_2^r) - \frac{3}{4}(15y_1 + 12y_2) \\ \text{s. t. } & x_1 \geq 27.2, \quad 3y_1^r + 2y_2^r \leq x_1, \quad 3y_1 + 2y_2 \leq x_1, \\ & x_2 \geq 41.6, \quad 2y_1^r + 5y_2^r \leq x_2, \quad 2y_1 + 5y_2 \leq x_2, \\ & 4.8 \leq y_1^r \leq 6, \quad .8\xi_1^k \leq y_1 \leq \xi_1^k, \\ & 6.4 \leq y_2^r \leq 8, \quad .8\xi_2^k \leq y_2 \leq \xi_2^k, \\ & y \geq 0. \end{aligned}$$

The bounds on x_1 and x_2 are induced by the feasibility for the reference scenarios.

Table 2 gives the solutions of the pairs subproblems for the three scenarios (other than the reference scenario), the $SPEV$, the $EVRS$, and the $EPEV$ values.

Table 2 Pairs subproblems solutions.

Pairs Subproblem	First-Stage Solution	Second-Stage under Reference Sc.	Second-Stage under ξ_k	Objective Value z^P
1. (4,4), r	(27.2, 41.6)	(4.8, 6.4)	(4,4)	46.6
2. (6,4), r	(27.2, 41.6)	(4.8, 6.4)	(6,4)	24.1
3. (4,8), r	(27.2, 41.6)	(4.8, 6.4)	(4, 6.72)	22.12
				$SPEV = 30.94$
		$EPEV = \min_k E_{\xi} z(\bar{x}(\xi^k), \xi) = E_{\xi} z(27.2, 41.6, \xi) = 30.94$		
		$EVRS = E_{\xi} z((30.8, 44), \xi) = 40.6$		

This time, the relations one can derive from this table are strongly conclusive:

$$WS = 9.2 \leq SPEV = 30.94 \leq RP \leq EPEV = 30.94 \leq EVRS = 40.6$$

implies $RP = 30.94$ and $(27.2, 41.6)^T$ is an optimal solution.

Exercises

1. Show that Proposition 1 still holds if some of the x and/or y must be integer.
2. Consider Example 3.5 (with recourse function given in (3.3.1)) with a single first-stage decision x with first-stage cost $c \cdot x$ and

$$Q(x, \xi) = \min \{2y_1 + y_2 \mid y_1 \geq x - \xi, y_2 \geq \xi - x, y \geq 0, \text{ integer}\}$$

with $\xi = 1$ or 2 with probability $1/2$ each. Show:

- (a) If x must be integer, then $EV > WS$ for any value of $c \geq 0$.
- (b) If x is continuous, then $EV = WS$ for $0 \leq c \leq 1$ and $EV > WS$ for $c > 1$.
Beware that y is always integer; the discussion here concerns the effect of x 's being integer or not.

3. Consider the following stochastic program

$$\min_{x \geq 0} 2x + E_\xi\{\xi \cdot y \mid y \geq 1 - x, y \geq 0\},$$

and ξ takes on values 1 and 3 with probability $3/4$ and $1/4$, respectively.
Show that in this case $EV > WS$.

4. Consider the following two-stage program:

$$\begin{aligned} \min & 2x_1 + x_2 + E_\xi(-3y_1 - 4y_2 \mid \\ & y_1 + 2y_2 \geq \xi_1, y_1 \leq x_1, y_2 \leq x_2, y_2 \leq \xi_2, y \geq 0) \\ \text{s. t. } & x_1 + x_2 \leq 7, x_1, x_2 \geq 0, \end{aligned}$$

where ξ can take the values $\binom{3}{2}, \binom{5}{3}, \binom{7}{3}$ with probability $1/3$ each.

- (a) Choose the scenario $\binom{7}{3}$ as the reference scenario. Define the problem $z(x, \xi)$ for this reference scenario. Its optimal solution gives the optimal first-stage decision $x_1 = 4, x_2 = 3$. Compute the $EVRS$ value.
- (b) State the pairs subproblem for $\binom{3}{2}$ and the reference scenario.
- (c) The solution of the pairs subproblem for $\binom{3}{2}$ and the reference scenario has first-stage optimal solutions $x_1 = 5, x_2 = 2$; the solution of the pairs subproblem for $\binom{5}{3}$ and the reference scenario has first-stage optimal solutions $x_1 = 4, x_2 = 3$. Compute the values of the two pairs subproblems. Compute the $SPEV$ value. What relation holds for the recourse problem value?

- 5. Adapt the proofs in Proposition 7 for the case where $r \notin \Xi$.
- 6. Prove that the bounds in this chapter remain valid under general constraints $x \in X$ and $y \in Y(x)$ that may, for example, involve integrality restrictions. (Sandikçi, Kong, and Schaefer [2009]).
- 7. Fill in the corresponding entries for Tables 1 and 2 with the added restriction that all decision variables must be integers.

Part III

Solution Methods

Chapter 5

Two-Stage Recourse Problems

Computation in stochastic programs with recourse has focused on two-stage problems with finite numbers of realizations. This problem was introduced in the farming example of Chapter 1. As we saw in the capacity expansion model, this problem can also represent multiple stages of decisions with block separable recourse and it provides a foundation for multistage methods. The two-stage problem is, therefore, our primary model for computation.

The general model is to choose some initial decision that minimizes current costs plus the expected value of future recourse actions. With a finite number of second-stage realizations and all linear functions, we can always form the full deterministic equivalent linear program or extensive form. With many realizations, this form of the problem becomes quite large. Methods that ignore the special structure of stochastic linear programs become quite inefficient (as some of the results in Section 5.1d. show). Taking advantage of structure is especially beneficial in stochastic programs and is the focus of much of the algorithmic work in this area.

The method used most frequently is based on building an outer linearization of the recourse cost function and a solution of the first-stage problem plus this linearization. This cutting plane technique is called the *L-shaped method* in stochastic programming. Section 5.1 describes the basic *L*-shaped method and describes the cut construction in some detail. Section 5.1c. gives a formal proof of convergence of the *L*-shaped method while the following subsections continue this development with a discussion of enhancements of the *L*-shaped method in terms of multicuts and bunching of realizations. Variants adding nonlinear regularized terms are studied in Section 5.2 and with quadratic objectives in Section 5.3. Other extensions of the L-shaped method include its use with bounding techniques, which will be considered in Chapter 8, and in combination with sampling methods, which will be studied in Chapter 9.

The remainder of this chapter discusses alternative algorithms. In Section 5.6, we describe alternative decomposition procedures. The first method is an inner linearization, or Dantzig-Wolfe decomposition approach, that solves the dual of the *L*-shaped method problem. The other approach is a primal form of inner linearization based on generalized programming. Section 5.5 considers direct approaches

to the extensive form through efficient extreme point and interior point methods. We discuss basis factorization and its relationship to decomposition methods. We also present interior point approaches and the use of a special stochastic programming structure for these algorithms. Methods based on nonlinear optimization of the Lagrangian appear in Section 5.8. Section 5.9 discusses additional methods and considerations of computational complexity.

5.1 The *L*-Shaped Method

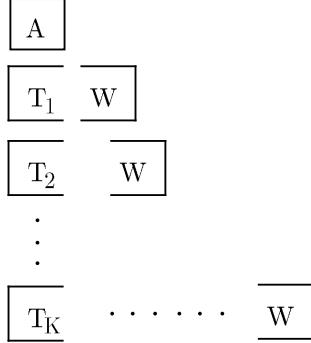
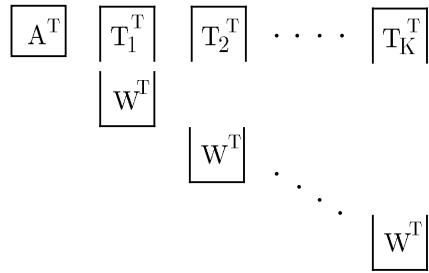
Consider the general formulation in (3.1.2) or (3.1.5). The basic idea of the *L*-shaped method is to approximate the nonlinear term in the objective of these problems. A general principle behind this approach is that, because the nonlinear objective term (the *recourse function*) involves a solution of all second-stage recourse linear programs, we want to avoid numerous function evaluations for it. We therefore use that term to build a master problem in x , but we only evaluate the recourse function exactly as a subproblem.

To make this approach possible, we assume that the random vector ξ has finite support. Let $k = 1, \dots, K$ index its possible realizations and let p_k be their probabilities. Under this assumption, we may now write the deterministic equivalent program in the extensive form. This form is created by associating one set of second-stage decisions, say, y_k , to each realization ξ , i.e., to each realization of q_k , h_k , and T_k . It is a large-scale linear problem that we can define as the *extensive form* (*EF*):

$$(EF) \quad \begin{aligned} & \min c^T x + \sum_{k=1}^K p_k q_k^T y_k \\ & \text{s. t.} \quad Ax = b, \\ & \quad T_k x + W y_k = h_k, \quad k = 1, \dots, K; \\ & \quad x \geq 0, \quad y_k \geq 0, \quad k = 1, \dots, K. \end{aligned} \tag{1.1}$$

An example of an extensive form has been given for the farmer's problem in Chapter 1 (Model (1.1.2)).

The block structure of the extensive form appears in Figure 1. This picture has given rise to the name, *L-shaped method* for the following algorithm. Taking the dual of the extensive form, one obtains a dual block-angular structure, as in Figure 2. Therefore it seems natural to exploit this dual structure by performing a Dantzig-Wolfe [1960] decomposition (inner linearization) of the dual or a Benders [1962] decomposition (outer linearization) of the primal. This method has been extended in stochastic programming to take care of feasibility questions and is known as Van Slyke and Wets's [1969] *L*-shaped method. It proceeds as follows.

**Fig. 1** Block structure of the two-stage extensive form.**Fig. 2** Block angular structure of the two-stage dual.

***L*-Shaped Algorithm**

Step 0. Set $r = s = v = 0$.

Step 1. Set $v = v + 1$. Solve the linear program (1.2)–(1.4)

$$\min \quad z = c^T x + \theta \quad (1.2)$$

$$\text{s. t.} \quad Ax = b,$$

$$D_\ell x \geq d_\ell, \quad \ell = 1, \dots, r, \quad (1.3)$$

$$E_\ell x + \theta \geq e_\ell, \quad \ell = 1, \dots, s, \quad (1.4)$$

$$x \geq 0, \quad \theta \in \Re.$$

Let (x^v, θ^v) be an optimal solution. If no constraint (1.4) is present, θ^v is set equal to $-\infty$ and is not considered in the computation of x^v .

Step 2. Check if $x \in K_2$. If not, add at least one cut (1.3) and return to Step 1. Otherwise, go to Step 3.

Step 3. For $k = 1, \dots, K$ solve the linear program

$$\begin{aligned} \min w &= q_k^T y \\ \text{s. t. } Wy &= h_k - T_k x^\nu, \\ y &\geq 0. \end{aligned} \tag{1.5}$$

Let π_k^ν be the simplex multipliers associated with the optimal solution of Problem k of type (1.5). Define

$$E_{s+1} = \sum_{k=1}^K p_k \cdot (\pi_k^\nu)^T T_k \tag{1.6}$$

and

$$e_{s+1} = \sum_{k=1}^K p_k \cdot (\pi_k^\nu)^T h_k. \tag{1.7}$$

Let $w^\nu = e_{s+1} - E_{s+1} x^\nu$. If $\theta^\nu \geq w^\nu$, stop; x^ν is an optimal solution. Otherwise, set $s = s + 1$, add to the constraint set (1.4), and return to Step 1.

The method consists of solving an approximation of (3.1.2) by using an outer linearization of \mathcal{Q} . This approximation is program (1.2)–(1.4). It is called the *master program*. It consists of finding a proposal x , sent to the second stage. Two types of constraints are sequentially added: (i) *feasibility cuts* (1.3) determining $\{x \mid \mathcal{Q}(x) < +\infty\}$ and (ii) *optimality cuts* (1.4), which are linear approximations to \mathcal{Q} on its domain of finiteness. We first illustrate the optimality cuts, in an example where $x \in K_2$ is always satisfied. We then provide details on how to obtain feasibility cuts.

a. Optimality cuts

Consider the following problem.

Example 1

Let

$$\begin{aligned} z &= \min 100x_1 + 150x_2 + \mathbf{E}_\xi(q_1y_1 + q_2y_2) \\ \text{s. t. } x_1 + x_2 &\leq 120, \\ 6y_1 + 10y_2 &\leq 60x_1, \\ 8y_1 + 5y_2 &\leq 80x_2, \\ y_1 &\leq d_1, \quad y_2 \leq d_2, \\ x_1 &\geq 40, \quad x_2 \geq 20, \quad y_1, y_2 \geq 0, \end{aligned}$$

where $\xi^T = (d_1, d_2, q_1, q_2)$ takes on the values $(500, 100, -24, -28)$ with probability 0.4 and $(300, 300, -28, -32)$ with probability 0.6.

Observe that, in this example, the second stage is always feasible ($y = (0, 0)^T$ is always feasible as $x \geq 0$ and $d \geq 0$). Thus $x \in K_2$ is always true and Step 2 can simply be omitted.

The example illustrates the optimality cuts in Step 3 and the effect on the master program. Steps 1 and 3 of the *L*-shaped method require the solution of a number of linear programs. They can easily be obtained through your favorite LP-solver. They can also be checked by constructing the optimal dictionaries (see Exercise 1). You may also trust the authors of this book.

The sequence of iterations of the *L*-shaped method is as follows:

Iteration 1:

Step 1. Ignoring θ , the master program is simply $z = \min\{100x_1 + 150x_2 \mid x_1 + x_2 \leq 120, x_1 \geq 40, x_2 \geq 20\}$ with solution $x^1 = (40, 20)^T$ and $\theta^1 = -\infty$.

Step 3.

- For $\xi = \xi_1$, solve the program

$$\begin{aligned} w = \min\{-24y_1 - 28y_2 \mid & 6y_1 + 10y_2 \leq 2400, 8y_1 + 5y_2 \leq 1600, \\ & 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 100\}. \end{aligned}$$

- The solution is $w_1 = -6100$, $y^T = (137.5, 100)$, $\pi_1^T = (0, -3, 0, -13)$.
- For $\xi = \xi_2$, solve the program

$$\begin{aligned} w = \min\{-28y_1 - 32y_2 \mid & 6y_1 + 10y_2 \leq 2400, 8y_1 + 5y_2 \leq 1600, \\ & 0 \leq y_1 \leq 300, 0 \leq y_2 \leq 300\}. \end{aligned}$$

The solution is $w_2 = -8384$, $y^T = (80, 192)$, $\pi_2^T = (-2.32, -1.76, 0, 0)$.

Using $h_1 = (0, 0, 500, 100)^T$ and $h_2 = (0, 0, 300, 300)^T$ in (1.7), one obtains

$$e_1 = 0.4 \cdot \pi_1^T \cdot h_1 + 0.6 \cdot \pi_2^T \cdot h_2 = 0.4 \cdot (-1300) + 0.6 \cdot (0) = -520.$$

The matrix T is identical in the two scenarios. It consists of two columns $(-60, 0, 0, 0)^T$ and $(0, -80, 0, 0)^T$. Thus, (1.6) gives

$$\begin{aligned} E_1 &= 0.4 \cdot \pi_1^T T + 0.6 \cdot \pi_2^T T = 0.4(0, 240) + 0.6(139.2, 140.8) \\ &= (83.52, 180.48). \end{aligned}$$

Finally, as $x^1 = (40, 20)^T$, $w^1 = -520 - (83.52, 180.48) \cdot x^1 = -7470.4$. Thus, $w^1 = -7470.4 > \theta^1 = -\infty$, add the cut

$$83.52x_1 + 180.48x_2 + \theta \geq -520 .$$

Iteration 2:

Step 1. Solve

$$\begin{aligned} z = \min\{ & 100x_1 + 150x_2 + \theta \mid x_1 + x_2 \leq 120, x_1 \geq 40, x_2 \geq 20, \\ & 83.52x_1 + 180.48x_2 + \theta \geq -520 \} \end{aligned}$$

with solution $z = -2299.2$, $x^2 = (40, 80)^T$, $\theta^2 = -18299.2$.

Step 3.

- For $\xi = \xi_1$ the program

$$\begin{aligned} w = \min\{ & -24y_1 - 28y_2 \mid 6y_1 + 10y_2 \leq 2400, 8y_1 + 5y_2 \leq 6400, \\ & 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 100 \} \end{aligned}$$

- has solution $w_1 = -9600$, $y^T = (400, 0)$, $\pi_1^T = (-4, 0, 0, 0)^T$.
- For $\xi = \xi_2$ the program

$$\begin{aligned} w = \min\{ & -28y_1 - 32y_2 \mid 6y_1 + 10y_2 \leq 2400, 8y_1 + 5y_2 \leq 6400, \\ & 0 \leq y_1 \leq 300, 0 \leq y_2 \leq 300 \} \end{aligned}$$

has solution: $w_2 = -10320$, $y^T = (300, 60)$, $\pi_2^T = (-3.2, 0, -8.8, 0)$.

Apply formulas (1.6) and (1.7) to obtain

$$\begin{aligned} e_2 &= 0.4 \cdot (0) + 0.6 \cdot (-2640) = -1584, \\ E_2 &= 0.4 \cdot (240, 0) + 0.6 \cdot (192, 0) = (211.2, 0). \end{aligned}$$

As $w_2 = -1584 - 211.2 \cdot 40 = -10032 > -18299.2$, add the cut

$$211.2x_1 + \theta \geq -1584 .$$

Iteration 3:

Step 1. Master program has solution $z = -1039.375$, $x^3 = (66.828, 53.172)^T$, $\theta^3 = -15697.994$.

Step 3. Add the cut

$$115.2x_1 + 96x_2 + \theta \geq -2104 .$$

Iteration 4:

Step 1. Master program has solution $z = -889.5$, $x^4 = (40, 33.75)^T$, $\theta^4 = -9952$.

Step 3. The second-stage program for $\xi = \xi_2$ has multiple solutions. Selecting one of them, we add the cut

$$133.44x_1 + 130.56x_2 + \theta \geq 0.$$

Iteration 5:

Step 1. Solve first stage program

$$\begin{aligned} z = \min\{ & 100x_1 + 150x_2 + \theta \mid x_1 + x_2 \leq 120, x_1 \geq 55, x_2 \geq 25, \\ & 83.52x_1 + 180.48x_2 + \theta \geq -520, 211.2x_1 + \theta \geq -1584, \\ & 115.2x_1 + 96x_2 + \theta \geq -2104, \\ & 133.44x_1 + 130.56x_2 + \theta \geq 0 \} . \end{aligned}$$

It has solution $z = -855.833$, $x^5 = (46.667, 36.25)^T$, $\theta^5 = -10960$.

Step 3.

- For $\xi = \xi_1$ the program

$$\begin{aligned} w = \min\{ & -24y_1 - 28y_2 \mid 6y_1 + 10y_2 \leq 2800, 8y_1 + 5y_2 \leq 2900, \\ & 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 100 \} \end{aligned}$$

has the solution $w_1 = -10000$, $y^T = (300, 100)$, $\pi_1^T = (0, -3, 0, -13)$.

- For $\xi = \xi_2$ the program

$$\begin{aligned} w = \min\{ & -28y_1 - 32y_2 \mid 6y_1 + 10y_2 \leq 2800, 8y_1 + 5y_2 \leq 2900, \\ & 0 \leq y_1 \leq 300, 0 \leq y_2 \leq 300 \} \end{aligned}$$

has the solution $w_2 = -11600$, $y^T = (300, 100)$, $\pi_2^T = (-2.32, -1.76, 0, 0)$.

Apply formulae (1.6) and (1.7) to obtain

$$\begin{aligned} e_5 &= 0.4 \cdot (-1300) + 0.6 \cdot (0) = -520, \\ E_5 &= 0.4 \cdot (0, 240) + 0.6 \cdot (139.2, 140.8) = (83.52, 180.48). \end{aligned}$$

As $w_5 = -520 - (83.52, 180.48) \cdot x_5 = -10960 = \theta^5$, stop.
 $x_5 = (46.667, 36.25)^T$ is the optimal solution.

Note that, as Example 1 is small, it is easy to write down the extensive form of Example 1 and solve it with an LP-solver to check whether $(46.667, 36.25)^T$ is

the optimal solution. Exercise 1 illustrates how optimality cuts are obtained through dictionaries and presents some simple and useful checks.

As indicated above, the second-stage program for $\xi = \xi_2$ at Iteration 4 has multiple solutions. An alternative cut is

$$165.12x_1 + 46.08x_2 + \theta \geq -1584.$$

Using this cut instead of the one used above, the algorithm also terminates at Iteration 5.

Example 2

Let

$$\begin{aligned} z &= \min E_{\xi}(y_1 + y_2) \\ \text{s. t. } &0 \leq x \leq 10, \\ &y_1 - y_2 = \xi - x, \\ &y_1, y_2 \geq 0, \end{aligned}$$

where ξ takes the values 1, 2 and 4 with probability 1/3 each.

Observe that $h = \xi$, $T = [1]$ and x are all scalars. Also observe that Step 2 can be omitted. As an exercise, we provide the calculations of Iteration 1. Take $x^1 = 0$ as starting point. Step 3 of Iteration 1 includes the following:

- For $\xi = \xi_1$, solve the program $w = \min\{y_1 + y_2 \mid y_1 - y_2 = 1, y_1, y_2 \geq 0\}$. The solution is $w_1 = 1$, $y^T = (1, 0)$, $\pi_1 = (1)$.
- For $\xi = \xi_2$, solve the program $w = \min\{y_1 + y_2 \mid y_1 - y_2 = 2, y_1, y_2 \geq 0\}$. The solution is $w_2 = 2$, $y^T = (2, 0)$, $\pi_2 = (1)$.
- For $\xi = \xi_3$, solve the program $w = \min\{y_1 + y_2 \mid y_1 - y_2 = 4, y_1, y_2 \geq 0\}$. The solution is $w_3 = 4$, $y^T = (4, 0)$, $\pi_3 = (1)$.
- Using $h_k = \xi_k$, one obtains $e_1 = 1/3 \cdot 1 \cdot (1+2+4) = 7/3$. Formula (1.6) gives $E_1 = 1/3 \cdot 1 \cdot (1+1+1) = 1$. Finally, as $x^1 = (0)$, $w^1 = 7/3 > -\infty$; add the cut, $\theta \geq 7/3 - x$.

Repeating these calculations, the iterations of the *L*-shaped method can be summarized as follows:

Iteration 1:

Step 1. $x^1 = 0$,

Step 3. x^1 is not optimal; add the cut $\theta \geq 7/3 - x$.

Iteration 2:

Step 1. $x^2 = 10$,

Step 3. x^2 is not optimal; add the cut $\theta \geq x - 7/3$.

Iteration 3:

Step 1. $x^3 = 7/3$,

Step 3. x^3 is not optimal; add the cut $\theta \geq (x+1)/3$.

Iteration 4:

Step 1. $x^4 = 1.5$,

Step 3. x^4 is not optimal; add the cut $\theta \geq (5-x)/3$.

Iteration 5:

Step 1. $x^5 = 2$,

Step 3. x^5 is optimal.

We now illustrate in this example that these cuts can be seen as supporting hyperplanes of $\mathcal{Q}(x)$.

To see this, recall that $\mathcal{Q}(x) = E_{\xi} Q(x, \xi) = \sum_{k=1}^K p_k Q(x, \xi_k)$, where

$$Q(x, \xi) = \min\{y_1 + y_2 \mid y_1 - y_2 = \xi - x, y_1, y_2 \geq 0\}.$$

In this very simple example, it is easy to see that if $x \leq \xi$, the second-stage optimal solution is $y^T = (\xi - x, 0)$ while it is $y^T = (0, x - \xi)$ if $x \geq \xi$. Thus

$$Q(x, \xi) = \begin{cases} \xi - x & \text{if } x \leq \xi, \\ x - \xi & \text{if } x \geq \xi. \end{cases}$$

Figure 3 represents the functions $Q(x, 1)$, $Q(x, 2)$, $Q(x, 4)$ as well as $\mathcal{Q}(x)$.

Now, consider again Iteration 1. $x^1 = 0$ is the starting point. Step 3 obtains the cut $\theta \geq 7/3 - x$. From our construction, we see that, for $x = x^1$, $Q(x, 1) = 1$, $Q(x, 2) = 2$, $Q(x, 4) = 4$ and $\mathcal{Q}(x) = 7/3$. But we can also conclude that “around $x = x^1$ ”, $Q(x, 1) = 1 - x$, $Q(x, 2) = 2 - x$, $Q(x, 4) = 4 - x$ and $\mathcal{Q}(x) = 7/3 - x$. In fact, “around $x = x^1$ ” is simply $0 \leq x \leq 1$. This can easily be seen from the construction of $Q(x, 1)$ where $Q(x, 1)$ changes when $x = 1$. In general, such a range can be obtained by linear programming sensitivity analysis around the second-stage optimal solutions.

We conclude that $\mathcal{Q}(x) = 7/3 - x$ within $0 \leq x \leq 1$. The optimality cut at the end of Iteration 1 is nothing other than $\theta \geq 7/3 - x$. It coincides with $\mathcal{Q}(x) =$

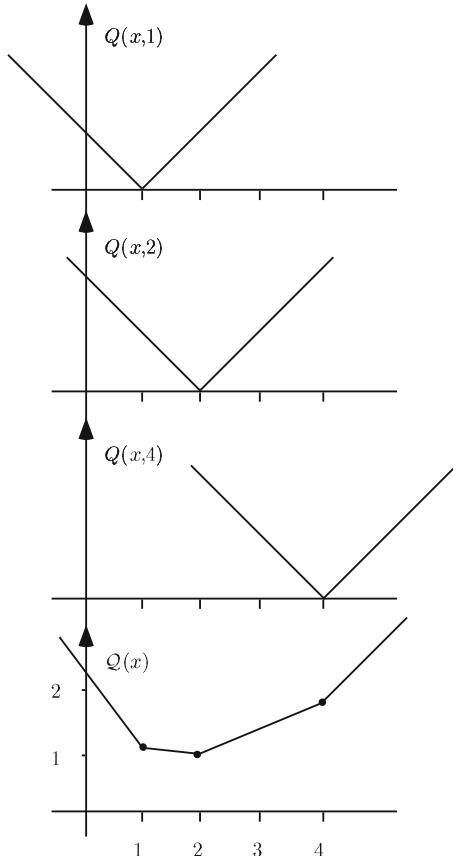


Fig. 3 Recourse functions for Example 2.

$7/3 - x$ within $0 \leq x \leq 1$ and is a lower bound on $\mathcal{Q}(x)$ elsewhere (see Section 5.1 for the proof). We say that the optimality cut is a supporting hyperplane of $\mathcal{Q}(x)$.

The *L*-Shaped algorithm successively adds four cuts which are supporting hyperplanes on the intervals $[0, 1]$, $[4, 10]$, $[2, 4]$ and $[1, 2]$, respectively. Thus, at the beginning of Iteration 5, a full description of $\mathcal{Q}(x)$ is available through the four cuts. Obviously, such a full description is not needed. In fact, the optimum is found here as soon as the supporting hyperplanes of the intervals $[1, 2]$ and $[2, 4]$ are known. If, by chance, these two intervals were to be considered in the first two iterations, then two cuts (and three iterations) would suffice to find the optimum. Thus, the efficiency of the *L*-Shaped algorithm can be influenced by an adequate choice of the starting point.

Finally, observe that, by linear programming duality, the cuts (1.4) can also be obtained from the primal second-stage solutions. Indeed, as we have seen in the proof of Proposition 3 of Chapter 3, solving the second-stage program

$$Q(x, \xi) = \min_y \{q(\omega)y \mid W(\omega)y = h(\omega) - T(\omega)x, y \geq 0\}$$

amounts to finding an optimal basis $B(\omega)$ (a square submatrix of W) such that $y_B = B(\omega)^{-1}(h(\omega) - T(\omega)x)$, $y_N = 0$ and $q_B(\omega)^T B(\omega)^{-1} \leq q(\omega)^T$, where y_B and y_N are the subvectors of y associated to the columns of $B(\omega)$ and to the remaining columns, respectively. It follows that

$$Q(x, \xi) = q_B(\omega)^T \cdot B(\omega)^{-1}(h(\omega) - T(\omega)x).$$

Sensitivity analysis shows that, for fixed ξ , this relation holds for all x 's such that $B(\omega)^{-1}(h(\omega) - T(\omega)x) \geq 0$. Noticing that $\pi^T = q_B(\omega)^T \cdot B(\omega)^{-1}$, one can show that the cut (1.4) is identical to

$$\theta \geq E_\xi \{q_B(\omega)^T \cdot B(\omega)^{-1}(h(\omega) - T(\omega)x)\}$$

and that the right-hand side of the cut coincides with $\mathcal{Q}(x)$ within

$$\cap_{\xi \in \Xi} \{x \mid B(\omega)^{-1}(h(\omega) - T(\omega)x) \geq 0\}.$$

The construction of the cuts from the primal second-stage solutions and the influence of the starting point are further discussed in Exercise 2.

b. Feasibility cuts

Step 2 of the L -shaped method consists of determining whether a first-stage decision $x \in K_1$ is also second stage feasible, i.e. $x \in K_2$. This step can be done as follows:

Step 2. For $k = 1, \dots, K$ solve the linear program

$$\min w' = e^T v^+ + e^T v^- \quad (1.8)$$

$$\text{s. t. } Wy + Iv^+ - Iv^- = h_k - T_k x^v, \quad (1.9)$$

$$y \geq 0, \quad v^+ \geq 0, \quad v^- \geq 0,$$

where $e^T = (1, \dots, 1)$, until, for some k , the optimal value $w' > 0$. In this case, let σ^v be the associated simplex multipliers and define

$$D_{r+1} = (\sigma^v)^T T_k \quad (1.10)$$

and

$$d_{r+1} = (\sigma^v)^T h_k \quad (1.11)$$

to generate a constraint (called a *feasibility cut*) of type (1.3). Set $r = r + 1$, add to the constraint set (1.3), and return to Step 1. If for all k , $w' = 0$, go to Step 3.

To illustrate the feasibility cuts, consider Example 4.2:

$$\begin{aligned} \min & 3x_1 + 2x_2 - E_{\xi}(15y_1 + 12y_2) \\ \text{s. t. } & 3y_1 + 2y_2 \leq x_1, \\ & 2y_1 + 5y_2 \leq x_2, \\ & .8\xi_1 \leq y_1 \leq \xi_1, \\ & .8\xi_2 \leq y_2 \leq \xi_2, \\ & x, y \geq 0, \text{ a.s.,} \end{aligned}$$

with $\xi_1 = 4$ or 6 and $\xi_2 = 4$ or 8 , independently, with probability $1/2$ each and $\xi = (\xi_1, \xi_2)^T$.

To keep the discussion short, assume the first considered realization of ξ is $(6, 8)^T$. If not, many cuts would be needed. Starting from an initial solution $x^1 = (0, 0)^T$, Program (1.8)–(1.9) reads as follows

$$\begin{aligned} w' = \min & v_1^+ + v_1^- + v_2^+ + v_2^- + v_3^+ + v_3^- \\ & + v_4^+ + v_4^- + v_5^+ + v_5^- + v_6^+ + v_6^- \\ \text{s. t. } & v_1^+ - v_1^- + 3y_1 + 2y_2 \leq 0, \\ & v_2^+ - v_2^- + 2y_1 + 5y_2 \leq 0, \\ & v_3^+ - v_3^- + y_1 \geq 4.8, \\ & v_4^+ - v_4^- + y_2 \geq 6.4, \\ & v_5^+ - v_5^- + y_1 \leq 6, \\ & v_6^+ - v_6^- + y_2 \leq 8, \\ & v^+, v^-, y \geq 0 \end{aligned}$$

The optimal solution is $w' = 11.2$ with non-zero variables $v_3^+ = 4.8$ and $v_4^+ = 6.4$. The corresponding dual variables are $\sigma^1 = (-3/11, -1/11, 1, 1, 0, 0)$. We observe that $h = (0, 0, 4.8, 6.4, 6, 8)^T$ and that T consists of the two columns $(-1, 0, 0, 0, 0, 0)^T$ and $(0, -1, 0, 0, 0, 0)^T$; thus, $D_1 = (-0.273, -0.091, 1, 1, 0, 0)$. $T = (0.273, 0.091)$, while $d_1 = (-0.273, -0.091, 1, 1, 0, 0) \cdot h = 11.2$, creating the feasibility cut $3/11x_1 + 1/11x_2 \geq 11.2$ or $3x_1 + x_2 \geq 123.2$.

The first-stage solution is then $x^2 = (41.067, 0)^T$. A second feasibility cut is $x_2 \geq 22.4$. The first-stage solution becomes $x^3 = (33.6, 22.4)^T$. A third feasibility cut $x_2 \geq 41.6$ is generated. The first-stage solution is:

$$x^4 = (27.2, 41.6)^T,$$

which yields feasible second-stage decisions.

This example also illustrates that generating feasibility cuts by a mere application of Step 2 of the L -shaped method may not be efficient. Indeed, a simple look at the problem reveals that, for feasibility when $\xi_1 = 6$ and $\xi_2 = 8$, it is at least necessary to have $y_1 \geq 4.8$ and $y_2 \geq 6.4$, which in turn implies $x_1 \geq 27.2$ and $x_2 \geq 41.6$.

We may then consider the following program as a reasonable initial problem:

$$\begin{aligned} \min \quad & 3x_1 + 2x_2 + \mathcal{Q}(x) \\ \text{s. t. } & x_1 \geq 27.2, \\ & x_2 \geq 41.6, \end{aligned}$$

which immediately appears to be feasible. Such situations frequently occur in practice and are now discussed.

In some cases, Step 2 can be simplified. A first case is when the second stage is always feasible. The stochastic program is then said to have complete recourse. Let, as in (1.1), the second-stage constraint be:

$$Wy = h - Tx, y \geq 0.$$

We repeat here the definition given in Section 3.1d. for complete recourse for convenience.

Definition. A stochastic program is said to have *complete recourse* when $\text{pos } W = \mathbb{R}^{m_2}$. It is said to have *relatively complete recourse* when $K_2 \supseteq K_1$, i.e., $x \in K_1$ implies $h - Tx \in \text{pos } W$ for any h, T realization of \mathbf{h}, \mathbf{T} .

If we consider the farmer's problem in Section 1.1, program (1.1.2) has complete recourse. The second stage just serves as a measure of the cost to the farmer of the decisions taken. Any lack of production can be covered by a purchase. Any production in excess can be sold. If we consider the power generation model (1.3.6), it has complete recourse if there exists at least one technology with zero lead time ($\Delta_i = 0$). If the demand in a given period t exceeds what can be delivered by the available equipment, an investment is made in this (usually expensive) technology to cover the needed demand.

A second case is when it is possible to derive some constraints that have to be satisfied to guarantee second-stage feasibility. These constraints are sometimes called *induced constraints*. They can be obtained from a good understanding of the model. A simple look at the second-stage program in the example reveals the conditions for feasibility. Constraints $x_1 \geq 27.2$ and $x_2 \geq 41.6$ are examples of induced constraints. In the power generation model (1.3.6) of Section 1.3, the total possible demand in a given stage t is obtained from (1.3.8) as $\sum_{j=1}^m d_j^t$. The maximal possible demand in stage t is thus $D^t = \max_{\xi \in \Xi} \sum_{j=1}^m d_j^t$. Stage t feasibility will thus require enough investments in the various technologies to cover the maximal demand, i.e.,

$$\sum_{i=1}^n a_i(w_i^{t-\Delta_i} + g_i) \geq D^t.$$

Again, with the introduction of these induced constraints, Step 2 of the *L*-shaped algorithm can be dropped.

A third case is when Step 2 is not required for all $k = 1, \dots, K$, but for one h_k . Assume T is deterministic. Also assume we can transform W so that for all $t \geq 0$, $t \in \text{pos } W$. This poses no difficulty for inequalities, as it is just a matter of taking the slack variables with a positive coefficient. In Example 4.2 discussed above, the following representation of W satisfies the desired requirement:

$$\begin{array}{lll} 3y_1 + 2y_2 + w_1 & & =x_1, \\ 2y_1 + 5y_2 + w_2 & & =x_2, \\ y_1 & +w_3 & =d_1, \\ -y_1 & +w_4 & =-0.8d_1, \\ y_2 & +w_5 & =d_2, \\ -y_2 & +w_6 & =-0.8d_2. \end{array}$$

For any $t \geq 0$, it suffices to take $w = t$ to have a second-stage feasible solution. Assume first some lower bound,

$$b(x) \leq h_k - T_k x, \quad k = 1, \dots, K,$$

exists. Then a sufficient condition for x to be feasible is that the linear system: $Wy = b(x)$, $y \geq 0$, is feasible. Indeed, if $Wy = b(x)$, $y \geq 0$ is feasible, then $Wy = b'(x)$, $y \geq 0$ is feasible for any $b'(x) \geq b(x)$ by construction of W .

Theorem 1. *Assume that W is such that $t \in \text{pos } W$ for all $t \geq 0$. Define $a_i = \min_{k=1, \dots, K} \{h_{ik}\}$ to be the componentwise minimum of h . Also assume there exists one realization $h_\ell, \ell \in \{1, \dots, K\}$ s.t. $a = h_\ell$. Then, $x \in K_2$ if and only if $Wy = a - Tx, y \geq 0$ is feasible.*

Proof: This is easily checked, as the condition was just seen to be sufficient. It is also necessary because $x \in K_2$ only if $Wy = a - Tx, y \geq 0$ is feasible. \square

Again taking the same example of the previous section, we observe that, with an appropriate choice of W , the vector $h = (0, 0, \xi_1, -0.8\xi_1, \xi_2, -0.8\xi_2)^T$. The componentwise minimum is $a = (0, 0, 4, -4.8, 4, -6.4)^T$. Unfortunately, no h coincides with a . The system $\{y \mid Wy = a - Tx, y \geq 0\}$ is infeasible.

On the other hand, the system is feasible only if $3y_1 + 2y_2 \leq x_1$, $2y_1 + 5y_2 \leq x_2$, $y_1 \geq 0.8\xi_1$, $y_2 \geq 0.8\xi_2$ is feasible (we just drop the upper bounds on y). This reduced system is feasible if and only if

$$3y_1 + 2y_2 \leq x_1 \quad 2y_1 + 5y_2 \leq x_2, \quad y_1 \geq 4.8, \quad y_2 \geq 6.4,$$

i.e., if and only if $x_1 \geq 27.2$ and $x_2 \geq 41.6$, which (as already seen intuitively) is a necessary condition for feasibility. Thus, even if in practice there does not always exist a realization h_ℓ such that $a = h_\ell$, the condition of Theorem 1 may still be helpful.

Exercises

1. Consider Step 3 of Iteration 1 within Example 1.

(a) For $\xi = \xi_1$ and $x = x^1$, the second-stage program is

$$\begin{aligned} w = \min\{ & -24y_1 - 28y_2 \mid 6y_1 + 10y_2 \leq 2400, \\ & 8y_1 + 5y_2 \leq 1600, 0 \leq y_1 \leq 500, 0 \leq y_2 \leq 100 \}. \end{aligned}$$

You may want to check that the optimal dictionary is

$$\begin{aligned} w &= -6100 + 3s_2 + 13s_4, \\ s_1 &= 575 + 6/8s_2 + 50/8s_4, \\ y_1 &= 137.5 - 1/8s_2 + 5/8s_4, \\ s_3 &= 362.5 + 1/8s_2 - 5/8s_4, \\ y_2 &= 100 - s_4, \end{aligned}$$

where s_1 and s_2 are the slacks of the two constraints and s_3 and s_4 the slacks of the upper bound constraints.

Check that this dictionary corresponds to the solution stated in Example 1.

Check that the optimal value $w = -6100$ is also obtained through the dual variables.

(b) For $\xi = \xi_1$, the optimal solution is $w_1 = -6100$ and for $\xi = \xi_2$, the optimal solution is $w_2 = -8384$. Check that $w^1 = 0.4w_1 + 0.6w_2$.

Prove by linear programming duality that $w = \sum_{k=1}^K p_k w_k$, where w_k denotes the solution of the second-stage program for realization k of ξ , $k = 1, \dots, K$.

(c) The optimal dictionary for $\xi = \xi_2$ is

$$\begin{aligned} w &= -8384 + 2.32s_1 + 1.76s_2, \\ y_2 &= 192 - 0.16s_1 + 0.12s_2, \\ y_1 &= 80 + 0.1s_1 - 0.2s_2, \\ s_3 &= 220 - 0.1s_1 + 0.2s_2, \\ s_4 &= 108 + 0.16s_1 - 0.12s_2. \end{aligned}$$

Obtain the two optimal dictionaries if $x^1 = (35, 25)^T$ instead of $(40, 20)^T$.

Show that the cut is unchanged.

(d) Consider again $x^1 = (40, 20)^T$. From the two dictionaries given in (a) and (c), construct the range of values of x where the same cut is obtained.

2. Consider the following problem:

$$\min 7x_1 + 11x_2 + E_\xi(\mathbf{q}_1 y_1 + \mathbf{q}_2 y_2)$$

$$\begin{aligned} \text{s. t.} \quad & \mathbf{y}_1 + 2\mathbf{y}_2 \geq \mathbf{d}_1 - \mathbf{x}_1, \\ & \mathbf{y}_1 \geq \mathbf{d}_2 - \mathbf{x}_2, \\ & 0 \leq \mathbf{x}_1 \leq 10, \quad 0 \leq \mathbf{x}_2 \leq 10, \quad \mathbf{y}_1, \mathbf{y}_2 \geq 0, \end{aligned}$$

where $\xi^T = (\mathbf{q}_1, \mathbf{q}_2, \mathbf{d}_1, \mathbf{d}_2)$ takes on the values $(26, 16, 6, 12)$ and $(14, 24, 10, 4)$ with probability 0.5 each.

- (a) In this example, the L -shaped method selects $x^1 = (0, 0)^T$ as starting point (Step 1 of Iteration 1). The L -shaped method can however be used with any other reasonable starting point. Take $x = (1, 5)^T$ as starting point. Show that the L -shaped then finds an optimal solution in three iterations (which means adding only two optimality cuts).
 - (b) Show that exactly the same steps are taken if the starting point is any point within the region $4 \leq x_2 \leq 6 + x_1$.
 - (c) Consider any stochastic program where the only first-stage constraints are bounds on the variables. Explain why the L -shaped method needs at least two cuts to terminate, unless at least one variable is at a bound at the optimum.
 - (d) Prove that the optimality cuts can also be constructed from the primal solutions of the second stage programs.
 - (e) Show that the first-stage feasibility set $K_1 = \{0 \leq x_1 \leq 10, 0 \leq x_2 \leq 10\}$ can be partitioned in four regions, each one yielding a different optimality cut. The regions are $R_1 = \{x \in K_1 \mid x_1 - 6 \leq x_2 \leq 4\}$, $R_2 = \{x \in K_1 \mid x_2 \leq x_1 - 6\}$, $R_3 = \{x \in K_1 \mid 4 \leq x_2 \leq 6 + x_1\}$, $R_4 = \{x \in K_1 \mid x_1 + 6 \leq x_2\}$.
3. Consider the problem of Exercise 2. Assume the second-stage includes the requirements: $y_1 \leq 15$, $y_2 \leq 2$. Obtain the feasibility cuts.
4. Feasibility cuts in Benders decomposition have an equivalent in Dantzig-Wolfe decomposition. What is it?

c. Proof of convergence

We now constructively prove that constraints of the type (1.4) defined in Step 3 are supporting hyperplanes of $\mathcal{Q}(x)$ and that the algorithm will converge to an optimal solution, provided the constraints (1.3) adequately define feasible points of K_2 . We then prove that at most finitely many cuts (1.3) are needed to guarantee $x \in K_2$.

First, observe that solving (3.1.3), namely,

$$\begin{aligned} & \min c^T x + \mathcal{Q}(x) \\ & \text{s. t. } x \in K_1 \cap K_2, \end{aligned} \tag{1.12}$$

is equivalent to solving

$$\min c^T x + \theta \quad (1.13)$$

$$\text{s. t. } \mathcal{Q}(x) \leq \theta , \quad (1.14)$$

$$x \in K_1 \cap K_2 ,$$

where, in both problems, $\mathcal{Q}(x)$ is defined as in (3.1.3),

$$\mathcal{Q}(x) = E_{\omega} Q(x, \xi(\omega))$$

and

$$Q(x, \xi(\omega)) = \min_y \{ q(\omega)^T y \mid Wy = h(\omega) - T(\omega)x, y \geq 0 \}$$

as in (3.1.4).

We are thus looking for a finitely convergent algorithm for solving (1.12) or (1.13). In Step 3 of the algorithm, problem (1.5) is solved repeatedly for each $k = 1, \dots, K$, yielding optimal simplex multipliers π_k^v , $k = 1, \dots, K$. It follows from duality in linear programming that, for each k ,

$$Q(x^v, \xi_k) = (\pi_k^v)^T (h_k - T_k x^v) .$$

Moreover, by convexity of $Q(x, \xi_k)$, it follows from the subgradient inequality that

$$Q(x, \xi_k) \geq (\pi_k^v)^T h_k - (\pi_k^v)^T T_k x .$$

We may now take the expectation of these two relations to obtain

$$\mathcal{Q}(x^v) = E(\pi^v)^T (\mathbf{h} - \mathbf{T}x^v) = \sum_{k=1}^K p_k \cdot (\pi_k^v)^T (h_k - T_k x^v)$$

and

$$\mathcal{Q}(x) \geq E(\pi^v)^T (\mathbf{h} - \mathbf{T}x) = \sum_{k=1}^K p_k (\pi_k^v)^T h_k - \left(\sum_{k=1}^K p_k (\pi_k^v)^T T_k \right) x ,$$

respectively. By $\theta \geq \mathcal{Q}(x)$, it follows that a pair (x, θ) is feasible for (1.13) only if $\theta \geq E(\pi^v)^T (\mathbf{h} - \mathbf{T}x)$, which corresponds to (1.4) where E_ℓ and e_ℓ are defined in (1.6) and (1.7).

On the other hand, if (x^v, θ^v) is optimal for (1.13), it follows that $\mathcal{Q}(x^v) = \theta^v$, because θ is unrestricted in (1.13) except for $\theta \geq \mathcal{Q}(x)$. This happens when $\theta^v = E(\pi^v)^T (\mathbf{h} - \mathbf{T}x^v)$, which justifies the termination criterion in Step 3.

This means that at each iteration either $\theta^v \geq \mathcal{Q}(x^v)$ implying termination or $\theta^v < \mathcal{Q}(x^v)$. In the latter case, none of the already defined optimality cuts (1.4) adequately imposes $\theta \geq \mathcal{Q}(x)$; so, a new set of multipliers π_k^v will be defined at x^v to generate an appropriate constraint (1.4). The finite convergence of the algorithm follows from the fact that there is only a finite number of different combinations of the K multipliers π_k , because each corresponds to one of the finitely many different bases of (1.5).

An alternative proof of convergence could be obtained by showing that Step 3 coincides with an iteration of the subproblems in the Dantzig-Wolfe decomposition of the dual of (1.12) while Step 1 coincides with the master problem. We will consider this approach in Section 5.6.

We now have to prove that at most a finite number of constraints (1.3) is needed to guarantee $x \in K_2$. Constraints (1.3) are generated in Step 2 of the algorithm. By definition, $x \in K_2$ is equivalent to

$$x \in \{x \mid \text{for } k = 1, \dots, K, \exists y \geq 0 \text{ s.t. } Wy = h_k - T_k x\}.$$

Referring to a previously introduced notation, this means

$$h_k - T_k x \in \text{pos } W, \text{ for } k = 1, \dots, K.$$

In Step 2, a subproblem (1.8) is solved that tests whether $h_k - T_k x^v$ belongs to $\text{pos } W$ for $k = 1, \dots, K$. If not, this means that for some $k = 1, \dots, K$, $h_k - T_k x^v \notin \text{pos } W$. Then, there must be a hyperplane separating $h_k - T_k x^v$ and $\text{pos } W$. This hyperplane must satisfy $\sigma^T t \leq 0$ for all $t \in \text{pos } W$ and $\sigma^T (h_k - T_k x^v) > 0$. In Step 2, this hyperplane is obtained by taking σ for the value σ^v of the simplex multipliers of the subproblem (1.8) solved in Step 2.

By duality, w' being strictly positive is the same as $(\sigma^v)^T (h_k - T_k x^v) > 0$. Also, $(\sigma^v)^T W \leq 0$ is satisfied because σ^v is an optimal simplex multiplier and, at the optimum, the reduced costs associated with y must be non-negative. Therefore, σ^v has the desired property. A necessary condition for x belonging to K_2 is that $(\sigma^v)^T (h_k - T_k x) \leq 0$. There is at most a finite number of such constraints (1.3) because there are only a finite number of optimal bases to the problem (1.8) solved in Step 2. This is no surprise because we already know from Theorem 3.5 that K_2 is polyhedral when ξ is a finite random variable. We thus have proved the following theorem.

Theorem 2. *When ξ is a finite random variable, the L-shaped algorithm finitely converges to an optimal solution when it exists or proves the infeasibility of Problem (3.1.2), namely,*

$$\begin{aligned} \min & c^T x + \mathcal{Q}(x) \\ \text{s. t. } & x \in K_1 \cap K_2. \end{aligned}$$

d. The multicut version

In Step 3 of the L-shaped method, all K realizations of the second-stage program are optimized to obtain their optimal simplex multipliers. These multipliers are then aggregated in (1.10) and (1.11) to generate one cut (1.4). The structure of stochastic

programs clearly allows placing several cuts instead of one. In the multicut version, one cut per realization in the second stage is placed. For those familiar with Dantzig-Wolfe decomposition (explored more deeply in Section 5.5), adding multiple cuts at each iteration corresponds to including several columns in the master program of an inner linearization algorithm (see, e.g., Lasdon [1970] for a general presentation and Birge [1985b] for an analysis of the stochastic case). We first give a presentation of the multicut algorithm, taken from Birge and Louveaux [1988].

The Multicut L -Shaped Algorithm

Step 0. Set $r = v = 0$ and $s_k = 0$ for all $k = 1, \dots, K$.

Step 1. Set $v = v + 1$. Solve the linear program (1.15)–(1.18):

$$\min \quad z = c^T x + \sum_{k=1}^K \theta_k \quad (1.15)$$

$$\text{s. t.} \quad Ax = b, \quad (1.16)$$

$$D_\ell x \geq d_\ell, \quad \ell = 1, \dots, r, \quad (1.17)$$

$$E_{\ell(k)} x + \theta_k \geq e_{\ell(k)}, \quad \ell(k) = 1, \dots, s_k, \quad (1.18)$$

$$x \geq 0, \quad k = 1, \dots, K,$$

Let $(x^v, \theta_1^v, \dots, \theta_K^v)$ be an optimal solution of (1.15)–(1.18). If no constraint (1.18) is present for some k , θ_k^v is set equal to $-\infty$ and is not considered in the computation of x^v .

Step 2. As before.

Step 3. For $k = 1, \dots, K$ solve the linear program (1.9). Let π_k^v be the simplex multipliers associated with the optimal solution of problem k . If

$$\theta_k^v < p_k(\pi_k^v)^T (h_k - T_k x^v), \quad (1.19)$$

define

$$E_{s_k+1} = p_k(\pi_k^v)^T T_k, \quad (1.20)$$

$$e_{s_k+1} = p_k(\pi_k^v)^T h_k, \quad (1.21)$$

and set $s_k = s_k + 1$. If (1.19) does not hold for any $k = 1, \dots, K$, stop; x^v is an optimal solution. Otherwise, return to Step 1.

We illustrate the multicut L -shaped method on Example 2. Starting from $x^1 = 0$, the sequence of iterations is as follows:

Iteration 1:

x^1 is not optimal, add the cuts

$$\theta_1 \geq \frac{1-x}{3}; \quad \theta_2 \geq \frac{2-x}{3}; \quad \theta_3 \geq \frac{4-x}{3}.$$

Iteration 2:

$x^2 = 10$, $\theta_1^2 = -3$, $\theta_2^2 = -8/3$, $\theta_3^2 = -2$ is not optimal; add the cuts

$$\theta_1 \geq \frac{x-1}{3}; \quad \theta_2 \geq \frac{x-2}{3}; \quad \theta_3 \geq \frac{x-4}{3}.$$

Iteration 3:

$x^3 = 2$, $\theta_1^3 = 1/3$, $\theta_2^3 = 0$, $\theta_3^3 = 2/3$ is the optimal solution.

Let us define a *major iteration* to consist of the operations performed between returns to Step 1 in both algorithms. By adding multiple cuts, a solution is found in two major iterations instead of four with the single-cut *L*-shaped method.

A few observations are necessary. By adding disaggregate cuts, more detailed information is given to the first stage. The number of major iterations is expected then to be fewer than in the single cut method. Because the two methods do not necessarily follow the same path, by chance, the *L*-shaped method can conceivably do better than the multicut approach. Exercise 1 provides such an example.

In general, however, as numerical experiments reveal, the number of major iterations is reduced. This is done at the expense of a larger first-stage program, because many more cuts are added. The balance between fewer major iterations but larger first-stage programs is problem-dependent. The results of numerical experiments are available in Birge and Louveaux [1988] and Gassmann [1990]. As a rule of thumb, the multicut approach is expected to be more effective when the number of realizations K is not significantly larger than the number of first-stage constraints m_1 .

Finally, some hybrid approach may be worthwhile, where subsets of the realizations are grouped to form a smaller number of combination cuts. Exercise 2 provides such an example.

Exercises

5. Assume $n_1 = 1$, $m_1 = 0$, $m_2 = 3$, $n_2 = 6$,

$$W = \begin{pmatrix} 1 & -1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix},$$

and $K = 2$ realizations of ξ with equal probability $1/2$. These realizations are $\xi^1 = (q^1, h^1, T^1)^T$ and $\xi^2 = (q^2, h^2, T^2)^T$, where $q^1 = (1, 0, 0, 0, 0, 0)^T$,

$q^2 = (3/2, 0, 2/7, 1, 0, 0)^T$, $h^1 = (-1, 2, 7)^T$, $h^2 = (0, 2, 7)^T$, and $T^1 = T^2 = (1, 0, 0)^T$. For the first value of ξ , $Q(x, \xi)$ has two pieces, such that

$$Q_1(x) = \begin{cases} -x - 1 & \text{if } x \leq -1, \\ 0 & \text{if } x \geq -1. \end{cases}$$

For the second value of ξ , $Q(x, \xi)$ has four pieces such that

$$Q_2(x) = \begin{cases} -1.5x & \text{if } x \leq 0, \\ 0 & \text{if } 0 \leq x \leq 2, \\ 2/7(x-2) & \text{if } 2 \leq x \leq 9, \\ x - 7 & \text{if } x \geq 9. \end{cases}$$

Assume also that x is bounded by $-20 \leq x \leq 20$ and $c = 0$. Starting from any initial point $x^1 \leq -1$, show that one obtains the following sequence of iterate points and cuts for the *L*-shaped method.

Iteration 1:

$$x^1 = -2, \theta^1 \text{ is omitted; new cut: } \theta \geq -0.5 - 1.25x.$$

Iteration 2:

$$x^2 = +20, \theta^2 = -25.5; \text{ new cut: } \theta \geq 0.5x - 3.5.$$

Iteration 3:

$$x^3 = 12/7, \theta^3 = -37/14; \text{ new cut: } \theta \geq 0.$$

Iteration 4:

$x^4 \in [-2/5, 7]$, $\theta^4 = 0$. If x^4 is chosen to be any value in $[0, 2]$, then the algorithm terminates at Iteration 4. The multicut approach would generate the following sequence.

Iteration 1:

$$x^1 = -2, \theta_1^1 \text{ and } \theta_2^1 \text{ omitted; new cuts: } \theta_1 \geq -0.5x - 0.5, \theta_2 \geq -3/4x.$$

Iteration 2:

$$x^2 = 20, \theta_1^2 = -10.5, \theta_2^2 = -15; \text{ new cuts: } \theta_1 \geq 0, \theta_2 \geq 0.5x - 3.5.$$

Iteration 3:

$$x^3 = 2.8, \theta_1^3 = 0, \theta_2^3 = -2.1; \text{ new cut: } \theta_2 \geq 1/7(x-2).$$

Iteration 4:

$$x^4 = 0.32, \theta_1^4 = 0, \theta_2^4 = -0.24; \text{ new cut: } \theta_2 \geq 0.$$

Iteration 5:

$$x^5 = 0, \theta_1^5 = \theta_2^5 = 0, \text{ stop.}$$

6. Consider Example 2, now with ξ taking values:

0.5, 1.0, 1.5 with probability 1/9 each,

2 with probability 1/3 ,

3,4,5 with probability 1/9 each.

As can be seen, the expectation of ξ is still $2\frac{1}{3}$, and new uncertainty is added around 1 and 4.

- (a) Show that the *L*-shaped method follows exactly the same path as before ($x^1 = 0, x^2 = 10, x^3 = 7/3, x^4 = 1.5, x^5 = 2$) provided that in Iteration 4, the support is chosen to describe the region $[1.5, 2]$. If it is chosen to describe the region $[1, 1.5]$, one more iteration is needed.
- (b) Show the multicut version also follows the same path as before ($x^1 = 0, x^2 = 10, x^3 = 2$).
- (c) Now consider an intermediate situation, where $\mathcal{Q}(x)$ is approximated by $\frac{1}{3}[\mathcal{Q}_1(x) + \mathcal{Q}_2(x) + \mathcal{Q}_3(x)]$, where $\mathcal{Q}_1(x)$ is the expectation over the three realizations 0.5, 1.0, and 1.5 (conditional on ξ being in the group $\{0.5, 1.0, 1.5\}$), $\mathcal{Q}_2(x) = Q(x, \xi = 2)$, and $\mathcal{Q}_3(x)$ is the (similarly conditional) expectation over the realizations 3, 4, and 5. Thus, the objective becomes $\frac{1}{3}(\theta_1 + \theta_2 + \theta_3)$. Show that in Iteration 1, the cuts at $x^1 = 0$ are $\theta_1 \geq 1 - x, \theta_2 \geq 2 - x$, and $\theta_3 \geq 4 - x$. In Iteration 2, $x^2 = 10$, and the cuts become $\theta_1 \geq x - 1, \theta_2 \geq x - 2$, and $\theta_3 \geq x - 4$. Show, without computations, that only two major iterations are needed. What conclusions can you draw from this example?

5.2 Regularized Decomposition

Regularized decomposition is a method that combines a multicut approach for the representation of the second-stage value function with the inclusion in the objective of a quadratic regularizing term. This additional term is included to avoid two classical drawbacks of the cutting plane methods. One is that initial iterations are often inefficient. The other is that iterations may become degenerate at the end of the process. Regularized decomposition was introduced by Ruszczyński [1986]. We present a somewhat simplified version of his algorithm using the notation of Section 5.1d.

The Regularized Decomposition Method

Step 0. Set $r = v = 0$, $s_k = 0$ for all $k = 1, \dots, K$. Select a^1 , a feasible solution.

Step 1. Set $v = v + 1$. Solve the regularized master program

$$\begin{aligned} & \min c^T x + \sum_{k=1}^K \theta_k + \frac{1}{2} \|x - a^v\|^2 \\ \text{s. t. } & Ax = b, \\ & D_\ell x \geq d_\ell, \quad \ell = 1, \dots, r, \\ & E_{\ell(k)} x + \theta_k \geq e_{\ell(k)}, \quad \ell(k) = 1, \dots, s_k, \quad k = 1, \dots, K, \\ & x \geq 0. \end{aligned} \tag{2.1}$$

Let (x^v, θ^v) be an optimal solution to (2.1) where $(\theta^v)^T = (\theta_1^v, \dots, \theta_K^v)^T$ is the vector of θ_k 's. If $s_k = 0$ for some k , θ_k^v is ignored in the computation. If $c^T x^v + e^T \theta^v = c^T a^v + \mathcal{Q}(a^v)$, stop; a^v is optimal.

Step 2. As before, if a feasibility cut (5.1.3) is generated, set $a^{v+1} = a^v$ (null infeasible step), and go to Step 1.

Step 3. For $k = 1, \dots, K$, solve the linear subproblem (5.1.9). Compute $\mathcal{Q}_k(x^v)$. If (5.1.19) holds, add an optimality cut (5.1.18) using formulas (5.1.20) and (5.1.21). Set $s_k = s_k + 1$.

Step 4. If (5.1.19) does not hold for any k , then $a^{v+1} = x^v$ (exact serious step); go to Step 1.

Step 5. If $c^T x^v + \mathcal{Q}(x^v) \leq c^T a^v + \mathcal{Q}(a^v)$, then $a^{v+1} = x^v$ (approximate serious step); go to Step 1. Else, $a^{v+1} = a^v$ (null feasible step), go to Step 1.

Observe that when a serious step is made, the value $Q(a^{v+1})$ should be memoized, so that no extra computation is needed in Step 1 for the test of optimality. Note also that a more general regularization would use a term of the form $\alpha \|x - a^v\|^2$ with $\alpha > 0$. This would allow tuning of the regularization with the other terms in the objective. As will be illustrated in Exercise 2, regularized decomposition works better when a reasonable starting point is chosen.

Example 1 (continued)

Consider Exercise 1 of Section 5.1d. Take $a^1 = -0.5$ as a starting point. It corresponds to the solution of the problems with $\xi = \bar{\xi}$ with probability 1. We have $\mathcal{Q}(a^1) = 3/8$.

Iteration 1: Cuts $\theta_1 \geq 0$, $\theta_2 \geq -\frac{3}{4}x$ are added. Let $a^2 = a^1$.

Iteration 2: The regularized master is

$$\begin{aligned} \min \theta_1 + \theta_2 + \frac{1}{2} (x + 0.5)^2 \\ \text{s. t. } \theta_1 \geq 0, \quad \theta_2 \geq -\frac{3}{4} x, \end{aligned}$$

with solution $x^2 = 0.25$: $\theta_1 = 0$, $\theta_2 = -3/16$. A cut $\theta_2 \geq 0$ is added. As $\mathcal{Q}(0.25) = 0 < \mathcal{Q}(a^1)$, $a^3 = 0.25$ (approximate serious step 1).

Iteration 3: The regularized master is

$$\begin{aligned} \min \theta_1 + \theta_2 + \frac{1}{2} (x - 0.25)^2 \\ \text{s. t. } \theta_1 \geq 0, \quad \theta_2 \geq -\frac{3}{4} x, \quad \theta_2 \geq 0, \end{aligned}$$

with solution $x^3 = 0.25$, $\theta_1 = 0$, $\theta_2 = 0$. Because $\theta^v = \mathcal{Q}(a^v)$, a solution is found.

In Exercise 1, the *L*-shaped and multicut methods are compared. The value of a starting point is given in Exercise 2.

We now describe the main results needed to prove convergence of the regularized decomposition to an optimal solution when it exists. For notational convenience, we drop the first-stage linear terms $c^T x$ in the rest of the section. This poses no theoretical difficulty, as we may either define $\theta_k = p_k(c^T x + Q_k(x))$, $k = 1, \dots, K$ or add a $(K+1)$ -th term $\theta_{K+1} = c^T x$. With this notation, the original problem can be written as

$$\begin{aligned} \min \mathcal{Q}(x) &= \sum_{k=1}^K p_k Q_k(x) \\ \text{s. t. (5.1.2), } &x \geq 0, \end{aligned} \tag{2.2}$$

and $Q_k(x) = \min\{q_k^T y \mid Wy = h_k - T_k x, y \geq 0\}$. This is equivalent to

$$\begin{aligned} \min e^T \theta &= \sum_{k=1}^K \theta_k \\ \text{s. t. (5.1.2), (5.1.3), (5.1.4), } &x \geq 0, \end{aligned} \tag{2.3}$$

provided all possible cuts (5.1.3) and (5.1.18) are included.

The regularized master program is

$$\begin{aligned} \min \eta(x, \theta, a^v) &= \sum_{k=1}^K \theta_k + \frac{1}{2} \|x - a^v\|^2 \\ \text{s. t. (5.1.2), (5.1.3), (5.1.18), } &x \geq 0. \end{aligned} \tag{2.4}$$

Note, however, that in the regularized master, only some of the potential cuts (5.1.3) and (5.1.18) are included. We follow the proof in Ruszczyński [1986].

Lemma 3. $e^T \theta^v \leq \eta(x^v, \theta^v, a^v) \leq \mathcal{Q}(a^v)$.

Proof: The first inequality simply comes from $\|x^v - a^v\|^2 \geq 0$. We then observe that a^v always satisfies (5.1.2), (5.1.3), as a^1 is feasible and the serious steps always pick feasible a^v 's. The solution $(a^v, \hat{\theta})$ obtained by choosing $\hat{\theta}_k = p_k Q_k(a^v)$, $k = 1, \dots, K$ necessarily satisfies all constraints (5.1.18) as θ_k is a lower bound on $p_k Q_k(\cdot)$. Thus, $\eta(x^v, \theta^v, a^v) \leq \eta(a^v, \hat{\theta}, a^v) = \mathcal{Q}(a^v)$. \square

Lemma 4. *If the algorithm stops at Step 1, then a^v solves the original problem (2.2).*

Proof: By Lemma 3 and the optimality criterion, $e^T \theta^v = \mathcal{Q}(a^v)$ (remember the linear term $c^T x$ has been dropped). It follows that $e^T \theta^v = \eta(x^v, \theta^v, a^v)$, which implies $\|x^v - a^v\|^2 = 0$, hence $x^v = a^v$. Thus, a^v solves the regularized master (2.4) with the cuts (5.1.3) and (5.1.18) available at iteration v . The cone of feasible directions at a^v does not include any direction of descent of $\eta(x, \theta, a^v)$. The cone of feasible directions at x^v for problem (2.3) is included in the cone of feasible directions at iterations v of the regularized master (2.4) (contains fewer cuts). Moreover, the gradient of the regularizing term vanishes at a^v . Thus, the descent directions of the regularized program (2.4) are the same as the descent directions of (2.3). Hence, a^v solves (2.3), which means a^v solves the original program (2.2). \square

Lemma 5. *If there is a null step at iteration v , then*

$$\eta(x^{v+1}, \theta^{v+1}, a^{v+1}) > \eta(x^v, \theta^v, a^v).$$

Proof: Because the objective function of the regularized master is strictly convex, program (2.4) has a unique solution. A null step at iteration v may be either a null infeasible step or a null feasible step. In the first case, a cut (5.1.3) is added that renders x^v infeasible. In the second case, a cut (5.1.18) is added that renders (x^v, θ^v) infeasible. Thus, as the previous solution becomes infeasible and the solution is unique, the objective function necessarily increases. \square

Lemma 6. *If the number of serious steps is finite, the algorithm stops at Step 1.*

Proof: If the number of serious steps is finite, there exists some v_0 such that $a^v = a^{v_0}$ for all $v \geq v_0$. By Lemma 5, this implies the objective function of the regularized master strictly increases at each iteration $v, v \geq v_0$. Because there are only finitely many possible cuts (5.1.3) and (5.1.18), the algorithm must stop. \square

Lemma 7. *The number of approximate serious steps is finite.*

Proof: By definition of Step 5, the value of $\mathcal{Q}(\cdot)$ does not increase in an approximate serious step (remember that the term $c^T x$ is dropped here). Approximate serious steps only happen when $\mathcal{Q}(x^v) \neq e^T \theta^v$. This can only happen finitely many times because the number of cuts (5.1.18) is finite. \square

Lemma 8. *If the algorithm does not stop, then either $\mathcal{Q}(a^v)$ tends to $-\infty$ as $v \rightarrow \infty$ or the sequence $\{a^v\}$ converges to a solution of the original problem.*

Proof: (i) Let us first consider the case in which the original problem has solution \hat{x} . Define $\hat{\theta}$ by $\hat{\theta}_k = p_k Q_k(\hat{x})$. Thus $(\hat{x}, \hat{\theta})$ solves (2.3). Also $(\hat{x}, \hat{\theta})$ must be feasible for the regularized master for all v . Because (x^v, θ^v) is the solution of the regularized master at iteration v , the derivative of η at (x^v, θ^v) in the direction $(\hat{x} - x^v, \hat{\theta} - \theta^v)$ must be non-negative, i.e.,

$$(x^v - a^v)^T (\hat{x} - x^v) + e^T \hat{\theta} - e^T \theta^v \geq 0$$

or

$$(x^v - a^v)^T (x^v - \hat{x}) \leq \mathcal{Q}(\hat{x}) - e^T \theta^v, \quad (2.5)$$

because $e^T \hat{\theta} = \mathcal{Q}(\hat{x})$.

Let S be the set of iterations at which serious steps occur. In view of Lemma 7, without loss of generality, we may consider such a set where all serious steps are exact. Because, for an exact serious step, $e^T \theta^v = \mathcal{Q}(x^v)$, (5.1.19) does not hold for any k , and $x^v = a^{v+1}$ by definition of the step, for all $v \in S$, (2.5) may be rewritten as

$$(a^{v+1} - a^v)^T (a^{v+1} - \hat{x}) \leq \mathcal{Q}(\hat{x}) - \mathcal{Q}(a^{v+1}).$$

By properties of sums of sequences,

$$\|a^{v+1} - \hat{x}\|^2 = \|a^v - \hat{x}\|^2 + 2(a^{v+1} - a^v)^T (a^{v+1} - \hat{x}) - \|a^{v+1} - a^v\|^2.$$

By dropping the last terms and using the inequality, for all $v \in S$,

$$\begin{aligned} \|a^{v+1} - \hat{x}\|^2 &\leq \|a^v - \hat{x}\|^2 + 2(a^{v+1} - a^v)^T (a^{v+1} - \hat{x}) \\ &\leq \|a^v - \hat{x}\|^2 + 2(\mathcal{Q}(\hat{x}) - \mathcal{Q}(a^{v+1})). \end{aligned} \quad (2.6)$$

Because $\mathcal{Q}(\hat{x}) \leq \mathcal{Q}(a^{v+1})$ for all v , $\|a^{v+1} - \hat{x}\| \leq \|a^v - \hat{x}\|$, i.e., the sequence $\{a^v\}$ is bounded.

Now (2.6) can be rearranged as

$$2(\mathcal{Q}(a^{v+1}) - \mathcal{Q}(\hat{x})) \leq \|a^v - \hat{x}\|^2 - \|a^{v+1} - \hat{x}\|^2.$$

Summing up both sides for $v \in S$, it can be seen that

$$\sum_{v \in S} (\mathcal{Q}(a^{v+1}) - \mathcal{Q}(\hat{x})) < \infty,$$

which implies $\mathcal{Q}(a^{v+1}) \rightarrow \mathcal{Q}(\hat{x})$ for some subsequence $\{a^v\}$, $v \in S_1$ where $S_1 \subseteq S$. Therefore, there must exist an accumulation point \hat{a} of $\{a^v\}$ with $\mathcal{Q}(\hat{a}) = \mathcal{Q}(\hat{x})$. All a^v are feasible, hence \hat{a} is feasible and \hat{a} may substitute for \hat{x} in (2.6) implying $\|a^{v+1} - \hat{a}\| \leq \|a^v - \hat{a}\|$, which shows that \hat{a} is the only accumulation point of $\{a^v\}$.

ii) Now assume that the original problem is unbounded but $\{\mathcal{Q}(a^v)\}$ is bounded. Thus one can find a feasible \hat{x} and an $\varepsilon > 0$ such that $Q(\hat{x}) \leq Q(a^v) - \varepsilon$, $\forall v$. Then (2.6) gives $\|a^{v+1} - \hat{x}\|^2 \leq \|a^v - \hat{x}\|^2 - 2\varepsilon$, which yields a contradiction as $v \rightarrow \infty$, $v \in S$. \square

Lemma 9. *If the algorithm does not stop and $\mathcal{Q}\{a^v\}$ is bounded, there exists v_0 such that if a serious step occurs at $v \geq v_0$, then the solution (x^v, θ^v) of (2.4) is also a solution of (2.4) without the regularizing term.*

Proof: Let K_v denote the set of (x, θ) that satisfy all constraints (5.1.2), (5.1.3), (5.1.18) at iteration v . The problem (2.4) without the regularizing term is thus:

$$\begin{aligned} & \min e^T \theta \\ & \text{s. t. } (x, \theta) \in K_v . \end{aligned} \tag{2.7}$$

Assume Lemma 9 is false. It is thus possible to find an infinite set S such that, for all $v \in S$, a serious step occurs and the solution (x^v, θ^v) to (2.4) is not optimal for (2.7).

Let K_v^* denote the normal cone to the cone of feasible directions for K_v at (x^v, θ^v) . Nonoptimality of (x^v, θ^v) means that the negative gradient of the objective in (2.7), $-d = \begin{bmatrix} 0 \\ -e \end{bmatrix} \notin K_v^*$. As this holds for all $v \in S$,

$$-d \notin \cup_{v \in S} K_v^* . \tag{2.8}$$

Now K_v is polyhedral. There can only be a finite number of constraints (5.1.2) and cuts (5.1.3) and (5.1.18). Thus, the right-hand-side of (2.8) is the union of a finite number of closed sets and, hence, is closed. There exists an $\varepsilon > 0$ such that

$$\mathcal{B}(-d, \varepsilon) \cap K_v^* = \emptyset, \quad \forall v \in S \tag{2.9}$$

where $\mathcal{B}(-d, \varepsilon)$ denotes the ball of radius ε centered at $-d$. On the other hand, (x^v, θ^v) solves (2.4); hence,

$$-\nabla \eta(x^v, \theta^v, a^v) \in K_v^*, \quad \forall v \in S . \tag{2.10}$$

By Lemma 8, $a^v \rightarrow \hat{x}$. By Lemma 7, there exists a v_0 such that for $v \geq v_0$, $e^T \theta^v = \mathcal{Q}(a^v)$ for all serious steps. Hence, at serious steps $v \geq v_0$, we have

$$\mathcal{Q}(a^v) \geq \eta(x^v, \theta^v, a^v) = \frac{1}{2} \|a^v - x^v\|^2 + e^T \theta^v$$

$$= \frac{1}{2} \|x^v - a^v\|^2 + \mathcal{Q}(a^v) .$$

This implies $x^v \rightarrow a^v$, $\forall v \in S$. Hence,

$$\nabla \eta(x^v, \theta^v, a^v) \rightarrow d \quad \forall v \in S ,$$

and (2.10) contradicts (2.9). \square

Theorem 10. *If the original problem has a solution, then the algorithm stops after a finite number of iterations. Otherwise, it generates a sequence of feasible points $\{a^v\}$ such that $\mathcal{Q}(a^v)$ tends to $-\infty$ as $v \rightarrow \infty$.*

Proof: By Lemma 6, the algorithm may only stop at a solution. Suppose the original problem has a solution but the algorithm does not stop. By Lemma 8, $\{a^v\}$ converges to a solution \hat{x} . Lemma 7 implies that for all v large enough, all serious steps are exact, i.e.,

$$\mathcal{Q}(a^{v+1}) = e^T \theta^v .$$

By Lemma 9, for v large enough, x^v also solves (2.4) without the regularizing term implying

$$e^T \theta^v \leq \mathcal{Q}(\hat{x}) ,$$

because problem (2.4) without the regularizing term is a relaxation of the original problem. Because $\mathcal{Q}(\hat{x}) \leq \mathcal{Q}(a^v)$ for all v , it follows that, for v large enough, $\mathcal{Q}(x^v) = \mathcal{Q}(\hat{x})$. Thus, no more serious steps are possible, which by Lemma 6 implies finite termination. The unbounded case was proved in Lemma 8. \square

Implementation of the regularized decomposition algorithm poses a number of practical questions, such as controlling the size of the master regularized problem and numerical stability. An implementation using a *QR* factorization and an active set strategy is described in Ruszczynski [1986]. On the problems tested by the author (see also Ruszczynski [1993b]) the regularized decomposition method outperforms all other methods. This includes a regularized version of the *L*-shaped method, the *L*-shaped method, or the multicut method and is confirmed in the experiments made by Kall and Mayer [1996].

Solving the regularized master program (2.1) is equivalent to solving

$$\begin{aligned} \min c^T x + \sum_{k=1}^K \theta_k \\ \text{s. t.} & \quad Ax = b , \\ & D_\ell x \geq d_\ell , \quad \ell = 1, \dots, r , \\ & E_{\ell(k)} x + \theta_k \geq e_{\ell(k)} , \quad \ell(k) = 1, \dots, s_k , k = 1, \dots, K , \\ & \|x - a^v\|_2 \leq \Delta_v , \\ & x \geq 0 . \end{aligned} \tag{2.11}$$

for some value of Δ_V (Exercise 4), which then suggests the general form of a *trust-region method* (see, e.g., Conn, Gould, and Toint [2000]). The norm as well as the centering point can also be varied in this approach. Linderooth and Wright [2003] use the ∞ -norm (maximum component deviation) to obtain a trust region algorithm for stochastic programs that also allows for significant parallelization and can achieve substantial computational efficiency.

Exercises

1. Check that, with the same starting point, both the L -shaped and the multicut methods require five iterations in Example 1.
2. The regularized decomposition only makes sense with a reasonable starting point. To illustrate this, consider the same example taking as starting point a highly negative value, e.g., $a^1 = -20$. At Iteration 1, the cuts $\theta_1 \geq -\frac{x-1}{2}$ and $\theta_2 \geq -\frac{3}{4}x$ are created. Observe that, for many subsequent iterations, no new cuts are generated as the sequence of trial points a^v move from -20 to $-\frac{75}{4}$, then $-\frac{70}{4}$, $-\frac{65}{4}$, ... each time by a change of $\frac{5}{4}$, until reaching 0 , where new cuts will be generated. Thus a long sequence of approximate serious steps is taken.
3. As we mentioned in the introduction of this section, the regularized decomposition algorithm works with a more general regularizing term of the form $\frac{\alpha}{2}\|x - a^v\|^2$.
 - (a) Observe that the proof of convergence relies on strict convexity of the objective function (Lemma 5), thus $\alpha > 0$ is needed. It also relies on $\nabla \frac{\alpha}{2}\|x^v - a^v\|^2 \rightarrow 0$ as $x^v \rightarrow a^v$, which is simply obtained by taking a finite α . The algorithm can thus be tuned for any positive α and α can vary within the algorithm.
 - (b) Taking the same starting point and data as in Exercise 2, show that by selecting different values of α , any point in $[-20, 20]$ can be obtained as a solution of the regularized master at the second iteration (where 20 is the upper bound on x and the first iteration only consists of adding cuts on θ_1 and θ_2).
 - (c) Again taking the same starting point and data as in Exercise 2, how would you take α to reduce the number of iterations? Discuss some alternatives.
 - (d) Let $\alpha = 1$ for Iterations 1 and 2. As of Iteration 2, consider the following rule for changing α dynamically. For each null step, α is doubled. At each exact step, α is halved. Show why this would improve the performance of the regularized decomposition in the case of Exercise 2. Consider the starting point $x^1 = -0.5$ as in Example 1 and observe that the same path as before is followed.
4. Show the equivalence of (2.1) and (2.11).

5. The choice of α in Exercise 3 has an analogy in the trust-region L -shaped method in terms of the size of the region Δ_V . Find a general expression for Δ_V as a function of α and the solution of (2.1) with weight α on the regularizing term. Find the corresponding value when $\alpha = 1$ for Example 1. What updating rule for Δ_V would be analogous to the rule in Exercise 3d. Starting with Δ_1 corresponding to $\alpha = 1$, follow that updating rule for the trust-region L -shaped method for Example 1.

5.3 The Piecewise Quadratic Form of the L -shaped Methods

In this section, we consider two-stage quadratic stochastic programs of the form

$$\begin{aligned} \min z(x) &= c^T x + \frac{1}{2} x^T C x + E_{\xi} [\min [q^T(\omega) y(\omega) + \frac{1}{2} y^T(\omega) D(\omega) y(\omega)]] \\ \text{s. t. } Ax &= b, \quad T(\omega)x + Wy(\omega) = h(\omega), \\ x &\geq 0, \quad y(\omega) \geq 0, \end{aligned} \tag{3.1}$$

where c , C , A , b , and W are fixed matrices of size $n_1 \times 1$, $n_1 \times n_1$, $m_1 \times n_1$, $m_1 \times 1$, and $m_2 \times n_2$, respectively and q , D , T , and h are random matrices of size $n_2 \times 1$, $n_2 \times n_2$, $m_2 \times n_1$, and $m_2 \times 1$, respectively. Compared to the linear case defined in (3.1.1), only the objective function is modified. As usual, the random vector ξ is obtained by piecing together the random components of q , D , T , and h . Although more general cases could be studied, we also make the following two assumptions.

Assumption 11. *The random vector ξ has a discrete distribution.*

Recall that an $n \times n$ matrix M is *positive semi-definite* if $x^T M x \geq 0$ for all $x \in \mathbb{R}^n$ and M is *positive definite* if $x^T M x > 0$ for all $0 \neq x \in \mathbb{R}^n$.

Assumption 12. *The matrix C is positive semi-definite and the matrices $D(\omega)$ are positive semi-definite for all ω . The matrix W has full row rank.*

The first assumption guarantees the existence of a finite decomposition of the second-stage feasibility set K_2 . The second assumption guarantees that the recourse functions are convex and well-defined.

We may again define the recourse function for a given $\xi(\omega)$ by:

$$\begin{aligned} Q(x, \xi(\omega)) &= \min \{ q^T(\omega) y(\omega) + \frac{1}{2} y^T(\omega) D(\omega) y(\omega) | \\ &\quad T(\omega)x + Wy(\omega) = h(\omega), y(\omega) \geq 0 \}, \end{aligned} \tag{3.2}$$

which is $-\infty$ or $+\infty$ if the problem is unbounded or infeasible, respectively. The expected recourse function is

$$\mathcal{Q}(x) = E_{\xi} Q(x, \xi) \quad (3.3)$$

with the convention $+\infty + (-\infty) = +\infty$.

The definitions of K_1 and K_2 are as in Section 3.5. Theorem 3.32 and Corollaries 3.33 and 3.34 apply, i.e., $\mathcal{Q}(x)$ is a convex function in x and K_2 is convex. Of greater interest to us is the fact that $\mathcal{Q}(x)$ is piecewise quadratic. Loosely stated, this means that K_2 can be decomposed in polyhedral regions called the *cells* of the decomposition and in addition to being convex, $\mathcal{Q}(x)$ is quadratic on each cell.

Example 2

Consider the following quadratic stochastic program

$$\begin{aligned} \min z(x) &= 2x_1 + 3x_2 + E_{\xi} \min \left\{ -6.5y_1 - 7y_2 + \frac{y_1^2}{2} + y_1 y_2 + \frac{y_2^2}{2} \right\} \\ \text{s. t. } &3x_1 + 2x_2 \leq 15, \quad y_1 \leq x_1, \quad y_2 \leq x_2 \\ &x_1 + 2x_2 \leq 8, \quad y_1 \leq \xi_1, \quad y_2 \leq \xi_2 \\ &x_1 + x_2 \geq 0, \quad x_1, x_2 \geq 0, \quad y_1, y_2 \geq 0. \end{aligned}$$

This problem consists of finding some product mix (x_1, x_2) that satisfies some first-stage technology requirements. In the second stage, sales cannot exceed the first-stage production and the random demand. In the second stage, the objective is quadratic convex because the prices are decreasing with sales. We might also consider financial problems where minimizing quadratic penalties on deviations from a mean value leads to efficient portfolios.

Assume that ξ_1 can take the three values 2, 4, and 6 with probability 1/3, that ξ_2 can take the values 1, 3, and 5 with probability 1/3, and that ξ_1 and ξ_2 are independent of each other. For very small values of x_1 and x_2 , it always is optimal in the second stage to sell the production, $y_1 = x_1$ and $y_2 = x_2$. More precisely, for $0 \leq x_1 \leq 2$ and $0 \leq x_2 \leq 1$, $y_1 = x_1, y_2 = x_2$ is the optimal solution of the second stage for all ξ . If needed, the reader may check this using the Karush-Kuhn-Tucker conditions.

Thus, $Q(x, \xi) = -6.5x_1 - 7x_2 + \frac{x_1^2}{2} + x_1 x_2 + \frac{x_2^2}{2}$ for all ξ and $\mathcal{Q}(x) = -6.5x_1 - 7x_2 + \frac{x_1^2}{2} + x_1 x_2 + \frac{x_2^2}{2}$. Here, the cell is $\{(x_1, x_2) \mid 0 \leq x_1 \leq 2, 0 \leq x_2 \leq 1\}$. Within that cell, $\mathcal{Q}(x)$ is quadratic.

Definition 13. A *finite closed convex complex* \mathcal{K} is a finite collection of closed convex sets, called the *cells* of \mathcal{K} , such that the intersection of two distinct cells has an empty interior.

Definition 14. A *piecewise convex program* is a convex program of the form $\inf\{z(x) \mid x \in S\}$ where f is a convex function on \mathbb{R}^n and S is a closed convex subset of the effective domain of f with nonempty interior.

Let \mathcal{K} be a finite closed convex complex such that

- (a) the n -dimensional cells of \mathcal{K} cover S ,
- (b) either f is identically $-\infty$ or for each cell C_v of the complex there exists a convex function $z_v(x)$ defined on S and continuously differentiable on an open set containing C_v which satisfies
 - (a) $z(x) = z_v(x) \forall x \in C_v$, and
 - (b) $\nabla z_v(x) \in \partial z(x) \forall x \in C_v$.

Definition 15. A *piecewise quadratic function* is a piecewise convex function where on each cell C_v the function z_v is a quadratic form.

Taking Example 2, we have both $\mathcal{Q}(x)$ and $z(x)$ piecewise quadratic. On $C_1 = \{0 \leq x_1 \leq 2, 0 \leq x_2 \leq 1\}$,

$$\begin{aligned}\mathcal{Q}_1(x) &= -6.5x_1 - 7x_2 + \frac{x_1^2}{2} + x_1x_2 + \frac{x_2^2}{2} \\ \text{and } z_1(x) &= -4.5x_1 - 4x_2 + \frac{x_1^2}{2} + x_1x_2 + \frac{x_2^2}{2}.\end{aligned}$$

Defining a polyhedral complex was first done by Walkup and Wets [1967] for the case of stochastic linear programs. Based on this decomposition, Gartska and Wets [1974] proved that the optimal solution of the second stage is a continuous, piecewise linear function of the first-stage decisions and showed that $Q(x, \xi)$ is piecewise quadratic in x . It follows that under Assumption 1, $\mathcal{Q}(x)$ and $z(x)$ are also piecewise quadratic in x .

For the sake of completeness, observe that $z(x)$ is not always $\max_v z_v(x)$. To this end, consider

$$z(x) = \begin{cases} z_1(x) = \frac{x}{2} & \text{when } 0 \leq x \leq 2, \\ z_2(x) = (x-1)^2 & \text{when } x \geq 2. \end{cases}$$

This function is easily seen to be piecewise quadratic. On $(0, 1/2)$, $z(x) = z_1(x)$ while $\max\{z_1(x), z_2(x)\} = z_2(x)$.

An algorithm

In this section, we study a finitely convergent algorithm for piecewise quadratic programs (Louveaux [1978]).

Algorithm PQP

Initialization: Let $S_1 = S$, $x^0 \in S$, $v = 1$.

Step 1. Obtain C_v , a cell of the decomposition of S containing x^{v-1} . Let $z_v(\cdot)$ be the quadratic form on C_v .

Step 2. Let $x^v \in \arg \min \{z_v(x) \mid x \in S_v\}$ and $w^v \in \arg \min \{z_v(x) \mid x \in C_v\}$. If w^v is the limiting point of a ray on which $z_v(x)$ is decreasing to $-\infty$, the original PQP is unbounded and the algorithm terminates.

Step 3. If

$$\nabla^T z_v(w^v)(x^v - w^v) = 0, \quad (3.4)$$

then stop; w^v is an optimal solution.

Step 4. Let $S_{v+1} = S_v \cap \{x \mid \nabla^T z_v(w^v)x \leq \nabla^T z_v(w^v)w^v\}$. Let $v = v + 1$; go to Step 1.

Thus, contrary to the L -shaped method in the linear case, the subgradient inequality is not applied at the current iterate point x^v . Instead, it is applied at w^v , a point where $z_v(\cdot)$ is minimal on C_v . Under some practical conditions on the constructions of the cells, the algorithm is finitely convergent.

We first prove that the condition,

$$\nabla^T z_v(w^v)x \leq \nabla^T z_v(w^v)w^v, \quad (3.5)$$

is a necessary condition for optimality of x .

Because $\nabla z_v(w^v) \in \partial z(w^v)$, the subgradient inequality applied at w^v implies that $z(x) \geq z(w^v) + \nabla^T z_v(w^v)(x - w^v)$ for all x . Now, x is a minimizer of $z(\cdot)$ only if $z(x) \leq z(w^v)$. This implies that x is a minimizer of $z(\cdot)$ only if $\nabla^T z_v(w^v)(x - w^v) \leq 0$, which is precisely (3.5). Thus, a solution $x \in \arg \min \{z(x) \mid x \in S_v\}$ is also a solution $x \in \arg \min \{z(x) \mid x \in S\}$.

We next show that any solution $\bar{x} \in \arg \min \{z_v(x) \mid x \in S_v\}$ is a solution $\in \arg \min \{z(x) \mid x \in S_v\}$ (and thus by the argument, a solution is in $\arg \min \{z(x) \mid x \in S\}$) if $\bar{x} \in C_v$.

By definition, $\bar{x} \in \arg \min \{z_v(x) \mid x \in S_v\}$ is a solution of a quadratic convex program whose objective is continuously differentiable on S_v ; it must satisfy the condition $\nabla^T z_v(\bar{x})(x - \bar{x}) \geq 0, \forall x \in S_v$. If $\bar{x} \in C_v$, then $\nabla z_v(\bar{x}) \in \partial z(\bar{x})$. Applying the subgradient inequality for $z(\cdot)$ at \bar{x} implies

$$z(x) \geq z(\bar{x}) + \nabla^T z_v(\bar{x})(x - \bar{x}) \geq z(\bar{x}) \quad \forall x \in S_v.$$

Thus, if $\bar{x} \in C_v$, it is a solution to the original problem.

Finally, if the optimality condition (3.4) holds, applying the gradient inequality to the quadratic convex function $z_v(\cdot)$ at w^v implies

$$z_v(x^v) \geq z_v(w^v) + \nabla^T z_v(w^v)(x^v - w^v) = z_v(w^v),$$

which proves $w^v \in \arg \min \{z_v(x) \mid x \in S_v\}$. Thus, w^v is (another) minimizer of $z_v(\cdot)$ on S_v . As $w^v \in C_v$, the conclusion implies it is a solution to the original problem. A more detailed proof, including properties of the successive sets S_v and a discussion of the construction of full dimensional cells of a piecewise quadratic program, can be found in Louveaux [1978].

Exercises

- For Example 2, consider the values $x_1 = 4.5$, $x_2 = 0$. Check that around these values, $y_2 = x_2$ for all ξ_2 , and

$$y_1 = \begin{cases} \xi_1 & \text{if } \xi_1 = 2 \text{ or } 4, \\ x_1 & \text{if } \xi_1 = 6, \end{cases}$$

are the optimal second-stage decisions. Check that the corresponding cell is defined as

$$\{(x_1, x_2) \mid 4 \leq x_1 \leq 6, 0 \leq x_2 \leq 1, x_1 + x_2 \leq 6.5\}$$

and

$$z(x) = -\frac{29}{3} - \frac{x_1}{6} - 2x_2 + \frac{x_1^2}{6} + \frac{x_1 x_2}{3} + \frac{x_2^2}{2}.$$

- We now apply the PQP algorithm to the problem of Example 2.

Initialization: $x^0 = (0, 0)$; $v = 1$

$$S_1 = S = \{x \mid 3x_1 + 2x_2 \leq 15, x_1 + 2x_2 \leq 8, x_1, x_2 \geq 0\}.$$

Iteration 1:

As we saw in the discussion of Example 2, $C_1 = \{x \mid 0 \leq x_1 \leq 2, 0 \leq x_2 \leq 1\}$ and $z_1(x) = -4.5x_1 - 4x_2 + \frac{x_1^2}{2} + x_1 x_2 + \frac{x_2^2}{2}$. Using the classical Karush-Kuhn-Tucker condition, we obtain $x^1 = (4.5, 0)^T$ and $w^1 = (2, 1)^T \in C_1$. Hence, $\nabla^T z_1(w^1) = (-1.5, -1)$, $\nabla^T z_1(w^1)(x^1 - w^1) = -2.75 \neq 0$, and

$$S_2 = S \cap \{x \mid -1.5x_1 - x_2 \leq -4\}.$$

Iteration 2:

As we saw in Exercise 1, $x^1 \in C_2 = \{x \mid 4 \leq x_1 \leq 6, 0 \leq x_2 \leq 1, x_1 + x_2 \leq 6.5\}$ and

$$z_2(x) = -\frac{29}{3} - \frac{x_1}{6} - 2x_2 + \frac{x_1^2}{6} + \frac{x_1 x_2}{3} + \frac{x_2^2}{2}.$$

We obtain $x^2 = \left(\frac{22}{19}, \frac{43}{19}\right)^T$, a point where the optimality constraint $-1.5x_1 - x_2 \leq -4$ is binding. We also obtain $w^2 = \left(4, \frac{2}{3}\right)^T \in C_2$, $\nabla^T z_2(w^2) = (25/18, 0)^T$,

and (3.3) does not hold.

$$S_3 = S_2 \cap \left\{ x \mid \frac{25}{18}x_1 \leq \frac{100}{18} \right\}.$$

Iteration 3:

- (a) We now obtain $x^2 \in C_3 = \{x \mid 0 \leq x_1 \leq 2, 1 \leq x_2 \leq 3\}$. In the second stage, $y_1 = x_1 \forall \xi_1, y_2 = x_2$ when $\xi_2 \geq 3$ and $y_2 = 1$ when $\xi_2 = 1$, so that

$$z_3(x) = -\frac{13}{6} - \frac{25}{6}x_1 - \frac{5}{3}x_2 + \frac{x_1^2}{2} + \frac{2x_1x_2}{3} + \frac{x_2^2}{3}.$$

- (b) $x^3 = (4, 0)^T$; $w^3 = w^1 = (2, 1)^T$.
(c) $S_4 = S_3 \cap \{x \mid -\frac{3}{2}x_1 + \frac{x_2}{3} \leq -\frac{8}{3}\}$.

Iteration 4:

- (a) $x^3 \in C_4 = \{x \mid 2 \leq x_1 \leq 4, 0 \leq x_2 \leq 1\}$.
 $z_4(x) = -\frac{11}{3} - \frac{7}{3}x_1 - \frac{10}{3}x_2 + \frac{x_1^2}{3} + \frac{2x_1x_2}{3} + \frac{x_2^2}{2}$.
(b) $x^4 \simeq (2.18, 1.81)^T$, a point where $-\frac{3}{2}x_1 + \frac{x_2}{3} = -\frac{8}{3}$.
 $w^4 = (2.5, 1)$.
(c) $S_5 = S_4 \cap \{x \mid -\frac{2x_2}{3} \leq -\frac{2}{3}\}$.

Iteration 5:

- (a) $x^4 \in C_5 = \{x \mid 2 \leq x_1 \leq 4, 1 \leq x_2 \leq 3\} \cap S$.
 $z_5(x) = -\frac{101}{18} - \frac{19}{9}x_1 - \frac{11}{9}x_2 + \frac{x_1^2}{3} + \frac{4x_1x_2}{9} + \frac{x_2^2}{3}$.
(b) $x^5 = w^5 = (2.5, 1)^T$ is an optimal solution to the problem.

The PQP iterations for the example are shown in Figure 4. The thinner lines represent the limits of cells and the constraints containing S . The heavier lines give the optimality cuts, OC_v , for $v = 1, 2, 3, 4$. A few comments are in order:

- (a) Observe that the objective values of the successive iterate points are not necessarily monotone decreasing. As an example, $z^1(w^1) = -8.5$ and $z^2(w^2) = -\frac{71}{9} > z^1(w^1)$.
(b) A stronger version of (3.4) can be obtained. Let $\bar{z} = \min_v \{z(w^v)\}$ be the best known solution at iteration v . Starting from the subgradient inequality at w^v ,

$$z(x) \geq z(w^v) + \nabla z_v^T(w^v)(x - w^v)$$

and observing that $z(x) \leq \bar{z}$ is a necessary condition for optimality, we obtain an updated cut,

$$\nabla^T z_v(w^v)x \leq \nabla^T z_v(w^v)w^v + \bar{z} - z(w^v). \quad (3.6)$$

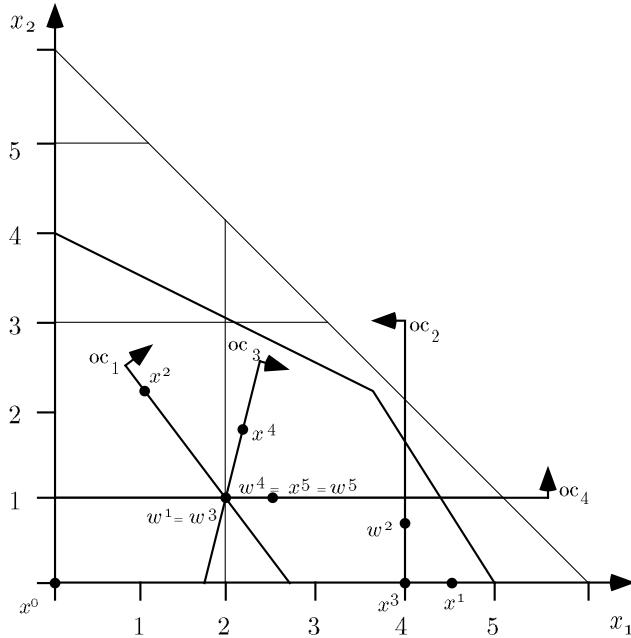


Fig. 4 The cells and PQP cuts of Example 2.

Updating is quite easy, as it only involves the right-hand sides of the cuts. As an example, at Iteration 2, the cut could be updated from

$$\frac{25x_1}{18} \leq \frac{100}{18} \quad \text{to} \quad \frac{25}{18}x_1 \leq \frac{100}{18} - 8.5 + \frac{71}{9},$$

namely, $\frac{25x_1}{18} \leq \frac{89}{18}$. Similarly, at Iteration 4, \bar{z} becomes $-\frac{103}{12}$ and the right-hand sides of all previously imposed cuts can be modified by $(-\frac{103}{12} + 8.5)$, i.e., by $-\frac{1}{12}$. In the example, the updating does not change the sequence of iterations.

- (c) The number of iterations is strongly dependent on the starting point. In particular, if one cell exists such that the minimizer of its quadratic form over S is in fact within the cell, then starting from that cell would mean that a single iteration would suffice. In Example 2, this is not the case. However, starting from $\{x \mid 2 \leq x_1 \leq 4, 1 \leq x_2 \leq 3\}$ would require only two iterations. This is in fact a reasonable starting cell. Indeed, the intersection of the two nontrivial constraints defining S ,

$$3x_1 + 2x_2 \leq 15, \quad x_1 + 2x_2 \leq 8,$$

is the point $(3.5, 2.25)$ that belongs to that cell. (An alternative would be to start from the minimizer of the mean value problem on S .)

- (d) If we observe the graphical representation of the cells and of the cuts, we observe that the cuts each time eliminate all points of a cell, except possibly the point w^v at which they are imposed, and possibly other points on a face of dimension strictly less than n_1 . (Working with updated cuts (3.6) sometimes also eliminates the point w^v at which it is imposed.) The finite termination of the algorithm is precisely based on the elimination of one cell at each iteration. (We leave aside the question of considering cells of full dimension n_1 .) There is thus no need at iteration v to start from a cell containing x^{v-1} . In fact, any cell not yet considered is a valid candidate. One reasonable candidate could be the cell containing $\frac{x^{v-1} + w^{v-1}}{2}$, for example, or any convex combination of x^{v-1} and w^{v-1} .

3. Consider the farming example of Section 1.1. As in Exercise 1.1, assume that prices are influenced by quantities. As an individual, the farmer has little influence on prices, so he may reasonably consider the current solution optimal. If we now consider that all farmers read this book and optimize their choice of crop the same way, increases of sales will occur in parallel for all farmers, bringing large quantities together on the market. Taking things to an extreme, this means that changes in the solution are replicated by all farmers. Assume a decrease in selling prices of \$0.03 per ton of grain and of \$0.06 per ton of corn brought into the market by each individual farmer. Assume the selling price of beets and purchase prices are not affected by quantities.

Show that the PQP algorithm reaches the solution in one iteration when the starting point is taken as $\{x_1, x_2, x_3 \mid 80 \leq x_2 \leq 100; 250 \leq x_3 \leq 300; x_1 + x_2 + x_3 = 500\}$. (Remark: Although only one iteration is needed, calculations are rather lengthy. Observe that constant terms are not needed to obtain the optimal solution.)

5.4 Bunching and Other Efficiencies

One big issue in the efficient implementation of the L -shaped method is in Step 3. The second-stage program (1.5) has to be solved K times to obtain the optimal multipliers, π_k^v . For a given x^v and a given realization k , let B be the optimal basis of the second stage. It is well-known from linear programming that B is a square submatrix of W such that $(\pi_k^v)^T = q_{k,B}^T B^{-1}$, $q_k^T - (\pi_k^v)^T W \geq 0$, $B^{-1}(h_k - T_k x^v) \geq 0$, where $q_{k,B}$ denotes the restriction of q_k to the selection of columns that define B . Important savings can be obtained in Step 3 when the same basis B is optimal for several realizations of k . This is especially the case when q is deterministic. Then, two different realizations that share the same basis also share the same multipliers π_k^v . We present the rest of the section, assuming q is deterministic.

To be more precise, define

$$\tau = \{t \mid t = h_k - T_k x^v \text{ for some } k = 1, \dots, K\} \quad (4.1)$$

as the set of possible right-hand sides in the second stage. Let B be a square submatrix and $\pi^T = q_B^T B^{-1}$. Assume B satisfies the optimality criterion $q^T - \pi^T W \geq 0$. Then define a *bunch* as

$$Bu = \{t \in \tau \mid B^{-1}t \geq 0\}, \quad (4.2)$$

the set of possible right-hand sides that satisfy the feasibility condition. Thus, π is an optimal dual multiplier for all $t \in Bu$. Note also that, by virtue of Step 2 of the L -shaped method, only feasible first-stage $x^v \in K_2$ are considered. This observation means that, by construction,

$$\tau \subseteq \text{pos } W = \{t \mid t = Wy, y \geq 0\}.$$

We now provide an introduction to possible implementations that use these ideas. For more details, the reader is referred to Gassmann [1990], Wets [1988], or Wets [1983b].

a. Full decomposability

One first possibility is to work out a full decomposition of $\text{pos } W$ into component bases. This can only be done for small problems or problems with a well-defined structure. As an example, consider the farming example of Section 1.1. The second-stage representation (1.1.4) is repeated here under the notation of the current chapter:

$$\begin{aligned} Q(x, \xi) &= \min 238y_1 - 170y_2 + 210y_3 - 150y_4 - 36y_5 - 10y_6 \\ \text{s. t. } &y_1 - y_2 - w_1 = 200 - \xi_1 x_1, \\ &y_3 - y_4 - w_2 = 240 - \xi_2 x_2, \\ &y_5 + y_6 + w_3 = \xi_3 x_3, \\ &y_5 + w_4 = 6000, \\ &y, w \geq 0, \end{aligned}$$

where w_1 to w_4 are slack variables. This second stage has complete recourse, so $\text{pos } W = \mathbb{R}^4$. The matrix $W =$

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix},$$

which has 4 rows and 10 columns; so that theoretically, $\binom{10}{4} = 210$ bases could be found. However, in practice w_1 , w_2 , and w_3 are never in the basis, as they are always dominated by y_2 , y_4 , and y_6 , respectively. The matrix where the columns w_1 , w_2 , and w_3 are removed is sometimes called the *support* of W (see Wallace and Wets [1992]). Also, y_5 is always in the basis (a fact of worldwide importance as it is one of the reasons that created tension between United States and Europe within the GATT negotiations). Moreover, y_1 or y_2 and y_3 or y_4 are always basic. In this case, not only is a full decomposition of pos W available, but an immediate analytical expression for the multipliers is also obtained. Thus,

$$\begin{aligned}\pi_1(\xi) &= \begin{cases} 238 & \text{if } \xi_1 x_1 < 200, \\ -170 & \text{otherwise;} \end{cases} \\ \pi_2(\xi) &= \begin{cases} 210 & \text{if } \xi_2 x_2 < 240, \\ -150 & \text{otherwise;} \end{cases} \\ \pi_3(\xi) &= \begin{cases} -36 & \text{if } \xi_3 x_3 < 6000, \\ 0 & \text{otherwise;} \end{cases} \\ \pi_4(\xi) &= \begin{cases} 10 & \text{if } \xi_3 x_3 > 6000, \\ 0 & \text{otherwise.} \end{cases}\end{aligned}$$

The dual multipliers are easily obtained because the problem is small and enjoys some form of separability. The decomposition is thus $(1,3,5,6)$, $(1,3,5,10)$, $(1,4,5,6)$, $(1,4,5,10)$, $(2,3,5,6)$, $(2,3,5,10)$, $(2,4,5,6)$, $(2,4,5,10)$, where the four variables in a basis are described by their indices (where the index is $6+j$ for the j -th slack variable). Another example is given in Exercise 1 and Wallace [1986a].

When applicable, full decomposability has proven very efficient. In general, however, it is expected to be applicable only for small problems.

b. Bunching

A relatively simple bunching procedure is as follows. Again let $\tau = \{t \mid t = h_k - T_k x^v \text{ for some } k = 1, \dots, K\}$ be the set of possible right-hand sides in the second stage. Consider some k . Denote $t_k = h_k - T_k x^v$. It might arbitrarily be $k = 1$, or, if available, a value of k such that $h_k - T_k x^v = \bar{t}$, the expectation of all $t_k \in \tau$. Let B_1 be the corresponding optimal basis and $\pi(1)$ the corresponding vector of simplex multipliers. Then, $Bu(1) = \{t \in \tau \mid B_1^{-1}t \geq 0\}$. Let $\tau_1 = \tau \setminus Bu(1)$.

We can now repeat the same operations. Some element of τ_1 is chosen. The corresponding optimal basis B_2 and its associated vector of multipliers $\pi(2)$ are formed. Then, $Bu(2) = \{t \in \tau_1 \mid B_2^{-1}t \geq 0\}$ and $\tau_2 = \tau_1 \setminus Bu(2)$. The process is repeated until all $t_k \in \tau$ are in one of b total bunches. Then, (1.6) and (1.7) are

replaced by

$$E_{s+1} = \sum_{\ell=1}^b \pi(\ell)^T \sum_{t_k \in Bu(\ell)} p_k T_k \quad (4.3)$$

and

$$e_{s+1} = \sum_{\ell=1}^b \pi(\ell)^T \sum_{t_k \in Bu(\ell)} p_k h_k. \quad (4.4)$$

This procedure still has some drawbacks. One is that the same $t_k \in \tau$ may be checked many times against different bases. The second is that a new optimization is restarted every time a new bunch is considered. It is obvious here that some savings can be obtained in organizing the work in such a way that the optimal basis in the next bunch is obtained by performing only one (or a few) dual simplex iterations from the previous one. As an example, consider the following second stage:

$$\begin{aligned} & \max 6y_1 + 5y_2 + 4y_3 + 3y_4 \\ \text{s. t. } & 2y_1 + y_2 + y_3 \leq \xi_1, \\ & y_2 + y_3 + y_4 \leq \xi_2, \\ & y_1 + y_3 \leq x_1, \\ & 2y_2 + y_4 \leq x_2, \\ & y \geq 0. \end{aligned}$$

Let $\xi_1 \in \{4, 5, 6, 7, 8\}$ with equal probability 0.2 each and $\xi_2 \in \{2, 3, 4, 5, 6\}$ with equal probability 0.2 each. There are theoretically $\binom{8}{4} = 70$ different possible bases. In view of the possible realizations of ξ , at most 25 different bases can be optimal.

Let t^1 to t^{25} denote the possible right-hand sides with

$$t^1 = \begin{pmatrix} 4 \\ 2 \\ x_1 \\ x_2 \end{pmatrix}, \quad t^2 = \begin{pmatrix} 4 \\ 3 \\ x_1 \\ x_2 \end{pmatrix}, \quad \dots, \quad t^{25} = \begin{pmatrix} 8 \\ 6 \\ x_1 \\ x_2 \end{pmatrix}.$$

Consider the case where $x_1 = 3.1$ and $x_2 = 4.1$. Let us start from $\xi = \bar{\xi} = (6, 4)^T$. Represent a basis again by the variable indices with $4+j$ the index of the j th slack. The optimal basis is $B_1 = \{1, 4, 7, 8\}$ with $y_1 = 3$, $y_4 = 4$, $w_3 = 0.1$, $w_4 = 0.1$, the values of the basic variables.

The optimal dictionary associated with B_1 is

$$\begin{aligned} z &= 3\xi_1 + 3\xi_2 - y_2 - 2y_3 - 3w_1 - 3w_2, \\ y_1 &= 1/2\xi_1 - 1/2y_2 - 1/2y_3 - 1/2w_1, \\ y_4 &= \xi_2 - y_2 - y_3 - w_2, \\ w_3 &= 3.1 - 1/2\xi_1 + 1/2y_2 - 1/2y_3 + 1/2w_1, \end{aligned}$$

$$w_4 = 4.1 - \xi_2 - y_2 + y_3 + w_2 .$$

This basis is optimal and feasible as long as $\xi_1/2 \leq 3.1$ and $\xi_2 \leq 4.1$, which in view of the possible values of ξ amounts to $\xi_1 \leq 6$ and $\xi_2 \leq 4$, so that $Bu(1) = \{t^1, t^2, t^3, t^6, t^7, t^8, t^{11}, t^{12}, t^{13}\}$. Neighboring bases can be obtained by considering either $\xi_1 \geq 7$ or $\xi_2 \geq 5$. Let us start with $\xi_2 \geq 5$. This means that w_4 becomes negative and a dual simplex pivot is required in Row 4. This means that w_4 leaves the basis, and, according to the usual dual simplex rule, y_3 enters the basis.

The new basis is $B_2 = \{1, 3, 4, 7\}$ with an optimal dictionary:

$$\begin{aligned} z &= 3\xi_1 + \xi_2 + 8.2 - 3y_2 - 3w_1 - w_2 - 2w_4 , \\ y_1 &= \frac{\xi_1}{2} - \frac{\xi_2}{2} + 2.05 - y_2 - \frac{w_1}{2} + \frac{w_2}{2} - \frac{w_4}{2} , \\ y_3 &= \xi_2 - 4.1 + y_2 - w_2 + w_4 , \\ y_4 &= 4.1 - 2y_2 - w_4 , \\ w_3 &= 5.15 - \frac{\xi_1}{2} - \frac{\xi_2}{2} + \frac{w_1}{2} + \frac{w_2}{2} - \frac{w_4}{2} . \end{aligned}$$

The condition $\xi_1 - \xi_2 + 4.1 \geq 0$ always holds. This basis is optimal as long as $\xi_2 \geq 5$ and $\xi_1 + \xi_2 \leq 10$, so that $Bu(2) = \{t^4, t^5, t^9\}$.

Neighboring bases are B_1 when $\xi_2 \leq 4$ and B_3 obtained when $w_3 < 0$, i.e., $\xi_1 + \xi_2 \geq 11$. This basis corresponds to w_3 leaving the basis and w_2 entering the basis. To keep a long story short, we just summarize the various steps in the following list:

$$B_1 = \{1, 4, 7, 8\} \quad Bu(1) = \{t^1, t^2, t^3, t^6, t^7, t^8, t^{11}, t^{12}, t^{13}\}$$

$$B_2 = \{1, 3, 4, 7\} \quad Bu(2) = \{t^4, t^5, t^9\}$$

$$B_3 = \{1, 3, 4, 6\} \quad Bu(3) = \{t^{10}, t^{14}, t^{15}\}$$

$$B_4 = \{1, 4, 5, 6\} \quad Bu(4) = \{t^{19}, t^{20}, t^{24}, t^{25}\}$$

$$B_5 = \{1, 2, 4, 5\} \quad Bu(5) = \{t^{18}, t^{22}, t^{23}\}$$

$$B_6 = \{1, 2, 4, 8\} \quad Bu(6) = \{t^{16}, t^{17}, t^{21}\}$$

$$B_7 = \{1, 2, 5, 8\} \quad Bu(7) = \emptyset .$$

Several paths are possible, as one may have chosen B_6 instead of B_2 as a second basis. Also, the graph may take the form of a tree, and more elaborate techniques for constructing the graph and recovering the bases can be used, see Gassmann [1988] and Wets [1983b].

Research has also been done to find an appropriate root of the tree (Haugland and Wallace [1988]) and to develop preprocessing techniques (Wallace and Wets [1992]). Other attempts include the sifting procedure, a sort of parametric analysis proposed by Gartska and Rutenberg [1973]. Finally, parallel processing may be helpful in the search of the optimal multipliers in the second stage. As an example, Ariyawansa and Hudson [1991] designed a parallel implementation of the L -shaped algorithm, in which the computation of the dual simplex multipliers in Step 3 is parallelized. Linderoth and Wright [2003] also took considerable advantage of parallel processing in their trust region version as noted above.

Exercise

1. Consider the capacity expansion example from Section 1.3. Order the equipment in increasing order of utilization cost $q_1 \leq q_2 \leq \dots$. Observe that it is always optimal to use the equipment in that order. Then obtain a full decomposition of pos W .

5.5 Basis Factorization and Interior Point Methods

As observed earlier in this chapter, the matrices in (1.1) and its dual have a special structure that may allow efficient specific basis factorizations. In this way, the extensive form of the problem may be more efficiently solved by either extreme point or interior point methods. There are similarities with the previous decomposition approaches. We discuss relative advantages and disadvantages at the end of this section.

Basis factorization for extreme point methods has generally been considered the dual structure, although the same ideas apply to either the dual or primal problems. For more details on this approach, we refer to Kall [1979] and Strazicky [1980]. We consider the primal approach because, generally, the number of columns ($n_1 + Kn_2$) is larger than the number of rows ($m_1 + Km_2$) in the original constraint matrix. In this case, we can write a basic solution as $(x_{I_0}, x_{I_1}, \dots, x_{I_K}, y_{J_1}, \dots, y_{J_k})$, where I_j , $j = 0, \dots, K$, and J_l , $l = 1, \dots, K$, are index sets that may be altered at each iteration. The constraints are also partitioned according to these index sets so that a basis is:

$$B = \begin{pmatrix} A_{I_0} & A_{I_1} & \dots & A_{I_K} & \\ T_{1,I_0} & T_{1,I_1} & \dots & T_{1,I_K} & W_{J_1} \\ \vdots & \vdots & \vdots & \vdots & \\ T_{K,I_0} & T_{K,I_1} & \dots & T_{K,I_K} & W_{J_K} \end{pmatrix}. \quad (5.1)$$

For Example 2 in Section 5.1, a basis B corresponding to $x = 0$, $y_j^1 = \xi_j$, $j = 1, 2, 3$, is

$$B^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (5.2)$$

where the first column corresponds to a slack variables $s \geq 0$ such that $x + s = 10$ and $W_{J_k} = [1]$ for $k = 1, 2, 3$.

The main observation in basis factorization is that we may permute the rows of B to achieve an efficient form. This is the result of the following proposition.

Proposition 16. *A basis matrix, B , for problem (1.1) is equivalent after a row permutation P to*

$$B' = PB = \begin{pmatrix} D & C \\ F & L \end{pmatrix}, \quad (5.3)$$

where D is square invertible and at most $n_1 \times n_1$ and L is an invertible matrix of K invertible blocks of sizes at most $m_2 \times m_2$ each.

Proof: We can perform the required permutation on B in (5.1). First, note that the number of columns in A_{I_0}, \dots, A_{I_K} is at most n_1 for B to be nonsingular. We must also be able to form a nonsingular submatrix from these columns if B is invertible. Suppose this matrix is composed of A_{I_0}, \dots, A_{I_K} and rows T_{ku, I_j} from each subproblem $j = 1, \dots, K$. In this case, we have constructed

$$D = \begin{pmatrix} A_{I_0} & A_{I_1} & \dots & A_{I_K} \\ T_{1u, I_0} & T_{1u, I_1} & \dots & T_{1u, I_K} \\ \vdots & \vdots & \vdots & \vdots \\ T_{Ku, I_0} & T_{Ku, I_1} & \dots & T_{Ku, I_K} \end{pmatrix}.$$

Hence,

$$C = \begin{pmatrix} 0 & 0 & \dots & 0 & 0 \\ W_{1u, J_1} & 0 & \dots & 0 & 0 \\ 0 & \ddots & 0 & \dots & 0 \\ \vdots & 0 & W_{ku, J_k} & 0 & \vdots \\ 0 & \dots & 0 & \ddots & 0 \\ 0 & 0 & \dots & 0 & W_{Ku, J_K} \end{pmatrix}.$$

Next, assume that the remaining rows of T_{k, I_j} are T_{kl, I_j} . We then obtain:

$$F = \begin{pmatrix} T_{1l, I_0} & T_{1l, I_1} & \dots & T_{1l, I_K} \\ \vdots & \vdots & \vdots & \vdots \\ T_{Kl, I_0} & T_{Kl, I_1} & \dots & T_{Kl, I_K} \end{pmatrix}$$

and

$$L = \begin{pmatrix} W_{1l,J_1} & 0 & 0 \\ 0 & \dots W_{kl,J_k} \dots & 0 \\ 0 & 0 & W_{Kl,J_K} \end{pmatrix}.$$

Because D has rank at least m_1 , each W_{kl,J_k} in L has rank at most m_2 . This gives the result. \square

For Example 2 from Section 5.1, the solution, $x^1 = 1$ and $y_k^1 = \xi_k - 1$, $k = 1, 2, 3$, corresponds to the basis:

$$B^1 = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}, \quad (5.4)$$

which already has the form in Proposition 5 with $D = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$, $C = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$, $F = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$, and $L = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$. Note in this case that $W_1 = W_{1u} = W_{1l} = \emptyset$, an empty matrix.

To show how the partition in Proposition 5 is used, consider the forward transformation to find the basic values of $(x_{I_0}, x_{I_1}, \dots, x_{I_K}, y_{J_1}, \dots, y_{J_K})$, which we write as (x_B, y_B) , that solve:

$$Dx_B + Cy_B = b' ; \quad Fx_B + Ly_B = h' , \quad (5.5)$$

where $b' = \begin{pmatrix} b \\ h_u \end{pmatrix}$, $h' = h_l$, h_u corresponds to the components of the right-hand side for rows of T in D , and h_l corresponds to the components with rows in F .

Note that L is invertible; so,

$$y_B = L^{-1}(h' - Fx_B) . \quad (5.6)$$

Substituting in the first system of equations yields

$$(D - CL^{-1}F)x_B = b' - CL^{-1}h' . \quad (5.7)$$

Hence, we use L to solve for the columns of $L^{-1}F$ and $L^{-1}h'$, then form the working basis, $(D - CL^{-1}F)$, to solve for x_B , and multiply x_B again by $L^{-1}F$ and subtract from $L^{-1}h'$ to obtain y_B . Because most of the work involves just the square block matrices in L and the working basis, substantial effort can be saved in the decomposition procedure (see Exercise 1). The backward transformation can also be performed by taking advantage of this structure (see Exercise 2). The other forward transformation in the simplex method to find the leaving column is, of course, the same as the operations used in (5.6) and (5.7).

For basis B^1 in the Example 2, $b' = [1, 0, 1]^T$, $D = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$, and $C = 0_{2 \times 2}$, yields a solution to 5.7 with $x_B = [1, 9]^T$, where again the second component corresponds to the first-period slack variable. Now, with $h' = [2, 4]$, $F = \begin{pmatrix} 1 & 0 \\ 1 & 0 \end{pmatrix}$, and $L = I$, the solution to (5.6) is $y_B = [1, 3]^T$.

The entire simplex method then has the following form.

Basic Factorization Simplex Method

Step 0. Suppose that $(x_{B^0}^0, y_{B^0}^0) = (x_{I_0^0}^0, \dots, x_{I_K^0}^0, y_{J_0^0}^0, \dots, y_{J_K^0}^0)$ is an initial basic feasible solution for (1.1), with initial indices partitioned according to $B^0 = \{\beta_1^0, \dots, \beta_{l^0}^0\} = \{I_i^0, i = 0, \dots, K\}$ and $B^{0'} = \{\beta_{1,1}^{0'}, \dots, \beta_{1,l'_1}^{0'}, \dots, \beta_{K,1}^{0'}, \dots, \beta_{K,l'_K}^{0'}\} = J_j^0, j = 1, \dots, K$. Let the initial permutation matrix be P^0 , and set $v = 0$.

Step 1. Solve $(\rho^T, \pi^T) \begin{pmatrix} D & C \\ F & L \end{pmatrix} = (c_{B^0}^T, \hat{q}_{\beta^0})$, where $\hat{q}_{k,i} = p_k q_{k,i}$.

Step 2. Find $\bar{c}_s = \min_j \{c_j - (\rho^T | \pi^T) P^v (A_{\cdot,j}^T | T_{1,j}^T | \dots | T_{K,j}^T)^T\}$ and $\bar{q}_{k',s'} = \min_{j,k} \{p_k q_{k,j} - (\rho^T | \pi^T) P^v (0 \dots W_{k,j} \dots 0)\}$. If $\bar{c}_s \geq 0$ and $\bar{q}_{k',s'} \geq 0$, then stop; the current solution is optimal. Otherwise, if $\bar{c}_s < \bar{q}_{k',s'}$, go to Step 4. If $\bar{c}_s \geq \bar{q}_{k',s'}$, go to Step 3.

Step 3. Solve for the entering column, $\begin{pmatrix} D & C \\ F & L \end{pmatrix} \bar{W}_{k',s'} = P^v (0 \dots W_{k',s'}^T \dots 0)^T$. Let

$$\theta = x_{B^v(r)}^v / \bar{W}_{k',rs'} = \min_{\bar{W}_{k',is'} > 0, 1 \leq i \leq l^v} \{x_{B^v(i)}^v / \bar{W}_{k',is'}\} \quad (5.8)$$

and

$$\theta' = y_{B^{v'}(r')}^v / \bar{W}_{k',r's'} = \min_{\bar{W}_{k',is'} > 0, l^v + 1 \leq i \leq m_1 + K m_2} \{y_{B^{v'}(i)}^v / \bar{W}_{k',is'}\}. \quad (5.9)$$

If no minimum exists in either (5.8) or (5.9), then stop; the problem is unbounded. Otherwise, if $\theta < \theta'$, go to Step 5. If $\theta \geq \theta'$, go to Step 6.

Step 4. Solve for the entering column, $\begin{pmatrix} D & C \\ F & L \end{pmatrix} \bar{A}_{\cdot,s'} = P^v (A_{\cdot,s'}^T | T_{1,s'}^T | \dots | T_{K,s'}^T)^T$.

Let

$$\theta = x_{B^v(r)}^v / \bar{A}_{rs} = \min_{\bar{A}_{is} > 0, 1 \leq i \leq l^v} \{x_{B^v(i)}^v / \bar{A}_{is}\} \quad (5.10)$$

and

$$\theta' = y_{B^{v'}(r')}^v / \bar{A}_{r's} = \min_{\bar{A}_{is} > 0, l^v + 1 \leq i \leq m_1 + K m_2} \{y_{B^{v'}(i)}^v / \bar{A}_{is}\}. \quad (5.11)$$

If no minimum exists in either (5.10) or (5.11), then stop; the problem is unbounded. Otherwise, if $\theta < \theta'$, go to Step 5. If $\theta \geq \theta'$, go to Step 6.

Step 5. Let $B^{v+1} = B^v$, $B^{v+1'} = B^{v'}$, $I_i^{v+1} = I_i^v$, and $J^{v+1} = J^v$. Suppose $B^v(r) = I_{j,w}^v = t$. If x_s is entering, then let $B^{v+1}(r) = I^{v+1}(j,w) = s$. If $y_{k's'}$ is entering, then let $B^{v+1}(i) = B^v(i+1)$, $i \geq r$, $I_{j,i}^{v+1} = I_{j,i+1}^v$, $i \geq w$, $J_{k',l'_k+1}^{v+1} = s'$, and $l'_k = l'_k + 1$. Update P^v to P^{v+1} , the factorization correspondingly, let $v = v + 1$, and go to Step 1.

Step 6. Let $B^{v+1} = B^v$, $B^{v+1'} = B^{v'}$, $I_i^{v+1} = I_i^v$, and $J^{v+1} = J^v$. Suppose $B^{v'}(r') = J_{k,w}^v = t$. If x_s is entering, then let $B^{v+1}(\sum_{j=1}^k l_j) = I^{v+1}(k,l_k+1) = s$, $B^{v+1}(i) = B^v(i-1)$, $i > \sum_{j=1}^k l_j$, $l_k = l_{k+1}$, $J_{k,i}^{v+1} = J_{k,i+1}^v$, $i \geq w$. If $y_{k's'}$ is entering, then let $B^{v+1}(i) = B^v(i+1)$, $i \geq r$, $I_{j,i}^{v+1} = I_{j,i+1}^v$, $i \geq w$, $J_{k',l'_k+1}^{v+1} = s'$, $J_{k,i}^{v+1} = J_{k,i+1}^{v+1}$, $i \geq w$, $l'_k = l'_k - 1$, and $l'_k = l'_k + 1$. Update P^v to P^{v+1} , the factorization correspondingly, let $v = v + 1$, and go to Step 1.

For updating a factorization of the basis as used in (5.6) and (5.7), several cases need to be considered according to the possibilities in Steps 5 and 6 (see Exercise 3). If the entering and leaving variables are both in x , then only D changes. Substantial effort can again be saved. In other cases, only one block of L is altered by any iteration so we can again achieve some savings by only updating the corresponding parts of $L^{-1}F$ and $L^{-1}h$.

As mentioned earlier, this procedure can also apply to the dual of (1.1) and the primal. In this case, the procedure can mimic decomposition procedures and entails essentially the same work per iteration as the L -shaped method (see Birge [1988b]) or the inner linearization method applied to the dual. If choices of entering columns are restricted in a special variant of a decomposition procedure, then factorization and decomposition follow the same path.

In general, decomposition methods have been favored for this class of problems because they offer other paths of solutions, require less overhead, and, by maintaining separate subproblems, allow for parallel computation. The extensive form offers little hope for efficient solution, so it is not surprising that even sophisticated factorizations would not prove beneficial. Because most commercial methods already have substantial capabilities for exploiting general matrix structure, it is difficult to see how substantial gains could be obtained from basis factorization alone for a direct extreme point approach. Combinations of decomposition and factorization approaches may, however, be beneficial, as observed in Birge [1985b].

Factorization schemes also offer substantial promise for interior point methods, where there is much speculation that the solution effort grows linearly in the size of the problem. This observation is supported by the results we present here. For this discussion, we assume that the interior point method follows a standard form version of Karmarkar's projective algorithm (Karmarkar [1984]). We also assume an unknown optimal objective value and use Todd and Burrell's [1986] method for updating a lower bound on the optimal objective value. We use an initial lower bound, as is often available in practice. An alternative is Anstreicher's [1989] method to obtain an initial lower bound.

Many other interior point methods are available (see, e.g., Roos, Terlaky, and Vial [2005] and Ye [1997]). In particular, many commercial solvers use the homogeneous self-dual formulation of the standard linear program (see, e.g., Andersen [2009]). Other interior point methods and interpretations include path-following, logarithmic barrier, and affine scaling (see Roos, Terlaky, and Vial [2005] for descriptions of alternatives). They all follow similar steps to the method given below.

We first describe the algorithm for a standard linear program:

$$\begin{aligned} & \min c^T x \\ \text{s. t. } & Ax = b, \\ & x \geq 0, \end{aligned} \tag{5.12}$$

where $x \in \mathbb{R}^n$, $c \in \mathbb{Z}^n$ (i.e., an n -vector of rationals), $b \in \mathbb{Z}^m$, $A \in \mathbb{Z}^{m \times n}$ with optimal value $c^T x^* = z^*$. In referring to the parameters in (5.12), we use ext as a modifier, e.g., c_{ext} , when necessary to distinguish the parameters in (5.12) from our standard stochastic program form in (1.1).

Suppose we have a strictly interior feasible point x^0 of (5.12), i.e.,

$$Ax^0 = b, \quad x^0 > 0, \tag{5.13}$$

a lower bound β^0 on z^* , and the set of optimal solutions in (5.12) is bounded. Note that if we do not have a feasible solution, we can solve a phase-one problem or use the self-dual form of the problem. In that case, the goal becomes finding (x, t, λ) to solve:

$$\begin{aligned} & \min 0 \\ \text{s. t. } & Ax - bt = 0, \\ & -A^T \lambda + ct \geq 0, \\ & b^T \lambda - c^T x \geq 0, \\ & x \geq 0, \quad t \geq 0, \end{aligned} \tag{5.14}$$

which can be solved by an interior point method initiated at any solution (x^0, t^0, λ^0) with $x^0 > 0$ and $t^0 > 0$ by iteratively choosing search directions to reduce the infeasibility of the system (5.14) with solution (x^k, t^k, λ^k) at iteration k .

For exposition here, we follow the standard form variant of the projective scaling algorithm, which creates a sequence of points x^0, x^1, \dots, x^k by the following steps.

Standard Form Projective Scaling Method

Step 0. Set $v = 0$ and lower bound $\beta^0 \leq z^*$.

Step 1. If $c^T x^v - \beta^v$ is small enough, i.e., less than a given positive number ε , then stop. Otherwise, go to Step 2.

Step 2. Let $D = \text{diag}\{x_1^v, \dots, x_n^v\}$, $\hat{A} := [AD, -b]$, and let $\Pi_{\hat{A}}$ be the projection onto the null space of \hat{A} . Find

$$u = \Pi_{\hat{A}} \begin{pmatrix} Dc \\ 0 \end{pmatrix}, \quad v = \Pi_{\hat{A}} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (5.15)$$

and let $\mu(\beta^v) = \min\{u_i - \beta^v v_i : i = 1, \dots, n+1\}$. If $\mu(\beta^v) \leq 0$, let $\beta^{v+1} = \beta^v$. Otherwise, let $\beta^{v+1} = \min\{u_i/v_i : v_i > 0, i = 1, \dots, n+1\}$. Go to Step 3.

Step 3. Let $c_p = u - \beta^{v+1}v - (c^T x^v - \beta^{v+1})e/(n+1)$, where $e = (1, \dots, 1)^T \in \mathbb{R}^{n+1}$. Let

$$g' = \frac{1}{n+1}e - \alpha \frac{c_p}{\|c_p\|_2}.$$

Let $\bar{g} \in \mathbb{R}^n$ consist of the first n components of g' . Then $x^{v+1} = D\bar{g}/g'_{n+1}$, $v = v + 1$, go to Step 1.

For the purpose of obtaining a worst-case bound, the step length α in the definition of g' may be set equal to $\frac{1}{3(n+1)}$, (Gay [1987]), but better performance is obtained by choosing α using a line search.

To show how the structure of a stochastic program can be exploited in these methods, we consider the number of arithmetic operations in a complexity analysis. The main computational effort in each iteration of the algorithm is to compute the projections in (5.15), which requires, for $n > m$, $O(n^3)$ arithmetic operations (and, on average, $O(n^{2.5})$, operations per iteration using a rank-one updating scheme). In $O(n/\epsilon)$ iterations, or with some modifications in $O(\sqrt{n}/\epsilon)$, the method finds a solution with $O(\epsilon)$ precision.

In our case, if we consider the stochastic program (1.1) in the extensive form (5.12), then $n = n_2 + Kn_2$, $m = m_1 + Km_2$, and $x_{ext} = \begin{pmatrix} x \\ y_1 \\ \vdots \\ y_K \end{pmatrix}$, $c_{ext} = \begin{pmatrix} c \\ p_1 q_1 \\ \vdots \\ p_K q_K \end{pmatrix}$,

$$b_{ext} = \begin{pmatrix} b \\ h_1 \\ \vdots \\ h_K \end{pmatrix}, \text{ and}$$

$$A_{ext} = \begin{pmatrix} A & 0 & \dots & 0 \\ T_1 & W & \dots & 0 \\ \vdots & 0 & \ddots & 0 \\ T_K & 0 & \dots & W \end{pmatrix}. \quad (5.16)$$

The main computational work at each step of the projective scaling algorithm is to compute the projection in (5.15), which can be written as

$$\Pi_{\hat{A}} = (I - \hat{A}^T (\hat{A}\hat{A}^T)^{-1}\hat{A}), \quad (5.17)$$

where $(\hat{A}\hat{A}^T) = AD^2A^T + bb^T := M + bb^T$. In this case, the work is dominated by computing M^{-1} (or solving systems with coefficient matrix, $M = AD^2A^T$). For the general A in the formulation in (5.12), using the specific A_{ext} in the stochastic program extensive form as in (5.16) and letting $D_0 = \text{diag}(x^v)$, $D_k = \text{diag}(y_k^v)$, $k = 1, \dots, K$, we would have

$$M = \begin{pmatrix} AD_0^2A^T & AD_0^2T_1^T & \dots & AD_0^2T_K^T \\ T_1D_0^2A^T & T_1D_0^2T_1^T + WD_1^2W^T & \dots & T_1D_0^2T_K^T \\ \vdots & \vdots & \ddots & \vdots \\ T_KD_0^2A^T & T_1D_0^2T_K^T & \dots & T_KD_0^2T_K^T + WD_K^2W^T \end{pmatrix}, \quad (5.18)$$

which is clearly much denser than the original constraint matrix in (1.1). In this case, a straightforward implementation of an interior point method that solves systems with M is quite inefficient.

To see the structure, we consider Example 2 from Section 5.1. Here,

$$A_{ext} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{pmatrix}. \quad (5.19)$$

Now, let $x_{ext}^0 = (3, 7, 1, 3, 1, 2, 2, 1)^T$ in (5.13) represent $x^0 = (3, 7)^T$, $y_1^0 = (1, 3)^T$, $y_2^0 = (1, 2)^T$, and $y_3^0 = (2, 1)^T$ in Example 2. We then have $D^0 = \text{diag}(3, 7)$, $D_1 = \text{diag}(1, 3)$, $D_2 = \text{diag}(1, 2)$, and $D_3 = \text{diag}(2, 1)$. The matrix M in this case is:

$$M = \begin{pmatrix} 58 & 9 & 9 & 9 \\ 9 & 19 & 9 & 9 \\ 9 & 9 & 14 & 9 \\ 9 & 9 & 9 & 14 \end{pmatrix}. \quad (5.20)$$

While M is dense, it in fact has a great deal of structure that can be exploited in any solution scheme. This is the object of the factorization scheme given by Birge and Qi [1988] (see also Birge and Holmes [1992] for an implementation discussion). The following proposition gives the essential characterization of that factorization.

Proposition 17. Let $S_0 = I_2 \in \Re^{m_1 \times m_1}$, $S_l = W_l D_l^2 W_l^T$, $l = 1, \dots, K$, $S = \text{diag}\{S_0, \dots, S_K\}$. Then $S^{-1} = \text{diag}\{S_0, S_1^{-1}, \dots, S_K^{-1}\}$. Let I_1 and I_2 be identity matrices of dimensions n_1 and m_1 , respectively. Let

$$G_1 = (D_0)^{-2} + A^T S_0^{-1} A + \sum_{l=1}^K T_l^T S_l^{-1} T_l, \quad G_2 = -A G_1^{-1} A^T, \quad (5.21)$$

$$U = \begin{pmatrix} A & I_2 \\ T_1 & 0 \\ \vdots & \vdots \\ T_K & 0 \end{pmatrix}, \quad V = \begin{pmatrix} A & -I_2 \\ T_1 & 0 \\ \vdots & \vdots \\ T_K & 0 \end{pmatrix}.$$

If A , $W_k, k = 1, \dots, K$ have full row rank, then G_2 and M are invertible and

$$\begin{aligned} M^{-1} &= S^{-1} - S^{-1}U \begin{pmatrix} I_1 & -G_1^{-1}A^T \\ 0 & I_2 \end{pmatrix} \begin{pmatrix} I_1 & 0 \\ 0 & -G_2^{-1} \end{pmatrix} \\ &\quad \begin{pmatrix} I_1 & 0 \\ A & I_2 \end{pmatrix} \begin{pmatrix} G_1^{-1} & 0 \\ 0 & I_2 \end{pmatrix} V^T S^{-1}. \end{aligned} \quad (5.22)$$

Proof: Exercise 6. \square

Following the assumptions, the number of arithmetic operations using this factorization can be reduced from $O((n_1 + Kn_2)^4)$ as in the general projective scaling method. Using the factorization, the effort is, in fact, dominated by $O(K(n_2^3 + n_2^2 n_1 + n_2 n_1^2))$. It is also possible to reduce this bound further with a partial rank-one updating scheme as mentioned earlier. In this case, for $n = n_1 + Kn_2$, the complexity using the factorization in (5.22) becomes $O((n^{0.5} n_2^2 + n \max\{n_1, n_2\} + n_1^3)/\varepsilon)$ for the entire algorithm, or, if $K \sim n_1 \sim n_2$, the full arithmetic complexity is $O(n^{2.5}/\varepsilon)$, compared to the general result of $O(n^{3.5}/\varepsilon)$. Thus, the factorization in (5.22) provides an order of magnitude improvement over a general solution scheme if the number of realizations K approaches the number of variables in the first and second stage.

In practice, we would not compute M^{-1} explicitly, but solve a set of systems as follows:

$$Mv = u \quad (5.23)$$

using

$$v = p - r, \quad (5.24)$$

where

$$Sp = u, \quad Gq = V^T p, \quad Sr = Uq, \quad (5.25)$$

where G is the inverse of the matrix between U and V^T in (5.22):

$$G = \begin{pmatrix} G_1 & A^T \\ -A & 0 \end{pmatrix} = \begin{pmatrix} G_1 & 0 \\ 0 & I_2 \end{pmatrix} \begin{pmatrix} I_1 & 0 \\ A & I_2 \end{pmatrix}^{-1} \begin{pmatrix} I_1 & 0 \\ 0 & -G_2 \end{pmatrix} \begin{pmatrix} I_1 & -G_1^{-1}A^T \\ 0 & I_2 \end{pmatrix}^{-1}. \quad (5.26)$$

The systems in (5.25) require solving systems with S_l , computation of G_1 and G_2 , and solving systems with G_1 and G_2 . In practice, we find a Cholesky factorization of each S_l , use them to find G_1 and G_2 , and find Cholesky factorizations of G_1 and G_2 .

For Example 2, with initial values $(x^0, y_1^0, \dots, y_K^0)$ given above, we have

$$S_0 = [1]; S_1 = [10]; S_2 = [5]; S_3 = [5]; \quad (5.27)$$

$$G_1 = \begin{pmatrix} \frac{1}{9} & 0 \\ 0 & \frac{1}{49} \end{pmatrix} + \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} + \begin{pmatrix} \frac{1}{10} & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} \frac{1}{5} & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} \frac{1}{5} & 0 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 1.61 & 1 \\ 1 & 1.02 \end{pmatrix}; \quad (5.28)$$

$$G2 = -[1 \ 1] \begin{pmatrix} 1.61 & 1 \\ 1 & 1.02 \end{pmatrix}^{-1} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = [-0.98]; \quad (5.29)$$

$$U = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}; V = \begin{pmatrix} 1 & 1 & -1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}. \quad (5.30)$$

To solve for v in $Mv = u$, we first solve $Sp = u$ as

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{pmatrix} \quad (5.31)$$

to obtain:

$$p_1 = u_1, p_2 = 0.1u_2; p_3 = 0.2u_3; p_4 = 0.2u_4. \quad (5.32)$$

Next, we find

$$V^T p = \begin{pmatrix} u_1 + 0.1u_2 + 0.2u_3 + 0.2u_4 \\ u_1 \\ -u_1 \end{pmatrix}. \quad (5.33)$$

Next, we solve $Gq = V^T p$ as follows:

- find q^1 such that $\begin{pmatrix} G_1 & 0 \\ 0 & I_2 \end{pmatrix} q^1 = V^T p$ as

$$\begin{pmatrix} 1.61 & 1 & 0 \\ 1 & 1.02 & 0 \\ 0 & 0 & 1 \end{pmatrix} q^1 = \begin{pmatrix} u_1 + 0.1u_2 + 0.2u_3 + 0.2u_4 \\ u_1 \\ -u_1 \end{pmatrix} \quad (5.34)$$

$$\text{to obtain } q^1 = \begin{pmatrix} 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.95u_1 - 0.16u_2 - 0.31u_3 - 0.31u_4 \\ -u_1 \end{pmatrix};$$

- find q^2 such that $q^2 = \begin{pmatrix} I_1 & 0 \\ A & I_2 \end{pmatrix} q^1$ as

$$q^2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} q^1 = \begin{pmatrix} 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.95u_1 - 0.16u_2 - 0.31u_3 - 0.31u_4 \\ -0.02u_1 + 0.003u_2 + 0.006u_3 + 0.001u_4 \end{pmatrix}; \quad (5.35)$$

- find q^3 such that $\begin{pmatrix} I_1 & 0 \\ 0 & -G_2 \end{pmatrix} q^3 = q^2$ as

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.98 \end{pmatrix} q^3 = q^2 \quad (5.36)$$

- to obtain $q^3 = \begin{pmatrix} 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.95u_1 - 0.16u_2 - 0.31u_3 - 0.31u_4 \\ -0.02u_1 + 0.003u_2 + 0.01u_3 + 0.01u_4 \end{pmatrix}$;
- find $q = q^4$ such that $q^4 = \begin{pmatrix} I_1 & -G_1^{-1}A^T \\ 0 & I_2 \end{pmatrix} q^3$ as
- $$q = \begin{pmatrix} 1 & 0 & -0.03 \\ 0 & 1 & -0.95 \\ 0 & 0 & 1 \end{pmatrix} q^3 = \begin{pmatrix} 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.97u_1 - 0.16u_2 - 0.32u_3 - 0.32u_4 \\ -0.02u_1 + 0.003u_2 + 0.01u_3 + 0.01u_4 \end{pmatrix}. \quad (5.37)$$

The next step is to solve for $Sr = Uq$ as

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 5 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{pmatrix} = \begin{pmatrix} 0.98u_1 + 0.003u_2 + 0.01u_3 + 0.01u_4 \\ 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \\ 0.03u_1 + 0.16u_2 + 0.32u_3 + 0.32u_4 \end{pmatrix} \quad (5.38)$$

or $r = \begin{pmatrix} 0.98u_1 + 0.003u_2 + 0.01u_3 + 0.01u_4 \\ 0.003u_1 + 0.02u_2 + 0.03u_3 + 0.03u_4 \\ 0.01u_1 + 0.03u_2 + 0.06u_3 + 0.06u_4 \\ 0.01u_1 + 0.03u_2 + 0.06u_3 + 0.06u_4 \end{pmatrix}$, which finally yields $v = p - r$ as

$$v = \begin{pmatrix} 0.02u_1 - 0.003u_2 - 0.01u_3 - 0.01u_4 \\ -0.003u_1 + 0.08u_2 - 0.03u_3 - 0.03u_4 \\ -0.01u_1 - 0.03u_2 + 0.14u_3 - 0.06u_4 \\ -0.01u_1 - 0.03u_2 - 0.06u_3 - 0.14u_4 \end{pmatrix}, \quad (5.39)$$

which can be seen as $M^{-1}u$ for M in (5.20).

Now, for the projection operation defined in (5.17), note that

$$(\hat{A}\hat{A}^T)^{-1}\hat{A} = M^{-1}\hat{A} - M^{-1}bb^TM^{-1}\hat{A}/(1+b^TM^{-1}b), \quad (5.40)$$

which requires finding V^1 and v^2 such that $MV^1 = \hat{A}$ and $Mv^2 = b$ where

$$\hat{A} = \begin{pmatrix} 3 & 7 & 0 & 0 & 0 & 0 & 0 & -10 \\ 3 & 0 & 1 & -3 & 0 & 0 & 0 & -1 \\ 3 & 0 & 0 & 0 & 1 & -2 & 0 & -2 \\ 3 & 0 & 0 & 0 & 0 & 0 & 2 & -4 \end{pmatrix} \text{ and } b = \begin{pmatrix} 10 \\ 1 \\ 2 \\ 4 \end{pmatrix}, \quad (5.41)$$

where note that v^2 is also the negative of the last column of V^1 .

Using (5.39) then yields

$$V_1 = [V_{11} \ -v_2] = \begin{pmatrix} 0.01 & 0.14 & -0.003 & 0.01 & -0.01 & 0.01 & -0.01 & 0.01 & -0.16 \\ 0.05 & -0.02 & 0.08 & -0.25 & -0.03 & 0.06 & -0.06 & 0.0 & 0.14 \\ 0.11 & -0.05 & -0.03 & 0.10 & 0.14 & -0.27 & -0.13 & 0.06 & 0.08 \\ 0.11 & -0.05 & -0.03 & 0.10 & -0.06 & 0.13 & 0.27 & -0.14 & -0.32 \end{pmatrix} \quad (5.42)$$

From (5.40), $(\hat{A}\hat{A}^T)^{-1}\hat{A} = V_1 - v_2 b^T V_1 / (1 + b^T v_2)$ or

$$(\hat{A}\hat{A}^T)^{-1}\hat{A} = \begin{pmatrix} -0.02 & 0.10 & 0.003 & -0.01 & -0.003 & 0.01 & -0.04 & 0.02 & -0.04 \\ 0.08 & 0.02 & 0.08 & -0.24 & -0.03 & 0.07 & -0.04 & 0.02 & 0.04 \\ 0.12 & -0.02 & -0.03 & 0.10 & 0.14 & -0.27 & -0.11 & 0.06 & 0.02 \\ 0.03 & -0.14 & -0.02 & 0.06 & -0.06 & 0.11 & 0.21 & -0.11 & -0.09 \end{pmatrix}. \quad (5.43)$$

Finally, the search direction components u and v in (5.15) are then:

$$u = (I - \hat{A}(\hat{A}\hat{A}^T)^{-1}\hat{A}) \begin{pmatrix} Dc_{ext} \\ 0 \end{pmatrix} = \begin{pmatrix} 0.31 \\ 0.17 \\ 0.53 \\ 0.42 \\ 0.43 \\ 0.47 \\ 0.24 \\ 0.55 \\ 0.22 \end{pmatrix}; v = (I - \hat{A}(\hat{A}\hat{A}^T)^{-1}\hat{A}) \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0.22 \\ 0.32 \\ -0.04 \\ 0.12 \\ -0.02 \\ 0.04 \\ 0.18 \\ -0.09 \\ 0.28 \end{pmatrix}. \quad (5.44)$$

Note how these operations only required solutions with G_1 ($n_1 \times n_1$), G_2 ($m_1 \times m_1$), and S (K solutions using $m_2 \times m_2$ matrices). After finding u and v , the other operations in the project scaling method only involve simple operations on vectors of the same size. Exercise 7 asks for completion of these operations until the objective value is within 0.01 of the bound.

Other factorizations or formulations can also yield advantages for interior point methods. These include the following approaches:

1. Schur complement updates;
2. Column splitting;
3. Solution of the dual.

The Schur complement approach is used in many interior point method implementations. The basic idea is to write M as the sum of a matrix with sparse columns, $A_s D_s^2 A_s^T$, and a matrix with dense columns, $A_d D_d^2 A_d^T$. Using a Cholesky factorization of the sparse matrix, $LL^T = A_s D_s^2 A_s^T$, the method involves solving $Mu = v$ by:

$$\begin{pmatrix} LL^T & -A_d D_d \\ D_d A_d^T & I \end{pmatrix} \begin{pmatrix} v \\ w \end{pmatrix} = \begin{pmatrix} u \\ 0 \end{pmatrix}, \quad (5.45)$$

which requires solving $[I + D_d A_d^T (LL^T)^{-1} A_d D_d]w = -D_d A_d^T (LL^T)^{-1} b$ and $LL^T v = b + A_d D_d w$, where $I + D_d A_d^T (LL^T)^{-1} A_d D_d$ is a Schur complement.

The Schur complement is thus quite similar to the factorization method given earlier. If every column of x is considered a dense column, then the remaining matrix is quite sparse but rank deficient. The factorization in (5.22) is a method for maintaining an invertible matrix when $A_s D_s^2 A_s^T$ is singular. It can thus be viewed as an extension of the Schur complement to the stochastic linear program. Because of the possible rank deficiency and the size of the Schur complement, the straightforward

Schur complement approach in (5.45) is quick but can lead to numerical instabilities (see Carpenter, Lustig, and Mulvey [1991]).

Carpenter et al. also propose the column splitting technique. The basic idea is to rewrite problem (1.1) with explicit constraints on nonanticipativity. The formulation then becomes:

$$\min \quad \sum_{k=1}^K p_k(c^T x_k + q_k^T y_k) \quad (5.46)$$

$$\text{s. t.} \quad Ax_k = b, \quad (5.47)$$

$$T_k x_k + W y_k = h_k, \quad k = 1, \dots, K, \quad (5.48)$$

$$x_k - x_{k+1} = 0, \quad k = 1, \dots, K-1, \quad (5.49)$$

$$x_k \geq 0, \quad y_k \geq 0, \quad k = 1, \dots, K. \quad (5.50)$$

The difference now is that the constraints in (5.47) and (5.48) separate into separate subproblems k and constraints (5.49) link the problems together. Alternating constraints, (5.47), (5.48) and (5.49) for each k in sequence, the full constraint matrix has the form:

$$\bar{A} = \begin{pmatrix} A & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ T_1 & W & 0 & 0 & 0 & 0 & 0 & 0 \\ I & 0 & -I & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & A & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & T_2 & W & 0 & 0 & 0 & 0 \\ 0 & 0 & I & 0 & -I & 0 & 0 & 0 \\ \vdots & \vdots & 0 & \ddots & \vdots & \ddots & 0 & \vdots \\ 0 & 0 & 0 & 0 & I & 0 & -I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & A & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & T_k & W \end{pmatrix}. \quad (5.51)$$

If we form $\bar{A}\bar{A}^T$, then we obtain $\bar{A}\bar{A}^T =$

$$\begin{pmatrix} AA^T & AT_1^T & A & 0 & 0 & 0 & 0 & 0 \\ T_1 A^T & T_1 T_1^T & T_1 & 0 & 0 & 0 & 0 & 0 \\ & WW^T & & & & & & \\ A^T & T_1^T & 2I & -A^T & 0 & 0 & 0 & 0 \\ 0 & 0 & -A & AA^T & AT_2^T & A & 0 & 0 \\ 0 & 0 & T_2 A^T & T_2 T_2^T & T_2 & 0 & 0 & 0 \\ & & & WW^T & & & & \\ 0 & 0 & 0 & T_2^T & 2I & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & 0 & \vdots \\ 0 & 0 & 0 & A^T & T_{K-1}^T & 2I & -A^T & 0 \\ 0 & 0 & 0 & 0 & 0 & -A & AA^T & AT_K^T \\ 0 & 0 & 0 & 0 & 0 & 0 & T_K A^T & T_K T_K^T \\ & & & & & & & WW^T \end{pmatrix}, \quad (5.52)$$

which is clearly much sparser than the original matrix in (5.18). It is, however, larger than the matrix in (5.18) (see Exercise 8) so there is some tradeoff for the reduced density.

The third additional approach is to form the dual of (1.1) and to solve that problem using the same basic interior point method we gave earlier. (In the self-dual form for (5.14), this corresponds to eliminating the primal variables first and then solving for the dual variables. The primal projective scaling method corresponds to eliminating the dual variables and then solving for the primal variables. Another alternative for the self-dual form is directly to solve the full system again taking advantage of the stochastic program constraint structure.) The dual approach considers the problem:

$$\max b^T \rho + \sum_{k=1}^K p_k \pi_k^T h_k \quad (5.53)$$

$$\text{s. t. } A^T \rho + \sum_{k=1}^K p_k T_k^T \pi_k \leq c, \quad k = 1, \dots, K, \quad (5.54)$$

$$W^T \pi_k \leq q, \quad k = 1, \dots, K, \quad (5.55)$$

where the variables are not restricted in sign. For this problem, we can achieve a standard form as in (5.12) by splitting the variables π_k and ρ into differences of non-negative variables and by adding slack variables to constraints (5.54) and (5.55)¹. In this way the constraint matrix for (5.54) and (5.55) becomes $A' =$

$$\begin{pmatrix} A^T & -A^T & T_1^T & -T_1^T & 0 & T_2^T & -T_2^T & \dots & T_K^T & -T_K^T & 0 & I \\ 0 & 0 & W^T & -W^T & I & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & W^T & -W^T & I & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & 0 & \ddots & \ddots & \ddots & 0 & 0 & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & W^T & -W^T & I & 0 \end{pmatrix}. \quad (5.56)$$

The matrix in (5.56) may again be much larger than the matrix in the original, but the gain comes in considering $A'A'^T$ which is now:

$$\begin{pmatrix} 2(A^T A + \sum_{k=1}^K T_k^T T_k) + I & 2T_1^T W & 2T_2^T W & \dots & 2T_K^T W \\ 2W^T T_1 & 2W^T W + I & 0 & 0 & 0 \\ 2W^T T_2 & 0 & 2W^T W + I & 0 & 0 \\ \vdots & 0 & 0 & \ddots & 0 \\ 2W^T T_K & 0 & 0 & 0 & 2W^T W \end{pmatrix}, \quad (5.57)$$

with an inherent sparsity of which an interior point method can take advantage. In fact, it is not necessary to take the dual to use this alternative factorization form by again using the Sherman-Morrison-Woodbury formula (see Birge, Freund, and

¹ The dual problem may no longer have a bounded set of optima causing some theoretical difficulties for convergence results. In practice, bounds are placed on the variables to guarantee convergence.

Vanderbei [1992]). In this way, the matrix in the form of (5.57) replaces the dense matrix in (5.18).

In addition to reducing total computational effort, the factorizations described in this section also allow significant parallel processing for the computations involving sub-matrices corresponding to the second-period subproblems (see, e.g., Yang and Zenios [1997] and Gondzio and Grothey [2009]). The interior point method factorization also can be extended to multistage problems using a recursive form (see Pfug and Halada [2003]).

An additional strategy for interior point methods is to combine the outer linearization and with an interior point method so that the interior point iterations are taken with an increasingly constrained region as in the standard L-shaped method to solve (1.2)–(1.4). This method may reduce the effort in solving subproblems to optimality while still obtaining refined information about the recourse function constraints without requiring full information as in the extensive form. Bahn, Goffin, du Merle, and Vial [1995] provide a description and computational results for this approach.

Exercises

1. Use the matrix structure in Proposition 5 to complete the simplex iterations starting from basis B^1 for Example 2 from Section 5.1.
2. Compare the number of operations to solve (5.5) using (5.6) and (5.7) compared to solving (5.5) as an unstructured linear system of equations.
3. Give a similar basis factorization scheme to (5.6) and (5.7) to solve the backward transformation, $(\sigma^T, \pi^T)B = (c_B^T, q_B^T)$, for a basis corresponding to columns B from the constraint matrix of (1.1).
4. Describe an efficient updating procedure for any possible combination of entering and leaving columns in the basis matrix of (5.5) using the factorization scheme in (5.6) and (5.7).
5. Find the number of arithmetic operations for a single step of the interior point method using (5.22). Compare this to the number of arithmetic operations if no special factorization is used.
6. Prove Proposition 6.
7. Assuming an initial lower bound $\beta^0 = 0$, follow the projective scaling algorithm for Example 2 starting from the $x_{ext}^0 = (3, 7, 1, 3, 1, 2, 2, 1)^T$ until $c_{ext}^T x_{ext}^v - \beta^v < \epsilon = 0.01$. (Note: the number of iterations required here in comparison to the L-shaped method may surprise you, but the number of iterations for interior point methods generally grows slowly as the problem size increases.)
8. Compare the sizes of the adjacency matrices in (5.18) and (5.51). Assuming that each matrix A , T_k , and W is completely dense, compare the number of nonzero entries in these two matrices.

5.6 Inner Linearization Methods and Special Structures

As mentioned earlier, the most direct alternative to an outer linearization, or cut generation, approach is an inner linearization or column generation approach (see Geoffrion [1970] for other basic approaches to large-scale problems). In fact, this was the first suggestion of Dantzig and Madansky [1961] for solving stochastic linear programs. They observed that the structure of the dual in Figure 2 fits the prototype for Dantzig-Wolfe decomposition. In fact, we can derive this approach from the L -shaped method by taking duals.

Consider the following dual linear program to (1.2)–(1.4).

$$\max \zeta = \rho^T b + \sum_{\ell=1}^r \sigma_\ell d_\ell + \sum_{\ell=1}^s \pi_\ell e_\ell \quad (6.1)$$

$$\text{s. t. } \rho^T A + \sum_{\ell=1}^r \sigma_\ell D_\ell + \sum_{\ell=1}^s \pi_\ell E_\ell \leq c^T, \quad (6.2)$$

$$\sum_{\ell=1}^s \pi_\ell = 1, \sigma_\ell \geq 0, \ell = 1, \dots, r, \pi_\ell \geq 0, \ell = 1, \dots, s. \quad (6.3)$$

The linear program in (6.1)–(6.3) includes multipliers σ_ℓ on extreme rays, or directions of recession, which cannot be produced with positive combinations of other distinct recession directions, of the duals of the subproblems and multipliers π_ℓ on the expectations of extreme points of the duals of the subproblems. To see this, suppose that (6.1)–(6.3) is solved to obtain a multiplier x^ν on constraint (6.2). Now, consider the following dual to (1.9):

$$\max w = \pi^T (h_k - T_k x^\nu) \quad \text{s.t. } \pi^T W \leq q^T. \quad (6.4)$$

If (6.4) is unbounded for any k , we then must have some σ^ν such that $\sigma^{\nu T} W \leq 0$ and $\sigma^{\nu T} (h_k - T_k x^\nu) > 0$, or (1.5)–(1.6) has a feasible dual solution (hence optimal primal solution) with a positive value. So, Step 2 of the L -shaped method is equivalent to checking whether (6.4) is unbounded for any k . In this case, we form D_{r+1} and d_{r+1} as in (1.7) and (1.8) of the L -shaped method and add them to (6.1)–(6.3).

Next, note that if (6.4) is infeasible, the stochastic program is not well-formulated (see Exercise 1). Consider when (6.4) has a finite optimal value for all k . In the L -shaped method, if (1.9) was solvable for all k , then we formed E_{s+1} and e_{s+1} and added them to (1.2)–(1.4). In this case in the inner linearization procedure, we again use (1.10) and (1.11) to form E_{s+1} and e_{s+1} and add them to (6.1)–(6.3).

Solving the duals in Steps 1 to 3 of the L -shaped algorithm then consists of solving (6.1)–(6.3) as a master problem and problems (6.4) as subproblems. Formally, this method is the following inner linearization method.

Inner Linearization Algorithm

Step 0. Set $r = s = v = 0$.

Step 1. Set $v = v + 1$ and solve the linear program in (6.1)–(6.3). Let the solution be $(\rho^v, \sigma^v, \pi^v)$ with a dual solution, (x^v, θ^v) .

Step 2. For $k = 1, \dots, K$, solve (6.4). If any infeasible problem (6.4) is found, stop and evaluate the formulation. If an unbounded solution with extreme ray σ^v is found for any k , then form new columns $(d_{r+1} = (\sigma^v)^T h_k, D_{r+1} = (\sigma^v)^T T_k)$, set $r = r + 1$, and return to Step 1.

If all problems (6.4) are solvable, then form new columns, E_{s+1} and e_{s+1} , as in (1.10) and (1.11). If $e_{s+1} - E_{s+1}x^v - \theta^v \leq 0$, then stop; $(\rho^v, \sigma^v, \pi^v)$ and (x^v, θ^v) are optimal in the original problem (1.2).

If $e_{s+1} - E_{s+1}x^v - \theta^v > 0$, set $s = s + 1$, and return to Step 1.

Clearly, the inner linearization method follows the same steps as the *L*-shaped method, except that we solve the duals of the problems instead of the primals. Hence, convergence follows directly from the *L*-shaped method. We could also view this approach directly as in Dantzig-Wolfe decomposition by stating that (6.1)–(6.3) is an inner linearization of the dual of the basic *L*-shaped problem in (1.2) and that the subproblems (6.4) generate new extreme points and rays to add to this inner linearization (see Exercise 2).

If, as in many problems, $n_1 \gg m_1$, the primal version has smaller basis matrices, at most of order $m_1 + m_2$, than the $n_1 \times n_1$ bases for the dual. Hence, the *L*-shaped implementation is usually preferred. Inner linearization can, however, be applied directly to the primal by assuming T is fixed using the form in (3.1.5), which we repeat here:

$$\begin{aligned} \min z &= c^T x + \Psi(\chi) \\ \text{s. t. } &Ax = b, \\ &Tx - \chi = 0, \\ &x \geq 0, \end{aligned} \tag{6.5}$$

where $\Psi(\chi) = E_\omega \psi(\chi, \xi(\omega))$ and $\psi(\chi, \xi(\omega)) = \min\{q(\omega)^T y \mid Wy = h(\omega) - \chi, y \geq 0\}$. Note that, in this form, we assume that T is fixed but q and h may still be functions of ω . For this reason, we revert to the use of Ψ for the recourse function.

In this case, we wish to build an inner linearization of the function $\Psi(\chi)$ using the generalized programming approach as in Dantzig [1963, Chapter 24]. The basic idea is to replace $\Psi(\chi)$ by the convex hull of points $\Psi(\chi^\ell)$ chosen in each iteration of the algorithm. Each iteration generates a new extreme point of a region of linearity for Ψ , which is polyhedral as we showed in Theorem 3.6. Thus, finite convergence is assured with finite numbers of realizations. The algorithm follows.

Generalized Programming Method for Two-Stage Stochastic Linear Programs

Step 0. Set $s = t = v = 0$.

Step 1. Set $v = v + 1$ and solve the linear program master problem:

$$\min z^v = c^T x + \sum_{i=1}^r \mu_i \Psi_0^+(\zeta^i) + \sum_{i=1}^s \lambda_i \Psi(\chi^i) \quad (6.6)$$

$$\text{s. t.} \quad Ax = b, \quad (6.7)$$

$$Tx - \sum_{i=1}^r \mu_i \zeta^i - \sum_{i=1}^s \lambda_i \chi^i = 0, \quad (6.8)$$

$$\sum_{i=1}^r \lambda_i = 1, \quad (6.9)$$

$$x, \mu_i \geq 0, \quad i = 1, \dots, r, \quad \lambda_i \geq 0, \quad i = 1, \dots, s.$$

If (6.6)–(6.9) is infeasible or unbounded, stop. Otherwise, let the solution be (x^v, μ^v, λ^v) with associated dual variables, $(\sigma^v, \pi^v, \rho^v)$.

Step 2. Solve the subproblem:

$$\min_{\chi} \Psi(\chi) + (\pi^v)^T \chi - \rho^v, \quad (6.10)$$

which we assume has value less than ∞ .

If (6.10) is unbounded, a recession direction ζ^{r+1} is obtained, such that for some χ , $\Psi(\chi + \alpha \zeta^{r+1}) + (\pi^v)^T (\chi + \alpha \zeta^{r+1}) \rightarrow -\infty$ as $\alpha \rightarrow \infty$. In this case, let $\Psi_0^+(\zeta^{r+1}) = \lim_{\alpha \rightarrow \infty} \frac{\Psi(\chi + \alpha \zeta^{r+1}) - \Psi(\chi)}{\alpha}$, $r = r + 1$, and return to Step 1.

If (6.10) is solvable, let the solution be χ^{s+1} . If $\Psi(\chi) + (\pi^v)^T \chi - \rho^v \geq 0$, then stop; (x^v, μ^v, λ^v) corresponds to an optimal solution to (6.5). Otherwise, set $s = s + 1$ and return to Step 1.

This algorithm generates columns in (6.6)–(6.9) corresponding to new proposals from the subproblem in (6.10). In the two-stage stochastic linear program form, (6.10) can be recast as:

$$\begin{aligned} \min & \sum_{k=1}^K p_k q_k^T y_k + (\pi^v)^T \chi - \rho^v \\ \text{s. t.} & W y_k + \chi = h_k, \quad k = 1, \dots, K, \\ & y_k \geq 0, \quad k = 1, \dots, K. \end{aligned} \quad (6.11)$$

This problem is not generally separable into different subproblems for each k . Hence, for general problems, the L -shaped method has an advantage. In some cases (notably simple recourse), $\Psi(\chi)$ is separable into components for each k , and (6.11) can again be divided into K independent subproblems. We discuss this possibility further in Section 5.7.

To see how the generalized programming form of inner linearization can be applied to a stochastic program, we again consider Example 2 from Section 5.1.

Iteration 1:

Suppose we start with an initial solution of $\chi^1 = 1$ and $\Psi(\chi^1) = \frac{7}{3}$ in (6.6), which then takes the form:

$$\min z^v = 0x_1 + \lambda_1 \Psi(\chi^1) \quad (6.12)$$

$$\text{s. t.} \quad x_1 + x_2 = 10, \quad (6.13)$$

$$x_1 - 0 \cdot \lambda_1 = 0, \quad (6.14)$$

$$\lambda_1 = 1, \quad (6.15)$$

$$x_1, x_2, \lambda_1 \geq 0,$$

which has an optimal solution $(x_1^1, x_2^1, \lambda^1) = (0, 10, 1)$ with dual multipliers $(\sigma^1, \pi^1, \rho^1) = (0, 0, \frac{7}{3})$.

Next, for Step 2, the solution is to find the minimum value of (6.10) or $\Psi(\chi) - \rho^1$ over χ , which is achieved at $\chi^2 = 2$ with $\Psi(\chi^2) = 1$. Since $\Psi(\chi^2) - \rho^1 = 1 - \frac{7}{3} = -\frac{4}{3} < 0$, the algorithm returns to Step 1 with $v = 2$.

Iteration 2:

The solution of (6.6) now is $(x_1^2, x_2^2, \lambda_1^2, \lambda_2^2) = (1, 9, 0, 1)$ with dual multipliers $(\sigma^2, \pi^2, \rho^2) = (0, 0, 1)$. In Step 2, the minimum value for (6.10) occurs at $\chi^3 = \chi^2 = 1$ and the objective value $\Psi(\chi^3) - \rho^2 = 0$, the termination condition.

The steps of this inner linearization algorithm can be viewed as taking the convex hull of an increasing numbers of extreme points of the epigraph of the recourse function. This can be seen for Example 2 in Figure 3. The solution starts at the point on the function corresponding to $x(\chi) = 0$ and then moves directly to including the point on the epigraph at $x = \chi = 1$, where no further descent is possible. The algorithm terminates virtually immediately for this example because the best candidate χ directly yields an overall optimal solution. This circumstance of course does not always occur, but the algorithm may be quite efficient when T is fixed and the subproblems (6.10) can be solved efficiently.

To show that the generalized programming method also converges finitely, we wish to demonstrate the property of generating extreme points on the epigraph of Ψ by showing that an extreme solution in (6.11) is an extreme value of linear regions of $\Psi(\chi)$. We do this for extreme points in the following proposition.

Proposition 18. *Every optimal extreme point, $(\bar{y}_1, \dots, \bar{y}_K, \bar{\chi})$, of the feasible region in (6.11) corresponds to an extreme point $\bar{\chi}$ of $\{\chi \mid \Psi(\chi) = \bar{\pi}^T \chi + \theta\}$, where $\bar{\pi} = \sum_{k=1}^K \bar{\pi}_k$, and each $\bar{\pi}_k$ is an extreme point of $\{\pi_k \mid \pi_k^T W \leq q_k^T\}$.*

Proof: Suppose $(\bar{y}_1, \dots, \bar{y}_K, \bar{\chi})$ is an optimal extreme point in (6.11). In this case, we must have $q_i^T \bar{y}_i \leq q_i^T y_i$ for all $W y_i = \xi_i - \bar{\chi}$. We must also have that \bar{y}_i is an extreme point of $\{y_i \mid W y_i = \xi_i - \bar{\chi}, y_i \geq 0\}$ because, otherwise, we could take $\bar{y}_i = (1/2)(y_i^1 + y_i^2)$ for distinct feasible y_i^1 and y_i^2 . So, \bar{y}_k has a complementary dual solution, $\bar{\pi}_k$, that is an extreme point of $\{\pi_k \mid \pi_k^T W \leq q_k^T\}$ and such that $(q_k^T - \bar{\pi}_k^T W)\bar{y}_k = 0$.

Now, suppose $\bar{\chi}$ is not an extreme point of the linearity region where $\Psi(\chi) = \pi^T \chi + \theta$ for $\theta = \Psi(\bar{\chi}) - \bar{\pi}^T \chi$ with $\bar{\pi} = \sum_{k=1}^K \bar{\pi}_k$. In this case, there exists χ^1 and χ^2 such that $\bar{\chi} = \lambda \chi^1 + (1-\lambda) \chi^2$ where $0 < \lambda < 1$, for $\Psi(\chi^1) = \bar{\pi}^T \chi^1 + \theta$ and $\Psi(\chi^2) = \bar{\pi}^T \chi^2 + \theta$. We also have that $\Psi(\chi^j) = \sum_{k=1}^K q_k^T y_k^j$, where $q_k^T y_k^j = \bar{\pi}_k^T (h_k - \chi^j)$ for $j = 1, 2$, because, by $\bar{\pi}_k^T$ feasible in the k -th recourse problem, the only other possibility is $q_k^T y_k^j > \bar{\pi}_k^T (\xi - \chi^j)$, which would imply $\Psi(\chi^j) > \bar{\pi}^T \chi^j + \theta$. This also implies that

$$(\bar{\pi}_k^T W - q_k^T)(\lambda y_k^1 + (1-\lambda)y_k^2) = 0, \quad (6.16)$$

which implies that $\lambda y_k^1 + (1-\lambda)y_k^2 = \bar{y}_k$ because \bar{y}_k is an extreme point of the feasible region in recourse problem k . In this case, $(\bar{y}_1, \dots, \bar{y}_K, \bar{\chi}) = \lambda(y_1^1, \dots, \bar{y}_K^1, \chi^1) + (1-\lambda)(y_1^2, \dots, \bar{y}_K^2, \chi^2)$, with both terms feasible in (6.11). This contradicts that $(\bar{y}_1, \dots, \bar{y}_K, \bar{\chi})$ is an extreme point. \square

A similar argument shows that any extreme ray found in solving (6.11) is an extreme ray of a region of linearity of $\Psi(\chi)$ (Exercise 3). Now, we can state the generalized programming finite convergence result.

Theorem 19. *The generalized programming applied to problem (6.5) with subproblem (6.11) solves (6.5) in a finite number of steps.*

Proof: At each solution of (6.11), a new linear region extreme value is generated. First for a new extreme ray, we must have $\Psi_0^+(\zeta^{r+1}) + (\pi^v)^T(\zeta^{r+1}) < 0$, while, for $1 \leq i \leq s$, $\Psi_0^+(\zeta^i) \geq -(\pi^v)^T \zeta^i$. For an extreme point, we only add that point if $\Psi(\chi^{s+1}) + (\pi^v)^T \chi^{s+1} - \rho^v < 0$, while, for $1 \leq i \leq s$, $\Psi(\chi^s) + (\pi^v)^T \chi^s - \rho^v \geq 0$. Because the number of such regions is finite and each has a finite number of extreme points and rays, the algorithm converges finitely.

The solution found solves (6.5) because if we reach the termination condition, then

$$\begin{aligned} (\sigma^v)^T b + \rho^v &\leq (\sigma^v)^T b + \Psi(\chi) + (\pi^v)^T \chi \\ &\leq (\sigma^v)^T A + (\pi^v)^T T x + \Psi(\chi), \quad (x, \chi) \text{ feasible in (6.5)}, \\ &\leq c^T x + \Psi(\chi), \end{aligned} \quad (6.17)$$

for all (x, χ) feasible in (6.5). \square

As with the L -shaped method, we can also modify the generalized linear programming approach to consider only active columns so that s and t can be bounded again by m_2 . Of course, this approach's greatest potential is in simple recourse problems as we mentioned earlier. It may also be advantageous if an algorithm can take advantage of the special matrix structure in (6.11). The most direct approach in this case is to construct a working basis and to try to perform most linear transformations with submatrices chosen from W . In this case, the procedure becomes quite similar to the procedures for directly attacking (3.1.2) that are given in the next section.

The generalized programming approach is also useful in considering the stochastic program as a procedure for combining *tenders* χ_i (see Nazareth and Wets [1986]) bid from the subproblems. In this case, the method may converge most quickly if the initial set of tenders is chosen well. A method for choosing such an initial set of tenders appears in Birge and Wets [1984]. This view of stochastic programs can also be quite useful for stochastic integer programs and is used to obtain efficiencies in branch-and-bound algorithms as discussed in Section 7.3.

Exercises

1. Suppose Problem (6.4) is infeasible for some k . What can be said about the original two-stage stochastic linear program? Find examples for these possible situations.
2. Prove directly that the inner linearization method converges to an optimal solution to the two-stage stochastic linear program (3.1.2).
3. Show that any extreme descending ray in (6.11) corresponds to an extreme ray of a linear piece of $\Psi(\chi)$.
4. Describe the steps of the generalized programming method for a modification of Example 2 in which the first period costs are δx , where $\delta = \{-2, -0.5, 0.5, 1, 2\}$. What differs in the path of the algorithm as δ changes?

5.7 Simple and Network Recourse Problems

In many stochastic programming problems, special structure provides additional computational advantages. The most common structures that allow for further efficiencies are simple recourse and network problems. The key features of these problems are separability of any nonlinear objective terms and efficient matrix computations.

Separability is the key to simple recourse computations. In Section 3.1 and Section 5.6, we described how these problems involve a recourse function that separates into components for each random variable. With simple recourse, the stochastic program in (6.5) can then be written with a separable recourse function as:

$$\begin{aligned} \min z &= c^T x + \sum_{i=1}^{m_2} \Psi_i(\chi_i) \\ \text{s. t.} \quad Ax &= b, \\ Tx - \chi &= 0, \\ x &\geq 0, \end{aligned} \tag{7.1}$$

where $\Psi_i(\chi_i) = \int_{h_i \leq \chi_i} q^- (\chi_i - h_i) dF(h_i) + \int_{h_i > \chi_i} q^+ (h_i - \chi_i) dF(h_i)$. Using this form of the objective in χ , we can substitute in (3.1.9) to obtain:

$$\Psi_i(\chi_i) = q_i^+ \bar{h}_i - (q_i^+ - q_i F_i(\chi_i)) \chi_i - q_i \int_{h_i \leq \chi_i} h_i dF(h_i), \quad (7.2)$$

where $\bar{h}_i = E[\mathbf{h}_i]$.

The separable objective terms in (7.1) offer advantages for computation. In general, we can use nonlinear programming techniques that apply even when the random variables are continuous. Linear programming-based procedures apply as well when the random variables have a finite number of values. In this section, we will first show how to use linear programming structure, assuming that each \mathbf{h}_i takes on the values, $h_{i,j}, j = 1, \dots, K_i$ with probabilities $p_{i,j}$. We then consider methods for general nonlinear problems.

Wets [1983a] gave the basic framework for computation of finitely distributed simple recourse problems as a linear program with upper bounded variables. The idea is to split χ_i into values corresponding to each interval, $[h_{i,j}, h_{i,j+1}]$, so that

$$\chi_i = \sum_{j=0}^{K_i} \chi_{i,j}, \quad \chi_{i,0} \leq h_{i,1}, \quad 0 \leq \chi_{i,j} \leq h_{i,j+1} - h_{i,j}, \quad 0 \leq \chi_{i,K_i}. \quad (7.3)$$

The objective coefficients correspond to the slope of $\Psi(\chi_i)$ in each of these intervals. They are:

$$d_{i,0} = -q_i^+, \quad d_{i,j} = -q_i^+ + q_i \left(\sum_{l=1}^j p_{i,l} \right), \quad j = 1, \dots, K_i. \quad (7.4)$$

The piecewise linear program with these objective coefficients and variables is:

$$\begin{aligned} \min z &= c^T x + \sum_{i=1}^{m_2} \left(\left(\sum_{j=0}^{K_i} d_{i,j} \chi_{i,j} \right) + q_i^+ \bar{h}_i \right) \\ \text{s. t.} \quad Ax &= b, \\ Tx - \chi &= 0, \\ x &\geq 0 \text{ and (7.3).} \end{aligned} \quad (7.5)$$

The equivalence of (7.1) and (7.5) is given in the following theorem.

Theorem 20. *Problems (7.1) and (7.5) have the same optimal values and sets of optimal solutions, (x^*, χ^*) .*

Proof: We first show any solution $(x, \chi_1, \dots, \chi_{m_2})$ to (7.1) corresponds to a solution $(x, \chi_1, \dots, \chi_{m_2}, \chi_{1,1}, \dots, \chi_{m_2, K_{m_2}})$ to (7.5) with the same objective value. We then also show the reverse to complete the proof. Suppose (x, χ) feasible in (7.1). If $h_{i,j} \leq \chi_i < h_{i,j+1}$ for some $1 \leq j \leq K_i$, then let $\chi_{i,0} = h_{i,1}$, $\chi_{i,l} = h_{i,l+1} - h_{i,l}$,

$1 \leq l \leq j-1$, $\chi_{i,j} = \chi_i - h_{i,j}$ and $\chi_{i,l} = 0$, $l \geq j+1$. If $\chi_i < h_{i,0}$, then let $\chi_{i,0} = \chi_i$, $\chi_{i,l} = 0$, $l \geq 1$. In this way, we satisfy (7.3).

If $\chi_i \geq h_{i,1}$, the variable i objective term in (7.5) with these values is then

$$\begin{aligned}
& q_i^+ \bar{h}_i - q_i^+ \left(h_{i,1} + \sum_{l=1}^{j-1} (h_{i,l+1} - h_{i,l}) + (\chi_i - h_{i,j}) \right) \\
& + q_i \left(\sum_{l=1}^{j-1} \left[\left(\sum_{k=1}^l p_{i,k} \right) (h_{i,l+1} - h_{i,l}) \right] + \sum_{k=1}^j p_{i,k} (\chi_i - h_{i,j}) \right) \\
& = q_i^+ \bar{h}_i - q_i^+ \chi_i + q_i \left(\sum_{k=1}^{j-1} p_{i,k} \left[\sum_{l=k}^{j-1} (h_{i,l+1} - h_{i,l}) - h_{i,j} \right] \right. \\
& \quad \left. - p_{i,j} h_{i,j} + \sum_{k=1}^j p_{i,k} \chi_i \right) \\
& = q_i^+ \bar{h}_i - q_i^+ \chi_i - q_i \left(\sum_{k=1}^j p_{i,k} h_{i,k} \right) + q_i \left(\sum_{k=1}^j p_{i,k} \right) \chi_i \\
& = q_i^+ \bar{h}_i - q_i^+ \chi_i - q_i \int_{h_i \leq \chi_i} h_i dF(h_i) + q_i F_i(\chi_i) \chi_i \\
& = \Psi_i(\chi_i), \tag{7.6}
\end{aligned}$$

where the last equality follows from substitution in (7.2).

If $\chi_i < h_{i,1}$, then the objective term is $q_i^+ \bar{h}_i - q_i^+ \chi_i$ which again agrees with $\Psi_i(\chi_i)$ from (7.2). Hence, any feasible (x, χ) in (7.1) corresponds to a feasible (x, χ) (where χ is extended into the components for each interval) in (7.5).

Suppose now that some (x^*, χ^*) is optimal in (7.5). Because each $q_i > 0$ and $p_{i,j} > 0$, for $h_{i,j} \leq \chi_i^* < h_{i,j+1}$ for some $1 \leq j \leq K_i$, we must have $\chi_{i,0}^* = h_{i,1}$, $\chi_{i,l}^* = h_{i,l+1} - h_{i,l}$, $1 \leq l \leq j-1$, $\chi_{i,j}^* = \chi_i^* - h_{i,j}$ and $\chi_{i,l}^* = 0$, $l \geq j+1$. If not, then $\chi_{i,l}^* < h_{i,l+1} - h_{i,l} - \delta$ for some $l \leq j-1$ and $\chi_{i,\bar{l}}^* > \delta > 0$ for some $\bar{l} \geq j+1$. A feasible change of increasing $\chi_{i,l}^*$ by δ and decreasing $\chi_{i,\bar{l}}^*$ by δ yields an objective decrease of $\delta q_i \sum_{s=\bar{l}+1}^{\bar{l}} p_{i,s}$ and would contradict optimality. Hence, we must have that the i th objective term in (7.5) is again $\Psi_i(\chi_i^*)$. Similarly, this must be true if $\chi_i^* < h_{i,1}$. Therefore, any optimal solution in (7.1) corresponds to a feasible solution with the same objective value in (7.5), and any optimal solution in (7.5) corresponds to a feasible solution with the same objective value in (7.1). Their optima must then correspond. \square

This formulation as an upper bounded variable linear program can lead to significant computational efficiencies. An implementation in Kallberg, White, and Ziembka [1982] uses this approach in a short-term financial planning model with 12 random variables with three realizations, each corresponding to uncertain cash requirements and liquidation costs. They solve the stochastic model with problem (7.5) in approximately 1.5 times the effort to solve the corresponding mean value linear program

with expected values substituted for all random variables. This result suggests that stochastic programs with simple recourse can be solved in a time of about the same order of magnitude as a deterministic linear program ignoring randomness.

Further computational advantages for these problems are possible by treating the special structure of the $\chi_{i,j}$ variables as χ_i variables with piecewise, linear convex objective terms. Fourer [1985, 1988] presents an efficient simplex method approach for these problems. This implementation lends further support to the similar mean value problem–stochastic program order of magnitude claim.

Decomposition methods can also be applied to the simple recourse problem with finite distributions, although solution times better than the mean-value linear programming solution would generally be difficult to obtain. As mentioned in Section 5.1d., the multicut approach offers some advantage for the L -shaped algorithm (in terms of major iterations), but solution times are generally at best comparable with the mean-value linear program time.

For generalized programming, because $\Psi(\chi) = \sum_{i=1}^{m_2} \Psi_i(\chi_i)$ and each $\Psi_i(\chi_i)$ is easily evaluated, the subproblem in (6.10) is equivalent to finding χ_i^v such that

$$-\pi_i^v \in \partial\Psi_i(\chi_i^v). \quad (7.7)$$

From (7.4) and the argument in Proposition 5.1, $\partial\Psi_i(\chi_i) = \{d_{i,j}\}$ for $h_{i,j} < \chi_i < h_{i,j+1}$ and $\partial\Psi_i(\chi_i) = [d_{i,j-1}, d_{i,j}]$ for $h_{i,j} = \chi_i$. Thus, we can choose $\chi_i^v = h_{i,j}$ for $d_{i,j-1} \leq -\pi_i^v \leq d_{i,j}$, $j = 1, \dots, K_i$. If $\pi_i^v < -q_i^+$, then the value in (6.10) is unbounded. The algorithm chooses $\zeta_i^{s+1} = -1$, and $\Psi_{0,i}^+(-1) = q_i^+$. In this way, generalized programming can be implemented easily, but would appear similar to the piecewise linear approach.

In network problems, the simple recourse formulation can be even more efficiently solved. Suppose, for example, that the random variables \mathbf{h}_i correspond to random demands at m_2 destinations, that the variables x_{st} are flows from s to t , $Ax = b$ corresponds to the network constraints for all source nodes, transshipment nodes, and destinations with known demands, and that Tx represents all the flows entering the destinations with random demand. By adding the constraint,

$$\sum_{i=1}^{m_2} \left(\sum_{j=1}^{l_i} \chi_{i,j} \right) - \sum_{\text{sources } s} \sum_t x_{st} = - \sum_{\substack{\text{known demand} \\ \text{destinations } r}} \text{demand}(r), \quad (7.8)$$

every variable in (7.5) corresponds to a flow so that (7.5) becomes a network linear program. Hence, efficient network codes can be applied directly to (7.5) in this case.

When T has gains and losses, (7.5) is a generalized network. This problem was one of the first types of practical stochastic linear programs solved when Ferguson and Dantzig [1956] used the generalized network form to give an efficient procedure for allocating aircraft to routes (fleet assignment). We describe this problem to show the possibilities inherent in the stochastic program structure.

The problem includes m_1 aircraft and m_2 routes. The decision variables are x_{sr} aircraft s allocated to route r . The number of aircraft s available is b_s , the passenger capacity of aircraft s on route r is t_{sr} , and the uncertain passenger

demand is \mathbf{h}_r . Hence, the i th row of $Ax = b$ is $\sum_{r=1}^{m_2} x_{ir} = b_i$. The j -th row of $Tx - \chi = 0$ is $\sum_{s=1}^{m_1} t_{sj} x_{sj} - \chi_j = 0$.

The key observation about this problem is that the basis corresponds to a pseudo-rooted spanning forest (see, e.g., Bazaraa, Jarvis, and Sherali [1990]). For this problem, the simplex steps solve with trees and one-trees in an efficient manner. For example, suppose $m_1 = 3$, $m_2 = 3$, $b = (2, 2, 2)$, $t_1 = (200, 100, 300)$, $t_2 = (300, 100, 200)$, and $t_3 = (400, 100, 150)$, $p_{i,j} = 0.5$, and $h_{1,1} = 500$, $h_{1,2} = 700$, $h_{2,1} = 200$, $h_{2,2} = 400$, $h_{3,1} = 200$, $h_{3,2} = 400$. A basic solution is $x_{1,1} = 1$, $x_{1,2} = 1$, $x_{2,1} = 1$, $x_{2,2} = 1$, $x_{3,3} = 2$, and $\chi_{3,1} = 100$ with all other variables nonbasic. This basis is illustrated in Figure 5. The forest consists of a cycle and a subtree. Exercises 1, 2, and 3 explore this example in more detail.

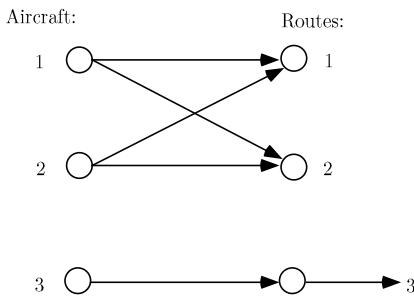


Fig. 5 Graph of basic arcs for aircraft-route assignment example.

For general network problems, Sun, Qi and Tsai [1993] describe a piecewise linear network method that allows the use of network methods and does not require adding the additional arcs that correspond to the $\chi_{i,j}$ values. Other generalizations for network structured problems allow continuous distributions and apply directly to the nonlinear problem. We discuss these methods in more detail in the next chapter.

The methods all apply to simple recourse problems in which the first-stage variables represent a network. Another class of problems includes network constraints in the second (and following) stages. These problems are called *network recourse* problems. In this case, some computational advantages are again possible.

Most computational experience with solving these problems directly has been with the *L*-shaped method. The efficiencies occur in constructing feasibility constraints, in generating facets of the polyhedral convex recourse function, and in solving multiple recourse problems using small Schur complement updates of a network basis. These procedures are described in Wallace [1986b]. Other methods for network recourse problems involve nonlinear programming-based procedures.

We suppose the simple recourse problem structure in (7.1). As noted earlier, the most direct methods for solving (7.1) use standard nonlinear programming techniques. We briefly describe some of the alternatives here. The most common procedures applied here are single-point linearization approaches, such as the

Frank-Wolfe method, multiple-point linearization, such as generalized linear programming as in Section 5.6, and active set or reduced variable methods, similar to simplex method extensions. Other methods are described in Nazareth and Wets [1986].

The Frank-Wolfe method for simple recourse problems appears in Wets [1966] and Ziembra [1970]. The basic procedure is to approximate the objective using the gradient and to solve a linear program to find a search direction. The algorithm contains the following basic steps. We assume that each random variable \mathbf{h}_i has an absolutely continuous distribution function F_i so that each Ψ_i is differentiable. In this case, the gradient of $\Psi(Tx)$ is easily calculated as $\nabla\Psi(Tx) = (q^+ - q)^T(\bar{F})T$, where $\bar{F} = \text{diag}\{F_i(T_i x)\}$, the diagonal matrix of the probability that \mathbf{h}_i is below $T_i x$.

Frank-Wolfe Method for Simple Recourse Problems

Step 0. Suppose a feasible solution x^0 to (7.1). Let $v = 0$. Go to Step 1.

Step 1. Let \hat{x}^v solve:

$$\begin{aligned} \min z &= (c^T + (q^+ - q)^T(\bar{F}^v)T)x \\ \text{s. t. } Ax &= b, \\ x &\geq 0, \end{aligned} \tag{7.9}$$

where $\bar{F}^v = \text{diag}\{F_i(T_i x^v)\}$.

Step 2. Find x^{v+1} to minimize $c^T(x^v + \lambda(\hat{x}^v - x^v)) + \sum_{i=1}^{m_2} \Psi_i(T(x^v + \lambda(\hat{x}^v - x^v)))$ over $0 \leq \lambda \leq 1$. If $x^{v+1} = x^v$, stop with an optimal solution. Otherwise, let $v = v + 1$ and return to Step 1.

The basis for this approach is that x^* is optimal in (7.1) if and only if x^* solves (7.9) with $x^* = x^v$. If x^v is not a solution of (7.1), then $x^{v+1} \neq x^v$, and descent occurs along $\hat{x}^v - x^v$. Exercise 1 asks for the details of this convergence result.

The *L*-shaped method and generalized linear programming can be considered extensions of the linearization approach that use multiple points of linearization. We have already considered the *L*-shaped method in some detail in the previous chapter. For generalized programming, the key advantage is that $\Psi(\chi)$ is separable. Williams [1966] and Beale [1961] observed the advantage of this property and gave generalized programming procedures for specific problems. In the case of the general problem in (7.1), the master problem of (3.4.9)–(3.4.10) becomes

$$\min z^v = c^T x + \sum_{j=1}^{m_2} \left(\sum_{i=1}^{r_j} \mu_{ji} \Psi_{0j}^+(\zeta_{ji}) + \sum_{i=1}^{s_j} \lambda_{ji} \Psi_j(\chi_{ji}) \right) \tag{7.10}$$

$$\text{s. t. } Ax = b, \tag{7.11}$$

$$T_i x - \sum_{i=1}^{r_j} \mu_{ji} \zeta_{ji} - \sum_{i=1}^{s_j} \lambda_{ji} \chi_{ji} = 0, \quad j = 1, \dots, m_2, \tag{7.12}$$

$$\sum_{i=1}^{s_j} \lambda_{ji} = 1, \quad x, \mu_{ji} \geq 0, \quad i = 1, \dots, r_j; \quad \lambda_{ji} \geq 0, \quad i = 1, \dots, s_j, \quad j = 1, \dots, m_2,$$
(7.13)

where we can divide the components of χ in the constraints because of the separability.

We then have a subproblem of the form in (3.5.12) for each j :

$$\min_{\chi_j} \Psi_j(\chi_j) + \pi_j^v \chi_j - \rho_j^v. \quad (7.14)$$

We can create an entering column whenever any of the values in (7.14) is negative. If all are non-negative, then the algorithm again terminates with an optimal value.

Example 4

As an example of generalized programming applied to a simple recourse problem, suppose the following situation. We have \$400 to buy boxes of blueberries (\$5 per box) and cherries (\$7 per box) from a farmer. We take the berries to the town market where we hope to sell them (\$11 per blueberry box and \$15 per cherry box). Any unsold berries at the end of the market day can be sold to a local baker (\$3 per blueberry box and \$5 per cherry box).

The demand for berries is stochastic. We assume that blueberry demand during market hours is uniformly distributed between 10 and 30 boxes and that cherry demand is uniformly distributed between 20 and 40 boxes. In the simple recourse problem, the correlation between these demands does not affect the recourse function value; so, we only need this marginal information.

The initial decisions are x_1 , the number of boxes of blueberries to buy, and x_2 , the number of boxes of cherries to buy. The full problem is then to find x^* , χ^* to

$$\begin{aligned} \min z &= 2x_1 + 2x_2 + \Psi_1(\chi_1) + \Psi_2(\chi_2) \\ \text{s. t. } &5x_1 + 7x_2 \leq 400, \\ &x_1 - \chi_1 = 0, \\ &x_2 - \chi_2 = 0, \\ &x_1, x_2 \geq 0, \end{aligned} \quad (7.15)$$

where

$$\Psi_1(\chi_1) = \begin{cases} -8\chi_1 & \text{if } \chi_1 \leq 10, \\ \frac{1}{5}\chi_1^2 - 12\chi_1 + 20 & \text{if } 10 \leq \chi_1 \leq 30, \\ -160 & \text{if } \chi_1 \geq 30, \end{cases}$$

$$\nabla \Psi_1(\chi_1) = \begin{cases} -8 & \text{if } \chi_1 \leq 10, \\ \frac{2}{5}\chi_1 - 12 & \text{if } 10 \leq \chi_1 \leq 30, \\ 0 & \text{if } \chi_1 \geq 30, \end{cases}$$

$$\Psi_2(\chi_2) = \begin{cases} -10\chi_2 & \text{if } \chi_2 \leq 20, \\ \frac{1}{4}\chi_2^2 - 20\chi_2 + 100 & \text{if } 20 \leq \chi_2 \leq 40, \\ -300 & \text{if } \chi_2 \geq 40, \end{cases}$$

and

$$\nabla \Psi_2(\chi_2) = \begin{cases} -10 & \text{if } \chi_2 \leq 20, \\ \frac{1}{2}\chi_2 - 20 & \text{if } 20 \leq \chi_2 \leq 40, \\ 0 & \text{if } \chi_2 \geq 40. \end{cases}$$

The generalized programming method follows these iterations.

Iteration 0:

Step 0. We start with (7.10)–(7.13) with $v = r^j = s^j = 0$.

Step 1. The obvious solution is $x^0 = (0, 0)^T$ with multipliers, $\pi^0 = \rho^0 = (0, 0)^T$.

Step 2. Setting $\pi_i^0 = -\nabla \Psi_i(\chi_{11})$, we obtain $\chi_{11} = 30$ and $\chi_{21} = 40$ with $\Psi_1(\chi_{11}) = -160$ and $\Psi_2(\chi_{21}) = -300$ and clearly $\Psi_j(\chi_{j,s_j+1}) + \pi_j^v \chi_{j,s_j+1} - \rho_j^v < 0$ for each $j = 1, 2$. Now, $s_1 = s_2 = 1$, $v = 1$ and we repeat.

Iteration 1:

Step 1. We assume that we can dispose of berries (to avoid creating an infeasibility in (7.10)–(7.13)). The master problem then has the form:

$$\begin{aligned} \min z &= 2x_1 + 2x_2 - 160\lambda_{11} - 300\lambda_{21} \\ \text{s. t.} \quad 5x_1 + 7x_2 &\leq 400, \\ x_1 - 30\lambda_{11} &\geq 0, \\ x_2 - 40\lambda_{21} &\geq 0, \\ \lambda_{11} &= 1, \\ \lambda_{21} &= 1, \\ x_1, x_2, \lambda_{11}, \lambda_{21} &\geq 0. \end{aligned} \tag{7.16}$$

The solution is $z^1 = -300$, $x^1 = (24, 40)^T$, $\lambda_{11} = 0.8$, $\lambda_{21} = 1.0$, $\pi^1 = (5.333, 6.667)^T$ and $\rho^1 = (0, -33.333)^T$.

Step 2. Setting $\pi_i^0 = -\nabla \Psi_i(\chi_{11})$, we obtain $\chi_{12} = 16.667$ and $\chi_{22} = 26.667$ with $\Psi_1(\chi_{11}) = -124.4$ and $\Psi_2(\chi_{22}) = -255.55$. Again, $\Psi_j(\chi_{j,s_j+1}) + \pi_j^v \chi_{j,s_j+1} - \rho_j^v < 0$ for each $j = 1, 2$ with $\Psi(\chi_{12}) + \pi_1^1 \chi_{12} - \rho_1^1 = -35.5$ and $\Psi(\chi_{22}) + \pi_2^1 \chi_{22} - \rho_2^1 = -44.4$. Now, $s_1 = s_2 = 2$, $v = 2$.

Iteration 2:

Step 1. The new master problem is:

$$\begin{aligned} \min z &= 2x_1 + 2x_2 - 160\lambda_{11} - 124.4\lambda_{12} - 300\lambda_{21} - 255.55\lambda_{22} \\ \text{s. t.} \quad & 5x_1 + 7x_2 \leq 400, \\ & x_1 - 30\lambda_{11} - 16.667\lambda_{12} \geq 0, \\ & x_2 - 40\lambda_{21} - 26.667\lambda_{22} \geq 0, \\ & \lambda_{11} + \lambda_{12} = 1, \\ & \lambda_{21} + \lambda_{22} = 1, \\ & x_1, x_2, \lambda_{11}, \lambda_{12}, \lambda_{21}, \lambda_{22} \geq 0. \end{aligned} \tag{7.17}$$

The solution is $z^2 = -316.0$, $x^2 = (24, 40)^T$, $\lambda_{11}^2 = 0.55$, $\lambda_{12}^2 = 0.45$, $\lambda_{21}^2 = 1.0$, $\pi^2 = (2.667, 2.934)^T$ and $\rho^2 = (-80.0, -182.6)^T$.

Step 2. Setting $\pi_i^2 = -\nabla\Psi_i(\chi_{i,s_i+1})$, we obtain $\chi_{13} = 23.33$ and $\chi_{23} = 34.13$ with $\Psi_1(\chi_{13}) = -151.1$ and $\Psi_2(\chi_{23}) = -291.4$. Here, $\Psi_1(\chi_{13}) + \pi_1^2 \chi_{13} - \rho_1^2 = -8.88$ and $\Psi_2(\chi_{23}) + \pi_2^2 \chi_{23} - \rho_2^2 = -8.61$. Now, $s_1 = s_2 = 3$, $v = 3$.

Iteration 3:

Step 1. The new master problem is:

$$\begin{aligned} \min z &= 2x_1 + 2x_2 - 160\lambda_{11} - 124.4\lambda_{12} - 151.1\lambda_{13} \\ & \quad - 300\lambda_{21} - 255.55\lambda_{22} - 291.4\lambda_{23} \\ \text{s. t.} \quad & 5x_1 + 7x_2 \leq 400, \\ & x_1 - 30\lambda_{11} - 16.667\lambda_{12} - 23.333\lambda_{13} \geq 0, \\ & x_2 - 40\lambda_{21} - 26.667\lambda_{22} - 34.133\lambda_{23} \geq 0, \\ & \lambda_{11} + \lambda_{12} + \lambda_{13} = 1, \\ & \lambda_{21} + \lambda_{22} + \lambda_{23} = 1, \\ & x_1, x_2, \lambda_{ij} \geq 0. \end{aligned} \tag{7.18}$$

The solution is $z^3 = -327.57$, $x^3 = (23.333, 34.133)^T$, $\lambda_{13}^3 = 1.00$, $\lambda_{23}^3 = 1.0$, $\pi^3 = (2.0, 2.0)^T$ and $\rho^3 = (-104.44, -223.13)^T$.

Step 2. Setting $\pi_i^3 = -\nabla\Psi_i(\chi_{i,s_i+1})$, we obtain $\chi_{14} = 25$ and $\chi_{24} = 36$ with $\Psi_1(\chi_{14}) = -155$ and $\Psi_2(\chi_{24}) = -296$. Here, $\Psi_1(\chi_{14}) + \pi_1^3 \chi_{14} - \rho_1^3 = -0.56$ and $\Psi_2(\chi_{24}) + \pi_2^3 \chi_{24} - \rho_2^3 = -0.87$. Now, $s_1 = s_2 = 4$, $v = 3$.

Iteration 4:

Step 1. We add λ_{14} and λ_{24} with their objective and constraint entries to (7.18) to obtain the same form of the master problem. The solution is now $z^4 = -329$, $x^4 = (25, 36)^T$, $\lambda_{14}^4 = 1.00$, $\lambda_{24}^4 = 1.0$, $\pi^4 = (2.0, 2.0)^T$ and $\rho^4 = (-105, -224)^T$.

Step 2. Because $\pi^4 = \pi^3$, we obtain $\chi_{i5} = \chi_{i4}$, and $\Psi_i(\chi_{i5}) + \pi_i^4 \chi_{i5} - \rho_i^4 = 0$ for $i = 1, 2$. Hence, no columns can be added. We stop with the optimal solution, $x^* = (25, 36)^T$ with objective value $z^* = 329$.

Notice that in this example the budget constraint is not binding. We only spend \$377 of the total possible, \$400. If we had solved this problem as separate news vendor problems in each type of berry, we would have obtained the same solution. In fact, this is one of the suggestions for initial tenders to start the generalized programming process (see Birge and Wets [1984] and Nazareth and Wets [1986]). In this case, we would terminate on the first step with this initial offer (just as in the case of Example 2 described in Section 5.6).

Notice also as in Section 5.6 that the algorithm appears to converge quite quickly here. In general, the retention of information about gradients at many points should improve convergence over techniques that use only local information. Second-order information is also valuable, assuming twice-differentiable functions. This is the motivation behind Beale's [1961] approach of quadratic approximation. This method is another form of the generalized programming approach for convex separable functions.

The other procedures specifically used on the simple recourse problem concern some form of active set or simplex based strategy. Wets [1966] and Ziembra [1970] give the basic reduced gradient or convex simplex method procedure. This method consists of computing a search direction corresponding to a change in the value of a nonbasic variable (assuming only basic variables change concomitantly). The basis is changed if the line search implies that basic variable becomes zero. Otherwise, the nonbasic variable's value is updated and other nonbasic variables are checked for possible descent.

A different approach is given by Qi [1986], who suggests alternating between the solution of a linear program with χ fixed and the solution of a reduced variable convex program. The linear program is to find

$$\begin{aligned} & \min_x c^T x + \Psi(\chi^v) \\ & \text{s. t. } Ax = b, \\ & \quad Tx = \chi^v, \\ & \quad x \geq 0, \end{aligned} \tag{7.19}$$

to obtain $x^{v+1} = (x_B^v, x_N^v)$, where $x_N^v = 0$. Then solve the reduced convex program:

$$\begin{aligned} & \min_{x, \chi} c^T x + \Psi(\chi) \\ & \text{s. t. } Ax = b, \quad Tx = \chi, \\ & \quad x_B \geq 0, \quad x_N = 0, \end{aligned} \tag{7.20}$$

to obtain $\hat{x}^{v+1}, \chi^{v+1}$. The algorithm is the following.

Alternating Algorithm for Simple Recourse Problems

Step 0. Let $v = 0$, choose a feasible solution x^0 to (7.20) and let χ^0 be part of a solution to (7.20) with N defined according to x^0 . Go to Step 1.

Step 1. Solve (7.19). Let $X^{v+1} = \{x \text{ optimal in (7.19)}\}$. Choose $x^{v+1} \in X^{v+1}$ such that $c^T x^{v+1} + \Psi(Tx^{v+1}) < c^T x^v + \Psi(Tx^v)$. If none exists, then stop. Otherwise, go to Step 2.

Step 2. Solve (7.20) with N defined for x^{v+1} to obtain χ^{v+1} . Let $v = v + 1$ and return to 1.

The algorithm converges to an optimal solution because x^{v+1} can always be found with $c^T x^{v+1} + \Psi(Tx^{v+1}) < c^T x^v + \Psi(Tx^v)$ whenever x^v is not optimal (Exercise 5). Of course, the algorithm's advantage is when the number of first-period variables n_1 is much greater than the number of second-period random variables m_2 , so that solving problem (7.20) provides a computational savings over solving (7.1) directly.

This algorithm (and indeed the convex simplex method) raises the possibility for multiple optima of the linear program (degeneracy). In this case, many solutions may be searched before improvement is found. In tests of partitioning in discretely distributed general stochastic linear programming problems (Birge [1985b]), this problem was found to overcome computational advantages of reducing the working problem size. The approach has, therefore, not been followed extensively in practice although it may, of course, offer efficient computation on some problems.

Other methods for simple recourse have built on the special structure. For transportation constraints, Qi [1985] gives a method based on using the forest structure of the basis to obtain a search direction and improved forest solution. This method only requires the solution of one-dimensional monotone equations apart from standard tree solutions. Piecewise linear techniques as in Sun, Qi, and Tsai [1990] can also be adapted here to general network structures and used in conjunction with Qi's forest procedure to produce a convergent algorithm.

Exercises

1. Show that any basis for the aircraft allocation problem consists of a collection of $m_1 + m_2$ basic variables that correspond to a collection of trees and one-trees.
2. Describe a procedure for finding the values of basic variables, multipliers, reduced costs, and entering and leaving basic variables for the structure in the aircraft allocation problem.
3. Solve the aircraft allocation problem using the procedure in (7.2) starting at the basis given with cost data corresponding to $c_{1.} = (300, 200, 100)$, $c_{2.} =$

$(400, 100, 300)$, $c_3 = (200, 100, 300)$, $q_i^+ = 25$, $q_i^- = 0$ for all i . You may find it useful to use the graph to compute the appropriate values.

4. Show that the Frank-Wolfe method for the simplex recourse problem converges to an optimal solution (assuming that one exists).
5. Solve the example in (7.15) using the L -shaped method.
6. Solve the example in (7.15) using the Frank-Wolfe method.
7. In the general stochastic linear programming model (with fixed T , (3.1.5)), show that solving (7.19) with $\chi^\nu = \chi^*$ yields an optimal solution x^* . Use this to show that there always exists a solution to (3.1.5) with at most $m_1 + m_2$ nonzero variables (Murty [1968]). What does this imply for retaining cuts in the L -shaped method?
8. Show that the alternating algorithm for simple recourse problems converges to an optimal solution assuming that the support of \mathbf{h} is compact. (Hint: From any x^ν , consider a path to x^* , use the convexity of Ψ , and consider the solution as x^ν is approached from x^* .)

5.8 Methods Based on the Stochastic Program Lagrangian

Again consider the general nonlinear stochastic program given in (3.5.1), which we repeat here without equality constraints to simplify the following discussion:

$$\begin{aligned} \inf z &= f^1(x) + \mathcal{Q}(x) \\ \text{s. t. } g_i^1(x) &\leq 0, \quad i = 1, \dots, m_1, \end{aligned} \tag{8.1}$$

where $\mathcal{Q}(x) = E_\omega[Q(x, \omega)]$ and

$$\begin{aligned} Q(x, \omega) &= \inf f^2(y(\omega), \omega) \\ \text{s. t. } g_i^2(x, y(\omega), \omega) &\leq 0, \quad i = 1, \dots, m_2, \end{aligned} \tag{8.2}$$

with the continuity assumptions mentioned in Section 3.5.

In general, we can consider a variety of approaches to (8.1) based on available nonlinear programming methods. For example, we may consider gradient projection, reduced gradient methods, and straightforward penalty-type procedures, but these methods all assume that gradients of \mathcal{Q} are available and relatively inexpensive to acquire. Clearly, this is not the case in stochastic programs because each evaluation may involve solving several problems (8.2). Lagrangian approaches have been proposed to avoid this problem.

The basic idea behind the Lagrangian approaches is to place the first- and second-stage links into the objective so that repeated subproblem optimizations are avoided in finding search directions. To see how this approach works, consider writing (8.1) in the following form:

$$\begin{aligned} \inf z &= f^1(x) + E_{\omega}[f^2(y(\omega), \omega)] \\ \text{s. t. } &g_i^1(x) \leq 0, i = 1, \dots, m_1, \\ &g_i^2(x, y(\omega), \omega) \leq 0, i = 1, \dots, m_2, \text{ a. s.} \end{aligned} \quad (8.3)$$

If we let (λ, π) be a multiplier vector associated with the constraints, then we can form a dual problem to (8.3) as:

$$\max_{\pi(\omega) \geq 0} w = \theta(\pi), \quad (8.4)$$

where

$$\begin{aligned} \theta(\pi) &= \inf_{x,y} z = f^1(x) + E_{\omega}[f^2(y(\omega), \omega)] \\ &\quad + E_{\omega}\left[\sum_{i=1}^{m_2} \pi(\omega)_i (g_i^2(x, y(\omega), \omega))\right] \\ \text{s. t. } &g_i^1(x) \leq 0, \quad i = 1, \dots, m_1. \end{aligned} \quad (8.5)$$

We show duality in the finite distribution case in the following theorem.

Theorem 21. Suppose the stochastic nonlinear program (8.1) with all functions convex has a finite optimal value and a point strictly satisfying all constraints, and suppose $\Omega = \{1, \dots, K\}$ with $P\{\omega = i\} = p_i$. Then $z \geq w$ for every feasible x, y_1, \dots, y_K in (8.1)–(8.2) and π_1, \dots, π_K feasible in (8.4), and their optimal values coincide, $z^* = w^*$.

Proof: From the general optimality conditions (see, e.g., Bazaraa and Shetty [1979, Theorem 6.2.1]), the result follows by noting that we may take x satisfying the first-period constraints as a general convex constraint set X so that only the second-period constraints are placed into the dual. We also divide any multipliers on the second-period constraints in (8.3) by p_i if they correspond to $\omega = i$. In this way, the expectation over ω in (8.5) is obtained. \square

Now, we can follow a dual ascent procedure in (8.4). This takes the form of a subgradient method. We note that

$$\partial \theta(\bar{\pi}) = \text{co} \{ (\zeta_1^1, \dots, \zeta_{m_2}^1)^T, \dots, (\zeta_1^K, \dots, \zeta_{m_2}^K)^T \}, \quad (8.6)$$

where again “co” denotes the convex hull,

$$\zeta_i^k = g_i^2(\bar{x}, \bar{y}_k, k), \quad (8.7)$$

and $(\bar{x}, \bar{y}_1, \dots, \bar{y}_K)$ solves the problem in (8.5) given $\pi = \bar{\pi}$. This again follows from standard theory as in, for example, Bazaraa and Shetty [1979, Theorem 6.3.7].

We can now describe a basic gradient method for the dual problem. For our purposes, we assume that (8.5) always has a unique solution.

Basic Lagrangian Dual Ascent Method

Step 0. Set $\pi^0 \geq 0$, $v = 0$ and go to Step 1.

Step 1. Given $\pi = \pi^v$ in (8.5), let the solution be $(x^v, y_1^v, \dots, y_K^v)$. Let $\hat{\pi}_i^k = 0$ if $\pi_i^{v,k} = 0$ and $g_i^2(x^v, y_k^v, k) \leq 0$, and $\hat{\pi}_i^k = g_i^2(x^v, y_k^v, k)$, otherwise. If $\hat{\pi}^k = 0$ for all k , stop.

Step 2. Let λ^v maximize $\theta(\pi^v + \lambda\hat{\pi})$ over $\pi^v + \lambda\hat{\pi} \geq 0, \lambda \geq 0$. Let $\pi^{v+1} = \pi^v + \lambda^v\hat{\pi}$, $v = v + 1$, and go to Step 1.

Assuming the unique solution property, this algorithm always produces an ascent direction in θ . The algorithm either converges finitely to an optimal solution or, assuming a bounded set of optima, produces an infinite sequence with all limit points optimal (see Exercise 1). For the case of multiple optima for (8.5), some nondifferentiable procedure must be used. In this case, one could consider finding the maximum norm subgradient to be assured of ascent or one could use various bundle-type methods (see Section 5.9).

The basic hope for computational efficiency in the dual ascent procedure is that the number of dual iterations is small compared to the number of function evaluations that might be required by directly attacking (8.1) and (8.2). Substantial time may be spent solving (8.2) but that should be somewhat easier than solving (8.1) because the linking constraints appear in the objective instead of as hard constraints. Overall, however, this type of procedure is generally slow due to our using only a single-point linearization of θ . This observation has led to other types of Lagrangian approaches to (8.1) that use more global or second-order information.

Rockafellar and Wets [1986] suggested one such procedure for a special case of (8.5) where $f^1(x) = c^T x + \frac{1}{2}x^T Cx$ and $y(\omega)$ can be eliminated so that the second and third objective terms become $\Phi(\pi, x)$ and the dual problem in (8.4) is then

$$\max_{\pi \geq 0} \inf_{\{x | g^1(x) \leq 0\}} [c^T x + \frac{1}{2}x^T Cx + \Phi(\pi, x)]. \quad (8.8)$$

Their approach is not to restrict the search to a single search direction but to allow optimization over a low dimensional set. Implementation of this method, called the Lagrangian finite-generation method for linear-quadratic stochastic programs, is described in King [1988a] and its application to solve practical water management problems concerning Lake Balaton in Hungary appears in Somlyódy and Wets [1988].

A similar method based on inner linearization approaches in nonlinear programming is restricted simplicial decomposition (Ventura and Hearn [1993]). This procedure replaces the line search in the Topkis-Veinott [1967] feasible direction method with a search over a simplex. The finite generation algorithm is analogously an enhancement over basic Lagrangian dual ascent methods that consider only gradient or subgradient steps. Both the finite-generation and restricted simplicial decomposition methods tend to avoid the zigzagging behavior that often occurs in methods based on single-point linearizations.

Another method for accelerating convergence is to enforce strictly convex terms in the objective. Rockafellar and Wets discussed methods for adding quadratic terms to the matrices C and $D(\omega)$ so that these matrices become positive definite. In this way, the finite generation method becomes a form of augmented Lagrangian procedure. We next discuss the basic premise behind these procedures.

In an augmented Lagrangian approach, one generally adds a penalty $r\|g_i^2(\bar{x}, \bar{y}_k, k)\|^2$ to $\theta(\pi)$ and performs the iterations including this term. The advantage (see the discussion in Dempster [1988]) is that Newton-type steps can be applied because we would obtain a nonsingular Hessian. The result should generally be that convergence becomes superlinear in terms of the dual objective without a significantly greater computational burden over the Lagrangian approach.

The computational experience reported by Dempster suggests that few dual iterations need be used but that a more effective alternative was to include explicit nonanticipative constraints as in (3.5.4) and to place these constraints into the objective instead of the full second-period constraints. In this way, θ becomes

$$\begin{aligned} \theta'(\rho) = \inf z = & f^1(x) + \sum_{k=1}^K p_k[f^2(y_k, k)] \\ & + \sum_{k=1}^K [\rho_k^T(x - x_k) + r/2\|x - x_k\|^2] \\ \text{s. t. } & g_i^1(x) \leq 0, \quad i = 1, \dots, m_1, \\ & g_i^2(x_k, y_k, k) \leq 0, \quad i = 1, \dots, m_2, \\ & \quad k = 1, \dots, K. \end{aligned} \tag{8.9}$$

Notice how in (8.9) the only links between the nonanticipative x decision and the scenario k decisions are in the $(x - x_k)$ objective terms. Dempster suggests solving this problem approximately on each dual iteration by iterating between searches in the x variables and search in the x_k, y_k variables. In this way, the augmented Lagrangian approach of solving (8.9) to find a dual ascent Newton-type direction achieves superlinear convergence in dual iterations. The only problem may come in the time to construct the search directions through solutions of (8.9).

This method also resembles the *progressive hedging algorithm* of Rockafellar and Wets [1991]. This method achieves a full separation of the individual scenario problems for each iteration and, therefore, has considerably less work in each iteration; however, the number of iterations as we shall see, may be greater. The method can offer many computational advantages, particularly for structured problems (see Mulvey and Vladimirou [1991a]). The key to this method's success is that individual subproblem structure is maintained throughout the algorithm. Related implementations by Nielsen and Zenios [1993a, 1993b] on parallel processors demonstrate possibilities for parallelism and the solution of large problems.

The basic progressive hedging method begins with a nonanticipative solution \hat{x}^v and a multiplier ρ^v . The nonanticipative (but not necessarily feasible) solution is used in place of x in (8.9). The first-period constraints are also split into each x_k .

In this way, we obtain a subproblem:

$$\begin{aligned} \inf z = & \sum_{k=1}^K p_k [f^1(x_k) + f^2(y_k, k) + \rho_k^{v,T}(x_k - \hat{x}^v) + r/2\|x_k - \hat{x}^v\|^2] \\ \text{s. t. } & g_i^1(x_k) \leq 0, \quad i = 1, \dots, m_1, \quad k = 1, \dots, K, \\ & g_i^2(x_k, y_k, k) \leq 0, \quad i = 1, \dots, m_2, \quad k = 1, \dots, K. \end{aligned} \quad (8.10)$$

Now (8.10) splits directly into subproblems for each k so these can be treated separately.

Supposing that (x_k^{v+1}, y_k^{v+1}) solves (8.10). We obtain a new nonanticipative decision by taking the expected value of x^{v+1} as \hat{x}^{v+1} and step in ρ by $\rho^{v+1} = \rho^v + (x^{v+1} - \hat{x}^{v+1})$.

The steps then are simply stated as follows.

Progressive Hedging Algorithm (PHA)

Step 0. Suppose some nonanticipative x^0 , some initial multiplier ρ^0 , and $r > 0$. Let $v = 0$. Go to Step 1.

Step 1. Let (x_k^{v+1}, y_k^{v+1}) for $k = 1, \dots, K$ solve (8.10). Let $\hat{x}^{v+1} = (\hat{x}^{v+1,1}, \dots, \hat{x}^{v+1,K})^T$ where $\hat{x}^{v+1,k} = \sum_{l=1}^K p_l x^{v+1,l}$ for all $k = 1, \dots, K$.

Step 2. Let $\rho^{v+1} = \rho^v + r(x^{v+1,k} - \hat{x}^{v+1})$. If $\hat{x}^{v+1} = \hat{x}^v$ and $\rho^{v+1} = \rho^v$, then, stop; \hat{x}^v and ρ^v are optimal. Otherwise, let $v = v + 1$ and go to Step 1.

The convergence of this method is based on Rockafellar's proximal point method [1976a]. The basis for this approach is not dual ascent but the contraction of the pair, $(\hat{x}^{v+1}, \rho^{v+1})$, about an optimal point. The key is that the algorithm mapping can be described as $(\Pi x^{v+1}, \rho^{v+1}/r) = (I - V)^{-1}(\Pi x^v, \rho^v/r)$, where V is a maximal monotone operator and Π is the diagonal matrix of probabilities corresponding to x^k and ρ^k , i.e, where $\Pi_{(k-1)n_1+i, (k-1)n_1+i} = p_k$ for $i = 1, \dots, n_1$ and $k = 1, \dots, K$.

To describe this approach we first define a maximal monotone operator at V (see Minty [1961] for more general details) such that for any pairs (w, z) where $z \in V(w)$ and (w', z') for $z' \in V(w')$, we have

$$(w - w')^T V(z - z') \geq 0. \quad (8.11)$$

The key point here is that if we have a Lagrangian function $l(x, y)$ that is convex in x and concave in y , then the subdifferential set of $l(x, y)$ at (\bar{x}, \bar{y}) defined by

$$\begin{aligned} \{(\zeta, \eta) \mid \zeta^T(x - \bar{x}) + l(\bar{x}, \bar{y}) \leq l(x, \bar{y}), \forall x; \\ \eta^T(y - \bar{y}) + l(\bar{x}, \bar{y}) \geq l(\bar{x}, y), \forall y\} \end{aligned} \quad (8.12)$$

yields a maximal monotone operator by

$$V(\bar{x}, \bar{y}) = \{(\zeta, \eta)\} \quad (8.13)$$

for $(\zeta, -\eta) \in \partial l(\bar{x}, \bar{y})$ (Exercise 3).

The second result that follows for maximal monotone operators is that a contraction mapping can be defined on it by taking $(I - V)^{-1}(x, y)$ to obtain (x', y') , or, equivalently, where $(x' - x, y' - y) \in V(x', y')$. The contraction result (Exercise 4) is that, if V is maximal monotone, then, for all $(x', y') = (I - (1/r)V)^{-1}(x, y)$ and $(\bar{x}', \bar{y}') = (I - V)^{-1}(\bar{x}, \bar{y})$,

$$\|(x' - \bar{x}', y' - \bar{y}')\|^2 \leq (x - \bar{x}, y - \bar{y})^T (x' - \bar{x}', y' - \bar{y}'). \quad (8.14)$$

These results then play the fundamental role in the following proof of convergence.

Theorem 22. *The progressive hedging algorithm, applied to (8.1) with the same conditions as in Theorem 14, converges to an optimal solution, x^*, ρ^* , (or terminates finitely with an optimal solution) and, at each iteration that does not terminate in Step 2,*

$$\|(\Pi\hat{x}^{v+1}, \rho^{v+1}/r) - (\Pi x^*, \rho^*/r)\| < \|(\Pi\hat{x}^v, \rho^v/r) - (\Pi x^*, \rho^*/r)\|. \quad (8.15)$$

Proof: As stated, the key is to find the associated Lagrangian and to show that the iterations follow the mapping as in (8.14). For the Lagrangian, define

$$\begin{aligned} l(\bar{x}, \bar{\rho}) &= \inf_x (1/r)z(x) + \bar{\rho}^T \Pi x \\ \text{s. t. } J\Pi x - \bar{x} &= 0, \end{aligned} \quad (8.16)$$

where $z(x)$ is defined as $\sum_{k=1}^K [f^1(x^k) + Q(x^k, k)]$ for feasible x^k and as $+\infty$ otherwise, Π is defined as the diagonal probability matrix, and J is the matrix corresponding to column sums, $J_{r,s}$ equal one if $r \pmod{n_1} = s \pmod{n_1}$ and zero otherwise. We want to show that

$$(\Pi(\hat{x}^v - \hat{x}^{v+1}), (\rho^v - \rho^{v+1})/r) \in \partial l(\Pi\hat{x}^{v+1}, \rho^{v+1}/r);$$

so, we can use the contraction property in (8.14) from the maximal monotone operator defined on $\partial l(\Pi\hat{x}^{v+1}, \rho^{v+1}/r)$.

Note that, for $\bar{x} = \Pi\hat{x}^v$ and $\bar{\rho} = \rho^v/r = \sum_{i=1}^V (x^i - \hat{x}^i)$, $\bar{x}^T \bar{\rho} = \hat{x}^{v,T} \Pi (\sum_{i=1}^V (x^i - \hat{x}^i)) = (x')^{v,T} J\Pi (\sum_{i=1}^V (x^i - \hat{x}^i))$ for $(x')^{v,T} = (1/K)\hat{x}^{v,T}$. Because $J\Pi x^i = \hat{x}^i$, we have $\bar{x}^T \bar{\rho} = 0$. We can thus add the term, $\bar{x}^T \bar{\rho}$ to the objective in (8.16) without changing the problem. We then obtain:

$$\eta \in \partial_{\bar{\rho}} l(\bar{x}, \bar{\rho}) \Leftrightarrow -J\Pi \bar{\rho} \in (1/r) \partial z(\Pi^{-1}(-\eta) + \bar{x}) + \pi^T J\Pi, \quad (8.17)$$

where $J\Pi(\Pi^{-1}(-\eta)) = \bar{x}$ and π is some multiplier. For $\partial_{\bar{x}} l(\bar{x}, \bar{\rho})$, $\zeta = -\pi^T J\Pi$, and some π ,

$$\zeta \in \partial_{\bar{x}} l(\bar{x}, \bar{\rho}) \Leftrightarrow \zeta - J\Pi \bar{\rho} \in (1/r) \partial Z(x'), \quad (8.18)$$

for some $J\Pi x' = \hat{x}$. We combine (8.17) and (8.18) to obtain that $(\zeta, \eta) \in \partial l(\bar{x}, \bar{\rho})$ if

$$\zeta - \Pi\bar{\rho} \in (1/r)\partial z(\Pi^{-1}(-\eta) + \bar{x}) . \quad (8.19)$$

We wish to show that

$$\Pi(\hat{x}^v - \hat{x}^{v+1}) - \Pi\rho^{v+1}/r \in (1/r)\partial z(\Pi^{-1}(\rho^{v+1} - \rho^v)/r + \hat{x}^{v+1}) . \quad (8.20)$$

From the algorithm,

$$-\Pi\rho^v \in \partial z(x^{v+1}) + r\Pi(x^{v+1} - \hat{x}^v) . \quad (8.21)$$

Substituting, $\rho^{v+1} = \rho^v + r(x^{v+1} - \hat{x}^{v+1})$, we obtain from (8.21),

$$-\Pi\rho^{v+1} + r\Pi(x^{v+1} - \hat{x}^{v+1}) \in \partial z(x^{v+1}) + r\Pi(x^{v+1} - \hat{x}^v) , \quad (8.22)$$

which, after eliminating $r\Pi x^{v+1}$ from both sides, coincides with (8.20).

By the nonexpansive property, there exists $(\Pi x^*, \rho^*/r)$, a fixed point of this mapping. By substituting into (8.14), with $(\Pi x^*, \rho^*/r) = (I - V)(\Pi x^*, \rho^*/r)$ and $(\Pi\hat{x}^{v+1}, \rho^{v+1}/r) = (I - V)(\Pi\hat{x}^v, \rho^v/r)$, we have (Exercise 5):

$$\|(\Pi\hat{x}^{v+1}, \rho^{v+1}/r) - (\Pi x^*, \rho^*/r)\| < \|(\Pi\hat{x}^v, \rho^{v+1}/r) - (\Pi x^*, \rho^*/r)\| . \quad (8.23)$$

Our result follows if (x^*, ρ^*) is indeed a solution of (8.1). Note that in this case, we must have $0 = x^{v+1} - \hat{x}^{v+1} = x^{v+1} - \hat{x}^v$; so, from (8.21), $-\Pi\rho^* \in \partial z(x^*)$. From Theorem 3.2.5, optimality in (8.1) is equivalent to $\rho^T \Pi \in \partial z(x^*)$ for some ρ , where $J\Pi\rho = 0$, which is true because $J\Pi(-\rho^*) = -\sum_v J\Pi(x^{v+1} - x^v) = 0$. Hence, we obtain optimality. The method converges as desired. \square

We note that Rockafellar and Wets obtained these results by defining an inner product as $\langle \rho, x \rangle = \rho^T \Pi x$ and using appropriate operations with this definition. They also show that, in the linear-quadratic case, the convergence to optimality is geometric.

Variants of this method are possible by considering other inner products and projection operators. For example, we can let \hat{x}^{v+1} be the standard orthogonal projection of x^{v+1} into the null space of $J\Pi$. This value is the simple average of x_k^{v+1} values, so that $\hat{x}_k^{v+1}(i) = (1/K) \sum_{k=1}^K x_k^{v+1}(i)$ for all $k = 1, \dots, K$. The multiplier update is then:

$$\rho^{v+1} = \rho^v + r\Pi^{-1}(x^{v+1} - \hat{x}^{v+1}) . \quad (8.24)$$

One can again obtain the maximal monotone operator property, and, observing that $Jx^{v+1} = J\hat{x}^{v+1}$, obtain $J\Pi\rho^* = 0$ and optimality.

Example 3

The algorithm's geometric convergence may require many iterations even on small problems as we show in the following small example. Suppose we can invest

\$10,000 in either of two investments, A or B. We would like a return of \$25,000, but the investments have different returns according to two future scenarios. In the first scenario, A returns just the initial investment while B returns 3 times the initial investment. In the second scenario, A returns 4 times the initial investment and B returns twice the initial investment. The two scenarios are considered equally likely. To reflect our goal of achieving \$25,000, we use an objective that squares any return less than \$25,000. The overall formulation is then:

$$\begin{aligned} \min z &= 0.5(y_1^2 + y_2^2) \\ \text{s. t. } &x_A + x_B \leq 10, \\ &x_A + 3x_B + y_1 \geq 25, \\ &4x_A + 2x_B + y_2 \geq 25, \\ &x_A, x_B, y_1, y_2 \geq 0. \end{aligned} \tag{8.25}$$

Clearly, this problem has an optimal solution at $x_A^* = 2.5$ and $x_B^* = 7.5$ with an objective value $z^* = 0$. A single iteration of Step 1 in the basic Lagrangian method is all that would be required to solve this problem for any positive π value. A single iteration is also all that would be necessary in the augmented Lagrangian problem in (8.9). The price for this efficiency is, however, the incorporation of all subproblems into a single master problem. Progressive hedging on the other hand maintains completely separate subproblems. We will follow the first two iterations of PHA for $r = 2$ here.

Iteration 0:

Step 0. Begin with a multiplier vector of $\rho^0 = 0$, and let $x_1^0 = (x_{1A}^0, x_{1B}^0) = (0, 10)^T$ and let $x_2^0 = (x_{2A}^0, x_{2B}^0) = (10, 0)^T$. The initial value of $\hat{x}^0 = (5, 5)^T$.

Step 1. We wish to solve:

$$\begin{aligned} \min(1/2)[y_1^2 + y_2^2 + (x_{1A}^1 - 5)^2 + (x_{1B}^1 - 5)^2 + (x_{2A}^1 - 5)^2 + (x_{2B}^1 - 5)^2] \\ \text{s. t. } &x_{1A}^1 + x_{1B}^1 \leq 10, \\ &x_{2A}^1 + x_{2B}^1 \leq 10, \\ &x_{1A}^1 + 3x_{1B}^1 - y_1 \geq 25, \\ &4x_{2A}^1 + 2x_{2B}^1 - y_2 \geq 25, \\ &x_{1A}^1, x_{1B}^1, x_{2A}^1, x_{2B}^1, y_1, y_2 \geq 0. \end{aligned} \tag{8.26}$$

This problem splits into separate subproblems for x_{1A}^1 , x_{1B}^1 , y_1 and x_{2A}^1 , x_{2B}^1 , y_2 , as mentioned earlier. For x_{1A}^1 , x_{1B}^1 , y_1 feasible in (8.26), the K-K-T conditions are that there exist $\lambda_1 \geq 0$, $\lambda_2 \geq 0$ such that

$$\begin{aligned} 2(x_{1A}^1 - 5) + \lambda_1 - \lambda_2 &\geq 0, \\ 2(x_{1B}^1 - 5) + \lambda_1 - 3\lambda_2 &\geq 0, \end{aligned}$$

$$\begin{aligned}
2y_1 + \lambda_2 &\geq 0, \\
(2(x_{1A}^1 - 5) + \lambda_1 - \lambda_2)x_{1A}^1 &= 0, \\
(2(x_{1B}^1 - 5) + \lambda_1 - 3\lambda_2)x_{1B}^1 &= 0, \\
(2y_1 + \lambda_2)y_1 &= 0, \\
(x_{1A}^1 + x_{1B}^1 - 10)\lambda_1 &= 0, \\
(x_{1A}^1 + 3x_{1B}^1 - y_1 - 25)\lambda_2 &= 0,
\end{aligned} \tag{8.27}$$

which has a solution of $(x_{1A}^1, x_{1B}^1, y_1) = (10/3, 20/3, 5/3)$ and $(\lambda_1, \lambda_2) = (20/3, 10/3)$. Similar conditions exist for the second subproblem, which has a solution $(x_{2A}^1, x_{2B}^1, y_2) = (5, 5, 0)$. We then let $(\hat{x}_{iA}^1, \hat{x}_{iB}^1) = (4\frac{1}{6}, 5\frac{5}{6})$ for $i = 1, 2$.

Step 2. The new multiplier is $\rho^1 = (\rho_{1A}^1, \rho_{1B}^1, \rho_{2A}^1, \rho_{2B}^1)^T = 2((10/3 - 25/6), (20/3 - 35/6), (5 - 25/6), (5 - 35/6))^T = (-5/3, 5/3, 5/3, -5/3)^T$.

Iteration 2:

Step 1. The first subproblem is now

$$\begin{aligned}
\min y_1^2 - (5/3)(x_{1A}^2 - 25/6) + (5/3)(x_{1B}^2 - 35/6) \\
+ (x_{1A}^2 - 25/6)^2 + (x_{1B}^2 - 35/6)^2 \\
\text{s. t. } x_{1A}^2 + x_{1B}^2 \leq 10, \\
x_{1A}^2 + 3x_{1B}^2 - y_1 \geq 25, \\
x_{1A}^2, x_{1B}^2, y_1 \geq 0,
\end{aligned} \tag{8.28}$$

which again has an optimal solution, $(x_{1A}^2, x_{1B}^2, y_1^2) = (10/3, 20/3, 5/3)$. Curiously, we also have the second subproblem solution of $(x_{2A}^2, x_{2B}^2, y_2^2) = (10/3, 20/3, 0)$. In this case, $(\hat{x}_{iA}^2, \hat{x}_{iB}^2) = (10/3, 20/3)$ for $i = 1, 2$.

Step 2. Because the subproblems returned the same solution, $\rho^2 = \rho^1$. We continue because the x values changed, even though we took no multiplier step.

The full iteration values are given in Table 1. Notice how the method achieves convergence in the x values before the ρ values have converged. Also, notice how the convergence appears to be geometric. This type of performance appears to be typical of PHA. It should be noted again, however, that the iterations are quite simple and that little overhead is required.

Exercises

1. Show that the basic dual ascent method converges to an optimal solution under the conditions given.

Table 1 PHA iterations for Example 3.

k	\hat{x}_A^k	\hat{x}_B^k	ρ_{1A}^k $= -\rho_{2A}^k$	ρ_{1B}^k $= -\rho_{2B}^k$	x_{1A}^k	x_{1B}^k	x_{2A}^k	x_{2B}^k
0	5.0	5.0	0.0	0.0	3.33	6.67	5.0	5.0
1	4.17	5.83	-1.67	1.67	3.33	6.67	3.33	6.67
2	3.33	6.67	-1.67	1.67	3.06	6.94	2.50	7.50
3	2.78	7.22	-1.11	1.11	2.78	7.22	2.41	7.59
4	2.59	7.41	-0.74	0.74	2.65	7.35	2.41	7.59
5	2.53	7.47	-0.49	0.49	2.59	7.41	2.43	7.57
6	2.50	7.50	-0.33	0.33	2.56	7.44	2.45	7.55
7	2.50	7.50	-0.22	0.22	2.54	7.46	2.46	7.54
8	2.50	7.50	-0.15	0.15	2.53	7.48	2.48	7.52
9	2.50	7.50	-0.10	0.10	2.52	7.48	2.48	7.52
10	2.50	7.50	-0.07	0.07	2.51	7.49	2.49	7.51
11	2.50	7.50	-0.04	0.04	2.51	7.49	2.49	7.51
12	2.50	7.50	-0.03	0.03	2.50	7.50	2.50	7.50

2. Show that (8.4) can be reduced to (8.8) when $g^2(y(\omega), \omega) = T(\omega)x + Wy(\omega) - h(\omega)$, $f^2(y(\omega), \omega) = q(\omega)^T y(\omega) + \frac{1}{2}y(\omega)^T D(\omega)y(\omega)$, and D is positive definite.
3. Show that V as defined in (8.13) is a maximal monotone operator.
4. Prove the contraction property in (8.14).
5. Use (8.14) to obtain (8.23).
6. Apply the dual ascent method and the augmented Lagrangian method with problem (8.9) to the example in (8.25). Start with zero multipliers (ρ), $\pi = 0$ or 1, and positive penalty r . Show that each obtains an optimal solution in at most one iteration.

5.9 Additional Methods and Complexity Results

In the previous sections, we considered cutting plane methods and Lagrangian methods for problems with discrete random variables and simple recourse-based techniques for problems with continuous random variables. Other nonlinear programming procedures can also be applied to stochastic programs, although these other procedures have not received as much attention in stochastic programming problems. A notable exception is Noël and Smeers' [1987] multistage combined inner linearization and augmented Lagrangian procedure, which we will describe in more detail in the next chapter.

A difficulty with discrete random variables is that Ψ or \mathcal{Q} generally loses differentiability. In this case, derivative-based methods cannot apply. As we saw,

the L -shaped method and other cutting plane approaches are a standard approach that requires only subgradient information. We also saw that augmented Lagrangian techniques can smooth nondifferentiable functions.

Explicit nondifferentiable methods include the nonmonotonic reduced subgradient procedure considered by Ermoliev [1983]. Another possibility is to use bundles of subgradients as in Lemaréchal [1978] and Kiwiel [1983]. Results by Plambeck et al. [1996], for example, show good performance for bundle methods in practical stochastic programs.

Nonsmooth generalizations of the Frank-Wolfe procedure are also possible. These and other options are described in detail in Demyanov and Vasiliev [1981]. With general continuous random variables or with large numbers of discrete random vector realizations, direct nonlinear programming procedures generally break down because of difficulties in evaluating function and derivative values. In these cases, one must rely on approximation. These approximations either take the form of bounds on the actual function values or are in some sense statistical estimates of the actual function values. We present these approaches in Chapters 8 to 10.

While models with discrete random variables inherit the complexity results of their deterministic equivalent forms with possible improvements due to problem structure as shown for interior point methods in Section 5.5, general distributions can present difficulties even in the two-stage case. For the common mean-variance objective, for example, the two-stage stochastic program is NP-hard (Ahmed [2006]). While exact solutions to general stochastic programs are difficult in general, bounds may be obtained efficiently using the methods in Chapter 8 and other approaches that can achieve a priori bounds on error in special cases. For example, Dye, Stougie, and Tomaszard [2003] consider a problem of a central resource serving facilities with random demands; Gupta, et al. [2007] provide bounds on the related stochastic Steiner tree problem to connect a source node to terminal nodes that are randomly revealed in the second period; Ravi and Sinha [2006] provide results for the stochastic shortest path version with generalizations to other combinatorial problems; and Flaxman, Frieze, and Krivelevich [2005] give a solution for a two-stage stochastic spanning tree problem, where instead of random demand, uncertainty is in the cost of edges which can be purchased for known costs in the first period and for random costs in the second period. Swamy and Shmoys [2006] provide a survey of these and other approaches including sampling methods which are discussed in Chapter 9.

Chapter 6

Multistage Stochastic Programs

As the Chapter 1 examples demonstrate, many operational and planning problems involve sequences of decisions over time. The decisions can respond to realizations of outcomes that are not known *a priori*. The resulting model for optimal decision making is then a multistage stochastic program. In Section 3.4, we gave some of the basic properties of multistage problems. In this chapter, we explore the variety of solution procedures that have been proposed specifically for multistage stochastic programs.

In general, the methods for two-stage problems generalize to the multistage case but include additional complications. Because of these difficulties, we will describe only those methods that have shown some success for obtaining fully-optimal solutions to problems in high dimension with a given finite set of possible scenarios. As in previous chapters, the focus here is also on problems with time-separable objectives (in contrast to the risk-sensitive utility in (10.7) of Chapter 2).

As stated in Section 3.4, the multistage stochastic linear program with a finite number of possible future scenarios has a deterministic equivalent linear program. However, as the graph in Figure 5 of Chapter 3 begins to suggest, the structure of this problem is somewhat more complex than that of the two-stage problem. The extensive form is not readily accessible to manipulations such as the factorizations for extreme or interior point methods that were described in Chapter 5, although some computational efficiencies are again possible as mentioned in Section 5.5. Generally, some special structure is required for efficient solution in the general case since these problems are *PSPACE-hard* (Dyer and Stougie [2006]) and require exponential effort in the horizon H for provably tight approximations with high probability (Swamy and Shmoys [2005] and Shmoys and Swamy [2006]).

In general, a variety of approximation approaches to multistage problems are possible, such as the following:

1. *value function approximation*: replacing \mathcal{D}^t with some simplified representation, such as an outer or inner linearization;
2. *constraint relaxation and dualization*: relaxing constraints into a Lagrangian or looking at dual forms that may not be implementable but may give bounds or guidelines for implementable policies;

3. *policy restriction*: restricting the set of alternative actions to a simplified form that allows for efficient computation;
4. *time, state, and path aggregation or scenario generation and reduction*: starting with a large set of possibilities and then combining (or selecting) them to form more tractable representations;
5. *Monte Carlo methods*: sampling to obtain smaller, more tractable representations.

This chapter will focus on approaches to the first two items above while the other approaches that relate more directly to approximation and sampling methods appear in Chapters 9 and 10. In Section 6.1, we describe the basic nested decomposition procedures for multistage stochastic linear programs, which represents value function approximation with outer (or inner) linearization. Section 6.2 shows how this approach extends to quadratic problems. Section 6.3 then considers the use of block separability and special problem structures. Section 6.4 describes approaches for multistage nonlinear problems based on constraint relaxation and the Lagrangian approach.

6.1 Nested Decomposition Procedures

Nested decomposition procedures were proposed for deterministic models by Ho and Manne [1974] and Glassey [1973]. These approaches are essentially inner linearizations that treat all previous periods as subproblems to a current period master problem. The previous periods generate columns that can be used by the current-period master problem.

A difficulty with these primal nested decomposition or inner linearization methods is that the set of inputs may be fundamentally different for different last period realizations. Because the number of last period realizations is the total number of scenarios in the problem, these procedures are not well adapted to the bunching procedures described in Section 5.4. Some success has been achieved, however, by Noël and Smeers [1987], as we will describe, by applying inner linearization to the dual, which is again outer linearization of the primal problem.

The general primal approach is, therefore, to use an outer linearization built on the two-stage *L*-shaped method. Louveaux [1980] first performed this generalization for multistage quadratic problems, as we discuss in Section 6.2. Birge [1985b] extended the two-stage method in the linear case as in the following description. The approach also appears in Pereira and Pinto [1985].

The basic idea of the nested *L*-shaped or Benders decomposition method is to place cuts on $\mathcal{Q}^{t+1}(x^t)$ in (3.4.3) and to add other cuts to achieve an x^t that has a feasible completion in all descendant scenarios. The cuts represent successive linear approximations of \mathcal{Q}^{t+1} . Due to the polyhedral structure of \mathcal{Q}^{t+1} , this process converges to an optimal solution in a finite number of steps.

In general, for every stage $t = 1, \dots, H - 1$ and each scenario at that stage, $k = 1, \dots, \mathcal{K}^t$,¹ we have the following master problem, which generates cuts to stage $t - 1$ and proposals for stage $t + 1$:

$$\min (c_k^t)^T x_k^t + \theta_k^t \quad (1.1)$$

$$\text{s. t. } W^t x_k^t = h_k^t - T_k^{t-1} x_{a(k)}^{t-1}, \quad (1.2)$$

$$D_{k,j}^t x_k^t \geq d_{k,j}^t, \quad j = 1, \dots, r_k^t, \quad (1.3)$$

$$E_{k,j}^t x_k^t + \theta_k^t \geq e_{k,j}^t, \quad j = 1, \dots, s_k^t, \quad (1.4)$$

$$x_k^t \geq 0, \quad (1.5)$$

where $a(k)$ is the ancestor scenario of k at stage $t - 1$, $x_{a(k)}^{t-1}$ is the current solution from that scenario, and where for $t = 1$, we interpret $b = h^1 - T^0 x^0$ as initial conditions of the problem. We may refer also to the stage H problem in which θ_k^H and constraints (1.3) and (1.4) are not present. To designate the period and scenario of the problem in (1.1)–(1.5), we also denote this subproblem, $NLDS(t, k)$.

We first describe a basic algorithm for iterating among these stages. We then discuss some enhancements of this basic approach. In the following, $\mathcal{D}^t(j)$ denotes the period t descendants of a scenario j at period $t - 1$. We assume that all variables in (3.4.1) have finite upper bounds to avoid complications presented by unbounded solutions (although, again, these can be treated as in Van Slyke and Wets [1969]).

Nested L-Shaped Method for Multistage Stochastic Linear Programs

Step 0. Set $t = 1$, $k = 1$, $r_k^t = s_k^t = 0$, add the constraint $\theta_k^t = 0$ to (1.1)–(1.5) for all t and k , and let $DIR = FORE$. Go to Step 1.

Step 1. Solve the current problem, $NLDS(t, k)$. If infeasible and $t = 1$, then stop; problem (3.4.1) is infeasible. If infeasible and $t > 1$, then let $r_{a(k)}^{t-1} = r_{a(k)}^{t-1} + 1$ and let $DIR = BACK$. Let the infeasibility condition (see Exercise 1) be obtained by a dual basic solution, $\pi_k^t, \rho_k^t \geq 0$, such that $(\pi_k^t)^T W^t + (\rho_k^t)^T D_k^t \leq 0$ but $(\pi_k^t)^T (h_k^t - T_k^{t-1} x_{a(k)}^{t-1}) + (\rho_k^t)^T d_k^t > 0$. Let $D_{a(k), r_{a(k)}^{t-1}}^{t-1} = (\pi_k^t)^T T_k^{t-1}$, $d_{a(k), r_{a(k)}^{t-1}}^{t-1} = \pi_k^t h_k^t + (\rho_k^t)^T d_k^t$.

Let $t = t - 1$, $k = a(k)$ and return to Step 1.

If feasible, update the values of x_k^t , θ_k^t , and store the value of the complementary basic dual multipliers on constraints (1.2)–(1.4) as $(\pi_k^t, \rho_k^t, \sigma_k^t)$, respectively. If $k < \mathcal{K}^t$, let $k = k + 1$, and return to Step 1. Otherwise, ($k = \mathcal{K}^t$), if $t = 1$, set $DIR = FORE$; if $DIR = FORE$ and $t < H$, let $t = t + 1$ and return. If $t = H$, let $DIR = BACK$. Go to Step 2.

¹ Instead of a fixed number of scenarios K as in the two-stage discussion, we use \mathcal{K}^t here to represent the number of distinct scenarios at stage t to avoid confusion with K^t which represents the feasibility set at stage t . Later in the text, we also use \mathcal{K}_t to represent the conditional number of outcomes at stage t , i.e., the maximum number of branches from a single node at stage $t - 1$.

Step 2. If $t = 1$, let $t = t + 1$, $k = 1$ and go to Step 1. Otherwise, for all scenarios $j = 1, \dots, \mathcal{K}^{t-1}$ at $t - 1$, compute

$$E_j^{t-1} = \sum_{k \in \mathcal{D}^t(j)} \frac{p_k^t}{p_j^{t-1}} (\pi_k^t)^T T_k^{t-1}$$

and

$$e_j^{t-1} = \sum_{k \in \mathcal{D}^t(j)} \frac{p_k^t}{p_j^{t-1}} [(\pi_k^t)^T h_k^t + \sum_{i=1}^{r_k^t} (\rho_{ki}^t)^T d_{ki}^t + \sum_{i=1}^{s_k^t} (\sigma_{ki}^t)^T e_{ki}^t].$$

The current conditional expected value of all scenario problems in $\mathcal{D}^t(j)$ is then $\bar{\theta}_j^{t-1} = e_j^{t-1} - E_j^{t-1} x_j^{t-1}$. If the constraint $\theta_j^{t-1} = 0$ appears in $NLDS(t-1, j)$, then remove it, let $s_j^{t-1} = 1$, and add a constraint (1.4) with E_j^{t-1} and e_j^{t-1} to $NLDS(t-1, j)$.

If $\bar{\theta}_j^{t-1} > \theta_j^{t-1}$, then let $s_j^{t-1} = s_j^{t-1} + 1$ and add a constraint (1.4) with E_j^{t-1} and e_j^{t-1} to $NLDS(t-1, j)$. If $t = 2$ and no constraints are added to $NLDS(1)$ ($j = \mathcal{K}^1 = 1$), then stop with x_1^1 optimal. Otherwise, let $t = t - 1$, $k = 1$. If $t = 1$, let $DIR = FORE$. Go to Step 1.

Many alternative strategies are possible in this algorithm in terms of determining the next subproblem (1.1)–(1.5) to solve. For feasible solutions, the preceding description explores all scenarios at t before deciding to move to $t - 1$ or $t + 1$. For feasible iterations, the algorithm proceeds from t in the direction of DIR until it can proceed no further in that direction. This is the “fast-forward-fast-back” procedure proposed by Wittrock [1983] for deterministic problems and implemented with success by Gassmann [1990] for stochastic problems. One may alternatively enforce a move from t to $t - 1$ (“fast-back”) or from t to $t + 1$ (“fast-forward”) whenever it is possible. From various experiments (e.g., Gassmann [1990], Morton [1996], and Birge et al. [1996]), fast-forward-fast-back sequencing protocol seems generally more efficient than the alternatives.

For infeasible solutions at some stage, this algorithm immediately returns to the ancestor problem to see whether a feasible solution can be generated. This alternative appears practical because subsequent iterations with a currently infeasible solution do not seem worthwhile.

We note that much of this algorithm can also run in parallel. We refer to Ruszczyński [1993a] who describes parallel procedures in detail. Again, one should pay attention in parallel implementations to the possible additional work for solving similar subproblems as we mentioned in Chapter 5. The convergence of this method is relatively straightforward, as given in the following.

Theorem 1. *If all Ξ^t are finite and all x^t have finite upper bounds, then the nested L-shaped method converges finitely to an optimal solution of (3.4.1).*

Proof: First, we wish to demonstrate that all cuts generated by the algorithm are valid outer linearizations of the feasible regions and objectives in (3.4.3). By induction on t , suppose that all feasible cuts (1.3) generated by the algorithm for periods t or greater are valid. For $t = H$, no cuts are present so this is true for the last period. In this case, for any $\pi_k^t, \rho_k^t \geq 0$ such that $(\pi_k^t)^T W^t + (\rho_k^t)^T D_k^t \leq 0$, we must have $(\pi_k^t)^T (h_k^t - T_k^{t-1} x_{a(k)}^{t-1}) + (\rho_k^t)^T d_k^t \leq 0$ to maintain feasibility. Because this is the cut added, these cuts are valid for $t - 1$. Thus, the induction is proved.

Now, suppose the cuts in (1.3)–(1.4) are an outer linearization of $\mathcal{Q}_k^{t+1}(x_k^t)$ for t or greater and all k . In this case, for any $(\pi_k^t, \rho_k^t, \sigma_k^t)$ feasible in (1.1)–(1.5) for t and k , $(\pi_k^t)^T (h_k^t - T_k^t x_{a(k)}^{t-1}) + \sum_{i=1}^{r_k} (\rho_{ki}^t)^T d_{ki}^t + \sum_{i=1}^{s_k} (\sigma_{ki}^t)^T e_{ki}^t$ is a lower bound on $\mathcal{Q}_{a(k)}^t(x_{a(k)}^{t-1}, k)$ for any $x_{a(k)}^{t-1}$, each k , and $a(k)$. Thus, we must have

$$\begin{aligned} \mathcal{Q}_{a(k)}^t(x_{a(k)}^{t-1}) &\geq \sum_{k \in \mathcal{D}^t(a(k))} \left(\frac{p_k^t}{p_{a(k)}^{t-1}} \right) \left((\pi_k^t)^T (h_k^t - T_k^t x_{a(k)}^{t-1}) \right. \\ &\quad \left. + \sum_{i=1}^{r_k} (\rho_{ki}^t)^T d_{ki}^t + \sum_{i=1}^{s_k} (\sigma_{ki}^t)^T e_{ki}^t \right), \end{aligned} \quad (1.6)$$

which says that $\theta_k^{t-1} \geq -E_{a(k)}^{t-1} x_{a(k)}^{t-1} + e_{a(k)}^{t-1}$, as found in the algorithm. Thus, again, we achieve a valid cut on $\mathcal{Q}_{a(k)}^{t-1}$ for any $a(k)$, completing the induction.

Now, suppose that the algorithm terminates. This can only happen if (1.1)–(1.5) is infeasible for $t = 1$ or if each subproblem for $t = 2$ has been solved and no cuts are generated. In the former case, the problem is infeasible, because the cuts (1.3) are all outer linearizations of the feasible region. In the latter case, we must have $\theta^1 = \mathcal{Q}^2(x^1)$, the condition for optimality.

For finiteness, proceed by induction. Suppose that at stage t , at most a finite number of cuts from stage $t + 1$ to H can be generated for each k at t . For H , this is again trivially true. Because at most a finite number of cuts are possible at each k , at most a finite number of basic solutions, $(\pi_k^t, \rho_k^t, \sigma_k^t)$, can be generated to form cuts for $a(k)$. Thus, at most a finite number of cuts can be generated for all $a(k)$ at $t - 1$, again completing the induction.

The proof is complete by noting that every iteration of Step 1 or 2 produces a new cut. Because there is only a finite number of possible cuts, the procedure stops finitely. \square

The nested L -shaped method has many features in common with the standard two-stage L -shaped algorithm. There are, however, peculiarities about the multistage method. We consider the following example in some detail to illustrate these features. In particular, we should note that the two-stage method always produces cuts that are supports of the function \mathcal{Q} if the subproblem is solved to optimality. In the multistage case, with the sequencing protocol just given, we may not actually generate a true support so that the cut may lie strictly below the function being approximated.

Example 1

Suppose we are planning production of air conditioners over a three month period. In each month, we can produce 200 air conditioners at a cost of \$100 each. We may also use overtime workers to produce additional air conditioners if demand is heavy, but the cost is then \$300 per unit. We have a one-month lead time with our customers, so that we know that in Month 1, we should meet a demand of 100. Orders for Months 2 and 3 are, however, random, depending heavily on relatively unpredictable weather patterns. We assume this gives an equal likelihood in each month of generating orders for 100 or 300 units.

We can store units from one month for delivery in a subsequent month, but we assume a cost of \$50 per unit per month for storage. We assume also that all demand must be met. Our overall objective is to minimize the expected cost of meeting demand over the next three months. (We assume that the season ends at that point and that we have no salvage value or disposal cost for any leftover items. This resolves the end-of-horizon problem here.)

Let x_k^t be the regular-time production in scenario k at month t , let y_k^t be the number of units stored from scenario k at month t , let w_k^t be the overtime production in scenario k at month t , and let d_k^t be the demand for month t under scenario k . The multistage stochastic program in deterministic equivalent form is:

$$\begin{aligned} \min & x^1 + 3.0w^1 + 0.5y^1 + \sum_{k=1}^2 p_k^2(x_k^2 + 3.0w_k^2 + 0.5y_k^2) \\ & + \sum_{k=1}^4 p_k^3(x_k^3 + 3.0w_k^3) \\ \text{s. t. } & x^1 \leq 2, \\ & x^1 + w^1 - y^1 = 1, \\ & y^1 + x_k^2 + w_k^2 - y_k^2 = d_k^2, \\ & x_k^2 \leq 2, \quad k = 1, 2, \\ & y_{a(k)}^2 + x_k^3 + w_k^3 - y_k^3 = d_k^3, \\ & x_k^3 \leq 2, \quad k = 1, \dots, 4, \\ & x_k^t, y_k^t, w_k^t \geq 0, \quad k = 1, \dots, \mathcal{K}^t, \quad t = 1, 2, 3, \end{aligned} \tag{1.7}$$

where $a(k) = 1$, if $k = 1, 2$ at period 3, $a(k) = 2$ if $k = 3, 4$ at period 3, $p_k^2 = 0.5$, $k = 1, 2$, $p_k^3 = 0.25$, $k = 1, \dots, 4$, $d_1^2 = 1$, $d_2^2 = 3$, and $d^3 = (1, 3, 1, 3)^T$.

The nested L-shaped method applied to (1.7) follows these steps for the first two iterations. We list an iteration at each change of DIR .

Step 0. All subproblems $NLDS(t, k)$ have the explicit $\theta_k^t = 0$ constraint. $DIR = FORE$.

Iteration 1:

Step 1. Here $t = 1$, $k = 1$. The subproblem $NLDS(1,1)$ is:

$$\begin{aligned} & \min x^1 + 3w^1 + 0.5y^1 + \theta^1 \\ \text{s. t. } & x^1 \leq 2, \\ & x^1 + w^1 - y^1 = 1, \\ & x^1, w^1, y^1 \geq 0, \\ & \theta^1 = 0, \end{aligned} \tag{1.8}$$

which has the solution $x^1 = 1$; other variables are zero.

Step 1. Now, $t = 2$, $k = 1$, and $NLDS(2,1)$ is

$$\begin{aligned} & \min x_1^2 + 3w_1^2 + 0.5y_1^2 + \theta_1^2 \\ \text{s. t. } & x_1^2 \leq 2, \\ & x_1^2 + w_1^2 - y_1^2 = 1, \\ & x_1^2, w_1^2, y_1^2 \geq 0, \\ & \theta_1^2 = 0, \end{aligned} \tag{1.9}$$

which has the solution, $x_1^2 = 1$; other variables are zero.

Step 1. Here, $t = 2$, $k = 2$, and $NLDS(2,2)$ is

$$\begin{aligned} & \min x_2^2 + 3w_2^2 + 0.5y_2^2 + \theta_2^2 \\ \text{s. t. } & x_2^2 \leq 2, \\ & x_2^2 + w_2^2 - y_2^2 = 3, \\ & x_2^2, w_2^2, y_2^2 \geq 0, \\ & \theta_2^2 = 0, \end{aligned} \tag{1.10}$$

which has the solution, $x_2^2 = 2$, $w_2^2 = 1$; other variables are zero.

Step 1. Next, $t = 3$, $k = 1$. $NLDS(3,1)$ is

$$\begin{aligned} & \min x_1^3 + 3w_1^3 + 0.5y_1^3 + \theta_1^3 \\ \text{s. t. } & x_1^3 \leq 2, \\ & x_1^3 + w_1^3 - y_1^3 = 1, \\ & x_1^3, w_1^3, y_1^3 \geq 0, \\ & \theta_1^3 = 0, \end{aligned} \tag{1.11}$$

which has the solution, $x_1^3 = 1$; other primal variables are zero. The complementary basic dual solution is $\pi_1^3 = (0, 1)^T$.

Step 1. Next, $t = 3$, $k = 2$. $NLDS(3,2)$ has the same form as $NLDS(3,1)$, except we replace the second constraint with $x_2^3 + w_2^3 - y_2^3 = 3$. It has the solution, $x_2^3 = 2$, $w_2^3 = 1$; other primal variables are zero. The complementary basic dual solution is $\pi_2^3 = (-2, 3)^T$.

Step 1. For $t = 3$, $k = 3$, we have the same subproblem and solution as $t = 3$, $k = 1$, so $x_3^3 = 1$; other primal variables are zero. The complementary basic dual solution is $\pi_3^3 = (0, 1)^T$.

Step 1. For $t = 3$, $k = 4$, we have the same subproblem and solution as $t = 3$, $k = 2$, $x_4^3 = 2$, $w_4^3 = 1$; other primal variables are zero. The complementary basic dual solution is $\pi_4^3 = (-2, 3)^T$. Now, $DIR = BACK$, and we go to Step 2.

Iteration 2:

Step 2. For scenario $j = 1$ and $t - 1 = 2$, we have

$$\begin{aligned} E_{11}^2 &= \left(\frac{0.25}{0.5} \right) (\pi_1^3 T_1^2 + \pi_2^3 T_2^2) \\ &= (0.5) (0 \ 1) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} + (0.5) (-2 \ 3) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &= (0 \ 0 \ 2) \end{aligned} \quad (1.12)$$

and

$$\begin{aligned} e_{11}^2 &= \left(\frac{0.25}{0.5} \right) (\pi_1^3 h_1^3 + \pi_2^3 h_2^3) \\ &= (0.5) (0 \ 1) \begin{pmatrix} 2 \\ 1 \end{pmatrix} + (0.5) (-2 \ 3) \begin{pmatrix} 2 \\ 3 \end{pmatrix} \\ &= 3, \end{aligned} \quad (1.13)$$

which yields the constraint, $2y_1^2 + \theta_1^2 \geq 3$, to add to $NLDS(2,1)$.

For scenario $j = 2$ at $t - 1 = 2$, we have the same, $E_{21}^2 = (0 \ 0 \ 2)$, $e_{21}^2 = 3$. Now $t = 2$ and $k = 1$.

Step 1. $NLDS(2,1)$ is now:

$$\begin{aligned} \min & x_1^2 + 3w_1^2 + 0.5y_1^2 + \theta_1^2 \\ \text{s. t.} & x_1^2 \leq 2, \\ & x_1^2 + w_1^2 - y_1^2 = 1, \\ & 2y_1^2 + \theta_1^2 \geq 3, \\ & x_1^2, w_1^2, y_1^2 \geq 0, \end{aligned} \quad (1.14)$$

which has an optimal basic feasible solution, $x_1^2 = 2$, $y_1^2 = 1$, $\theta_1^2 = 1$, $w_1^2 = 0$, with complementary dual values, $\pi_1^2 = (-0.5, 1.5)^T$, $\sigma_{11}^2 = 1$.

Step 1. *NLDS(2,2)* has the same form as (1.14) except that the demand constraint is $x_2^2 + w_2^2 - y_2^2 = 3$. The optimal basic feasible solution found to this problem is $x_2^2 = 2$, $w_2^2 = 1$, $\theta_2^2 = 3$, $y_2^2 = 0$, with complementary dual values, $\pi_2^2 = (-2, 3)^T$, $\sigma_{21}^2 = 1$. We continue in *DIR = BACK* to Step 2.

Step 2. For scenario $t-1=1$, we have

$$\begin{aligned} E_1^1 &= (0.5)(\pi_1^2 T_1^2 + \pi_2^2 T_2^2) \\ &= (0.5)(-0.5 \ 1.5) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} + (0.5)(-2 \ 3) \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &= (0 \ 0 \ 2.25) \end{aligned} \quad (1.15)$$

and

$$\begin{aligned} e_1^1 &= (0.5)(\pi_1^2 h_1^2 + \pi_2^2 h_2^2) + (0.5)(\sigma_{11}^2 e_{11}^2 + \sigma_{21}^2 e_{21}^2) \\ &= (0.5)(-0.5 \ 1.5) \begin{pmatrix} 2 \\ 1 \end{pmatrix} + (0.5)(-2 \ 3) \begin{pmatrix} 2 \\ 3 \end{pmatrix} + (0.5)((1)(3) + (1)3) \\ &= (0.5)(0.5 + 5 + 6) = 5.75, \end{aligned} \quad (1.16)$$

which yields the constraint, $2.25y^1 + \theta^1 \geq 5.75$, to add to *NLDS(1)*.

Step 1. *NLDS(1)* is now:

$$\begin{aligned} \min & x^1 + 3w^1 + 0.5y^1 + \theta^1 \\ \text{s. t.} & x^1 \leq 2, \\ & x^1 + w^1 - y^1 = 1, \\ & 2.25y^1 + \theta^1 \geq 5.75, \\ & x^1, w^1, y^1 \geq 0, \end{aligned} \quad (1.17)$$

with optimal basis feasible solution, $x^1 = 2$, $y^1 = 1$, $w^1 = 0$, $\theta^1 = 3.5$. *DIR = FORE*.

This procedure continues through six total iterations to solve the problem. At the last iteration, we obtain $\bar{\theta}^1 = 3.75 = \theta^1$, so no new cuts are generated for Period 1. We stop with a current solution as optimal, $x^{1*} = 2$, $y^{1*} = 1$, $z^* = 2.5 + 3.75 = 6.25$. In Exercise 2, we ask the reader to generate each of the cuts.

Following the nested *L*-shaped method completely takes many steps in this example, six iterations or changes of direction corresponding to three forward passes and three backward passes. Figure 1 illustrates the process and provides some insight into nested decomposition performance.

In Figure 1, the solid line gives the objective value in (1.7) as a function of total production $prod^1 = x^1 + w^1$ in the first period. The dashed lines correspond to the cuts made by the algorithm (Cut 1,2). The first cut was $2.25y^1 + \theta \geq 5.75$ from (1.15)–(1.16) on Iteration 2. Because $y^1 = x^1 + w^1 - 1$, we can substitute for y^1 to obtain, $2.25x^1 + 2.25w^1 + \theta \geq 8$. The objective in (1.17) is $z^1 = x^1 + 3w^1 + 0.5y^1 + \theta$, so, combined with $1 \leq x^1 \leq 2$, we can substitute $\theta \geq 8 - 2.25(prod^1)$ to obtain $z^1(prod^1) = 7.5 + (1.5)\min\{2, prod^1\} + 3.5(prod^1 - 2)^+ - 2.25prod^1$, where $prod^1 \geq 1$. This can also be written as:

$$z^1(prod^1) = \begin{cases} 7.5 - 0.75prod^1 & \text{if } prod^1 \leq 2, \\ 3.5 + 1.25prod^1 & \text{if } prod^1 > 2, \end{cases} \quad (1.18)$$

which corresponds to the wide dashed line (*Cut 1*) in Figure 1.

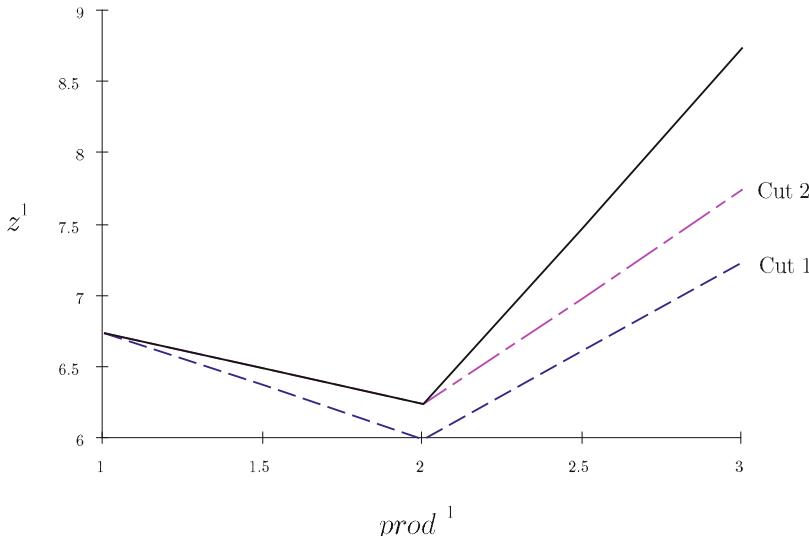


Fig. 1 The first period objective function (solid line) for the example and cuts (dashed lines) generated by the nested L -shaped method.

The second cut occurs on Iteration 4 (verify this in Exercise 2) as $2x^1 + 2w^1 + \theta \geq 7.75$, which yields $z^1(prod^1) = x^1 + 3w^1 + 0.5y^2 + \theta \geq 7.25 + (1.5)\min\{2, prod^1\} + 3.5(prod^1 - 2)^+ - 2prod^1$ or

$$z^1(prod^1) \geq \begin{cases} 7.25 - 0.5prod^1 & \text{if } prod^1 \leq 2, \\ 3.25 + 1.5prod^1 & \text{if } prod^1 > 2. \end{cases} \quad (1.19)$$

This cut corresponds to the narrow width dashed line (*Cut 2*) in Figure 1.

The optimal value and solution in terms of prod^1 can be read from Figure 1 as each cut is added. With only Cut 1, the lowest value of z^1 occurs when $\text{prod}^1 = 2$. With Cuts 1 and 2, the minimum is also achieved at $\text{prod}^1 = 2$. Note that the first cut is not, however, a facet of the objective function's graph. The cuts meet the objective at $\text{prod}^1 = 1$ and $\text{prod}^1 = 2$, respectively, but they need not even do this, as we mentioned earlier (see Exercise 3). The other parts of the Period 1 cuts are generated from bounds on Q_2^2 .

This example illustrates some of the features of the nested L -shaped method. Besides our not being guaranteed of obtaining a support of the function at each step, another possible source of delay in the algorithm's convergence is degeneracy. As the example illustrates, the solutions at each step occur at the links of the piecewise linear pieces generated by the method (Exercises 5 and 5). At these places, many bases may be optimal so that several bases may be repeated. Some remedies are possible, as in Birge [1980] and, for deterministic problems, Abrahamson [1983].

As with the standard two-stage L -shaped method, the nested L -shaped method acquires its greatest gains by combining the solutions of many subproblems through bunching (or sifting). In addition, multicuts are valuable in multistage as well as two-stage problems. Infanger [1991, 1994] has also suggested the uses of generating many cuts simultaneously when future scenarios all have similar structure. This procedure may make bunching efficient for periods other than H by making every constraint matrix identical for all scenarios in a given period. In this way, only objective and right-hand side constraint coefficients vary among the different scenarios.

In terms of primal decomposition, we mentioned the work of Noël and Smeers at the outset of this chapter. They apply nested Dantzig-Wolfe decomposition to the dual of the original problem. As we saw in Chapter 5, this is equivalent to applying outer linearization to the primal problem. The only difference is that they allow for some nonlinear terms in their constraints, which would correspond to a nonlinear objective in the primal model. Because the problems are still convex, nonlinearity does not really alter the algorithm. The only problem may be in the finiteness of convergence.

The advantage of a primal or dual implementation generally rests in the problem structure, although primal or dual simplex may be used in either method, making them indistinguishable. Gassmann [1990] presents some indication that dual iterations may be preferred in bunching. In general, many primal columns and few rows would tend to favor a primal approach (outer linearization as in the L -shaped method) while few columns and many rows would tend to favor a dual approach. In any case, the form of the algorithm and all proofs of convergence apply to either form.

While nested decomposition (and other linearization methods) are particularly well-suited for linear problems, the general methods apply equally well for convex nonlinear problems (i.e., problems with convex, time-separable objectives and convex constraints, see Exercise 8). Birge and Rosa [1996] describe a nested decomposition of this form applied to global energy-economy-environment interaction

models. They use an active set approach for the subproblems, but interior point methods might also be used.

Exercises

1. Verify that the infeasibility condition is as given in Step 1 of the nested L -shaped method. (Hint: note that if x_k^t satisfies (1.2) and (1.3), then there exists θ_k^t such that (x_k^t, θ_k^t) satisfy (1.4).)
2. Continue Example 1 with the nested L -shaped method until you obtain an optimal solution.
3. Construct a multistage example in which a cut generated by the second period in following the nested L -shaped method does not meet $\mathcal{Q}^1(x^1)$ for any value of x^1 , i.e., $-E_1^1 x^1 + e_1^1 < \mathcal{Q}(x^1)$.
4. Show that the situation in (1.1) is not possible if the fast-forward protocol is always followed.
5. Suppose a feasibility cut (1.3) is active for x_k^t for any t and k . Show that every basic feasible solution of $NLDS(t+1, j)$ with input x_k^t for some scenario $j \in \mathcal{D}^{t+1}(k)$ must be degenerate.
6. Suppose two optimality cuts (1.4) are active for (x_k^t, θ_k^t) for any t and k . Show that either the subproblems generate a new cut with $\bar{\theta}_k^t > \theta_k^t$ or an optimal solution of $NLDS(t+1, j)$ with input x_k^t for some scenario $j \in \mathcal{D}^{t+1}(k)$ must be degenerate.
7. Using four processors, what efficiency can be gained by solving the preceding example in parallel? Find the utilization of each processor and the speed-up of elapsed time, assuming each subproblem requires the same solution time.
8. Suppose θ^1 is broken into separate components for Q_1^2 and Q_2^2 as in the two-stage multicut approach. How does that alter the solution of the example?
9. Suppose that the objective in each period t for each scenario k is a general convex function $f_k^t(x_k^{t-1}, x_k^t)$ and, in addition to the linear constraints, there is an additional convex constraint, $g_k^t(x_k^{t-1}, x_k^t) \leq 0$. Assuming relatively complete recourse for simplicity and that your solver can return the primal solution and dual multipliers for the K-K-T system of equations, describe how you would modify the nested decomposition steps to accommodate these nonlinear functions.

6.2 Quadratic Nested Decomposition

Decomposition techniques for multistage nonlinear programs are available for the case in which the objective function is quadratic convex, the constraint set polyhe-

dral, and the random variables discrete. For the sake of clarity, we repeat the recursive definition of the deterministic equivalent program, already given in Section 3.4.

$$(MQSP) \quad \begin{aligned} \min z_1(x^1) &= (c^1)^T x^1 + (x^1)^T D^1 x^1 + \mathcal{Q}^2(x^1) \\ \text{s. t. } W^1 x^1 &= h^1, \\ x^1 &\geq 0, \end{aligned} \tag{2.1}$$

where $Q^t(x^{t-1}, \xi^t(\omega)) =$

$$\begin{aligned} \min (c^t(\omega))^T x^t(\omega) + (x^t(\omega))^T D^t(\omega) x^t(\omega) + \mathcal{Q}^{t+1}(x^{t+1}) \\ \text{s. t. } W^t x^t(\omega) = h^t(\omega) - T^{t-1}(\omega) x^{t-1}, \\ x^t(\omega) \geq 0, \end{aligned} \tag{2.2}$$

$$\mathcal{Q}^{t+1}(x^t) = E_{\xi^{t+1}} Q^{t+1}(x^t, \xi^{t+1}(\omega)), \quad t = 1, \dots, H-1, \tag{2.3}$$

and

$$\mathcal{Q}^H(x^{H-1}) = 0. \tag{2.4}$$

In $MQSP$, D^t is an $n_t \times n_t$ matrix. All other matrices have the dimensions defined in the linear case. The random vector, $\xi^t(\omega)$, is formed by the elements of $c^t(\omega)$, $h^t(\omega)$, $T^{t-1}(\omega)$, and $D^t(\omega)$. We keep the notation that ξ^t is an N^t -vector on (Ω, W^t, P) , with support Ξ^t . Finally, we again define

$$K^t = \{x^t \mid \mathcal{Q}^{t+1}(x^t) < \infty\}.$$

We also define $z^t(x^t) = (c^t)^T x^t + (x^t)^T D^t x^t + \mathcal{Q}^{t+1}(x^t)$.

Theorem 2. *If the matrices $D^t(\omega)$ are positive semi-definite for all $\omega \in \Omega$ and $t = 1, \dots, H$, then the sets K^t and the functions $\mathcal{Q}^{t+1}(x^t)$ are convex for $t = 1, \dots, H-1$. If Ξ^t is also finite for $t = 2, \dots, H$, then K^t is polyhedral. Moreover $z^t(x^t)$ is either identically $-\infty$ or there exists a decomposition of K^t into a polyhedral complex such that the t^{th} -stage deterministic equivalent program (2.2) is a piecewise quadratic program.*

Proof: The piecewise quadratic property of (2.2) is obtained by inductively applying to each cell of the polyhedral complex of K^t the result that if $z^t(\cdot)$ is a finite positive semi-definite quadratic form, there exists a piecewise affine continuous optimal decision rule for (2.2). All others results were given in Section 3.4. \square

We now describe a nested decomposition algorithm for $MQSP$ first presented in Louveaux [1980]. For simplicity in the presentation of the algorithms, we assume relatively complete recourse. This means that we skip the step that consists of generating feasibility cuts. If needed, those cuts are generated exactly as in the multistage linear case. We keep the notation of $a(k)$ for the ancestor scenario of k at stage

$t - 1$. As in Section 6.1, c_k^t , D_k^t , and \mathcal{Q}_k^{t+1} represent realizations of c^t , D^t , and \mathcal{Q}^{t+1} for scenario k and x_k^t is the corresponding decision vector. In Stage 1, we use the notations, z_1 and z_1^1 and x_1 and x_1^1 , as equivalent.

Nested PQP Algorithm for MQSP

Step 0. Set $t = 1$, $k = 1$, $C_1 = S_1 = K_1$. Choose $x_1^1 \in K_1$.

Step 1. If $t = H$, go to Step 2. For $i = t + 1, \dots, H$, let $k = 1$, $z_1^i(x_1^i) = (c_1^i)^T x_1^i + (x_1^i)^T D_1^i x_1^i$ and $C_1^i(x_{a(1)}^{i-1}) = S_1^i(x_{a(1)}^{i-1}) = K^i(x_{a(1)}^{i-1})$. Choose $x_1^i \in K^i(x_{a(1)}^{i-1})$. Set $t = H$.

Step 2. Find $v \in \arg\min\{z_k^t(x_k^t) \mid x_k^t \in S_k^t(x_{a(k)}^{t-1})\}$. Find $w \in \arg\min\{z_k^t(x_k^t) \mid x_k^t \in C_k^t(x_{a(k)}^{t-1})\}$. If w is the limiting point on a ray on which $z_k^t(\cdot)$ is decreasing to $-\infty$, then $(DEP)_k^t$ is unbounded and the algorithm terminates.

Step 3. If $\nabla^T z_k^t(w)(v - w) = 0$, go to Step 4. Otherwise, redefine

$$S_k^t(x_{a(k)}^{t-1}) \leftarrow S_k^t(x_{a(k)}^{t-1}) \cap \{x_k^t \mid \nabla^T z_k^t(w)(x_k^t - w) \leq 0\}.$$

Let $x_k^t = v$, $z_k^t = (c_k^t)^T x_k^t + (x_k^t)^T D_k^t x_k^t$ and $C_k^t = K^t(x_{a(k)}^{t-1})$. Go to Step 1.

Step 4. If $t = 1$, stop; w is an optimal first-period decision. Otherwise, find the cell $G_k^t(x_{a(k)}^{t-1})$ containing w and the corresponding quadratic form $Q_k^t(x_{a(k)}^{t-1})$. Redefine

$$\begin{aligned} z_{a(k)}^{t-1}(x_{a(k)}^{t-1}) &\leftarrow z_{a(k)}^{t-1}(x_{a(k)}^{t-1}) + p_k^t Q_k^t(x_{a(k)}^{t-1}) \\ C_{a(k)}^{t-1}(x_{a(k)}^{t-1}) &\leftarrow C_{a(k)}^{t-1}(x_{a(k)}^{t-1}) \cap G_{a(k)}^t(x_{a(k)}^{t-1}). \end{aligned}$$

If $k = \mathcal{K}^t$, let $t \leftarrow t - 1$, go to Step 2. Otherwise, let $k \leftarrow k + 1$, $z_k^t(x_k^t) = (c_k^t)^T x_k^t + (x_k^t)^T D_k^t x_k^t$, $C_k^t = S_k^t(x_{a(k)}^{t-1}) = K^t(x_{a(k)}^{t-1})$. Choose $x_k^t \in S_k^t(x_{a(k)}^{t-1})$. Go to Step 1.

Theorem 3. *The nested PQP algorithm terminates in a finite number of steps by either detecting an unbounded solution or finding an optimal solution of the multistage quadratic stochastic program with relatively complete recourse.*

Proof: The proof of the finite convergence of the PQP algorithm in Section 5.3 amounts to showing that Step 2 of the algorithm can be performed at most a finite number of times. The same result holds for a given piecewise quadratic program (2.2) in the nested sequence. The theorem follows from the observations that there is only a finite number of different problems (2.2) and that all other steps of the algorithm are finite. \square

Numerical experiments are reported in Louveaux [1980]. It should be noted that the *MQSP* easily extends to the multistage piecewise convex case. The limit there is that the objective function and the description of the cell are usually much more difficult to obtain. One simple example is proposed in Exercise 3.

It is interesting to observe that the *MQSP* method has a tendency to require few iterations when the quadratic terms play a significant role and a good starting point is chosen. (This probably relates to the good behavior of regularized decomposition.)

Example 1 (continued)

Assume that the cost of overtime is now quadratic (for example, larger increases of salary are needed to convince more people to work overtime). We replace everywhere $3.0w_k^t$ by $2.0w_k^t + (w_k^t)^2$. Assume all other data are unchanged. Take as the starting point a situation where $0 \leq y_1^1 \leq 1$, $0 \leq y_k^2 \leq 1$, $k = 1, 2$. (It is relatively easy to see what the corresponding values for the other first- and second-stage variables should be.) We now proceed backward. Let $t = 3$.

i) $t = 3$, $k = 1$. We solve

$$\begin{aligned} & \min x_1^3 + 2w_1^3 + (w_1^3)^2 \\ & \text{s. t. } y_1^2 + x_1^3 + w_1^3 = 1, \quad x_1^3 \leq 2, \\ & \quad x_1^3, w_1^3 \geq 0, \end{aligned}$$

where inventory at the end of Period 3 has been omitted for simplicity. The solution is easily seen to be $x_1^3 = 1 - y_1^2$, $w_1^3 = 0$ and is valid for $0 \leq y_1^2 \leq 1$. It follows that

$$\mathcal{Q}_1^3(y_1^2) = 1 - y_1^2.$$

ii) $t = 3$, $k = 2$. We solve

$$\begin{aligned} & \min x_2^3 + 2w_2^3 + (w_2^3)^2 \\ & \text{s. t. } y_1^2 + x_2^3 + w_2^3 = 3, \quad x_2^3 \leq 2, \\ & \quad x_2^3, w_2^3 \geq 0. \end{aligned}$$

The solution is now $x_2^3 = 2$, $w_2^3 = 1 - y_1^2$, valid for $0 \leq y_1^2 \leq 1$. It yields $\mathcal{Q}_2^3(y_1^2) = 4 - 2y_1^2 + (1 - y_1^2)^2$.

Combining (i) and (ii), we obtain

$$\mathcal{Q}_1^2(y_1^2) = \frac{1}{2}\mathcal{Q}_1^3(y_1^2) + \frac{1}{2}\mathcal{Q}_2^3(y_1^2) = \frac{5}{2} - \frac{3}{2}y_1^2 + \frac{(1 - y_1^2)^2}{2}$$

and

$$C_1^2(y_1^2) = \{y_1^2 \mid 0 \leq y_1^2 \leq 1\}.$$

iii) and iv) Because the randomness is only in the right-hand side, we conclude that cases (iii) and (iv) are identical to (i) and (ii), respectively. Hence,

$$\mathcal{Q}_2^2(y_2^2) = \frac{5}{2} - \frac{3}{2}y_2^2 + \frac{(1 - y_2^2)^2}{2} \quad \text{and} \quad C_2^2(y_2^2) = \{y_2^2 \mid 0 \leq y_2^2 \leq 1\}.$$

Next, we have $t = 2$.

i) $t = 2$, $k = 1$. The objective z_1^2 is computed as

$$z_1^2 = x_1^2 + 2w_1^2 + (w_1^2)^2 + 0.5y_1^2 + \frac{5}{2} - \frac{3}{2}y_1^2 + \frac{(1-y_1^2)^2}{2},$$

i.e.,

$$z_1^2 = \frac{5}{2} + x_1^2 + 2w_1^2 + (w_1^2)^2 - y_1^2 + \frac{(1-y_1^2)^2}{2}.$$

The constraint sets are

$$S_1^2 = \{x_1^2, w_1^2, y_1^2 \mid y^1 + x_1^2 + w_1^2 - y_1^2 = 1, 0 \leq x_1^2 \leq 2, x_1^2, w_1^2, y_1^2 \geq 0\}$$

and

$$C_1^2 = S_1^2 \cap \{0 \leq y_1^2 \leq 1\}.$$

The solution v of minimizing $z_1^2(\cdot)$ over S_1^2 is

$$y_1^2 = 1, \quad x_1^2 = 2 - y^1.$$

Because the solution belongs to C_1^2 , we can take $w = v$. (Beware that w without superscript and subscript corresponds to the optimal solution on a cell defined in Step 2, while w with superscript and subscript corresponds to overtime.) Thus, this point satisfies the optimality criterion in Step 3. It yields

$$\mathcal{D}_1^2(y^1) = \frac{5}{2} + 2 - y^1 - 1 = \frac{7}{2} - y^1$$

and

$$C_1^2(y^1) = \{y^1 \mid 0 \leq y^1 \leq 2\}.$$

ii) $t = 2$, $k = 2$. The objective z_2^2 is similarly computed as

$$z_2^2 = \frac{5}{2} + x_2^2 + 2w_2^2 + (w_2^2)^2 - y_2^2 + \frac{(1-y_2^2)^2}{2}.$$

The constraint set

$$S_2^2 = \{x_2^2, w_2^2, y_2^2 \mid y^1 + x_2^2 + w_2^2 - y_2^2 = 3, 0 \leq x_2^2 \leq 2, x_2^2, w_2^2, y_2^2 \geq 0\}$$

only differs in the right-hand side of the inventory constraint with

$$C_2^2 = S_2^2 \cap \{0 \leq y_2^2 \leq 1\}.$$

The solution v is now $x_2^2 = 2$, $w_2^2 = 1 - y^1$, $y_2^2 = 0$. Again $v \in C_2^2$, so that we have $w = v$, which satisfies the optimality criterion in Step 3. It yields

$$\begin{aligned}\mathcal{Q}_2^2(y^1) &= \frac{5}{2} + 2 + 2(1 - y^1) + (1 - y^1)^2 + \frac{1}{2} = 7 - 2y^1 + (1 - y^1)^2 \text{ and} \\ C_2^2(y^1) &= \{y^1 \mid 0 \leq y^1 \leq 1\}.\end{aligned}$$

Next is the case for $t = 1$.

The current objective function is computed as

$$z_1 = 21/4 - y^1 + \frac{(1 - y^1)^2}{2} + x^1 + 2w^1 + (w^1)^2.$$

The constraint sets are

$$\begin{aligned}S_1^1 &= \{x^1, w^1, y^1 \mid x^1 + w^1 - y^1 = 1, x^1 \leq 2, x^1, w^1, y^1 \geq 0\}, \\ C_1^1 &= S_1^1 \cap \{0 \leq y^1 \leq 1\}.\end{aligned}$$

The solution v of minimizing z_1 over S_1^1 is

$$x^1 = 2, \quad y^1 = 1, \quad w^1 = 0,$$

with objective value $z_1 = \frac{25}{4}$. Because this solution belongs to C_1^1 , it is the optimal solution of the problem. Thus, no cut was needed to optimize the problem.

Exercises

- Consider Example 1 with quadratic terms as in this section and take $1 \leq y^1 \leq 2$, $1 \leq y_1^2 \leq 2$, $0 \leq y_2^2 \leq 1$ as a starting point. Show that the following steps are generated. Obtain $0.5Q_1^3(y_1^2) + 0.5Q_2^3(y_1^2) = \frac{5}{4} - \frac{1}{4}y_1^2$. In $t = 2$, $k = 1$, solution v is $x_1^2 = 0$, $y_1^2 = y^1 - 1$ while w is $y_1^2 = 1$, $x_1^2 = 2 - y^1$, both with $w_1^2 = 0$. A cut $x_1^2 + 2w_1^2 + \frac{1}{4}y_1^2 \leq \frac{9}{4} - y^1$ is added. The new starting point is v , which corresponds to $0 \leq y_1^2 \leq 1$. Then the case $t = 2$, $k = 1$ is as in the text, yielding

$$\mathcal{Q}_1^2(y^1) = \frac{7}{2} - y^1 \quad \text{and} \quad C_1^2(y^1) = \{0 \leq y^1 \leq 2\}.$$

In $t = 2$, $k = 2$ (see the calculations in the text), we obtain $\mathcal{Q}_2^2(y^1) = 6 - y^1$ and $C_2(y^1) = \{1 \leq y^1 \leq 3\}$. Thus, in $t = 1$, $z_1 = x^1 + 2w^1 + (w^1)^2 + \frac{19}{4} - y^1/2$ and $C = \{1 \leq y^1 \leq 2\}$. Again, the solution $v : x^1 = 1$, $y^1 = 0$, $w^1 = 0$ does not coincide with $w : x^1 = 2$, $y^1 = 1$, $w^1 = 0$. A cut $x^1 - \frac{y^1}{2} + w^1 \leq 3/2$ is generated. The new starting point now coincides with the one in the text and the solution is obtained in one more iteration.

6.3 Block Separability and Special Structure

The definition of block separability was given in Section 3.4. It permits separate calculation of the recourse functions for the aggregate level decisions and the detailed level decisions. This is an advantage in terms of the number of variables and constraints, but often it makes the computation of the recourse functions and of the cells of the decomposition much easier in the case of a quadratic multistage program. This has been exploited in Louveaux [1986] and Louveaux and Smeers [2011].

We will illustrate a further benefit. It also consists of separating the random vectors. Consider the production of a single product. Now, assume the product cannot be stored (as in the case of a perishable good) or that the policy of the firm is to use a just-in-time system of production so that only a fixed safety stock is kept at the end of each period.

Assume that units are such that one worker produces exactly one product per stage. Two elements are uncertain: labor cost and demand. Labor cost is currently 2 per period. Next period, labor cost may be 2 or 3, with equal probability. Current revenue is 5 per product in normal time and 4 in overtime. Overtime is possible for up to 50% of normal time. Demand is a uniform continuous random variable within (0, 200) and (0, 100), respectively, for the next two periods. The original workforce is 50. Hiring and firing is possible once a period, at the cost of one unit each. Clearly, the labor decision is the aggregate level decision.

To keep notation in line with Section 3.4, we consider a three-stage model. In Stage 1, the decision about labor is made, say for Year 1. Stage 2 consists of production of Year 1 and decision about labor for Year 2. Stage 3 only consists of production of Year 2. Let ξ_1^t be labor cost in stage t , while ξ_2^t is the demand in stage t . Let w^t be the workforce in stage t . Then,

$$Q_w^t(w^{t-1}, \xi_1^t) = \min |w^t - w^{t-1}| + \xi_1^t w^t + \mathcal{Q}^{t+1}(w^t), \quad (3.1)$$

$$\mathcal{Q}^{t+1}(w^t) = E_{\xi^{t+1}}[Q_w^{t+1}(w^t, \xi_1^{t+1}) + Q_y^{t+1}(w^t, \xi_2^{t+1})], \quad (3.2)$$

and $Q_y^{t+1}(w^t, \xi_2^{t+1})$ is minus the expected revenue of production in stage $t+1$ given a workforce w^t and a demand scenario ξ_2^{t+1} . It is obtained as follows.

Let D^t represent the maximal demand in stage t (200 for $t=2$, 100 for $t=3$). Observe that the expectation of ξ_2^t is $D^t/2$ because ξ_2^t is uniformly continuous over $[0, D^t]$. If $w^t \geq D^t$, all demand can be satisfied with normal time. If $w^t \leq D^t \leq 1.5w^t$, demand up to w^t is satisfied with normal time, the rest in overtime. Finally, if $D^t \geq 1.5w^t$, normal time is possible up to a demand of w^t , overtime from w^t to $1.5w^t$, and extra demand is lost. Taking expectations over these cases, we obtain

$$\mathcal{Q}_y^{t+1}(w^t) = E_{\xi^{t+1}}[Q_y^{t+1}(w^t, \xi_2^{t+1})] = \begin{cases} -2.5D^t & \text{if } w^t \geq D^t, \\ \frac{(w^t)^2}{2D^t} - w^t - 2D^t & \text{if } w^t \leq D^t \leq 1.5w^t, \\ \frac{5(w^t)^2}{D^t} - 7w^t & \text{if } 1.5w^t \leq D^t. \end{cases}$$

This problem can now be solved with the *MQSP* algorithm. Assume $w^0 = 50$, $w^1 \geq 50$.

Let Stage (2, 1) represent the first labor scenario in Stage 2, i.e., $\xi_1^2 = 2$. The problem consists of finding

$$\begin{aligned} & \min |w^2 - w^1| + 2w^2 + \mathcal{Q}^3(w^2) \\ & \text{s. t. } w^2 \geq 0 . \end{aligned}$$

We compute $\mathcal{Q}^3(w^2) = \mathcal{Q}_y^3(w^2) = \frac{5(w^2)^2}{100} - 7w^2$, for $w^2 \leq \frac{200}{3}$, because $D^3 = 100$. We also replace $|w^2 - w^1|$ by an explicit expression in terms of hiring (h^2) and firing (f^2). The problem in Stage (2, 1) now reads:

$$\begin{aligned} Q_w^2(w^1, 1) &= \min h^2 + f^2 - 5w^2 + \frac{5(w^2)^2}{100} \\ &\text{s. t. } w^2 - h^2 + f^2 = w^1, \\ & \quad w^2 \geq 0, \quad h^2 \geq 0, \quad f^2 \geq 0 . \end{aligned}$$

Under this form, the problem is clearly quadratic convex (remember w^2 is w in Stage 2, not the square of w). Classical Karush-Kuhn-Tucker conditions give the optimal solution $w^2 = w^1$, as long as $40 \leq w^1 \leq 60$. Then

$$Q_w^2(w^1, 1) = -5w^1 + \frac{5(w^1)^2}{100} .$$

Similarly, in Scenario (2, 2) where $\xi_1^2 = 3$, the solution of

$$\begin{aligned} & \min |w^2 - w^1| + 3w^2 + \mathcal{Q}^3(w^2) \\ & \text{s. t. } w^2 \geq 0 \end{aligned}$$

is $w^2 = 50$, $f^2 = w^1 - 50$, as long as $w^1 \geq 50$. Then

$$Q_w^2(w^1, 2) = w^1 - 125 ,$$

and

$$\mathcal{Q}_w^2(w^1) = -\frac{125}{2} - 2w^1 + \frac{2.5(w^1)^2}{100} ,$$

which is valid within $C^2 = \{50 \leq w^1 \leq 60\}$.

The Stage 1 objective is:

$$\min h^1 + f^1 + 2w^1 + \mathcal{Q}_y^2(w^1) + \mathcal{Q}_w^2(w^1) ,$$

so that the Stage 1 problem reads:

$$\min h^1 + f^1 - 7w^1 + \frac{(w^1)^2}{20} - \frac{125}{2}$$

$$\begin{aligned} \text{s. t. } w^1 - h^1 + f^1 &= 50, \\ w^1, h^1, f^1 &\geq 0. \end{aligned}$$

Its optimal solution, $w^1 = 60$, $h^1 = 10$, belongs to C^2 and is thus also the optimal solution of the global problem with objective value -292.5 .

Many two-stage methods may also be enhanced for multiple stages using some form of block separability. One such approach assumes deviations from some mean value can be corrected by a penalty only relating to the current period. This method basically applies a simple recourse strategy in every period. For example, in Kallberg, White and Ziemba [1982] and Kusy and Ziemba [1986], penalties are imposed to meet financial requirements in each period of a short-term financial planning model. With this type of penalty, the various simple recourse methods may be applied to achieve efficient computation.

Exercises

1. Does the block separable property depend on having a single product? To help answer this question, take the example in the block separability paragraph and assume a second product with revenue 0.6 in normal time and 0.3 in overtime. One worker produces 10 such products in one stage. Obtain $\mathcal{Q}_y^{t+1}(w^t)$,
 - (a) if demand in Period t is known to be 400;
 - (b) if demand in Period t is uniform continuous within $[0, 500]$ and $[0, 100]$, respectively, for the two periods.
2. In the case of one product, obtain $\mathcal{Q}_y^{t+1}(w^t)$ if demand follows a negative exponential distribution with known parameter λ . Based on Louveaux [1978], extend the *MQSP* to the piecewise convex case, then solve the problem with $\lambda = 0.01$ and 0.02 for the two periods.

6.4 Lagrangian-Based Methods for Multiple Stages

The general goal in Lagrangian methods as in Section 5.8 is to relax a difficult constraint and place it in the objective to obtain a more efficient subproblem to solve. In stochastic programming, candidate constraints to relax include those that enforce nonanticipativity when the formulation imposes this restriction explicitly as in the progressive hedging algorithm (PHA). PHA is easily adapted for multiple stages by simply defining the projection, Π , to project onto the space of nonanticipative solutions by defining it as the conditional expectation of all solutions at time t that correspond to the same history up to t .

The main subproblem for the H -period case is a direct extension of (5.8.10) as follows.

$$\begin{aligned} \inf z = & \sum_{k=1}^K p_k [f^0(x_0, x_k^1) + \sum_{t=1}^H f^t(x_k^t, x_k^{t+1}, k) + \rho_k^{v,T}(x_k - \hat{x}^v) + r/2 \|x_k - \hat{x}^v\|^2] \\ \text{s. t. } & g_i^0(x_0, x_k^1) \leq 0, \quad i = 1, \dots, m_1, \quad k = 1, \dots, K, \\ & g_i^t(x_k^t, x_k^{t+1}, k) \leq 0, \quad i = 1, \dots, m_t; t = 1, \dots, H, \quad k = 1, \dots, K, \end{aligned} \quad (4.1)$$

where x_0 represents given initial conditions.

This formulation leads then to the PHA for multistage problems.

Multistage Progressive Hedging Algorithm

Step 0. Suppose some nonanticipative $x^0 = (x_k^t, k = 1, \dots, K; t = 1, \dots, H)$, $\hat{x}^0 = x^0$, initial multiplier ρ^0 , and $r > 0$. Let $v = 0$. Go to Step 1.

Step 1. Let (x_k^{v+1}) for $k = 1, \dots, K$ solve (4.1). Let $\hat{x}^{v+1} = \Pi(x^{v+1})$, so that $\hat{x}_k^{v+1}(i) = \hat{x}_{k'}^{v+1}(i)$ in all components i corresponding to decisions x^t at time t whenever k and k' share the same history until time t .

Step 2. Let $\rho^{v+1} = \rho^v + r(x^{v+1,k} - \hat{x}^{v+1})$. If $\hat{x}^{v+1} = \hat{x}^v$ and $\rho^{v+1} = \rho^v$, then, stop; \hat{x}^v and ρ^v are optimal. Otherwise, let $v = v + 1$ and go to Step 1.

To see how the algorithm applies to multiple stages, consider an extended version of Example 3 in Chapter 5. Suppose a three-stage example with the same returns on investments A and B in each period as in that example, with a goal of achieving \$55,000 at the start of the third period, and quadratic penalty for missing the goal as before. Suppose the initial solution corresponds to equal investments in the two assets without re-balancing after the first period. With four future scenarios possible, that yields $x^0 = (x_1^1, x_2^1, x_3^1, x_4^1, x_1^2, x_2^2, x_3^2, x_4^2) = ((5, 5), (5, 5), (5, 5), (5, 5), (5, 10), (5, 10), (20, 15), (20, 15))$. The first steps appear below.

Iteration 0:

Step 0. Begin with a multiplier vector of $\rho^0 = 0$, and let $\hat{x}^0 = ((5, 5), (5, 5), (5, 10), (20, 15))$. Let $r = 1$.

Step 1. We wish to solve:

$$\begin{aligned} \min(1/2) [\sum_{k=1}^4 y_k^2 + (x_{kA}^1 - 5)^2 + (x_{kB}^1 - 5)^2 + (x_{kA}^2 - 5(1 + 3 \cdot \mathbf{1}_{k=3,4}))^2 + \\ (x_{kB}^2 - 5(2 + \mathbf{1}_{k=3,4}))^2] \end{aligned} \quad (4.2)$$

$$\begin{aligned}
\text{s. t.} \quad & x_{kA}^1 + x_{kB}^1 \leq 10, k = 1, \dots, 4; \\
& (1 + 3 \cdot \mathbf{1}_{k=3,4})x_{kA}^1 + (2 + \mathbf{1}_{k=3,4})x_{kB}^1 - x_{kA}^2 - x_{kB}^2 = 0, k = 1, \dots, 4; \\
& (1 + 3 \cdot \mathbf{1}_{k=2,4})x_{kA}^2 + (2 + \mathbf{1}_{k=2,4})x_{kB}^2 - y_k \geq 55, k = 1, \dots, 4; \\
& x_{kA}^1, x_{kB}^1, x_{kA}^2, x_{kB}^2, y_k \geq 0, k = 1, \dots, 4,
\end{aligned} \tag{4.3}$$

where $\mathbf{1}_{k=X}$ has value 1 when k is in X and is 0 otherwise.

As in the two-stage case, this problem again separates into subproblems for each scenario k . The solution in this case is

$$\begin{aligned}
x^1 = & ((0, 10), (3.91, 6.09), (6.25, 3.75), (5, 5), (0, 20), \\
& (5.94, 10.16), (19.6, 16.7), (20, 15)),
\end{aligned}$$

which then yields

$$\begin{aligned}
\hat{x}^1 = & ((3.79, 6.23), (3.79, 6.23), (3.79, 6.23), (3.79, 6.23), \\
& (2.97, 15.08), (2.97, 15.08), (19.8, 15.8), (19.8, 15.8)).
\end{aligned}$$

Step 2. We then have

$$\begin{aligned}
\rho^1 &= 0 + 1(x^1 - \hat{x}^1) \\
&= ((-3.79, 3.79), (0.12, -0.12), (2.46, -2.46), (1.21, -1.21), \\
&\quad (-2.97, 4.92), (2.97, -4.92), (-0.21, 0.83), (0.21, -0.83)),
\end{aligned}$$

(where we use the same groupings of variables to show the relationship to x^1) and return for the next iteration. Exercise 1 asks you to complete the iterations until convergence to within 0.01 in each component of the iterates.

As discussed in Chapter 5, PHA is particularly well-adapted for problems, such as networks, where maintaining the original problem structure in each scenario problem leads to efficiency (see Mulvey and Vladimirou [1991b]). Although PHA is not necessarily convergent for stochastic integer problems, it and other Lagrangian methods can be used to solve the convex relaxation with additional branching to obtain integer solutions. This approach has been effective for *unit commitment* problems for planning electric power generation (see Takriti and Birge [2000a]). The structure in these problems also allows for close approximations of the integer program with the continuous-relaxation solution for large-scale problems with many resources (see Takriti and Birge [2000b]).

A different approach for multistage problems that performs well for nonlinear problems is a method from Mulvey and Ruszczyński [1995] called diagonal quadratic approximation (DQA). This method approximates quadratic penalty terms in a Lagrangian type of objective so that each subproblem is again easy to solve and can be spread across a wide array of distributed processors. DQA requires few assumptions on the problem structure and can be competitive also for linear problems.

Exercises

1. Complete the PHA iterations for the three-period version of Example 5.3 until convergence within 0.01 in every component of \hat{x}^v and ρ^v .
2. Show how to implement PHA on Example 1. Follow three iterations of the algorithm.

Chapter 7

Stochastic Integer Programs

As seen in Section 3.3, properties of stochastic integer programs are scarce. The absence of general efficient methods reflects this difficulty. Several techniques have been proposed in the recent years. As in deterministic integer programs, many of them are based on either a branching scheme or a reformulation scheme. The reader unfamiliar with either concept will find a brief introduction in the Short Reviews, Section 7.8 of this chapter. Section 7.1 recalls the links with the continuous case. Sections 7.2 and 7.3 consider two solution procedures that use a branching scheme. Section 7.4 considers the use of reformulation of the second-stage constraints by disjunctive cuts. Sections 7.5 to 7.7 consider simple integer recourse, feasibility cuts and the decomposition of the extensive form. Approximations can also be used, as indicated at the end of Section 9.5. Note also that Sections 7.2 to 7.7 can be read independently of each other.

7.1 Stochastic Integer Programs and LP-Relaxation

Consider the definition of a stochastic integer program, as in Section 3.3,

$$\begin{aligned} (\text{SIP}) \quad & \min_{x \in X} c^T x + \mathbb{E}_{\xi} \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x, y \in Y\} \\ & \text{s. t. } Ax = b, \end{aligned} \tag{1.1}$$

where the definitions of c , b , ξ , A , W , T , q and h are as before.

In this chapter, Y always contains integrality restrictions on y . In some cases, X also contains integrality restrictions on x . The second-stage program is

$$Q(x, \xi) = \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x, y \in Y\}, \tag{1.2}$$

and its expectation $\mathcal{Q}(x) = \mathbb{E}_{\xi} Q(x, \xi)$ can be used to obtain a deterministic equivalent program

$$(DEP) \quad \begin{aligned} & \min_{x \in X} c^T x + \mathcal{Q}(x) \\ & \text{s. t. } Ax = b . \end{aligned}$$

Even if it does look very similar to the deterministic equivalent program in the continuous case, we know from Section 3.3 that $\mathcal{Q}(x)$ does not possess appropriate properties for an easy solution procedure. Moreover, the computation of $\mathcal{Q}(x)$ for a given x is usually a much more difficult task than in the continuous case. In the case of a discrete random variable, assuming the solution of (1.2) has been obtained for one realization of ξ does not help solving the same program for another value of ξ . Indeed, the integrality restrictions imply that the usual forms of duality are lost. In the continuous case, a few dual iterations generally suffice to find the solution of (1.2) from one ξ to the other. In the integer case, (1.2) must typically be restarted from scratch for each ξ . Thus, finding $\mathcal{Q}(x)$ for a given x may be a challenge in itself. Yet, this evaluation is unavoidable (at least a few times) and the assumption is made that, for fixed x , $\mathcal{Q}(x)$ is computable in a finite number of steps.

Now, let \bar{Y} be the continuous or LP-relaxation of Y . For instance, if one considers a stochastic program with a binary second-stage, then $Y = \{y \mid y \in \{0, 1\}^{m_2}\}$ and $\bar{Y} = \{y \mid 0 \leq y \leq e\}$, where $e^T = (1, \dots, 1)$ is the unit vector of dimension m_2 . Similarly, let \bar{X} be the LP-relaxation of X . We introduce the following notation for the LP-relaxation of the second-stage program

$$C(x, \xi) = \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x, y \in \bar{Y}\}, \quad (1.3)$$

with

$$C(x) = E_\xi C(x, \xi). \quad (1.4)$$

with the usual conventions for infeasible and unbounded cases.

Proposition 1. *L-shaped optimality cuts of the form (5.1.4) calculated on the continuous relaxation (1.3)–(1.4) are valid cuts for (SIP).*

Proof: By definition of \bar{Y} , $C(x, \xi) \leq Q(x, \xi)$ holds for all x and ξ , where this result also holds if some problem is unbounded or infeasible. Taking expectations implies $C(x) \leq \mathcal{Q}(x)$. Following the proof in Section 5.1, an L-Shaped optimality cut calculated on (1.3)–(1.4) is an expression of the form $E_\xi(\pi^\nu(h - Tx)) = e_l - E_l x \leq C(x)$, where π^ν represent the optimal simplex multipliers for the second-stage programs at iteration ν , i.e. for some $x = x^\nu$. The result then follows from $e_l - E_l x \leq C(x) \leq \mathcal{Q}(x) \leq \theta$. \square

Based on this observation, solving (SIP) usually starts from solving its LP-relaxation (the program where X is replaced by \bar{X} and Y by \bar{Y}). This can typically be done by way of the L-Shaped method and results in a program of the form

$$(CP) \quad \min c^T x + \theta \quad (1.5)$$

$$\text{s. t.} \quad Ax = b, \quad (1.6)$$

$$D_l x \geq d_l, \quad l = 1, \dots, r, \quad (1.7)$$

$$E_l x + \theta \geq e_l , \quad l = 1, \dots, s , \quad (1.8)$$

$$x \geq 0 , \quad \theta \in \Re . \quad (1.9)$$

where (CP) stands for “Current Problem.”

Branching schemes typically consist of solving a sequence of (CP), each one being defined on a different subspace of the first-stage feasibility set. Finiteness of the procedure comes from the finite number of possible subspaces that are created. *Reformulation* means that optimality cuts in (CP) are reformulated to take integrality restrictions in the second-stage into account. Finiteness of the procedure comes from the limited number of possible reformulations, combined or not with a second-stage branching scheme.

7.2 First-stage Binary Variables

When the first-stage variables are binary variables, it is possible to derive specific optimality cuts in order to obtain a finitely convergent algorithm based on a branching system. The proposed method easily extends to the case of mixed first-stage variables, provided the tender variables are binary. We assume the existence of a lower bound on $\mathcal{Q}(x)$.

Assumption 2. *There exists a finite lower bound L satisfying*

$$L \leq \min_x \{\mathcal{Q}(x) \mid Ax = b, x \in X\}.$$

In Assumption 2, no requirement is made that the bound L should be tight, although it is desirable to have L as large as possible. Examples of how to find L will be given later.

Proposition 3. *Let $x_i = 1$, $i \in S$, and $x_i = 0$, $i \notin S$ be some first-stage feasible solution. Let $q_S = \mathcal{Q}(x)$ be the corresponding recourse function value. The optimality cut*

$$\theta \geq (q_S - L) \left(\sum_{i \in S} x_i - \sum_{i \notin S} x_i \right) - (q_S - L)(|S| - 1) + L \quad (2.1)$$

is valid.

Proof: Define

$$\delta(x, S) = \sum_{i \in S} x_i - \sum_{i \notin S} x_i . \quad (2.2)$$

Now, $\delta(x, S)$ is always less than or equal to $|S|$. It is equal to $|S|$ only if $x_i = 1$, $i \in S$, and $x_i = 0$, $i \notin S$. In that case, the right-hand side of (2.1) takes the value q_S and the constraint $\theta \geq q_S$ is valid as $\mathcal{Q}(x) = q_S$. In all other cases, $\delta(x, S)$ is smaller than or equal to $|S| - 1$, which implies that the right-hand side of (2.1) takes

a value smaller than or equal to L , which by Assumption 2 is a valid lower bound on $\mathcal{Q}(x)$ for all feasible x .

Readers more familiar with geometrical representations may see (2.1) as a half-space, in the (δ, θ) space, situated above a line going through the two points $(|S|, q_S)$ and $(|S| - 1, L)$. \square

Example 1

Consider a two-stage program, where the second stage is given by

$$\begin{aligned} & \min -2y_1 - 3y_2, \\ & \text{s. t. } y_1 + 2y_2 \leq \xi_1 - x_1, \\ & \quad y_1 \leq \xi_2 - x_2, \\ & \quad y \geq 0, \text{ integer.} \end{aligned}$$

Assume $\xi = (2, 2)^T$ or $(4, 3)^T$ with equal probability $1/2$ each. Find a lower bound L on $\mathcal{Q}(x)$ and derive a cut of type (2.1) if the current iterate point is $x = (0, 1)^T$.

1. The second stage is equivalent to: $-\max 2y_1 + 3y_2$. Because the first-stage decisions are binary, largest values of y are obtained with $x = (0, 0)^T$. To obtain a lower bound L , we simply drop the requirement that y should be integer and solve

$$\begin{aligned} & \min -2y_1 - 3y_2 \\ & \text{s. t. } y_1 + 2y_2 \leq \xi_1, \\ & \quad y_1 \leq \xi_2, \\ & \quad y_1, y_2 \geq 0. \end{aligned}$$

For $\xi = (2, 2)^T$, the solution is $y = (2, 0)^T$ and $Q(x, \xi) = -4$, while for $\xi = (4, 3)^T$, the solution is $y = (3, 0.5)^T$ with $Q(x, \xi) = -7.5$. This results in $L = 0.5 * (-4) + 0.5 * (-7.5) = -5.75$. (Alternatively, in this simple example, we may have maintained the requirement that y is integer and obtained the better bound $L = -5.5$. In general, this approach seems more difficult to implement. We continue here with $L = -5.75$.)

2. Here, $\delta(x, S) = x_2 - x_1$ because $x_1 = 0$ and $x_2 = 1$. For $\xi = (2, 2)^T$, the second stage becomes

$$\begin{aligned} & \min -2y_1 - 3y_2 \\ & \text{s. t. } y_1 + 2y_2 \leq 2, \\ & \quad y_1 \leq 1, \\ & \quad y_1, y_2 \geq 0, \text{ integer,} \end{aligned}$$

with solution $y = (0, 1)^T$ and $Q(x, \xi) = -3$. For $\xi = (4, 3)^T$, the second stage becomes

$$\begin{aligned} & \min -2y_1 - 3y_2 \\ \text{s. t. } & y_1 + 2y_2 \leq 4, \\ & y_1 \leq 2, \\ & y_1, y_2 \geq 0, \text{ integer}, \end{aligned}$$

with solution $y = (2, 1)^T$ and $Q(x, \xi) = -7$. We conclude that $q_S = -5$ and that the optimality cut (3.1) reads

$$\theta \geq 0.75(x_2 - x_1) - 5.75.$$

The integer L -shaped method was first proposed by Laporte and Louveaux [1993]. We now present a simplified version for the case of relatively complete recourse. If needed, feasibility cuts may be added at Step 3, using the methods of Section 7.6 for example.

Integer L -shaped Method

Step 0. Set $s = v = 0$, $\bar{z} = \infty$. The value of θ is set to $-\infty$ or to an appropriate lower bound and is ignored in the computation. A list is created that contains only a single pendant node corresponding to the initial subproblem.

Step 1. Set $v = v + 1$. Select some pendant node in the list as the current problem; if none exists, stop.

Step 2. Solve the current problem. If the current problem has no feasible solution, fathom the current node; go to Step 1. Otherwise, let (x^v, θ^v) be an optimal solution.

Step 3. If $c^T x^v + \theta^v > \bar{z}$, fathom the current problem and go to Step 1.

Step 4. Check for integrality restrictions. If a restriction is violated, create two new branches following the usual branch and cut procedure. Append the new nodes to the list of pendant nodes, and go to Step 1.

Step 5. Compute $\mathcal{Q}(x^v)$ and $z^v = c^T x^v + \mathcal{Q}(x^v)$. If $z^v < \bar{z}$, update $\bar{z} = z^v$.

Step 6. If $\theta^v \geq \mathcal{Q}(x^v)$, then fathom the current node and return to Step 1. Otherwise, impose one optimality cut (2.1) with $q_S = \mathcal{Q}(x^v)$, set $s = s + 1$, and return to Step 2.

Proposition 4. *Under Assumption 2, the integer L -shaped method yields an optimal solution of a (SIP) with relatively complete recourse and first-stage binary variables (when one exists) in a finite number of steps.*

Proof: Finiteness comes from the fact that there are at most 2^{n_1} different first-stage solutions. If not eliminated at Step 3, the current solution is eliminated at Step 6, either by fathoming or by adding the optimality cut (2.1). All other steps are finite. \square

In the rest of this section, we first show how the optimality cut (2.1) can be improved when more information is available on $\mathcal{Q}(x)$. We then illustrate how the integer L -shaped method can be implemented in a specific application (routing problems). Both subsections can be considered independently.

a. Improved optimality cuts

Define the set $N(s, S)$ of so-called s -neighbors of S as the set of solutions $\{x \mid Ax = b, x \in X, \delta(x, S) = |S| - s\}$, where $\delta(x, S)$ is as in (2.2). Let $\lambda(s, S) \leq \min_{x \in N(s, S)} \mathcal{Q}(x)$, $s = 0, \dots, |S|$ with $\lambda(0, S) = q_S$.

Proposition 5. Let $x_i = 1$, $i \in S$, $x_i = 0$, $i \notin S$ be some solution with $q_S = \mathcal{Q}(x)$. Define $a = \max \{q_S - \lambda(1, S), (q_S - L)/2\}$. Then

$$\theta \geq a \left(\sum_{i \in S} x_i - \sum_{i \notin S} x_i \right) + q_S - a|S| \quad (2.3)$$

is a valid optimality cut.

Proof: For an s -neighbor, the right-hand side of (2.3) is equal to $q_S - as$. This is a valid lower bound on $\mathcal{Q}(x)$. This is obvious for $s = 0$. When $s = 1$, $q_S - a$ is, by construction, bounded above by $q_S - (q_S - \lambda(1, S)) = \lambda(1, S)$, which by definition is a lower bound on 1-neighbors. When $s = 2$, $q_S - 2a \leq q_S - 2(q_S - L)/2 = L$. Finally, for $s \geq 3$, $q_S - as \leq q_S - 2a$, because $a \geq 0$. Hence, $q_S - as \leq L$. Convergence is again guaranteed by $\theta \geq q_S$ when $\delta(x, S) = |S|$ and (2.3) improves on (2.1) for all 1-neighbors. The reader more familiar with geometrical representations may now see (2.3) as a half-space in the (δ, θ) space, situated above a line going through the two points $(|S|, q_S)$ and $(|S| - 1, \lambda(1, S))$ when $a = q_S - \lambda(1, S)$, or the two points $(|S|, q_S)$ and $(|S| - 2, L)$ when $a = (q_S - L)/2$. \square

A further improvement for s -neighbors is sometimes possible.

Proposition 6. Let $x_i = 1$, $i \in S$, $x_i = 0$, $i \notin S$ be some solution with $q_S = \mathcal{Q}(x)$. Let $1 \leq t \leq |S|$ be some integer. Then (2.3) holds with

$$a = \max \left\{ \max_{s \leq t} (q_S - \lambda(s, S))/s; (q_S - L)/(t + 1) \right\}. \quad (2.4)$$

Proof: As before, for an s -neighbor, the right-hand side of (2.3) is $q_S - as$. By (2.4), $as \geq q_S - \lambda(s, S)$, for all $s \leq t$. Thus, $q_S - as \leq \lambda(s, S)$, which, by definition,

is a lower bound on $\mathcal{Q}(x)$ for all s -neighbors. When $s \geq t + 1$, $q_S - as \leq L$, and (2.3) remains valid. \square

As computing $\lambda(s, S)$ for $s \leq t$ with t large may prove difficult, the following proposition is sometimes useful.

Proposition 7. Define $\lambda(0, S) = q_S$. Assume $q_S > \lambda(1, S)$. Then, if $\lambda(s-1, S) - \lambda(s, S)$ is nonincreasing in s for all $1 \leq s \leq \lfloor (q_S - L)/(q_S - \lambda(1, S)) \rfloor$, (2.3) holds with $a = q_S - \lambda(1, S)$.

Proof: We have to show that in applying Proposition 6, the maximum in (2.4) is obtained for $q_S - \lambda(1, S)$. Let $t = \lfloor (q_S - L)/(q_S - \lambda(1, S)) \rfloor$. For $s \leq t$, $q_S - \lambda(s, S) = \sum_{i=1}^s (\lambda(i-1, S) - \lambda(i, S))$. By assumption, each term of the sum is smaller than the first term of the sum, i.e., $\lambda(0, S) - \lambda(1, S) = q_S - \lambda(1, S)$ so the total is less than s times this first term. By definition of t , we have $t+1 \geq (q_S - L)/(q_S - \lambda(1, S))$, or $q_S - \lambda(1, S) \geq (q_S - L)/(t+1)$. \square

Clearly, much of the implementation is problem-dependent. We illustrate here the use of these propositions in one example.

Example 2

Let $i = 1, \dots, n$ denote n inputs and $j = 1, \dots, m$ denote m outputs. Each input can be used to produce various outputs. First-stage decisions are represented by binary variables x_{ij} with costs c_{ij} and are equal to 1 if i is used to produce j and equal to 0 otherwise. If input i is used for at least one output, some fixed cost f_i is paid. To this end, the auxiliary variable z_i is defined equal to 1 if input i is used and 0 otherwise. The level of output j obtained when $x_{ij} = 1$ is a non-negative random variable ξ_{ij} . A penalty r_j is incurred whenever the level of output j falls below a required threshold d_j . This is represented by the second-stage variable y_j^ξ taking the value 1.

The problem can be defined as:

$$\min \sum_{i=1}^n f_i z_i + \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} + E_\xi \left(\sum_{j=1}^m r_j y_j^\xi \right) \quad (2.5)$$

$$\text{s. t.} \quad x_{ij} \leq z_i, \quad i = 1, \dots, n, \quad j = 1, \dots, m, \quad (2.6)$$

$$\sum_{i=1}^n \xi_{ij} x_{ij} + d_j y_j^\xi \geq d_j, \quad j = 1, \dots, m, \quad (2.7)$$

$$x_{ij}, z_i, y_j^\xi \in \{0, 1\}, \quad i = 1, \dots, n, \quad j = 1, \dots, m, \quad (2.8)$$

where, in practice, the x_{ij} variables are only defined for the possible combinations of inputs and outputs. In this problem, the second-stage recourse function only depends on the x decisions so that the z variables may be left over in our analysis of

optimality cuts. Moreover, the second stage is easily computed as

$$\mathcal{Q}(x) = \sum_{j=1}^m r_j P \left(\sum_{i \in S(j)} \xi_{ij} < d_j \right), \quad (2.9)$$

where

$$S(j) = \{i \mid x_{ij} = 1\}.$$

Let $S = \cup_{j=1}^m \{(i, j) \mid i \in S(j)\}$. To apply the propositions, we search for lower bounds, $\lambda(s, S)$, on the recourse function for all s -neighbors. To bound $q_S - \lambda(1, S)$, observe that 1-neighbors can be obtained in two distinct ways. The first way is to have one x_{ij} , with $(i, j) \in S$, going from one to zero and all other x_{ij} being unchanged. This implies for that particular j that, in (2.9), $P(\sum_{i \in S(j)} \xi_{ij} < d_j)$ increases in the neighboring solution, as $S(j)$ would contain one fewer term. Thus, for this type of 1-neighbor, $\mathcal{Q}(x)$ is increased.

The second way is to have one x_{ij} , with (i, j) not in S , going from zero to one, all other x_{ij} being unchanged. For that particular j , $P(\sum_{i \in S(j)} \xi_{ij} < d_j)$ decreases in the neighboring solution. To bound the decrease of $\mathcal{Q}(x)$, we simply assume $P(\sum_{i \in S(j)} \xi_{ij} < d_j)$ vanishes so that

$$q_S - \lambda(1, S) \leq \max_j \left\{ r_j P \left(\sum_{i \in S(j)} \xi_{ij} < d_j \right) \right\}. \quad (2.10)$$

Also observe that in this example, Proposition 7 applies. Indeed, $q_S - \lambda(s, S)$ can be taken as the sum of the s largest values of $\{r_j P(\sum_{i \in S(j)} \xi_{ij} < d_j)\}$. It follows that $\lambda(s-1, S) - \lambda(s, S)$ is nonincreasing in s .

Moreover, in this example, we can also find lower bounding functionals. By looking at (2.7), the optimal solution of the continuous relaxation of the second stage is easily seen to be

$$y_j^\xi = r_j \left(d_j - \sum_{i=1}^n \xi_{ij} x_{ij} \right)^+ / d_j, \quad j = 1, \dots, m,$$

and therefore,

$$C(x) = E_\xi \left[\sum_j r_j \left(d_j - \sum_{i=1}^n \xi_{ij} x_{ij} \right)^+ / d_j \right]. \quad (2.11)$$

In fact, we just need to compute

$$C(x) = E_\xi \sum_j r_j \left(d_j - \sum_{i \in S(j)} \xi_{ij} \right)^+ / d_j. \quad (2.12)$$

From (2.11), we may immediately apply Proposition 1 as

$$\theta \geq q_S + \sum_{ij \in S} a_{ij}(x_{ij} - 1) + \sum_{ij \notin S} a_{ij}x_{ij} \quad (2.13)$$

with

$$\begin{aligned} a_{ij} &= -r_j/d_j E_{\xi} \left[\xi_{ij} P \left(\sum_{l \in S(j) \setminus i} \xi_{lj} \leq d_j - \xi_{ij} \right) \right], & i \in S(j), \\ a_{ij} &= -r_j/d_j E_{\xi} \left[\xi_{ij} P \left(\sum_{l \in S(j)} \xi_{lj} < d_j \right) \right], & i \notin S(j) \end{aligned}$$

and

$$q_S = C(x) \quad \text{as in (2.12).}$$

Example 2 (continued)

We take Example 2 and consider the following numerical data. Let $n = 4$, $m = 6$, $f_i = 10$, for all i , $r_j = 40$ for all j . Let the c_{ij} coefficients take values between 5 and 15 as follows:

$$\begin{array}{ccccccc} j = & 1 & 2 & 3 & 4 & 5 & 6 \\ i = & 1 & 10 & 12 & 8 & 6 & 5 & 14 \\ & 2 & 8 & 5 & 10 & 15 & 9 & 12 \\ & 3 & 7 & 14 & 4 & 11 & 15 & 8 \\ & 4 & 5 & 8 & 12 & 10 & 10 & 10. \end{array}$$

Assume the ξ_{ij} are independent Poisson random variables with parameters

$$\begin{array}{ccccccc} j = & 1 & 2 & 3 & 4 & 5 & 6 \\ i = & 1 & 4 & 4 & 5 & 3 & 3 & 8 \\ & 2 & 5 & 2 & 4 & 8 & 5 & 6 \\ & 3 & 2 & 8 & 3 & 4 & 7 & 5 \\ & 4 & 3 & 5 & 6 & 4 & 6 & 5 \end{array}$$

and, finally, let the demands d_j be given by

$$\begin{array}{ccccccc} j = & 1 & 2 & 3 & 4 & 5 & 6 \\ d_j & 8 & 4 & 6 & 3 & 5 & 8. \end{array}$$

As already said, we may apply Proposition 7 to this example. A second possibility is to use the separability of $\mathcal{Q}(x)$ as

$$\mathcal{Q}(x) = \sum_{j=1}^m \mathcal{Q}_j(x) \quad (2.14)$$

with

$$\mathcal{Q}_j(x) = r_j P \left(\sum_{i \in S(j)} \xi_{ij} < d_j \right). \quad (2.16)$$

Bounding each $\mathcal{Q}_j(x)$ separately, we define

$$\theta = \sum_{j=1}^m \theta_j \quad (2.17)$$

and use Propositions 6 or 7 to define a valid set of cuts for each θ_j separately. Indeed, for one particular j , we have

$$\theta_j = r_j P \left(\sum_{i \in S(j)} \xi_{ij} < d_j \right) \quad (2.18)$$

and

$$\lambda_j(1, S) = r_j \min_{t \notin S(j)} P \left(\sum_{i \in S(j)} \xi_{ij} + \xi_{tj} < d_j \right), \quad (2.19)$$

where $\lambda_j(1, S)$ denotes a lower bound on $\mathcal{Q}_j(x)$ for 1-neighbors of the current solution obtained by changing x_{ij} s for that particular j only. Note that in practice finding t is rather easy. Indeed, because all random variables are independent Poisson, t is simply given by the random variable ξ_{tj} , $t \notin S(j)$, with the largest parameter value.

We illustrate the generation of cuts for $j = 1$. First, a lower bound is obtained by letting $x_{i1} = 1$, for all i . This gives $L_1 = 1.265$.

Assume a starting solution $x_{ij} = 0$, all i, j . For $j = 1$, the probability in the right-hand side of (2.16) is 1. Thus, $\mathcal{Q}_1(x) = r_1 = 40$. Cut (2.3) becomes $\theta_1 \geq 40 - 19.368(x_{11} + x_{21} + x_{31} + x_{41})$ with the coefficient $a = 19.368$ obtained from $(q_{S,1} - L_1)/2$, where $q_{S,1}$ is the notation for the value of $\mathcal{Q}_1(x)$. The continuous cut (2.13) is

$$\theta_1 \geq 40 - 20x_{11} - 25x_{21} - 10x_{31} - 15x_{41}.$$

The next iterate point is, e.g., $x_{11} = 1$, $x_{21} = 0$, $x_{31} = 0$, $x_{41} = 1$. Cut (2.3) becomes $\theta_1 \geq -16.788 + 20.368(x_{11} - x_{21} - x_{31} + x_{41})$ with the coefficient $a = 20.368$ now obtained from $(q_{S,1} - \lambda_1(1, S))$ while the continuous cut (2.13) is

$$\theta_1 \geq 29.164 - 11.974x_{11} - 14.968x_{21} - 5.987x_{31} - 8.981x_{41}.$$

Cut (2.3) is stronger than (2.13) at the current iterate point with value 23.948 instead of 8.309. Also, as the coefficient a comes from $(q_{S,1} - \lambda_1(1, S))$ and $\lambda_1(1, S)$ is obtained when x_{21} becomes 1, (2.3) gives an exact bound on the solution $x_{11} = 1$, $x_{21} = 1$, $x_{31} = 0$, $x_{41} = 1$. It provides a nontrivial but nonbinding bound for other cases, such as $x_{11} = 0$, $x_{21} = x_{31} = x_{41} = 1$. On the other hand, (2.13)

provides a nontrivial (but nonbinding) bound for some cases such as $x_{11} = 0$, $x_{21} = 1$, $x_{31} = 1$, $x_{41} = 0$, where (2.3) does not.

The algorithm for the full example with six outputs was simulated by adding cuts each time a new iterate point was found, then restarting the branch and bound. Cuts (2.3) and (2.13) were added each time the amount of violation exceeded 0.1. The number of iterate points is dependent on the strategies used in the branch and bound. For this example, the largest number of iterate points was 21. In that case, the mean number of cuts per output was 6.833 cuts of type (2.13) and 2.5 cuts (2.3). As extreme cases, 10 improved optimality cuts were imposed for Output 1 and only 4 for Output 2, while 4 continuous cuts were imposed for Output 3 and only 1 for Output 5.

The optimal solution is $x_{11} = x_{13} = x_{15} = x_{16} = x_{21} = x_{22} = x_{24} = x_{41} = x_{42} = x_{43} = x_{45} = x_{46} = 1$; all other x_{ij} s are zero with first-stage cost 140 and penalty 13.26, for a total of 153.26. It strongly differs from the solution of the deterministic problem where outputs equal expected values: $x_{11} = x_{12} = x_{13} = x_{14} = x_{16} = x_{21} = x_{23} = x_{25} = 1$ with first-stage cost 97. The reason is that in the stochastic case, even if the expected output exceeds demand, the probability that the demand is not met is nonzero. In fact, the solution of the deterministic problem has a penalty of 87.59 for a total cost of 184.59 and a VSS of 31.33.

b. Example with continuous random variables

Consider the vehicle routing problem of Section 1.6. Assume now there are n clients, each having an unknown demand. We are given a graph $G = (V, E)$ which consists of a set V of vertices (or nodes) and a set E of edges (or arcs). Here, the nodes correspond to the set of clients plus the depot $V = \{0, 1, 2, \dots, n, 0\}$ where 0 is the depot. Arc (i, j) corresponds to traveling from node i to node j . Arcs may be traveled in either direction, with a cost $c_{ij} = c_{ji}$. The graph is complete (the vehicle can travel from any point, client or depot, to another).

Each client i has a random demand ξ_i . This demand is not known when the tour starts. It becomes known only when the vehicle arrives at the client. The sum of the demands of a group of clients is a random variable. It is assumed that the cumulative distribution function of the sum is computable. This is the case for discrete random variables with a very small number of realizations or for demands following such distributions as Poisson or normal. The vehicle has a known capacity D . Given that the demands are random, the cumulative demand may at some point exceed the vehicle capacity. This situation is called a *failure*.

The simplest version of the stochastic TSP with random demands consists of finding, in the first-stage, a so-called *a priori route*. This route must be a Hamiltonian tour, in the sense that it starts from the depot, visits all clients exactly once, then returns to the depot. In the second-stage, the route is followed in the prescribed order. In case of failure, the vehicle returns to the depot, unloads and resumes its trip where the failure occurred. We have seen already in Section 1.6 that they are other

strategies, such as preventive returns, that may be more efficient. For simplicity in the presentation, we do not discuss these strategies here.

An a priori route can be represented by a sequence of clients, e.g., $\{0, v_1, v_2, \dots, v_n, 0\}$. Alternatively, let x_{ij} be a binary variable taking the value 1 if arc (i, j) is in the a priori route and 0 otherwise. Then $x = (x_{ij})$ is an a priori route. It is a vector of values for the x_{ij} 's that satisfy the conditions of a Hamiltonian tour. These conditions include the well-known subtour elimination constraints (see, for instance, Wolsey [1998]). We simply represent these conditions as $x \in X$, as we do not explicitly need them in this section. Thus, an a priori route can be represented either as a sequence of clients or as a vector of binary variables. It is easy to go from one representation to the other.

Define $\mathcal{Q}(x)$ to be the expected cost of failures. The problem then consists of finding an a priori route which minimizes $c^T x + \mathcal{Q}(x)$.

To apply the integer L-shaped method to this problem, we need to calculate $\mathcal{Q}(x)$ for a given x . Assume an a priori route $x = \{0, v_1, v_2, \dots, v_n, 0\}$ is given. It can be traveled in two orientations (starting at v_1 and ending at v_n , or the opposite.) We represent by $\mathcal{Q}^\lambda(x)$ the expected penalty for traveling in orientation λ , $\lambda = 1, 2$. Thus, $\mathcal{Q}(x) = \min\{\mathcal{Q}^1(x), \mathcal{Q}^2(x)\}$. Consider orientation 1, starting at v_1 and ending at v_n . Then,

$$\mathcal{Q}^1(x) = \sum_{j=1}^n P\{\text{a failure occurs at } v_j\} \cdot 2c_{j0},$$

where, by abuse of notation, $2c_{j0}$ represents the cost of the return trip from v_j to the depot.

For the sake of simplicity, assume that the probability of exact stockout is negligible. (Exact stockout means that the sum of the demands at a given point exactly coincides with the vehicle capacity.) This is always the case with a continuous random variable. Also assume that the probability of having two failures is negligible. This assumption is reasonable if the vehicle capacity is not too small compared with the total demand.

For a given tour, define the event

$$E_j = \{\text{the sum of demands up to } v_j \text{ exceeds the vehicle capacity}\}.$$

The event $\{\text{a failure occurs at } v_j\}$ corresponds to $E_j \cap \bar{E}_{j-1}$. Now,

$$P(E_j) = P(E_j \cap \bar{E}_{j-1}) + P(E_j \cap E_{j-1}) = P(E_j \cap \bar{E}_{j-1}) + P(E_{j-1}),$$

since E_{j-1} implies E_j . Thus,

$$P(E_j \cap \bar{E}_{j-1}) = P(E_j) - P(E_{j-1})$$

and

$$\begin{aligned}\mathcal{Q}^1(x) &= \sum_{j=1}^n \left[P\left(\sum_{k=1}^j \xi_k > D\right) - P\left(\sum_{k=1}^{j-1} \xi_k > D\right) \right] 2c_{j0} \\ &= \sum_{j=1}^n \left[P\left(\sum_{k=1}^{j-1} \xi_k \leq D\right) - P\left(\sum_{k=1}^j \xi_k \leq D\right) \right] 2c_{j0}.\end{aligned}$$

This expression can be calculated for summable distributions. These include continuous distributions such as normal distributions which are often easier to use than discrete distributions.

Two other aspects may be stressed. First, while $\mathcal{Q}(x)$ can be calculated for a given x as we have seen, expressing $\mathcal{Q}(x)$ as a mathematical program in terms of second stage variables representing the failures is much more difficult. Thus, the methods that we present in Sections 7.3 or 7.4 would be ineffective. Second, a lower bound on $\mathcal{Q}(x)$ is needed for the integer L -shaped method. One such lower bound is proposed as Exercise 4.

This problem has stimulated a stream of research. A first implementation is due to Gendreau, Laporte and Séguin [1995]. Hjörring and Holt [1999] have proposed improved optimality cuts which are valid at fractional solutions. Laporte, Louveaux, and Vanhamme [2002] have extended this approach for the VRP problem with m vehicles of limited capacity. Rei et al. [2009] show how to accelerate Benders decomposition and the integer L -shaped method by local branching techniques. Re-optimization approaches have been studied by Secomandi and Margot [2009]. For specific problem structures, such as in crew scheduling problems, Yen and Birge [2006] show that alternative branching schemes, in that case based on the crews' plane changes, can also lead to efficiencies.

Besides these computational examples, a full characterization of the integer L -shaped method based on general duality theory can be found in Carøe and Tind [1998]. A stochastic version of the branch and cut method based on statistical estimation of the recourse function can be found in Norkin, Ermolieva and Ruszczyński [1998] and Norkin, Pflug and Ruszczyński [1998]. A simple description of the sample average approximation method for the stochastic integer programs is given at the end of Section 9.5.

Exercises

1. Construct the cuts from the integer L -shaped method for Example 1, associated with the point $(0, 1)^T$.
Compare the results by checking the bound on $\theta_1 + \theta_2$ by the integer L -shaped method and the bound in Example 1 on θ by (2.1) for the four possible points, $(0, 0)$, $(0, 1)$, $(1, 0)$, $(1, 1)$ and, for some continuous points, $(1/2, 1/2)$, $(1.2, 0)$, $(0, 1.2)$, for example.
2. Extending (2.19), we obtain

$$\lambda_j(s, S) = r_j P \left(\sum_{i \in S(j)} \xi_{ij} + \sum_{t \in J} \xi_{tj} < d_j \right), \quad (2.20)$$

where J contains the indices of the s pairs ij , $i \notin S(j)$, with largest parameter values. Show that the assumptions of Proposition 6 hold.

3. Indicate why the wait-and-see solution cannot be reasonably computed in Example 2.
4. Consider the TSP with stochastic demands of Section 7.2b. Order the clients in increasing distance from the depot. Examine whether

$$L = \sum_{j=1}^n q_j \cdot c_{j0}$$

is a valid lower bound if q_j is the probability of having at least/exactly j failures.

5. Consider the TSP with stochastic demands of Section 7.2b. Show that, if the demand of the client can be split, having at most one failure corresponds to the total demand being less than or equal to $2D$; then, obtain a condition on D if the demands of the clients are independently distributed according to $N(\mu_i, \sigma_i^2)$ in order to obtain a $1 - \alpha$ probability that the total demand is less than $2D$.

7.3 Second-stage Integer Variables

We consider the case where the second-stage decisions are integer, the random variable has a discrete distribution, the technology matrix T is fixed and the recourse matrices W_k have integer coefficients. The latter can always be achieved by rescaling the second-stage constraints if the initial coefficients are rational.

The value of the second-stage program for one realization ξ_k reads as

$$Q(x, \xi_k) = \min_y \{q_k^T y \mid W_k y \geq h_k - Tx, y \in \mathbb{Z}^{n_2}\}. \quad (3.1)$$

The corresponding value function based on the tenders is

$$\psi(\chi, \xi_k) = \min_y \{q_k^T y \mid W_k y \geq h_k + \chi, y \in \mathbb{Z}^{n_2}\} \quad (3.2)$$

(where, for the sake of presentation in this section, the usual sign of the tender is reversed).

As usual, $\mathcal{Q}(x) = E_\xi Q(x, \xi) = \sum_{k=1}^K p_k Q(x, \xi_k)$. Similarly,

$$\psi(\chi) = E_\xi \psi(\chi, \xi) = \sum_{k=1}^K p_k \psi(\chi, \xi_k). \quad (3.3)$$

For any x , $\mathcal{Q}(x) = \psi(-Tx)$. The classical problem $\min_x \{c^T x + \mathcal{Q}(x) \mid x \in X\}$ is thus equivalent to

$$z^* = \min_{x, \chi} \{c^T x + \psi(\chi) \mid x \in X, \chi = -Tx\}. \quad (3.4)$$

The idea of *branching on tenders* is to partition the space of tenders $\chi = -Tx$ in an orthogonal partition and to use the non-decreasing property of the value function as a function of one component of the tender.

a. Looking in the space of tenders

We first show in an example why it is fruitful to look at the tender space instead of the x space.

Example 3

Consider the following second-stage program for one particular value of ξ

$$\begin{aligned} Q(x, \xi) &= \min 5y_1 + 3y_2 \\ \text{s. t. } &2y_1 + 3y_2 \geq -3 + x_1 + 2x_2, \\ &4y_1 + y_2 \geq -2.4 + x_1 + x_2, \\ &y_1, y_2 \geq 0, \text{ integer.} \end{aligned}$$

Due to the integer y , $Q(x, \xi)$ can only take finitely many different values. In such a small example, it is easy to describe the regions where $Q(x, \xi)$ takes a given value.

- $Q(x, \xi)$ takes the value 0 whenever $y = (0, 0)^T$ is optimal, i.e. in region $R_1 = \{x \mid x_1 + 2x_2 \leq 3, x_1 + x_2 \leq 2.4\}$. This is a convex polyhedron.
- It takes the value 3 whenever $y = (0, 1)^T$ is optimal, i.e. in region $R_2 = \{x \mid x_1 + 2x_2 \leq 6, x_1 + x_2 \leq 3.4\} \setminus R_1$. This is a nonconvex region, due to the $x \notin R_1$ condition.
- Next values are 5, 6 and 8 in regions $R_3 = \{x \mid x_1 + 2x_2 \leq 5\} \setminus R_1 \setminus R_2$, $R_4 = \{x \mid x_1 + x_2 \leq 4.4\} \setminus R_1 \setminus R_2 \setminus R_3$ and $R_5 = \{x \mid x_1 + 2x_2 \leq 8, x_1 + x_2 \leq 7.4\} \setminus R_1 \setminus R_2 \setminus R_3 \setminus R_4$, respectively. And so on. It turns out that R_3 and R_5 are convex but R_4 is not. This is easily seen on a graph of these regions. Figure 1 illustrates the above regions, which are identified by the value taken by $Q(x, \xi)$.

Some of the regions being nonconvex is already a problem. Describing the intersection of the regions for several realizations of ξ is clearly another one. Now, let us look at the same description in the χ space

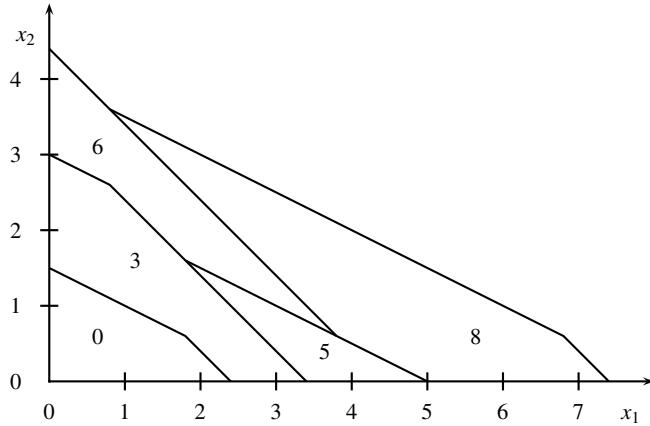


Fig. 1 Value of the second-stage solution in Example 3 in the x -space.

$$\begin{aligned}\Psi(\chi, \xi) = & \min 5y_1 + 3y_2 \\ \text{s. t. } & 2y_1 + 3y_2 \geq -3 + \chi_1, \\ & 4y_1 + y_2 \geq -2.4 + \chi_2, \\ & y_1, y_2 \geq 0, \quad \text{integer},\end{aligned}$$

with $\chi_1 = x_1 + 2x_2$ and $\chi_2 = x_1 + x_2$.

The corresponding regions become $R_1 = \{\chi \mid \chi_1 \leq 3, \chi_2 \leq 2.4\}$, $R_2 = \{x \mid \chi_1 \leq 6, \chi_2 \leq 3.4\} \setminus R_1$, $R_3 = \{\chi \mid \chi_1 \leq 5, \chi_2 \leq 6.4\} \setminus R_1 \setminus R_2$, $R_4 = \{x \mid \chi_1 \leq 9, \chi_2 \leq 4.4\} \setminus R_1 \setminus R_2 \setminus R_3$ and $R_5 = \{\chi \mid \chi_1 \leq 8, \chi_2 \leq 7.4\} \setminus R_1 \setminus R_2 \setminus R_3 \setminus R_4$. Figure 2 shows the regions in the χ space, each region being identified by the value of $\Psi(\cdot)$. Here, R_4 and R_5 are nonconvex. But now, all regions have orthogonal boundaries.

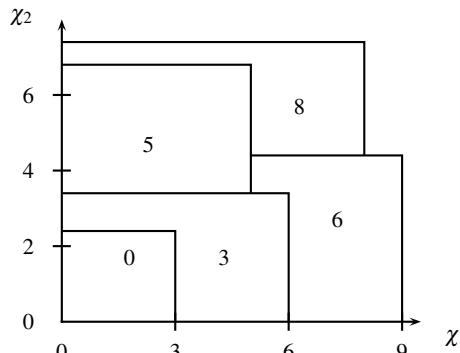


Fig. 2 Value of the second-stage solution in Example 3 in the space of tenders.

To obtain orthogonal boundaries and convex regions, the *branching on tenders* method constructs *hypercubes* of the form $H = \prod_{j=1}^{m_2} (l_j, u_j]$. Here l_j is either a

point of discontinuity of $\Psi(\cdot)$ as a function of χ_j or a lower bound on χ_j . Similarly, u_j is either a point of discontinuity of $\Psi(\cdot)$ as a function of χ_j or an upper bound on χ_j .

In Example 3 with $m_2 = 2$, hypercubes are rectangles. For instance, $(0,6] \times (2.4,6.4]$ is a hypercube since $\Psi(\chi, \xi)$ has a discontinuity point at $\chi_1 = 6$, at $\chi_2 = 2.4$ and at $\chi_2 = 6.4$. Note that this hypercube contains several other discontinuity points of $\Psi(\chi, \xi)$. The *smallest* hypercubes are those where, for all j , l_j and u_j are successive discontinuity points. One such hypercube is $(3,5] \times (2.4,3.4]$ for example. The hypercubes based on discontinuity points of $\Psi(\chi, \xi)$ lead themselves to easy intersections for different realizations of ξ and are also a good way to exploit the property of $\Psi(\chi)$ being nondecreasing as a function of one particular component χ_j .

b. Discontinuity points

Let $\Psi(\chi_j, \xi_k)$ denote $\Psi(\chi, \xi_k)$ as a function of the j -th component of χ , $j = 1, \dots, m_2$.

Proposition 8. For any $k = 1, \dots, K$ and $j = 1, \dots, m_2$, $\Psi(\chi_j, \xi_k)$ is lower semi-continuous and nondecreasing in χ_j . For any $a \in Z$, $\Psi(\chi_j, \xi_k)$ is constant over $(a - h_{kj} - 1, a - h_{kj}]$, for any $k = 1, \dots, K$ and $j = 1, \dots, m_2$ where h_{kj} denotes the j -th component of h_k .

Proof: The first part of the proposition comes from Proposition 3.20. Now, consider the j -th constraint $(W_k y)_j \geq h_{kj} + \chi_j$. As W_k is integral, it implies $(W_k y)_j \geq \lceil h_{kj} + \chi_j \rceil$. Thus $\Psi(\chi_j, \xi_k)$ is constant for all χ_j s.t. $h_{kj} + \chi_j = \lceil h_{kj} + \chi_j \rceil$. Taking $a = \lceil h_{kj} + \chi_j \rceil$ provides the desired result. \square

Consider now $\Psi(\chi) = E_\xi \Psi(\chi, \xi)$ and, as above, let $\Psi(\chi_j)$ denote $\Psi(\chi)$ as a function of the j -th component of χ , $j = 1, \dots, m_2$.

Proposition 9. There exists a finite number $S \geq 1$ of distinct values f_s , $s = 1, \dots, S$ s.t. for any $a \in Z$, $\Psi(\chi_j)$ is constant over $(a + f_s, a + f_{s+1}]$, $s = 1, \dots, S$, where $f_{S+1} = f_1 + 1$.

Proof: Consider a given j . For any $a \in Z$, $\Psi(\chi_j, \xi_k)$ is constant over $(a - h_{kj} - 1, a - h_{kj}]$, for any $k = 1, \dots, K$. Let $f_k = a - h_{kj} - \lfloor (a - h_{kj}) \rfloor$ be the fractional part of $a - h_{kj}$. Let S be the number of different such fractional parts. Clearly $1 \leq S \leq K$. Reordering the f_k 's in increased order yields the desired result. \square

Thus, all discontinuity points of $\Psi(\chi_j)$ are of the form $a + f_{s_j}$, $s_j = 1, \dots, S_j$, $a \in Z$. A special case is $S_j = 1$ when, for instance, h_j only takes on integer values.

Example 3 (continued)

Assume h take the values $(-3, -2.4)^T$, $(-3.8, -2.5)^T$ and $(-2.6, -4.4)^T$ with equal probability $1/3$. For $j = 1$, the fractional values in increasing order are 0, 0.6 and 0.8. For any $a \in Z$, successive discontinuity points exist at a , $a + 0.6$, $a + 0.8$, $a + 1$, and so on. For $j = 2$, the fractional values in increasing order are 0.4 and 0.5 and successive discontinuity points are of the form $a + 0.4$, $a + 0.5$, $a + 1.4$ and so on, for $a \in Z$.

Now consider some particular discontinuity point of $\Psi(\chi_j)$, say l_j . $\Psi(\chi_j)$ is constant over $(l_j, l'_j]$ where l'_j is the next discontinuity point. To know $\Psi(\chi_j)$ over this interval, it suffices to calculate $\Psi(l_j + \varepsilon)$ for some ε . The chosen ε must be large enough to avoid numerical problems but smaller than $l'_j - l_j$. The smallest interval where $\Psi(\chi_j)$ is constant for any j is $\min\{f_{s_j+1} - f_{s_j}, s_j = 1, \dots, S_j, j = 1, \dots, m_2\}$. Thus ε can be any nonzero value strictly smaller than this minimum (for instance half the minimum). In Example 3, the smallest interval is 0.1 between $a + 0.4$ and $a + 0.5$ (for $s_2 = 1$). Thus $\varepsilon = 0.05$ does the job.

c. Algorithm

Current problem

Consider a hypercube $H = \prod_{j=1}^{m_2}(l_j, u_j]$, where for each j , l_j is a point of discontinuity of $\Psi(\chi)$ as a function of χ_j . Define the current problem as

$$\begin{aligned} CP(l, u) = \min c^T x + \theta & \quad (3.5) \\ \text{s. t. } x \in X, \chi = -Tx, l \leq \chi \leq u, \\ \theta \geq \Psi(l + \varepsilon e). \end{aligned}$$

$CP(l, u)$ is a lower bound on $\min_{x, \chi} \{c^T x + \Psi(\chi) \mid x \in X, \chi = -Tx, l \leq \chi \leq u\}$. Indeed, $\Psi(\chi) = \Psi(l + \varepsilon e)$ for all $l \leq \chi \leq u$, if $\Psi(\cdot)$ has no discontinuity points within H . And $\Psi(\chi) \geq \Psi(l + \varepsilon e)$ otherwise (with the inequality being strict if χ_j is larger than at least one discontinuity point of $\Psi(\cdot)$ within H , for at least one j).

The $CP(l, u)$ problem can be strengthened by any lower bounding functionals, such as the Bender's cuts. We now present the *branching on tenders algorithm*, assuming relatively complete recourse. If needed, feasibility cuts may be added at Step 3, using the technique of Section 7.6.

Branching on Tenders Algorithm

Step 0. Set $v = 0$ and $\bar{z} = \infty$. Set $(l, u]$ to any relevant values s.t. $\{\chi \mid l < \chi \leq u\} \supset \{\chi \mid x \in X, \chi = -Tx\}$. A list is created that contains the single hypercube $\prod_{j=1}^{m_2}(l_j, u_j]$, with a dummy lower bound. There is no incumbent solution.

Step 1. Set $v = v + 1$. Select one hypercube in the list (one with the smallest lower bound for example). Remove it from the list. Denote it $H^v = \prod_{j=1}^{m_2}(l_j^v, u_j^v]$. If none exists, stop: the incumbent solution is the optimal solution.

Step 2. Solve the current problem $CP(l^v, u^v)$. If it has no feasible solution, go to Step 1.

Step 3. Let x^v, χ^v be a solution to $CP(l^v, u^v)$. Calculate $z^v = z(x^v, \chi^v)$.

Step 4. (Update and fathom) If $z^v < \bar{z}$, update $\bar{z} = z^v$, let (x^v, χ^v) be the incumbent solution and remove from the list all the hypercubes having a lower bound larger than \bar{z} .

Step 5. (Fathom or Branch) If $CP(l^v, u^v) \geq \bar{z}$, go to Step 1. Find some component j having a discontinuity point of $\Psi(\cdot)$, say δ_j , within (l_j, u_j) . If none exists, go to Step 1. Otherwise, partition H^v in two hypercubes, one having interval $(l_j, \delta_j]$ in the j -th component, the other having interval $(\delta_j, u_j]$ in the j -th component (with the intervals for the other components unchanged). Insert the two hypercubes in the list with a lower bound of $CP(l^v, u^v)$ each. Go to Step 1.

Proposition 10. *The branching on tenders algorithm terminates with a global minimum (when one exists) in a finite number of steps.*

Proof: Assume X contains at least one solution. Partitioning (or branching) occurs at Step 5. This operation is finite if X is compact. Indeed, there can only be a finite number of discontinuity points for each component, thus a finite number of partitions. At each iteration, at least one hypercube is fathomed. Thus, there can only be a finite number of iterations. Now, consider an optimal solution, say x^*, χ^* with objective value z^* and let H^* be the smallest hypercube containing χ^* . This is a hypercube such that $\Psi(\cdot)$ does not contain any discontinuity. Thus, $\Psi(\cdot)$ is constant on H^* and the solution of the LB problem on H^* must be χ^* (or another χ with equal z^* value). Otherwise there would be another χ within H^* with strictly smaller $c^T x + \Psi(\chi)$ value, contradicting the optimality of χ^* . Within the list of hypercubes, there will always be one hypercube containing H^* , unless the optimum is found at step 4, in which case the proposition holds. Along the iterations, the hypercube containing H^* will be partitioned (at most a finite number of times) up to the point where H^* enters the list. When H^* is selected in Step 1, the optimum is found in Step 4. \square

Example 4

Consider the following stochastic integer program

$$\begin{aligned} \min_{x \geq 0} \quad & -2.5x_1 - 2x_2 + E_\xi \min\{4.4y_1 + 3y_2\} \\ \text{s. t. } \quad & 4x_1 + 5x_2 \leq 15, \quad 2y_1 + 3y_2 \geq h_1 + \chi_1, \\ & x_1 + x_2 \geq 1.5, \quad 4y_1 + y_2 \geq h_2 + \chi_2, \\ & \chi_1 = x_1 + 2x_2, \quad y \geq 0, \quad \text{integer}, \\ & \chi_2 = 2x_1 + x_2, \end{aligned}$$

where $h^T = (-2.8, -1.2)$ and $(-2, -3)$ with equal probability $\frac{1}{2}$.

We use the following notation. The list of remaining hypercubes is denoted by Λ . An upper index on a hypercube represents the iteration number, while a lower index represents its place in the list. β_i represents the lower bound associated to a particular hypercube. Thus,

$$\begin{aligned} H^v &= \text{hypercube selected at iteration } v; \\ H_i &= i\text{-th hypercube in the list, with lower bound } \beta_i. \end{aligned}$$

Given the possible values of h , ε can take any value $0 < \varepsilon < 0.2$. We choose $\varepsilon = 0.1$. We use the first-stage feasibility set to find the feasibility intervals $1.5 \leq \chi_1 \leq 6$, $1.5 \leq \chi_2 \leq 7.5$. As the left intervals of hypercubes are open, we subtract ε on the left part to make sure no feasible point is omitted. The initial hypercube is $H_0 = (1.4, 6] \times (1.4, 7.5]$. Set $\bar{z} = 0$ and $v = 0$.

Iteration 1:

Step 1. $v = 1$. Select $H^1 = H_0$. Λ is empty.

Step 2. $l^1 = (1.4, 1.4)^T$ and $u^1 = (6, 7.5)^T$.

Compute $\Psi(l^1 + \varepsilon e) = \Psi(1.5, 1.5)$: for $h = (-2.8, -1.2)^T$, the second-stage solution is $y = (0, 1)^T$ and $\Psi(\chi, \xi) = 3$, for $h = (-2, -3)^T$, it is $y = (0, 0)^T$ with $\Psi(\chi, \xi) = 0$. Taking the expectation, we obtain $\Psi(1.5, 1.5) = 1.5$. The current problem reads as follows:

$$\begin{aligned} CP(l^1, u^1) = \min \quad & -2.5x_1 - 2x_2 + \theta \\ \text{s. t. } \quad & 4x_1 + 5x_2 \leq 15, \quad x_1 + x_2 \geq 1.5, \\ & \chi_1 = x_1 + 2x_2, \quad \chi_2 = 2x_1 + x_2, \\ & 1.5 \leq \chi_1 \leq 6, \quad 1.5 \leq \chi_2 \leq 7.5, \quad x_1, x_2 \geq 0, \\ & \theta \geq 1.5, \quad \theta \geq -4.62 + 2.2\chi_2, \\ & \theta \geq -2.32 + 0.5\chi_1 + 1.1\chi_2, \quad \theta \geq -2.64 + 0.7\chi_1 + 0.9. \end{aligned}$$

The last three constraints are Benders' cuts expressed in terms of χ_1 and χ_2 . The reader may check that the solution of the current problem with these three cuts is also the solution of the continuous LP-relaxation of the problem. The current problems in the next iterations only differ by the bounds on χ and the corresponding $\theta \geq \Psi(l^v + \varepsilon e)$ bound. Some of the current problems have multiple solutions. A different selection than ours would alter the course of the iterations.

Step 3. The solution of the current problem is $x^1 = (0.096, 1.696)^T$, $\chi^1 = (3.488, 1.887)^T$ and $CP(l^1, u^1) = -2.131$. Compute the value of $\Psi(\chi^1) = \Psi(3.8, 2)$. For $h = (-2.8, -1.2)^T$, $h + \chi = (1, 0.8)^T$. The second-stage solution is $y = (0, 1)^T$ and $\Psi(\chi, \xi) = 3$. For $h = (-2, -3)^T$, $h + \chi = (1.8, 0)^T$, $y = (0, 1)^T$ with $\Psi(\chi, \xi) = 3$. Taking the expectation, we obtain $\Psi(\chi^1) = 3$. Thus, $z^1 = z(x^1, \chi^1) = c^T x^1 + \Psi(\chi^1) = -3.631 + 3 = -0.631$.

Step 4. Set $\bar{z} = z^1 = -0.631$.

Step 5. Find discontinuity points of $\Psi(\cdot)$. For χ_1 , discontinuity points are all integers and all integers $+0.8$. Thus, from $\chi_1 = 3.488$, we may branch at 3 or 3.8. For χ_2 , discontinuity points are all integers and all integers $+0.2$. Thus, from $\psi_2 = 1.887$, we may only branch at 2 (since 1.2 is outside the bounds). Say, we branch at $\chi_1 = 3$. Create two new hypercubes, both having the same lower bound: $H_1 = (3, 6] \times (1.4, 7.5]$, with $\beta_1 = -2.131$ and $H_2 = (1.4, 3] \times (1.4, 7.5]$, with $\beta_2 = -2.131$. $\Lambda = \{H_1, H_2\}$.

Iteration 2:

Step 1. $v = 2$. Select $H^2 = H_1$ and remove it from the list.

Step 2. $l^2 = (3, 1.4)^T$ $u^2 = (6, 7.5)^T$. $\Psi(l^2 + \varepsilon e) = \Psi(3.1, 1.5) = \Psi(3.8, 2) = \Psi(\chi_1) = 3$.

Step 3. Create a new current problem, with a lower bound of 3.1 for χ_1 (instead of 1.5) and a lower bound 3 for θ . The solution of the current problem is $x^2 = (0.408, 2.008)^T$, $\chi_2 = (4.425, 2.825)^T$ and $CP(l^2, u^2) = -2.037$. Compute the value of $\Psi(\chi^2) = \Psi(4.425, 2.825) = \Psi(4.8, 3)$. For $h = (-2.8, -1.2)^T$, $h + \chi = (2, 1.8)^T$. The second-stage solution is $y = (1, 0)^T$ and $\Psi(\chi, \xi) = 4.4$. For $h = (-2, -3)^T$, $h + \chi = (2.8, 0)^T$, $y = (1, 0)^T$ with $\Psi(\chi, \xi) = 3$. Taking expectation, we get $\Psi(\chi^2) = 3.7$. Thus, $z^2 = z(x^2, \chi^2) = c^T x^2 + \Psi(\chi^2) = -5.037 + 3.7 = -1.337$.

Step 4. Set $\bar{z} = z^2 = -1.337$.

Step 5. Find discontinuity points of $\Psi(\cdot)$. From $\chi_1 = 4.425$, we may branch at 4 or 4.8. From $\chi_2 = 2.825$, we may branch at 2.2 or 3. Say, we branch at $\chi_2 = 2.2$. Create two new hypercubes $H_3 = (3, 6] \times (2.2, 7.5]$ and $H_4 = (3, 6] \times (1.4, 2.2]$, with $\beta_3 = \beta_4 = -2.037$. $\Lambda = \{H_2, H_3, H_4\}$.

Iteration 3:

Step 1. $v = 3$. Select $H^3 = H_2$ and remove it from the list.

Step 2. $l^3 = (1.4, 1.4)^T$ $u^3 = (3, 7.5)^T$. $\Psi(l^3 + \varepsilon e) = \Psi(1.5, 1.5) = 1.5$.

Step 3. The solution of the current problem is $x^3 = (0.406, 1.297)^T$, $\chi^3 = (3, 2.109)^T$ and $CP(l^3, u^3) = -2.109$. $\Psi(\chi^3) = \Psi(3, 2.2) = 3$ and $z^3 = z(x^3, \chi^3) = -0.609$.

Step 4. \bar{z} is unchanged.

Step 5. Find discontinuity points of $\Psi(\cdot)$. From $\chi_2 = 2.109$, we may branch at 2 or 2.2. Say we branch on $\chi_2 = 2$. Create two new hypercubes $H_5 = (1.4, 3] \times (2, 7.5]$ and $H_6 = (1.4, 3] \times (1.4, 2]$, with $\beta_5 = \beta_6 = -3.25$. $\Lambda = \{H_3, H_4, H_5, H_6\}$.

Iteration 4:

Step 1. $v = 4$. Select $H^4 = H_3$ and remove it from the list.

Step 2. $l^4 = (3, 2.2)^T$ $u^4 = (6, 7.5)^T$. $\Psi(l^4 + \varepsilon e) = \Psi(3.1, 2.3) = 3.7$.

Step 3. The solution of the current problem is $x^4 = (0.554, 2.154)^T$, $\chi^4 = (4.863, 3.262)^T$ and $CP(l^4, u^4) = -1.994$.

$\Psi(\chi^4) = \Psi(5, 4) = 5.2$ and $z^4 = z(x^4, \chi^4) = -5.694 + 5.2 = -0.494$.

Step 4. \bar{z} is unchanged.

Step 5. From $\chi_1 = 4.863$, we may branch at 4.8 or 5. From $\chi_2 = 3.262$, we may branch at 3.2 or 4. Say, we branch at $\chi_1 = 4.8$. Create two new hypercubes $H_7 = (4.8, 6] \times (2.2, 7.5]$ and $H_8 = (3, 4.8] \times (2.2, 7.5]$, with $\beta_7 = \beta_8 = -1.994$. $\Lambda = \{H_4, H_5, H_6, H_7, H_8\}$.

Iteration 5:

Step 1. $v = 5$. Select $H^5 = H_4$ and remove it from the list.

Step 2. $l^5 = (3, 1.4)^T$ $u^5 = (6, 2.2)^T$. $\Psi(l^5 + \varepsilon e) = \Psi(3.1, 1.5) = 3$.

Step 3. The solution of the current problem is $x^5 = (0, 2.2)^T$, $\chi^5 = (4.4, 2.2)^T$ and $CP(l^5, u^5) = -1.4$. $\Psi(\chi^5) = \Psi(4.8, 2.2) = 3$ and $z^5 = z(x^5, \chi^5) = -4.4 + 3 = -1.4$.

Step 4. Set $\bar{z} = z^5 = -1.4$.

Step 5. $CP(l^5, u^5) \geq \bar{z}$. Fathom. This is the situation described in Exercice 1 below. $\Lambda = \{H_5, H_6, H_7, H_8\}$.

Iteration 6:

Step 1. $v = 6$. To speed up things, select $H^6 = H_8$ and remove it from the list.

Step 2. $l^6 = (3, 2.2)^T$ $u^6 = (4.8, 7.5)^T$. $\Psi(l^6 + \varepsilon e) = \Psi(3.1, 2.3) = 3.7$.

Step 3. The solution of the current problem is $x^6 = (0.594, 2.103)^T$, $\chi^6 = (4.8, 3.291)^T$ and $CP(l^6, u^6) = -1.991$. $\Psi(\chi^6) = \Psi(4.8, 4) = 5.2$ and $z^6 = z(x^6, \chi^6) = -5.691 + 5.2 = -0.491$.

Step 4. \bar{z} is unchanged.

Step 5. Find discontinuity points of $\Psi(\cdot)$. From $\chi_2 = 3.291$, we branch at $\chi_2 = 3.2$. Create two new hypercubes $H_9 = (3, 4.8] \times (3.2, 7.5]$ and $H_{10} = (3, 4.8] \times (2.2, 3.2]$, with $\beta_9 = \beta_{10} = -1.993$. $\Lambda = \{H_5, H_6, H_7, H_9, H_{10}\}$.

Iteration 7:

Step 1. $v = 7$. To speed up things, select $H^7 = H_{10}$ and remove it from the list.

Step 2. $l^7 = (3, 2.2)^T$ $u^7 = (4.8, 3.2)^T$. $\Psi(l^7 + \varepsilon e) = \Psi(3.1, 2.3) = 3.7$

Step 3. The solution of the current problem is $x^7 = (0.533, 2.133)^T$, $\chi^7 = (4.8, 3.2)^T$ and $CP(l^7, u^7) = -1.9$. $\Psi(\chi^7) = \Psi(4.8, 3.2) = 3.2$ and $z^7 = z(x^7, \chi^7) = -5.6 + 3.7 = -1.9$.

Step 4. Set $\bar{z} = z^7 = -1.9$.

Step 5. $CP(l^7, u^7) \geq \bar{z}$. Fathom. $\Lambda = \{H_5, H_6, H_7, H_9\}$.

Subsequent iterations.

The current solution $\chi^7 = (4.8, 3.2)^T$ with $z^7 = -1.9$ is in fact the optimal one. (In a small problem like this one, this can be checked by solving the full deterministic equivalent.) Of the remaining hypercubes, only H_9 will be fathomed directly by the value of the current problem (-1.664). The other three hypercubes will need extra branchings. Note that the lower bounds β_i 's have not been used for the selection of the hypercubes in Step 1, as this would have yet augmented the number of iterations. Also, they could not be used to fathom hypercubes, as all lower bounds were smaller than the optimum.

Faster fathoming is expected if better bounds can be obtained. One way to get those would be to have a full description of the second-stage continuous recourse function, for instance by sending all possible Benders cuts. In the current example, the value of the current problem would have been improved only on H_9 .

A number of implementation aspects have been omitted in the presentation as well as in the example. They can be found in Ahmed, Tawarmalani and Sahinidis [2004]. This includes how to find the smallest initial hypercube or how to choose an effective partitioning component. Earlier work on integer second-stage includes decomposition of test sets (Hemmecke and Schultz [2003]) or Gröbner basis reduction techniques (Schultz, Stougie, van der Vlerk [1998]). For the case of integer first- and second-stage and random right-hand side only, Kong, Schaefer and Hunsaker [2006] develop a superadditive dual approach.

Exercises

1. In the branching-on-tenders algorithm, show that if $\Psi(\chi^v) = \Psi(l^v + \varepsilon e)$, then no branching occurs in Step 5.
2. Consider the second-stage constraints as in Example 3. Compare two situations:
 - h is a random vector with two independent components, each taking all integer values between -1 and -6 independently;
 - h can take four values: $(-2, -2.4)^T$, $(-3.8, -3.5)^T$, $(-4.6, -4.1)^T$ and $(-5.2, -5.3)^T$.

Which one is likely to create more discontinuity points in $\Psi(\cdot)$?

7.4 Reformulation

To illustrate reformulation, assume a discrete random variable and a fixed recourse matrix. Also assume binary second-stage decision variables. The value of the second-stage program for one realization ξ_k reads as

$$Q(x, \xi_k) = \min_y \{q_k^T y \mid Wy \geq h_k - T_k x, y \in \{0, 1\}^{n_2}\} \quad (4.1)$$

where, as usual, the index $k = 1, \dots, K$ is used for the K realizations of ξ . The LP-relaxation of this program is

$$C(x, \xi_k) = \min_y \{q_k^T y \mid Wy \geq h_k - T_k x, 0 \leq y \leq e\}. \quad (4.2)$$

The idea of reformulation is to modify the original formulation of $\{y \mid Wy \geq h_k - T_k x, 0 \leq y \leq e\}$ by adding a number of so-called *valid inequalities* or cuts that will reduce the number of fractional solutions. A large variety of valid inequalities have been proposed in integer programming. The choice of an appropriate class of valid inequalities depends on the structure of the LP-relaxation. Valid inequalities are routinely used in so-called *branch-and-cut systems*. Section 7.8b. provides simple examples of valid inequalities in deterministic models. We use these examples to illustrate the extra difficulties in stochastic integer programs.

a. Difficulties of reformulation in stochastic integer programs

Example 5

Consider the following second-stage program:

$$\begin{aligned} Q(x, \xi) = & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } & 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq h - Tx, \\ & y_1, \dots, y_4 \in \{0, 1\}. \end{aligned}$$

Assume two realizations of $\xi = (h, T)$, $h - Tx = 10 - 2x_1 - 4x_2$ and $11 - 4x_1 - 3x_2$ for $k = 1, 2$, with probability 0.25 and 0.75, respectively. Consider a current iterate $x^v = (0.3, 0.6)^T$. The second-stage program for $x = x^v$ and $\xi = \xi_1$ is

$$\begin{aligned} Q(x^v, \xi_1) = & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } & 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ & y_1, \dots, y_4 \in \{0, 1\}. \end{aligned} \tag{4.3}$$

The LP-relaxation of (4.3) has a fractional solution $y = (1, 1, 0.2, 0)^T$. The cover inequality $y_3 + y_4 \geq 1$ is a valid inequality and, as shown in Section 7.8b., it suffices to provide an extended LP-relaxation

$$\begin{aligned} C(x^v, \xi_1) = & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } & 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ & y_3 + y_4 \geq 1, \\ & 0 \leq y_1, \dots, y_4 \leq 1, \end{aligned} \tag{4.4}$$

having an integer optimal solution $y = (1, 0, 1, 0)^T$.

If we consider $\xi = \xi_2$, the r.h.s. of the initial constraint becomes 8. The LP-relaxation has a fractional solution $y = (1, 1, 0.4, 0)^T$ and two cuts are needed to obtain the extended LP-relaxation:

$$\begin{aligned} C(x^v, \xi_2) = & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } & 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 8, \\ & y_2 + y_3 + y_4 \geq 2, \\ & y_1 + y_3 \geq 1, \\ & 0 \leq y_1, \dots, y_4 \leq 1, \end{aligned} \tag{4.5}$$

having an integer optimal solution $y = (0, 0, 1, 1)^T$.

The extra difficulty in stochastic integer program is that the second stage program is dependent on x . For $\xi = \xi_1$, we have obtained reformulation (4.4) for $x = (0.3, 0.6)^T$. If we consider another iterate point, say $x^v = (0.5, 0.25)^T$, then the knapsack constraint in $Q(x^v, \xi_1)$ obtains a r.h.s. of 8 and the appropriate reformulation is the same as in (4.5).

Thus, the reformulation of the second-stage of a stochastic integer program is faced with two difficulties: a reformulation is needed for each realization of the random vector and the reformulation must be made dependent on the value of the first-stage variables.

b. Disjunctive cuts

One way to overcome these difficulties is through the use of disjunctive cuts, as we now explain. Section 7.8c. provides a short reminder of disjunctive cuts, with a proof and some examples.

Proposition 11. *If $P^i = \{x \in \mathbb{R}_+^n \mid A^i x \geq b^i\}$ for $i = 0, 1$ are two nonempty polyhedra, then $\pi^T x \geq \pi_0$ is a valid inequality for $\text{co}(P^0 \cup P^1)$ if and only if there exists $u^0, u^1 \geq 0$ such that $\pi \geq (u^i)^T A^i$ and $\pi_0 \leq (u^i)^T b^i$ for $i = 0, 1$.*

This proposition provides a way of convexifying the union of two sets. It will be used in this form at the end of this section. It is also used to realize the disjunction for a fractional variable.

Let $P = \{y \in \mathbb{R}_+^{n_2} \mid Wy \geq d, y \leq e\}$ be a particular second stage LP-relaxation (i.e. for one particular ξ_k and one particular $h - Tx^v = d$). Assume that, at the solution of the second-stage LP-relaxation, some second-stage binary variable y_j takes a fractional value. Instead of a classical branching $y_j \leq 0$ versus $y_j \geq 1$, one can consider the disjunction $P^0 = P \cap \{y \in \mathbb{R}_+^{n_2} \mid y_j \leq 0\}$ and $P^1 = P \cap \{y \in \mathbb{R}_+^{n_2} \mid y_j \geq 1\}$. Specializing Proposition 11 (with specific dual variables for each type of constraint and with each constraint under the \geq format), one obtains the following.

Proposition 12. *The inequality $\pi^T y \geq \pi_0$ is valid if and only if there exists $u^i, v^i, w^i \geq 0$ for $i = 0, 1$ such that*

$$\begin{aligned}\pi &\geq (u^0)^T W - v^0 - w^0 e_j, \\ \pi &\geq (u^1)^T W - v^1 + w^1 e_j, \\ \pi_0 &\leq (u^0)^T d - e^T v^0, \\ \pi_0 &\leq (u^1)^T d - e^T v^1 + w^1.\end{aligned}$$

A *disjunctive cut* is obtained by solving an LP consisting of maximizing the violation $\pi_0 - \pi^T y^v$, under the constraints defined in Proposition 12, where y^v is the current solution of the second stage LP. To be bounded, this LP needs some normalizing. One possibility is to take $-1 \leq \pi_0 \leq 1$, $-e \leq \pi \leq e$.

Proposition 12 is used in deterministic integer programs to generate one disjunctive cut. It is desired now to find one such cut for each realization ξ_k . The idea of the so-called *common cut coefficient technique* consists of obtaining an inequality $\pi^T y \geq \pi_0^k$ where the coefficients π for the variables remain the same independently of k and only the r.h.s.'s are dependent on k .

Proposition 13 (Common Cut Coefficient or C³). *The inequality $\pi^T y \geq \pi_0^k$ is valid for $k = 1, \dots, K$ if and only if there exists $u^i, v^i, w^i \geq 0$ for $i = 0, 1$ such that*

$$\begin{aligned}\pi &\geq (u^0)^T W - v^0 - w^0 e_j, \\ \pi &\geq (u^1)^T W - v^1 + w^1 e_j,\end{aligned}$$

$$\begin{aligned}\pi_0^k &\leq (u^0)^T d^k - e^T v^0, \\ \pi_0^k &\leq (u^1)^T d^k - e^T v^1 + w^1\end{aligned}$$

where $d^k = h^k - T^k x^v$.

In practice, the cut is obtained by solving an LP consisting of maximizing the expected violation $\sum_{k=1}^K p_k (\pi_0^k - \pi^T y^k)$ under the constraints defined in Proposition 13, where y^k is the second-stage solution associated to d^k . As above, we may use the normalization $-1 \leq \pi_0^k \leq 1$, $-e \leq \pi \leq e$. This LP is called the C^3-LP or $C^3-LP(W, d^k)$ if one needs to specify the problem data.

Example 5 (continued)

Consider again the second-stage program:

$$\begin{aligned}Q(x, \xi) &= \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } &2y_1 + 4y_2 + 5y_3 + 3y_4 \geq h - Tx, \\ &y_1, \dots, y_4 \in \{0, 1\},\end{aligned}$$

with the two realizations $h - Tx = 10 - 2x_1 - 4x_2$ and $11 - 4x_1 - 3x_2$ for $k = 1, 2$, with probability 0.25 and 0.75, respectively.

Consider the current iterate $x^v = (0.3, 0.6)^T$. The corresponding second-stage r.h.s. values $d^k = h^k - T^k x^v$ are 7 and 8, respectively for $k = 1, 2$. The solutions of the second-stage LP relaxations are $y = (1, 1, 0.2, 0)^T$ and $y = (1, 1, 0.4, 0)^T$ for the two realizations. The disjunction is made on y_3 as it is fractional in both cases. Taking the objective of maximizing the expected violation $\sum_{k=1}^K p_k (\pi_0^k - \pi^T y^k)$ and the normalization as above, the (C^3 -LP) problem reads as follows:

$$\begin{aligned}(C^3\text{-LP}) \quad z &= \max 0.25\pi_0^1 + 0.75\pi_0^2 - \pi_1 - \pi_2 - 0.35\pi_3 \\ \text{s. t. } &\pi_1 \geq 2u^0 - v_1^0, \quad \pi_1 \geq 2u^1 - v_1^1, \\ &\pi_2 \geq 4u^0 - v_2^0, \quad \pi_2 \geq 4u^1 - v_2^1, \\ &\pi_3 \geq 5u^0 - v_3^0 - w^0, \quad \pi_3 \geq 5u^1 - v_3^1 + w^1, \\ &\pi_4 \geq 3u^0 - v_4^0, \quad \pi_4 \geq 3u^1 - v_4^1, \\ &\pi_0^1 \leq 7u^0 - v_1^0 - v_2^0 - v_3^0 - v_4^0, \\ &\pi_0^1 \leq 7u^1 - v_1^1 - v_2^1 - v_3^1 - v_4^1 + w^1, \\ &\pi_0^2 \leq 8u^0 - v_1^0 - v_2^0 - v_3^0 - v_4^0, \\ &\pi_0^2 \leq 8u^1 - v_1^1 - v_2^1 - v_3^1 - v_4^1 + w^1, \\ &-e \leq \pi \leq e, \quad -1 \leq \pi_0^1, \quad \pi_0^2 \leq 1, \quad u, v, w \geq 0.\end{aligned}$$

Its optimal solution is $z = 0.35$, $u^0 = 1/3$, $v^0 = (2/3, 4/3, 0, 0)^T$, $u^1 = 0$, $v^1 = (0, 0, 0, 0)^T$, $w^0 = 1$, $w^1 = 2/3$, $\pi = (0, 0, 2/3, 1)^T$, $\pi_0^1 = 1/3$, $\pi_0^2 = 2/3$. The two cuts are $2/3y_3 + y_4 \geq 1/3$, $2/3y_3 + y_4 \geq 2/3$, for $k = 1, 2$, respectively.

At the current second-stage solutions, the two cuts are violated by an amount of $0.6/3$ and $1.2/3$, respectively. The expected violation corresponds to the value 0.35 of the objective of $C^3 - LP$. Given the u , v , w values, one can check that $\pi_0^1 \leq \min\{1/3, 2/3\}$ and $\pi_0^2 \leq \min\{2/3, 2/3\}$.

We now look of how to make these values dependent on the first-stage decision variables.

c. First-stage dependence

Consider the C^3 cut for a given k . We have seen that $\pi^T y \geq \pi_0^k$ is valid for

$$\begin{aligned}\pi_0^k &\leq (u^0)^T d^k - e^T v^0, \\ \pi_0^k &\leq (u^1)^T d^k - e^T v^1 + w^1,\end{aligned}$$

where $d^k = h^k - T^k x^v$.

If instead of considering a fixed d^k , we let x vary, we obtain a cut $\pi^T y \geq \pi_0^k(x)$ whose r.h.s depends on x . This cut remains valid for

$$\begin{aligned}\pi_0^k(x) &\leq (u^0)^T (h^k - T^k x) - e^T v^0, \\ \pi_0^k(x) &\leq (u^1)^T (h^k - T^k x) - e^T v^1 + w^1.\end{aligned}$$

With π , u , v and w unchanged, it suffices indeed to take a sufficiently small value of $\pi_0^k(x)$ to obtain a valid cut.

To simplify notations, let $\alpha^0 = (u^0)^T h^k - e^T v^0$, $\alpha^1 = (u^1)^T h^k - e^T v^1 + w^1$ and $\beta^i = (u^i)^T T^k$ for $i = 0, 1$. Thus,

$$\pi^T y \geq \min\{\alpha^0 - \beta^0 x, \alpha^1 - \beta^1 x\}$$

where the index k is omitted in the r.h.s. even if the data are dependent on k .

Due to the min operation, the cut is nonlinear and needs convexification. This can be achieved through a disjunction with the two sets

$$\begin{aligned}P^0 &= \{x \in \Re^n_+, \gamma \geq 0 \mid Ax \geq b, \gamma \geq \alpha^0 - \beta^0 x\}, \\ P^1 &= \{x \in \Re^n_+, \gamma \geq 0 \mid Ax \geq b, \gamma \geq \alpha^1 - \beta^1 x\},\end{aligned}$$

where γ is an extra variable representing the minimum of the two expressions.

The RHS(k) problem consists of finding $r^i, s^i \geq 0$ for $i = 0, 1$ and (ρ, ρ_0) s.t.

$$\rho \geq (r^0)^T A + \beta^0 s^0,$$

$$\begin{aligned}\rho &\geq (r^1)^T A + \beta^1 s^1, \\ \rho_0 &\leq (r^0)^T b + \alpha^0 s^0, \\ \rho_0 &\leq (r^1)^T b + \alpha^1 s^1, \\ s^0, s^1 &\leq 1.\end{aligned}$$

These inequalities are written down assuming a value of 1 for the coefficient of γ , to form a cut $\gamma \geq \rho_0 - \rho^T x$. This is an appropriate form of normalization. The solution is obtained from an LP with the objective of maximizing $\max \rho_0 - \rho^T x^v$. The final cut is $\pi^T y \geq \rho_0 - \rho^T x$.

The notation $RHS(k)$ is a reminder that the r.h.s. of the resulting cut is valid for one given k . When needed, the notation $\pi^T y \geq \rho_{0k} - \rho_k^T x$ is then used to represent the cut obtained for one specific k .

Example 5 (continued)

Assume a single first stage constraint $4x_1 + 6x_2 \leq 5$ and, as above, a current iterate $x^v = (0.3, 0.6)^T$. The solution of the C^3-LP includes $u^0 = 1/3$, $v^0 = (2/3, 4/3, 0, 0)^T$, $u^1 = 0$, $v^1 = (0, 0, 0, 0)^T$, $w^0 = 1$, $w^1 = 2/3$.

Consider $k = 1$. Thus, $h - Tx = 10 - 2x_1 - 4x_2$. We obtain $\alpha^0 = 1/3p10 - 6/3 = 4/3$ and $\beta^0 = (2/3, 4/3)^T$ for $i = 0$ and $\alpha^1 = 2/3$ and $\beta^1 = (0, 0)^T$ for $i = 1$. $RHS(1)$ consists in convexifying $\min\{4/3 - 2/3x_1 - 4/3x_2, 2/3\}$ under $4x_1 + 6x_2 \leq 5$, $x_1, x_2 \geq 0$. Using the objective $\max \rho_0 - \rho^T x^v$ and the proposed normalization of the coefficient of γ , we obtain:

$$\begin{aligned}RHS(1) \quad z = \max \rho_0 - 0.3\rho_1 - 0.6\rho_2 \\ \text{s. t. } \rho_1 &\geq -4r^0 + 2/3s^0, \quad \rho_1 \geq -4r^1, \\ \rho_2 &\geq -6r^0 + 4/3s^0, \quad \rho_2 \geq -6r^1, \\ \rho_0 &\leq -5r^0 + 4/3s^0, \quad \rho_0 \leq -5r^1 + 2/3s^1, \\ 0 &\leq r^0, r^1, \quad 0 \leq s^0, s^1 \leq 1.\end{aligned}$$

The optimal solution is $z = 0.92/3$, $\rho_0 = 2/3$, $\rho_1 = 0.4/3$, $\rho_2 = 1.6/3$. The disjunctive cut for $k = 1$ is thus $2/3y_3 + y_4 \geq 2/3 - 0.4/3x_1 - 1.6/3x_2$.

d. An algorithm

For simplicity, we present an algorithm with second-stage reformulation for the case when the first-stage variables are continuous and assuming relatively complete recourse. Such an algorithm is a direct extension of the *L*-shaped method of Chapter 5, with an extended Step 3 for the construction of the Benders cuts.

L -Shaped Algorithm with Second-stage Reformulation

Step 0. Set $s = v = 0$. Set $W_1 = W$, $h_1 = h$, $T_1 = T$.

Step 1 and *Step 2*: unchanged.

Step 3.

- (a) Solve the LP-relaxation $C(x, \xi_k) = \min_y \{q_k^T y \mid W_v y \geq h_{vk} - T_{vk}x^v, 0 \leq y \leq e\}$ for $k = 1, \dots, K$.
- (b) Select some component j s.t. y_j is fractional for at least one k . (If none exists, let $W_{v+1} = W_v$, $h_{v+1} = h_v$, $T_{v+1} = T_v$ and go to (f) with unchanged multipliers).
- (c) Solve $C^3 - LP(W_v, d_v^k)$ with $d_v^k = h_{vk} - T_{vk}x^v$. Append the solution π^T to the matrix W_v to form W_{v+1} .
- (d) Solve $RHS(k)$ for $k = 1, \dots, K$. For each k , append ρ_{0k} to h_{vk} to form $h_{v+1,k}$ and append ρ_k^T to the matrix T_{vk} to form $T_{v+1,k}$.
- (e) Solve the LP-relaxation $C(x, \xi_k) = \min_y \{q_k^T y \mid W_{v+1}y \geq h_{v+1,k} - T_{v+1,k}x^v, 0 \leq y \leq e\}$ for $k = 1, \dots, K$.
- (f) Use the dual multipliers to generate an *L*-Shaped cut as in (5.1.6) and (5.1.7), based on $h_{v+1,k}$ and $T_{v+1,k}$.
- (g) Test of optimality or addition of the cut (as in the end of Step 3 in the *L*-shaped method).

If one compare the above steps with those of the *L*-shaped method, the extra work consists of solving one $C^3 - LP$, solving K times a $RHS(k)$ and reoptimizing the K second-stage relaxations with one extra constraint each. The $C^3 - LP$ has $2m_2 + 2n_2 + 2K + 2$ variables and $2n_1 + 2K$ constraints. Each $RHS(k)$ has $2m_1 + 2$ variables and $2n_1 + 2$ constraints. The alternative of finding one possibly different disjunctive cut for each k not only in the r.h.s. but also in the l.h.s. would request the solution of K successive LP's, each having $2m_2 + 2n_2 + 2$ variables and $2n_1 + 2$ constraints. The convexification of the r.h.s.'s would still require the solution of K $RHS(k)$ programs having the same dimension as above.

The above algorithm was developed by Sen and Higle [2005]. It can be seen as an integer *L*-shaped type of method, with more elaborate steps for the construction of the cuts. The case of continuous first-stage may present some technicalities that are studied in Ntiamo and Sen [2006a]. The second stage reformulation may fail to produce natural integer solutions in the second-stage for all $k = 1, \dots, K$ in a sufficiently fast manner. In such a case, an extra branch-and-bound step in the second-stage may be needed. A description of this extra feature can be found in Sen and Sherali [2002]. Reports on computational experiments can be found in Ntiamo and Sen [2006b].

Exercises

1. Take Example 5 and problem $RHS(1)$. For the current iterate point with $y_3 = 0.2$, $y_4 = 0$ and $x^v = (0.3, 0.6)^T$, compare the violation of the disjunctive cut after convexification and the one in the C^3-LP solution.
2. Take Example 5. Solve $RHS(2)$ and obtain the cut $2/3y_3 + y_4 \geq 2/3 - 1.6/3x_1$.
3. Take Example 5. Suppose that the first-stage constraints are $x_1 \leq 1$, $x_2 \leq 1$ (instead of the single constraint $4x_1 + 6x_2 \leq 5$). Solve $RHS(1)$ and $RHS(2)$ to obtain the r.h.s. of the disjunctive cuts. What are the violations at the current iterate point $x^v = (0.3, 0.6)^T$?

7.5 Simple Integer Recourse

As seen in Section 3.3, a two-stage stochastic program with simple integer recourse can be transformed into

$$\begin{aligned} & \min c^T x + \sum_{i=1}^m \Psi_i(\chi_i) \\ & \text{s. t. } Ax = b, \quad Tx = \chi, \quad x \in X \subset \mathbb{Z}_+^{n_1}, \end{aligned} \quad (5.1)$$

where

$$\Psi_i(\chi_i) = q_i^+ u_i(\chi_i) + q_i^- v_i(\chi_i) \quad (5.2)$$

with

$$u_i(\chi_i) = E[\xi_i - \chi_i]^+, \quad (5.3)$$

defined as the expected shortage, and

$$v_i(\chi_i) = E[\chi_i - \xi_i]^+, \quad (5.4)$$

defined as the expected surplus. As before, we assume

$$q_i^+ \geq 0, q_i^- \geq 0.$$

Also from Section 3.3, we know that the values of the expected shortage and surplus can be computed in finitely many steps, either exactly or within a prespecified tolerance ϵ .

Before turning to algorithms, we still need some results concerning the functions Ψ_i ; for simplicity in the exposition we omit the index i . As we also know from Section 3.3, the function Ψ is generally not convex and is even discontinuous when ξ has a discrete distribution. It turns out, however, that some form of convexity exists between function values evaluated in (not necessarily integer) points that are

integer length apart. Thus, let $x^0 \in \mathfrak{R}$ be an arbitrary point. Let $i \in \mathbb{Z}$ be some integer.

Define $x^1 = x^0 + i$, and for any $j \in \mathbb{Z}$, $j \leq i$, $x^\lambda = x^0 + j$. Equivalently, we may define

$$\begin{aligned} x^\lambda &= \lambda x^0 + (1 - \lambda)x^1, \\ \lambda &= (i - j)/i. \end{aligned}$$

In the following, we will use x as an argument for Ψ as if $Tx = Ix = \chi$ without losing generality. We make T explicit again when we speak of a general problem and not just the second stage.

Proposition 14. *Let $x^0 \in \mathfrak{R}$, $i, j \in \mathbb{Z}$ with $j \leq i$, $x^1 = x^0 + i$, $x^\lambda = x^0 + j$. Then*

$$\Psi(x^\lambda) \leq \lambda \Psi(x^0) + (1 - \lambda) \Psi(x^1) \quad (5.5)$$

with $\lambda = (i - j)/i$.

Proof: We prove that $\Psi(x+1) - \Psi(x)$ is a nondecreasing function of x . We leave it as an exercise to infer that this is a sufficient condition for (5.5) to hold. Using (3.3.16) and (3.3.17), we respectively obtain $u(x+1) - u(x) = -(1 - F(x))$ and $v(x+1) - v(x) = \hat{F}(x+1)$, where F is again the cumulative distribution function of ξ and \hat{F} is defined as in Section 3.3. With this,

$$\Psi(x+1) - \Psi(x) = q^- \hat{F}(x+1) - q^+(1 - F(x)).$$

The result follows as $q^+ \geq 0$, $q^- \geq 0$ and \hat{F} and F are nondecreasing. \square

This means that we can draw a piecewise linear convex function through points that are integer length apart. Such a convex function is called a ρ -approximation rooted at x if it is drawn at points $x \pm \kappa$, κ integer. In Figures 3 and 4, we provide the ρ -approximations rooted at $x = 0$ and $x = 0.5$, respectively, for the case in Example 3.1.

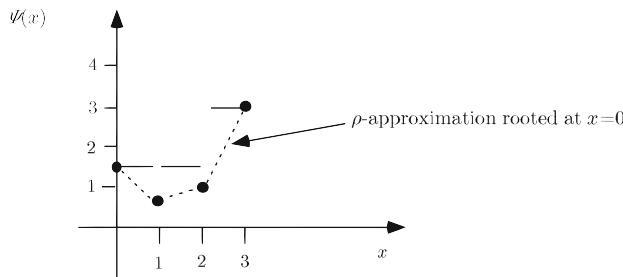


Fig. 3 The ρ -approximation rooted at $x = 0$.

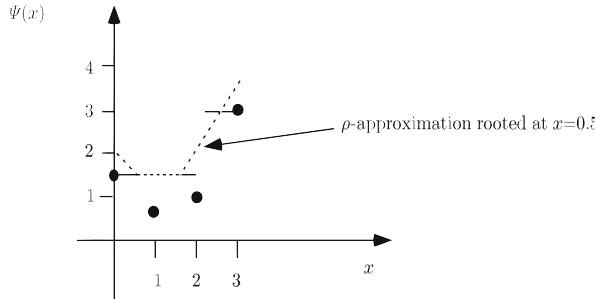


Fig. 4 The ρ -approximation rooted at $x = 0.5$.

If we now turn to discrete random variables, we are interested in the different possible fractional values associated with a random variable. As an example, if ξ can take on the values 0.0, 1.0, 1.2, 1.6, 2.0, 2.2, 2.6, and 3.2 with some given probability, then the only possible fractional values are 0.0, 0.2, and 0.6. Let $f_1 < f_2 < \dots < f_S$ denote the S ordered possible fractional values of ξ . Define $f_{S+1} = 1$. Let the extended list of fractionals be all points of the form $a + f_s$, $a \in \mathbb{Z}$, $1 \leq s \leq S$. This extended list is a countable list that contains many more elements than the possible values of ξ . In the example, 0.2, 0.6, 3.0, 3.6, 4.0, 4.2, ... are in the extended list of fractionals but are not possible values of ξ .

Lemma 15. *Let ξ be a discrete random variable. Assume that S is finite. Let $a \in \mathbb{Z}$. Then*

$$\begin{aligned} \Psi(x) &\text{ is constant within the open interval } (a + s_j, a + s_{j+1}), \\ \Psi(x) &\geq \max \{\Psi(a + s_j), \Psi(a + s_{j+1})\}, \\ \text{for all } x &\in (a + s_j, a + s_{j+1}). \end{aligned}$$

Proof: The proof can be found in Louveaux and van der Vlerk [1993]. \square

The lemma states that $\Psi(x)$ is piecewise constant in the open interval between two consecutive elements of the extended list of fractionals and that the values in points between two consecutive elements of that list are always greater than or equal to the values of $\Psi(\cdot)$ at these two consecutive elements in the extended list. The reader can easily observe this property in the examples that have already been given.

Corollary 16. *Let ξ be a random variable with $S = 1$. Let $\rho(\cdot)$ be a ρ -approximation of $\Psi(\cdot)$ rooted at some point in the support of ξ . Then*

$$\rho(x) \leq \Psi(x), \forall x \in \mathfrak{R}.$$

Moreover, ρ is the convex hull of the function Ψ .

Proof: By Lemma 15,

$$\forall x \in (a, a+1), \\ \rho(x) \leq \max\{\rho(a), \rho(a+1)\} = \max\{\Psi(a), \Psi(a+1)\} \leq \Psi(x).$$

Thus, ρ is a lower bound for Ψ . It is the convex hull of Ψ because it is convex, piecewise linear, and it coincides with Ψ in all points at integer distance from the root. \square

Among the cases where $S = 1$, the most natural one in the context of simple integer recourse is when ξ only takes on integer values. A well-known such case is the Poisson distribution. Then the ρ -approximation rooted at any integer point is the piecewise linear convex hull of Ψ that coincides with Ψ at all integer points.

We use Proposition 14 and Corollary 16 to derive finite algorithms for two classes of stochastic programs with simple integer recourse.

a. χ restricted to be integer

Integral χ is a natural assumption, because one would typically expect first-stage variables to be integer when second-stage variables are integer. It suffices then for T to have integer coefficients. By definition of a ρ -approximation rooted at an integer point, solving (5.1) is thus equivalent to solving

$$\min\{c^T x + \sum_{i=1}^{m_2} \rho_i(\chi_i) \mid Ax = b, \chi = Tx, x \in X\}, \quad (5.6)$$

where T is such that $x \in X$ implies χ is integer, and ρ_i is a ρ -approximation of Ψ_i rooted at an integer point.

Because the objective in (5.6) is piecewise linear and convex, problem (5.6) can typically be solved by a dual decomposition method such as the L -shaped method. We recommend using the multicut version because we are especially concerned with generating individual cut information for each subproblem that may require many cuts. This amounts to solving a sequence of current problems of the form

$$\min_{x \in X, \theta \in \Re^{m_2}} \left\{ c^T x + \sum_{i=1}^{m_2} \theta_i \mid Ax = b, \chi = Tx, \right. \\ \left. E_{il}\chi_i + \theta_i \geq e_{il}, i = 1, \dots, m_2, l = 1, \dots, s_i \right\}. \quad (5.7)$$

In this problem, the last set of constraints consists of optimality cuts. They are used to define the epigraph of Ψ_i , $i = 1, \dots, m_2$. Optimality cuts are generated only as needed. If χ_i^v is a current iterate point with $\theta_i^v < \Psi_i(\chi_i^v)$, then an additional

optimality cut is generated by defining

$$E_{ik} = \Psi_i(\chi_i^v) - \Psi_i(\chi_i^v + 1) \quad (5.8)$$

and

$$e_{ik} = (\chi_i^v + 1)\Psi_i(\chi_i^v) - \chi_i^v\Psi_i(\chi_i^v + 1), \quad (5.9)$$

which follows immediately by looking at a linear piece of the graph of Ψ_i . The algorithm iteratively solves the current problem (5.7) and generates optimality cuts until an iterate point (χ^v, θ^v) is found such that $\theta_i^v = \Psi_i(\chi_i^v)$, $i = 1, \dots, m_2$. It is important to observe that the algorithm is applicable for any type of random variable for which Ψ_i 's can be computed.

Example 6

Consider two products, $i = 1, 2$, which can be produced by two machines $j = 1, 2$. Demand for both goods follows a Poisson distribution with expectation 3. Production costs (in dollars) and times (in minutes) of the two products on the two machines are as follows:

Machine:			
	1	2	
Product: 1	3	2	
	2	4	5
Cost/Unit			

Machine: Finishing:				
	1	2	1	2
Product: 1	20	25	4	7
	2	30	25	6
Time/Unit				

The total time for each machine is limited to 100 minutes. After machining, the products must be finished. Finishing time is a function of the machine used, with total available finishing time limited to 36 minutes. Production and demand correspond to an integer number of products. Product 1 sells at \$4 per unit. Product 2 sells at \$6 per unit. Unsold goods are lost.

Define x_{ij} = number of units of product i produced on machine j and $y_i(\xi) =$ amount of product i sold in state ξ . The problem reads as follows:

$$\min 3x_{11} + 2x_{12} + 4x_{21} + 5x_{22} + E_\xi \{-4y_1(\xi) - 6y_2(\xi)\}$$

$$\begin{aligned}
\text{s. t.} \quad & 20x_{11} + 30x_{21} \leq 100, \quad y_1(\xi) \leq \xi_1 \\
& 25x_{12} + 25x_{22} \leq 100, \quad y_2(\xi) \leq \xi_2, \\
& 4x_{11} + 7x_{12} + 6x_{21} + 5x_{22} \leq 36, \quad y_1(\xi) \leq x_{11} + x_{12}, \\
& x_{ij} \geq 0 \text{ integer}, \quad y_2(\xi) \leq x_{21} + x_{22}, \\
& y_1(\xi), y_2(\xi) \geq 0 \text{ integer}.
\end{aligned}$$

Letting $y_i^+(\xi) = \xi_i - y_i(\xi)$, one obtains an equivalent formulation,

$$\begin{aligned}
\min & 3x_{11} + 2x_{12} + 4x_{21} + 5x_{22} + E_\xi\{4y_1^+(\xi) + 6y_2^+(\xi)\} - 30 \\
\text{s. t.} \quad & 20x_{11} + 30x_{21} \leq 100, \quad y_1^+(\xi) \geq \xi_1 - \chi_1, \\
& 25x_{12} + 25x_{22} \leq 100, \quad y_2^+(\xi) \geq \xi_2 - \chi_2, \\
& 4x_{11} + 7x_{12} + 6x_{21} + 5x_{22} \leq 36, \\
& y_1^+(\xi), y_2^+(\xi) \geq 0 \text{ and integer}, \\
& x_{11} + x_{12} = \chi_1, \quad x_{21} + x_{22} = \chi_2, \\
& x_{ij} \geq 0 \text{ and integer}.
\end{aligned}$$

This representation puts the problem under the form of a simple recourse model with expected shortage only.

Let us start with the null solution, $x_{ij} = 0$, $\chi_i = 0$, $i, j = 1, 2$ with $\theta_i = -\infty$, $i = 1, 2$. We compute $u(0) = E[\xi]^+ = \mu^+ = 3$; hence $\Psi_1(0) = 12$, $\Psi_2(0) = 18$, where we have dropped the constant, -30 , from the objective for these computations. To construct the first optimality cuts, we also compute $u(1) = u(0) - 1 + F(0) = 2 + 0.05 = 2.05$. Thus, $E_{11} = 4(3 - 2.05) = 3.8$, $e_{11} = 4(1 * 3 - 0 * 2.05) = 12$, defining the optimality cut $\theta_1 + 3.8\chi_1 \geq 12$. As $\chi_2 = \chi_1$, E_{21} and e_{21} are just 1.5 times E_{11} and e_1 , respectively, yielding the optimality cut $\theta_2 + 5.7\chi_2 \geq 18$.

The current problem becomes

$$\begin{aligned}
\min & 3x_{11} + 2x_{12} + 4x_{21} + 5x_{22} - 30 + \theta_1 + \theta_2 \\
\text{s. t.} \quad & 20x_{11} + 30x_{21} \leq 100, \quad 25x_{12} + 25x_{22} \leq 100, \\
& 4x_{11} + 7x_{12} + 6x_{21} + 5x_{22} \leq 36, \\
& x_{11} + x_{12} = \chi_1, \quad x_{21} + x_{22} = \chi_2, \\
& \theta_1 + 3.8\chi_1 \geq 12, \quad \theta_2 + 5.7\chi_2 \geq 18, \\
& x_{ij} \geq 0, \text{ integer}.
\end{aligned}$$

We obtain the solution $x_{11} = 0$, $x_{12} = 4$, $x_{21} = 1$, $x_{22} = 0$, $\theta_1 = -3.2$, $\theta_2 = 12.3$. We compute $u(4) = u(0) + \sum_{l=0}^3 (F(l) - 1) = 0.31936$ and $\Psi_1(4) = 1.277 > \theta_1$. A new optimality cut is needed for $\Psi_1(\cdot)$. Because $\Psi(5) = 0.5385$, the cut is $0.739\chi_1 + \theta_1 \geq 4.233$. We also have $u(1) = 2.05$, hence $\Psi_2(1) = 12.3 = \theta_2$, so no new cut is generated for $\Psi_2(\cdot)$.

At the next iteration, with the extra optimality cut on θ_1 , we obtain a new solution of the current problem as $x_{11} = 0$, $x_{12} = 2$, $x_{21} = 3$, $x_{22} = 0$, $\theta_1 = 4.4$, $\theta_2 = 0.9$. Here, two new optimality cuts are needed:

$$2.312\chi_1 + \theta_1 \geq 9.623$$

and

$$2.117\chi_2 + \theta_2 \geq 10.383.$$

The next iteration gives $x_{11} = 0$, $x_{12} = 3$, $x_{21} = 2$, $x_{22} = 0$, $\theta_1 = 2.688$, $\theta_2 = 6.6$ as a solution of the current problem. Because $\Psi_2(2) = 7.5 > \theta_2$, a new cut is generated, i.e., $3.467\chi_2 + \theta_2 \geq 14.435$. The next iteration point is $x_{11} = 0$, $x_{12} = 3$, $x_{21} = 2$, $x_{22} = 0$, $\theta_1 = 2.688$, $\theta_2 = 7.5$, which is the optimal solution with total objective value -5.812 .

b. The case where $S = 1$, χ not integral

Details can again be found in Louveaux and van der Vlerk [1993]; we illustrate the results with an example. Consider Example 6 but with the x_{ij} 's continuous. Because we still assume the random variables follow a Poisson distribution, the example indeed falls into the category $S = 1$; only integer realizations are possible.

For a given component i , the ρ_i -approximation rooted at an integer defines the convex hull of the function $\Psi_i(\cdot)$. All optimality cuts defined at integer points are thus valid inequalities. If we take Example 6 again and impose all optimality cuts at integer points, the continuous solution is $x_{11} = 0$, $x_{12} = 3$, $x_{21} = 2$, $x_{22} = 0$, and no extra cuts are needed here. Now assume the objective coefficients of x_{12} and x_{21} are 1 and 4.5 (instead of 2 and 4). The solution of the stochastic program with continuous first-stage decisions and all optimality cuts imposed at integer points becomes $x_{11} = 0$, $x_{12} = 4$, $x_{21} = 1.33$, $x_{22} = 0$, and thus, $\chi_1 = 4$, $\chi_2 = 1.33$.

We now illustrate how to deal with a noninteger value of some χ_i . Now, $u(1.33) = 3 - 1 + F(0) = 2.05$ and therefore $\Psi_2(1.33) = 12.3 > \theta_2$. This requires imposing a new optimality cut. By Lemma 15, we know $\Psi_2(\cdot)$ is constant within $(1, 2)$ with value 12.3. Let

$$\begin{aligned}\delta_a &= 1 \text{ if } \chi_2 > 1 \text{ and } 0 \text{ otherwise,} \\ \delta_b &= 1 \text{ if } \chi_2 < 2 \text{ and } 0 \text{ otherwise.}\end{aligned}$$

The cut imposes that $\theta_2 \geq 12.3$ if $1 < \chi_2 < 2$, i.e., if $\delta_a = \delta_b = 1$. This is realized by the following constraints:

$$\begin{aligned}\chi_2 &\leq 1 + 10\delta_a, & \chi_2 &\geq (1 + \varepsilon)\delta_a, \\ \chi_2 &\leq 10 - (8 + \varepsilon)\delta_b, & \chi_2 &\geq 2 - 2\delta_b, \\ \theta_2 &\geq 12.3 - 100(2 - \delta_a - \delta_b),\end{aligned}$$

where 10 and 100 are sufficiently large numbers to serve as bounds on χ_2 and $-\theta_2$, respectively, and ε is a very small number. Thus, defining a function $\Psi_i(\cdot)$ to be constant in some interval requires two extra binary variables and three extra

constraints. It is thus reasonable to first consider optimality cuts that define the convex hull.

Continuing the example, we solve the current problem with the three additional constraints. The solution is $x_{11} = 0$, $x_{12} = 3.43$, $x_{21} = 2$, $x_{22} = 0$ with $\chi_1 = 3.43$, $\chi_2 = 2$, $\theta_1 = 2.08$, $\theta_2 = 7.5$. Thus, one more set of cuts is needed to define Ψ_1 in the interval $(3,4)$. The final solution is $x_{11} = 0$, $x_{12} = 3$, $x_{21} = 1$, $x_{22} = 0$, $\theta_1 = 2.689$, $\theta_2 = 12.3$, and $z = -7.51$.

Exercises

1. The definition (3.3.5) of a two-stage stochastic program with simple recourse shows that it is a particular case of a two-stage stochastic program with integer second-stage. Explain why Lemma 15 is not identical to Propositions 8 and 9.
2. Similarly, for case where $S = 1$ and χ is not integral, explain why the branching on tenders algorithm of Section 7.3 does not apply directly.

7.6 Cuts Based on Branching in the Second Stage

We now show how branching on the second-stage variables may create feasibility or optimality cuts.

a. Feasibility cuts

As usual, let $K_2(\xi)$ denote the second-stage feasibility set for a given ξ and $K_2 = \bigcap_{\xi \in \Xi} K_2(\xi)$. Let also $C_2(\xi)$ denote the set of first-stage decisions that are feasible for the continuous relaxation of the second stage, i.e.,

$$C_2(\xi) = \{x \mid \exists y \text{ s. t. } Wy = h(\omega) - T(\omega)x, y \geq 0\}.$$

Clearly, $K_2(\xi) \subset C_2(\xi)$, and any induced constraint valid for $C_2(\xi)$ is also valid for $K_2(\xi)$. Also, detecting that some point $x \in C_2(\xi)$ does not belong to $K_2(\xi)$ amounts to solving a phase one problem:

$$\begin{aligned} (P1) \quad w(x, \xi) &= \min e^T v^+ + e^T v^- \\ \text{s. t. } Wy + v^+ - v^- &= h(\omega) - T(\omega)x, \\ y &\in Z_+^{n_2}, \quad v^+, v^- \geq 0. \end{aligned} \tag{6.1}$$

As usual, $x \in K_2(\xi)$ if and only if $w(x, \xi) = 0$. If $x \notin K_2(\xi)$, we would like to generate a feasibility cut. Let (y, v^+, v^-) be a solution to (P1), and because

$x \notin K_2(\xi)$, we have $w(x, \xi) = e^T v^+ + e^T v^- > 0$. If $y \in Z_+^{n_2}$, then a cut of the form: (5.1.3) can be generated. If $y \notin Z_+^{n_2}$, then some of the components of y are not integer. A branch and bound algorithm can be applied to (P1). This will generate a branching tree where, at each node, additional simple upper or lower bounds are imposed on some variables.

Let $\rho = 1, \dots, R$ index all *terminal nodes*, i.e., nodes that have no successors, of the second-stage branching tree. Let Y^ρ be the corresponding subregions. They form a partition of $\mathfrak{R}_+^{n_2}$, i.e., $\mathfrak{R}_+^{n_2} = \cup_{\rho=1, \dots, R} Y^\rho$ and $Y^\rho \cap Y^\sigma = \emptyset$, $\rho \neq \sigma$. Now, $x \in K_2(\xi)$ if and only if $x \in \cup_{\rho=1, \dots, R} K_2^\rho(\xi)$, where

$$K_2^\rho(\xi) = \{x \mid \exists y \in Y^\rho \text{ s. t. } Wy \leq h(\omega) - T(\omega)x, y \geq 0\}.$$

However, because Y^ρ is obtained from $\mathfrak{R}_+^{n_2}$ by some branching process, it is defined by adding a number of bounds to some components of y . Thus, $K_2^\rho(\xi)$ is a polyhedron for which linear cuts are obtained through a classical separation or duality argument. It follows that $x \in K_2(\xi)$ if and only if at least one among R sets of cuts is satisfied.

In practice, one constructs the branching tree of the second stage associated with one particular \bar{x} and generates one cut per terminal node of the restricted tree. This means that one first-stage feasibility cut (8.1.3) corresponds to the requirement that one out of R cuts is satisfied. As expected, this takes the form of a *Gomory function*. It can be embedded in a linear programming scheme by the addition of extra binary variables, one for each of the R cuts, as follows. Assume the ρ th cut is represented by $u_\rho^T x \leq d_\rho$. One introduces R binary variables, $\delta_1, \dots, \delta_R$. The requirement that at least one of the R cuts is satisfied is equivalent to

$$\begin{aligned} u_\rho^T x \leq d_\rho + M_\rho(1 - \delta_\rho), & \quad \rho = 1, \dots, R, \\ \sum_{\rho=1}^R \delta_\rho \geq 1, & \\ \delta_\rho \in \{0, 1\}, & \quad \rho = 1, \dots, R, \end{aligned}$$

where M_ρ is a large number such that $u_\rho^T x \leq d_\rho + M_\rho$, $\forall x \in K_1$.

Finally, observe that $x \in K_2$ if and only $x \in K_2(\xi)$, $\forall \xi \in \Xi$. As in the continuous case (Section 5b.), it is sometimes enough to consider $x \in K_2(\xi)$ for one particular ξ .

Example 7

Consider again Example 3.3, when the second stage is defined as

$$\begin{aligned} -y_1 + y_2 &\leq \xi - x_1, \\ y_1 + y_2 &\leq 2 - x_2, \quad y_1, y_2 \geq 0 \text{ and integer,} \end{aligned}$$

where ξ takes on the values 1 and 2 with equal probability 0.5. It suffices here to consider $x \in K_2(1)$ because $K_2(1) \subset K_2(2)$. First, consider $x = (2, 2)^T$. From Section 5.1, we find a violated continuous induced constraint:

$$x_1 + x_2 \leq 3.$$

Next, consider $x = (1.4, 1.6)^T$. Problem (P1) is

$$\begin{aligned} & \min v_1 + v_2 \\ \text{s. t. } & -y_1 + y_2 - v_1 \leq -0.4, \\ & y_1 + y_2 - v_2 \leq 0.4, \\ & y_1, y_2 \geq 0 \text{ and integer}, \end{aligned}$$

where v_1 and v_2 correspond to v^- in (6.1) and v^+ is not needed due to the inequality form of the constraints. The optimal solution of the continuous relaxation of (P1) is given by the following dictionary:

$$\begin{aligned} w &= v_1 + v_2, \\ y_1 &= 0.4 + y_2 + s_1 - v_1, \\ s_2 &= 0 - 2y_2 - s_1 + v_1 + v_2. \end{aligned}$$

Its solution is $w = 0$, which implies $x \in C_2(1)$. However, y_1 is not integer. Branching creates two nodes, $y_1 \leq 0$ and $y_1 \geq 1$, respectively. In the first branch, the bound $y_1 \leq 0$ creates the additional constraint $y_1 + s_3 = 0$. After one dual iteration, the following optimal dictionary is obtained:

$$\begin{aligned} w &= 0.4 + y_2 + s_1 + s_3 + v_2, \\ y_1 &= 0 - s_3, \\ s_2 &= 0.4 - y_2 + s_3 + v_2, \\ v_1 &= 0.4 + y_2 + s_1 + s_3. \end{aligned}$$

Associating the dual variables $(-1, 0, -1)$ with the right-hand sides $(1 - x_1, 2 - x_2, 0)$, one obtains the feasibility cut, $x_1 - 1 \leq 0$, for this branch.

Similarly, in the second branch, the bound $y_1 \geq 1$ creates a constraint $y_1 - s_3 = 1$. After two dual iterations, the optimal dictionary is:

$$\begin{aligned} w &= 0.6 + y_2 + s_2 + s_3 + v_1, \\ y_1 &= 1 + s_3, \\ v_2 &= 0.6 + y_2 + s_2 + s_3, \\ s_1 &= 0.6 - y_2 + s_3 + v_1. \end{aligned}$$

Associating the dual variables $(0, -1, 1)$ to the right-hand sides $(1 - x_1, 2 - x_2, 1)$, one obtains the feasibility cut, $x_2 - 1 \leq 0$, for the second branch. Thus, $R = 2$, as the solutions in the two nodes satisfy the integrality requirement and are thus

terminal. The feasibility cut is thus that either $x_1 - 1 \leq 0$ or $x_2 - 1 \leq 0$ must be satisfied. Because we also have $x_1 \leq 2$ and $x_2 \leq 2$, we may take $M_1 = M_2 = 1$ so that we have to impose the following set of conditions:

$$\begin{aligned} x_1 &\leq 2 - \delta_1, \\ x_2 &\leq 2 - \delta_2, \\ \delta_1 + \delta_2 &\geq 1, \\ \delta_1, \delta_2 &\in \{0, 1\}. \end{aligned}$$

b. Optimality cuts

We consider here a multicut approach,

$$\theta = \sum_{k=1}^K \theta_k,$$

where, as usual, K denotes the cardinality of Ξ . We search for optimality cuts on a given θ_k . Based on branching on the second-stage problem, one obtains a partition of $\Re_+^{n_2}$ into R terminal nodes $Y^\rho = \{y \mid a^\rho \leq y \leq b^\rho\}$, $\rho = 1, \dots, R$. The objective value of the second-stage program over Y^ρ is

$$Q^\rho(x^v, \xi_k) = \min\{q^T y \mid Wy = h(\xi^k) - T(\xi^k)x^v, a^\rho \leq y \leq b^\rho\}.$$

It is the solution of a linear program that by classical duality theory is also

$$Q^\rho(x^v, \xi_k) = (\pi^\rho)^T(h(\xi^k) - T(\xi^k)x^v) + (\underline{\pi}^\rho)^T a^\rho + (\bar{\pi}^\rho)^T b^\rho,$$

where π^ρ , $\underline{\pi}^\rho$, and $\bar{\pi}^\rho$ are the dual variables associated with the original constraints, lower and upper bounds on $y \in Y^\rho$, respectively.

To simplify notation, we represent this expression as:

$$Q^\rho(x^v, \xi_k) = (\sigma_k^\rho)^T x^v + \tau_k^\rho,$$

with $(\sigma_k^\rho)^T = -(\pi^\rho)^T T(\xi^k)$ and $\tau_k^\rho = (\pi^\rho)^T h(\xi^k) + (\underline{\pi}^\rho)^T a^\rho + (\bar{\pi}^\rho)^T b^\rho$. By duality theory, we know that $Q^\rho(x, \xi^k) \geq (\sigma_k^\rho)^T x^v + \tau_k^\rho$. Moreover, $Q(x, \xi^k) = \min_{\rho=1, \dots, R} Q^\rho(x, \xi^k)$. Thus,

$$\theta_k \geq p_k \left(\min_{\rho=1, \dots, R} (\sigma_k^\rho)^T x^v + \tau_k^\rho \right). \quad (6.2)$$

Note that some of the terminal nodes may be infeasible, in which case their dual solutions contain unbounded rays with dual objective values going to ∞ so that the minimum is in practice restricted to the feasible terminal nodes.

This expression takes the form of a Gomory function, as expected. Again, it unfortunately requires R extra binary variables to be included in a mixed integer linear representation. This makes the branching on the second-stage very often computationally unattractive.

Example 8

Consider the second-stage program

$$E_{\xi} \min\{-8y_1 - 9y_2 \text{ s. t. } 3y_1 + 2y_2 \leq \xi, -y_1 + y_2 \leq x_1, y_2 \leq x_2, y \geq 0, \text{ integer}\}.$$

Consider the value $\xi_1 = 8$ and $\bar{x} = (0, 6)^T$. The optimal dictionary of the continuous relaxation of the second-stage program is:

$$\begin{aligned} z &= -136/5 + 17s_1/5 + 11s_2/5, \\ y_1 &= 8/5 - s_1/5 + 2s_2/5, \\ y_2 &= 8/5 - s_1/5 - 3s_2/5, \\ s_3 &= 22/5 + s_1/5 + 3s_2/5, \end{aligned}$$

where s_1 , s_2 , and s_3 are the slack variables of the three constraints. Branching on y_1 gives two nodes, $y_1 \leq 1$ and $y_1 \geq 2$, which turn out to be the only two terminal nodes. For the first node, adding the constraint $y_1 + s_4 = 1$ yields the following dictionary after one dual iteration:

$$\begin{aligned} z &= -17 + 9s_2 + 17s_4, \\ s_1 &= 3 + 2s_2 + 5s_4, \\ y_2 &= 1 - s_2 - s_4, \\ s_3 &= 5 + s_2 + s_4, \\ y_1 &= 1 - s_4. \end{aligned}$$

We thus have dual variables $(0, -9, 0)$ associated with the right-hand side $(8, x_1, x_2)$ of the constraints and -17 associated with the bound 1 on y_1 . Hence, $Q^1(\bar{x}, 8) = -9x_1 - 17$.

Similarly, we add $y_1 - s_4 = 2$ for the second node. We obtain:

$$\begin{aligned} z &= -25 + 9/2s_1 + 11/2s_4, \\ y_1 &= 2 + s_4, \\ y_2 &= 1 - s_1/2 - 3/2s_4, \\ s_3 &= 5 + s_1/2 + 3/2s_4, \\ s_2 &= 1 + s_1/2 + 5/2s_4. \end{aligned}$$

We now have dual variables, $(-9/2, 0, 0)$, associated with the right-hand side $(8, x_1, x_2)$ of the constraints and $11/2$ associated with the lower bound 2 on y_1 . Hence, $Q^2(\bar{x}, 8) = -25$. Applying (6.2), we conclude that

$$\theta_1 \geq p_1 \min(-9x_1 - 17, -25), \quad (6.3)$$

where p_1 is the probability of $\xi = \xi_1$.

Exercises

1. Consider Example 7. We have seen that $x \in K_2(1)$ if $x_1 \leq 2$, $x_2 \leq 2$ and either $x_1 \leq 1$ or $x_2 \leq 1$. Thus, the feasibility set is the union of two sets.

Apply Proposition 11 in two cases:

- (a) if $x = (2, 2)^T$;
- (b) if $x = (1.4, 1.6)^T$.

- Show that the disjunctive cut formed in (a) is the same as the continuous induced cut: $x_1 + x_2 \leq 3$. (This example can be found in Section 7.8b.)
- Show that no violated disjunctive cut is obtained in (b).

2. Consider Example 8.

- (a) Compare the cut (6.3) with the one obtained by L -shaped cut for $\xi_1 = 8$. Show that (6.3) is stronger for $x_1 \leq 1.5$.
- (b) Assume $2x_1 + x_2 \leq 6$. Convexifying (6.3) over $2x_1 + x_2 \leq 6$, $x_1 \geq 0$, $x_2 \geq 0$ gives a line passing by $(0, -25p_1)$ and $(3, -44p_1)$ in the (x_1, θ_1) space, namely $\theta_1 \geq p_1(-25 - \frac{19}{3}x_1)$. This convexification is stronger than the L -shaped cut only for $x_1 \leq 33/62$.

7.7 Extensive Forms and Decomposition

Problems with mixed integer second-stage can sometimes be solved by decomposing the second-stage variables into their discrete parts and continuous parts. Assuming a mixed second stage with binary variables, one can divide $y(\omega)^T = (y_B(\omega)^T, y_C(\omega)^T)$ where $y_B(\omega)$ is the vector of binary variables and $y_C(\omega)$ the vector of continuous variables. Partitioning q and W in a similar fashion, the classical two-stage program becomes

$$\begin{aligned} \min z &= c^T x + E_{\xi} q_B^T(\omega) y_B(\omega) + E_{\xi} Q(x, y_B(\omega), \omega) \\ \text{s. t. } Ax &= b, \\ x &\in X, \quad y_B(\omega) \in Y_B(\omega), \end{aligned}$$

where

$$\begin{aligned} Q(x, y_B(\omega), \omega) &= \min\{q_C^T(\omega)y_C(\omega) \\ &\quad \text{s. t. } W_C y_C(\omega) \leq h(\omega) - T(\omega)x - W_B y_B(\omega), y_C(\omega) \in Y_C(\omega)\}. \end{aligned}$$

When ξ is a discrete random variable, this amounts to writing down the extensive form for the second-stage binary variables. When the number of realizations of ξ remains low, such a program is still solvable by the ordinary L -shaped method. An extension of this idea to a three-stage problem in the case of acquisition of resources can be found in Bienstock and Shapiro [1988].

The same idea applies for multistage stochastic programs having the block separable property defined in Section 3.4, provided the discrete variables correspond to the aggregate level decisions and the continuous variables correspond to the detailed level decisions. Then the multistage program is equivalent to a two-stage stochastic program, where the first stage is the extensive form of the aggregate level problems and the value function of the second stage for one realization of the random vector is the sum, weighted by the appropriate probabilities of the detailed level recourse functions for that realization and all its successors. This result is detailed in Louveaux [1986], where examples are provided.

Example 9

As an illustration, consider the warehouse location problem similar to those studied in Section 2.4. As usual, let

$$x_j = \begin{cases} 1 & \text{if plant } j \text{ is open,} \\ 0 & \text{otherwise,} \end{cases}$$

with fixed-cost c_j , and v_j , the size of plant j , with unit investment cost g_j , be the first-stage decision variables. Assume $k = 1, \dots, K$ realizations of the demands d_i^k in the second stage. Let y_{ij}^k be the fraction of demand d_i^k served from j , with unit revenue q_{ij} (see Section 2.4c). Now, assume the possibility exists in the second stage to extend open plants by an extra capacity (size) of fixed value e_j at cost r_j . For simplicity, assume this extension can be made immediately available (zero construction delay).

To this end, let

$$w_j^k = \begin{cases} 1 & \text{if extra capacity is added to } j \\ & \quad \text{when the second-stage realization is } k, \\ 0 & \text{otherwise.} \end{cases}$$

The two-stage stochastic program would normally read as

$$\begin{aligned}
& \max - \sum_{j=1}^n c_j x_j - \sum_{j=1}^n g_j v_j + \sum_{k=1}^K p_k \left(\max \sum_{i=1}^m \sum_{j=1}^n q_{ij} y_{ij}^k - \sum_{j=1}^n r_j w_j^k \right) \\
\text{s. t. } & \sum_{j=1}^n y_{ij}^k \leq 1, \quad k = 1, \dots, K, \quad i = 1, \dots, m, \\
& x_j \in \{0, 1\}, \quad v_j \geq 0, \quad j = 1, \dots, n, \\
& \sum_{i=1}^m d_i^k y_{ij}^k - e_j w_j^k \leq v_j, \quad k = 1, \dots, K, \quad j = 1, \dots, n, \\
& 0 \leq y_{ij}^k \leq x_j, \quad i = 1, \dots, m, \quad j = 1, \dots, n, \\
& \quad k = 1, \dots, K, \\
& w_j^k \leq x_j, \quad j = 1, \dots, n, \quad k = 1, \dots, K, \\
& w_j^k \in \{0, 1\}, \quad j = 1, \dots, n, \quad k = 1, \dots, K.
\end{aligned}$$

Using the extensive form for the binary variables, w_j^k 's transforms it into

$$\begin{aligned}
& \max - \sum_{j=1}^n c_j x_j - \sum_{j=1}^n g_j v_j - \sum_{j=1}^n \sum_{k=1}^K p_k r_j w_j^k + \sum_{k=1}^K p_k \max \sum_{i=1}^m \sum_{j=1}^n q_{ij} y_{ij}^k \\
\text{s. t. } & x_j \in \{0, 1\}, \quad v_j \geq 0, \quad j = 1, \dots, n, \\
& \sum_{j=1}^n y_{ij}^k \leq 1, \quad i = 1, \dots, m, \quad k = 1, \dots, K, \\
& w_j^k \leq x_j, \quad \sum_{i=1}^m d_i^k y_{ij}^k \leq v_j + e_j w_j^k, \quad j = 1, \dots, n, \\
& \quad k = 1, \dots, K, \\
& w_j^k \in \{0, 1\}, \quad 0 \leq y_{ij}^k \leq x_j, \quad i = 1, \dots, m, \\
& \quad j = 1, \dots, n, \quad k = 1, \dots, K.
\end{aligned}$$

Thus, at the price of expanding the first-stage program, one obtains a second stage that enjoys the good properties of continuous programs.

When the stochastic programs with mixed-integer second-stage cannot be efficiently decomposed as above, then it can be solved through a scenario decomposition approach. In this method, the nonanticipativity constraints are subjected to Lagrangian relaxation to create mixed-integer programs which are separable in the realizations of the random vector. Details on the method can be found in Carøe and Schultz [1999].

Exercises

1. In Example 9, assume a given construction delay for the warehouses in the second stage. Is it still possible to decompose the second stage?

7.8 Short Reviews

a. Branch-and-bound

Consider the following integer program

$$\begin{aligned} z &= \min 3y_1 + 2y_2 \\ \text{s. t. } &2y_1 + 3y_2 \geq 9, \\ &-3y_1 + 3y_2 \leq 5, \\ &y_1, y_2 \geq 0, \text{ integer} \end{aligned}$$

Optimize: First consider the LP-relaxation, i.e. the same problem where the requirement “ y integer” is removed. Its solution is easily obtained through your favorite LP-solver or through a graphical method. It is $z = 7.333$, $y = (0.8, 2.467)^T$. Let Y denote the second-stage polyhedron for this relaxation.

Bounding: On any polyhedron, the integer solution is no better than the continuous one. The objective value of the LP-relaxation is thus a lower bound on the solution of the integer program. It can be rounded down as the objective must be integer, so $\underline{z} = 8$. We may take $\bar{z} = \infty$, where \underline{z} and \bar{z} denote lower and upper bounds on the optimal solution.

Branching: as y is fractional, we may branch on either component. Say we branch on y_2 . The current value is $y_2 = 2.467$. Any integer solution must satisfy either $y_2 \leq 2$ or $y_2 \geq 3$. This dichotomy excludes the current solution. It does not eliminate any integer point. Branching consists of considering two nodes: $Y_1 = Y \cap \{y_2 \leq 2\}$ and $Y_2 = Y \cap \{y_2 \geq 3\}$. The list of nodes is denoted by $\Lambda = \{Y_1, Y_2\}$.

Select a Node and Reoptimize: We (arbitrarily) select Y_1 and reoptimize the LP-relaxation on Y_1 . Its solution is $z = 8.5$, $y = (1.5, 2)^T$.

Branching: as $y_1 = 1.5$ is fractional, we create two new nodes: $Y_3 = Y_1 \cap \{y_1 \leq 1\}$ and $Y_4 = Y_1 \cap \{y_1 \geq 2\}$. Y_1 is removed from the list. $\Lambda = \{Y_2, Y_3, Y_4\}$.

Select a Node and Reoptimize: We select Y_3 and reoptimize the LP-relaxation on Y_1 . It has no feasible solution. Y_3 is *fathomed*. This means it is removed from the list and does not need any further branching (which would not help in creating a feasible solution). $\Lambda = \{Y_2, Y_4\}$.

Select a Node and Reoptimize: We select Y_4 and reoptimize the LP-relaxation. Its solution is $z = 9.333$, $y = (2, 1.667)^T$.

Branching: as $y_2 = 1.667$ is fractional, we create two new nodes: $Y_5 = Y_4 \cap \{y_2 \leq 1\}$ and $Y_6 = Y_4 \cap \{y_2 \geq 2\}$. Y_4 is removed from the list. $\Lambda = \{Y_2, Y_5, Y_6\}$.

Select a Node and Reoptimize: We select Y_5 and reoptimize the LP-relaxation. Its solution is $z = 11$, $y = (3, 1)^T$.

Updating the Incumbent: As y is integer and $z < \bar{z}$, the best feasible solution becomes $y = (3, 1)^T$ and $\bar{z} = 11$. Y_5 is fathomed (as they are no better integer solutions in Y_5). $\Lambda = \{Y_2, Y_6\}$.

Select a Node and Reoptimize : We select Y_6 and reoptimize the LP-relaxation. Its solution is $z = 10$, $y = (2, 2)^T$.

Updating the Incumbent: As y is integer and $z < \bar{z}$, the best feasible solution becomes $y = (2, 2)^T$ and $\bar{z} = 10$. Node Y_6 is fathomed. $\Lambda = \{Y_2\}$.

Select a Node and Reoptimize: We select Y_2 and reoptimize the LP-relaxation. Its solution is $z = 10$, $y = (1.333, 1)^T$. Y_2 is fathomed: no solution in Y_2 can be better than 10, which is the value of the current best solution. The list is empty. The algorithm terminates with optimal solution $y = (2, 2)^T$ and $z = 10$.

To summarize, branching occurs at nodes having a fractional solution. Fathoming occurs when the LP-relaxation of a node is infeasible, has an integer solution or has a solution whose value is worse than the current incumbent. Branch-and-bound is only a part of the techniques used for solving large MIP's. It is combined with cut generation, reduced cost fixing, preprocessing, special-ordered set (SOS) or generalized upper bound (GUB) branching, and primal heuristics to cite some of the most important.

b. A simple example of valid inequalities

Consider the following binary program

$$\begin{aligned} & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ & \text{s. t. } 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ & \quad y_1, \dots, y_4 \in \{0, 1\}. \end{aligned}$$

The so-called *cover inequalities* can be found by a simple reasoning. If we consider a solution s.t. $y_3 = y_4 = 0$, the constraint cannot be satisfied. Thus, at least one of the two variables must be 1. This can be expressed as

$$y_3 + y_4 \geq 1,$$

which is a cover inequality. It is said to be valid as it must be satisfied by any binary solution. At the same time, it cannot replace the original constraint.

Similarly, the original constraint cannot be satisfied if $y_2 = y_3 = 0$, or if $y_1 = y_2 = y_4 = 0$ implying

$$\begin{aligned} y_2 + y_3 &\geq 1, \\ y_1 + y_2 + y_4 &\geq 1. \end{aligned}$$

respectively.

Three comments are in line here. First, there are more valid inequalities than the above three. For instance, $y_1 + y_2 + y_3 \geq 1$ is also valid. However, it is implied by $y_2 + y_3 \geq 1$. Second, reformulation of an integer program with several constraints may lead to a very large number of valid inequalities. In practice, the idea is to only add those which are violated at the current iterate point. Going back to our example, its LP solution is $y = (1, 1, 0.2, 0)^T$. (This is easily checked as the variables in the example are put in increasing order ($3/2 \leq 7/4 \leq 9/5 \leq 6/3$) of the ratio between the objective coefficient and the constraint coefficient.) Of the four valid inequalities, only the first one $y_3 + y_4 \geq 1$ is violated, as $y_3 + y_4 = 0.2$. Adding $y_3 + y_4 \geq 1$ reduces the number of fractional solutions without changing the binary solutions. It turns out that the LP with the addition of the cut $y_3 + y_4 \geq 1$ has a spontaneous optimal integer solution $y = (1, 0, 1, 0)^T$ which is thus the optimal solution of the integer program.

Third, the valid inequalities depend on the r.h.s. Consider the solution of the same problem where the right-hand side is 8 :

$$\begin{aligned} \min \quad & 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ \text{s. t. } & 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 8, \\ & y_1, \dots, y_4 \in \{0, 1\}. \end{aligned}$$

The non-dominated valid inequalities become: $y_2 + y_3 + y_4 \geq 2$ and $y_1 + y_3 \geq 1$. The first inequality can be justified as follows: if $y_1 = 1$, then $4y_2 + 5y_3 + 3y_4 \geq 6$ must hold, which requires at least two variables to be 1. Note that this inequality is not valid when the r.h.s. is 7. A constraint like $y_3 + y_4 \geq 1$ is still valid but dominated by $y_2 + y_3 + y_4 \geq 2$.

The solution of the LP relaxation is $y = (1, 1, 0.4, 0)^T$. It violates the cut $y_2 + y_3 + y_4 \geq 2$. The LP relaxation with the addition of this single cut gives a fractional solution. The LP relaxation with the addition of $y_2 + y_3 + y_4 \geq 2$ and $y_1 + y_3 \geq 1$ gives the optimal solution $y = (0, 0, 1, 1)^T$.

c. Disjunctive cuts

c.1 Union of Sets

Proposition 17. If $P^i = \{x \in \mathbb{R}_+^n \mid A^i x \geq b^i\}$ for $i = 0, 1$ are two nonempty polyhedra, then $\pi^T x \geq \pi_0$ is a valid inequality for $\text{co}(P^0 \cup P^1)$ if and only if there exists $u_0, u_1 \geq 0$ such that $\pi \geq (u^i)^T A^i$ and $\pi_0 \leq (u^i)^T b^i$ for $i = 0, 1$.

Proof: Let $P^i = \{x \in \Re^n_+ \mid A^i x \geq b^i\}$ for $i = 0, 1$ be two nonempty polyhedra. We search for a valid inequality for $\text{co}(P^0 \cup P^1)$. Any nonnegative combination of the constraints in one of the P^i 's gives a valid constraint for that P^i . Let $u^i \geq 0$ be the vector representing this combination. Thus $(u^i)^T A^i x \geq (u^i)^T b^i$ is valid for P^i . If we do the same in both sets, we may construct a valid inequality for $\text{co}(P^0 \cup P^1)$ of the form $\pi^T x \geq \pi_0$ by taking $\pi \geq (u^i)^T A^i$ and $\pi_0 \leq (u^i)^T b^i$ for $i = 0, 1$. Indeed, if $x \in \text{co}(P^0 \cup P^1)$, it must belong to one of the two polyhedra. Say it belongs to P^i . Then, $\pi^T x \geq (u^i)^T A^i x \geq (u^i)^T b^i \geq \pi_0$ which proves the validity of the cut. \square

Example: Let $P^0 = \{x \in \Re^2_+ \mid x_1 \leq 1, x_2 \leq 3\}$ and $P^1 = \{x \in \Re^2_+ \mid 4x_1 + 2.5x_2 \leq 10\}$.

Say, we want a disjunctive cut that separates the current point $x^v = (1.8, 2.4)^T$. Then, the cut is obtained by solving an LP consisting of maximizing the violation $\pi_0 - \pi^T x^v$, under the constraints of Proposition 11. To be bounded, this LP needs some normalizing. One possibility is to take $-1 \leq \pi_0 \leq 1, -e \leq \pi \leq e$. We obtain:

$$\begin{aligned} z &= \max \pi_0 - 1.8\pi_1 - 2.4\pi_2 \\ \text{s. t. } \pi_1 &\geq -u_1^0, & \pi_1 &\geq -4u^1, \\ \pi_2 &\geq -u_2^0, & \pi_2 &\geq -2.5u^1, \\ \pi_0 &\leq -u_1^0 - 3u_2^0, & \pi_0 &\leq -10u^1, \\ u &\geq 0, & -e &\leq \pi \leq e, & -1 &\leq \pi_0 \leq 1. \end{aligned}$$

The solution is $z = 0.2$, $u_1^0 = 0.4$, $u_2^0 = 0.2$, $u^1 = 0.1$, $\pi = (-0.4, -0.2)^T$, $\pi_0 = -1$. The disjunctive cut is $-0.4x_1 - 0.2x_2 \geq -1$. At $x^v = (1.8, 2.4)^T$, the cut is violated by 0.2 which is the value of z . The cut can also be written as $2x_1 + x_2 \leq 5$. The line $2x_1 + x_2 = 5$ passes through $(1, 3)^T$ and $(2.5, 0)^T$, which are extreme points of P^0 and P^1 , respectively.

c.2 Disjunction on a binary variable

We consider the disjunction $P^0 = Y \cap \{y \in \Re^{n_2}_+ \mid y_j \leq 0\}$ and $P^1 = Y \cap \{y \in \Re^{n_2}_+ \mid y_j \geq 1\}$ for some fractional variable.

Proposition 18. *The inequality $\pi^T y \geq \pi_0$ is valid if and only if there exists $u^i, v^i, w^i \geq 0$ for $i = 0, 1$ such that*

$$\begin{aligned} \pi &\geq (u^0)^T W - v^0 - w^0 e_j, \\ \pi &\geq (u^1)^T W - v^1 + w^1 e_j, \\ \pi_0 &\geq (u^0)^T d - e^T v^0, \\ \pi_0 &\geq (u^1)^T d - e^T v^1 + w^1. \end{aligned}$$

The cut is obtained by solving an LP consisting of maximizing the violation $\pi_0 - \pi^T y^v$, under the constraints defined in Proposition 12, where y^v is the current fractional solution. To be bounded, this LP needs some normalizing. One possibility is to take $-1 \leq \pi_0 \leq 1$, $-e \leq \pi \leq e$.

Example: Consider again the program:

$$\begin{aligned} & \min 3y_1 + 7y_2 + 9y_3 + 6y_4 \\ & \text{s. t. } 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ & \quad y_1, \dots, y_4 \in \{0, 1\}. \end{aligned}$$

Its LP relaxation has solution $y = (1, 1, 0.2, 0)^T$ (see Section 7.8b.). y_3 is the only fractional variable and is thus used for the disjunction:

$$\begin{aligned} P^0 &= \{y \geq 0 \mid 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ &\quad y_1 \leq 1, y_2 \leq 1, y_3 \leq 1, y_4 \leq 1, y_3 \leq 0\} \\ \text{and } P^1 &= \{y \geq 0 \mid 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ &\quad y_1 \leq 1, y_2 \leq 1, y_3 \leq 1, y_4 \leq 1, y_3 \geq 1\} \end{aligned}$$

or

$$\begin{aligned} P^0 &= \{y \geq 0 \mid 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ &\quad -y_1 \geq -1, -y_2 \geq -1, -y_3 \geq -1, -y_4 \geq -1, -y_3 \geq 0\} \\ \text{and } P^1 &= \{y \geq 0 \mid 2y_1 + 4y_2 + 5y_3 + 3y_4 \geq 7, \\ &\quad -y_1 \geq -1, -y_2 \geq -1, -y_3 \geq -1, -y_4 \geq -1, y_3 \geq 1\}. \end{aligned}$$

The disjunctive cut is obtained through the solution of:

$$\begin{aligned} z &= \max \pi_0 - \pi_1 - \pi_2 - 0.2\pi_3 \\ \text{s. t. } & \pi_1 \geq 2u^0 - v_1^0, \quad \pi_1 \geq 2u^1 - v_1^1, \\ & \pi_2 \geq 4u^0 - v_2^0, \quad \pi_2 \geq 4u^1 - v_2^1, \\ & \pi_3 \geq 5u^0 - v_3^0 - w^0, \quad \pi_3 \geq 5u^1 - v_3^1 + w^1, \\ & \pi_4 \geq 3u^0 - v_4^0, \quad \pi_4 \geq 3u^1 - v_4^1, \\ & \pi_0 \leq 7u^0 - v_1^0 - v_2^0 - v_3^0 - v_4^0, \\ & \pi_0 \leq 7u^1 - v_1^1 - v_2^1 - v_3^1 - v_4^1 + w^1, \\ & u, v, w \leq 0, \quad -e \leq \pi \leq e, \quad -1 \leq \pi_0 \leq 1. \end{aligned}$$

The solution is $z = 0.8/3$, $u^0 = 1/3$, $v^0 = (2/3, 4/3, 0, 0)^T$, $w^0 = 4/3$, $u^1 = 0$, $v^1 = (0, 0, 0, 0)^T$, $w^1 = 1/3$, $\pi = (0, 0, 1/3, 1)^T$, $\pi_0 = 1/3$. The disjunctive cut is $1/3y_3 + y_4 \geq 1/3$, which is currently violated by $0.8/3$. Note however that it is dominated by the cut $y_3 + y_4 \geq 1$ (the cover inequality in Section 7.8b.).

Part IV

Approximation and Sampling Methods

Chapter 8

Evaluating and Approximating Expectations

The evaluation of the recourse function or the probability of satisfying a set of constraints can be quite complicated. This problem is basically one of numerical integration in high dimensions corresponding to the random variables. The general problem requires some form of approximation, such as quadrature formulas, which typically apply to smooth functions in low dimensions without using known convexity properties. In Section 8.1 of this chapter, we review some of these basic procedures, but note that stochastic programs often do not have differentiability as assumed in many numerical schemes but generally do have useful convexity properties.

In the remaining sections of this chapter, we consider approximations that give lower and upper bounds on the expected recourse function value in two-stage problems. The intent of these procedures is to provide progressively tighter bounds until some a priori tolerance has been achieved. This chapter focuses on such deterministic approximation results for two-stage problems. In Chapter 9, we describe approximations for two-stage problems built on Monte Carlo sampling. Chapter 10 discusses both deterministic and random approximation methods for the multistage case.

Section 8.2 in this chapter discusses the most common type of approximations built on discretizations of the probability distribution. The lower bounds are extensions of midpoint approximations, while the upper bounds are extensions of trapezoidal approximations. The bounds are refined using partitions of the region. Other improvements are possible using more tightly constrained moment problem models of the approximation, as described in Section 8.5.

Section 8.3 discusses computational uses for bounds. The goal is to place the bounds effectively into computational methods. We present uses of the bounds in the L -shaped method, inner linearizations, and separable nonlinear programming procedures. Section 8.4 discusses some basic bounding approaches for probabilistic constraints. General forms are presented briefly. These methods are based on fundamental inequalities from probability.

Section 8.5 presents a variety of extensions of the previous bounding approaches. It presents bounds based on approximations of the recourse function. The basic idea

is to bound the objective function above and below by functions that are simply integrated, such as separable functions. We present the basic separable piecewise linear upper bounding function and various methods based on this approach. We also discuss results for particular moment problem solutions. We consider bounds based on second moment information and allowances for unbounded support regions. Finally, Section 8.6 concludes this chapter with basic results on convergence of approximations and bounding procedures. Most of the following results are based on these convergence ideas.

8.1 Direct Solutions with Multiple Integration

In this section, we again consider the basic stochastic program in the form

$$\min_x \{c^T x + \mathcal{Q}(x) \mid Ax = b, x \geq 0\}, \quad (1.1)$$

where \mathcal{Q} is the expected recourse function, $\int_{\Omega} [Q(x, \omega)] P(d\omega)$, where we use $P(d\omega)$ in place of $dF(\omega)$ to allow for general probability measure convergence. We again have

$$Q(x, \omega) = \min_{y(\omega)} \{q(\omega)^T y(\omega) \mid Wy(\omega) = h(\omega) - T(\omega)x, y(\omega) \geq 0\}, \quad (1.2)$$

where we assume two stages and no probabilistic constraints for now.

As we mentioned previously, we can always treat (1.1) as a standard nonlinear program if we can evaluate $\mathcal{Q}(x)$ and perhaps its derivatives. The major difficulty of stochastic programming is, of course, just such an evaluation. These function evaluations all involve multiple integration with potentially large numbers (on the order of 1000 or more) of random variables. This section considers some of the basic techniques from numerical integration that have been attempted in the context of stochastic programming. Remaining sections consider various approximations that lead to computable problems.

Numerical integration procedures are generally built around formulas that apply only in small dimensions (see, e.g., Stroud [1971]). For some special functions defined over specific regions, efficient computations are possible, but these results do not generally carry over to the more general setting of the integrand, $Q(x, \omega)$. This function is piecewise linear in (1.2) as a function of ω and, hence, has many nondifferentiable points. The error analysis from standard smooth integrations (built on Peano's rule) cannot apply. In fact, quadrature formulas built on low-order polynomials may produce poor results when other simple calculations are exact (Exercise 1).

Generalizations of the basic trapezoid and midpoint approaches in numerical integration obtain bounds, however, when convexity properties of Q are exploited. Problem structure is in fact a key to obtaining computable approximations of the multiple integral.

The simple recourse example is the best case for exploitation of problem structure. In this case, $Q(x, \omega)$ becomes separable into functions of each component of $h(\omega)$, the right-hand side vector in (1.2). We obtain $\mathcal{Q}(x) = \sum_{i=1}^{m_2} \mathcal{Q}_i(x)$ as in (3.1.9), which only requires integration with respect to each \mathbf{h}_i separately. As we described in Chapter 5, this allows the use of general nonlinear programming algorithms.

In general, the stochastic linear program recourse function can also be written in terms of bases in W . Suppose the set of bases in W is $\{B_i, i \in I\}$. Let $\pi_i(\omega)^T = q_{B_i}^T B_i^{-1}$. Then

$$Q(x, \omega) = \max_i \{\pi_i(\omega)^T (h(\omega) - T(\omega)x) \mid \pi_i(\omega)^T W \leq q(\omega)^T\}, \quad (1.3)$$

where, if $q(\omega)$ is constant (i.e., not random), the evaluation reduces to finding the maximum value of the inner product over the same feasible set for all ω . With $q(\omega)$ constant,

$$\mathcal{Q}(x) = \sum_{i \in I} \int_{\Omega_i} \{\pi_i^T (h(\omega) - T(\omega)x)\} P(d\omega), \quad (1.4)$$

where $\Omega_i = \{\omega \mid \pi_i^T (h(\omega) - T(\omega)x) \geq \pi_j^T (h(\omega) - T(\omega)x), j \neq i\}$. The integrand in (1.4) is linear; so, we have

$$\mathcal{Q}(x) = \sum_i \pi_i^T (\bar{h}_i - \bar{T}_i x), \quad (1.5)$$

where $\bar{h}_i = \int_{\Omega_i} \mathbf{h}_i P(d\omega)$ and $\bar{T}_i = \int_{\Omega_i} \mathbf{T}_i P(d\omega)$. Thus, if each Ω_i can be found, then the numerical integration reduces to finding the expectations of the random parameters over the regions Ω_i , i.e., the *conditional expectation* on Ω_i . In this case, we can also define a basis B_i from W so that $B_i^T \pi_i = q$ and then $\Omega_i = \{\omega \mid q^T (B_i^{-1} (h(\omega) - T(\omega)x)) \geq q^T (B_j^{-1} (h(\omega) - T(\omega)x)), \forall j \neq i\}$. If integration over the regions Ω_i defined by B_i is sufficiently straightforward, then (1.5) can be used directly. We illustrate this with the following example, which we will also use for bounding approximations in the following sections.

Example 1

Consider the following recourse problem with only \mathbf{h} random:

$$\begin{aligned} Q(x, \xi) &= \min \mathbf{y}_1^+ + \mathbf{y}_1^- + \mathbf{y}_2^+ + \mathbf{y}_2^- + \mathbf{y}_3 \\ \text{s. t. } &\mathbf{y}_1^+ - \mathbf{y}_1^- + \mathbf{y}_3 = \mathbf{h}_1 - x_1, \\ &\mathbf{y}_2^+ - \mathbf{y}_2^- + \mathbf{y}_3 = \mathbf{h}_2 - x_2, \\ &\mathbf{y}_1^+, \mathbf{y}_1^-, \mathbf{y}_2^+, \mathbf{y}_2^-, \mathbf{y}_3 \geq 0, \end{aligned}$$

where \mathbf{h}_i is independently uniformly distributed on $[0, 1]$ for $i = 1, 2$.

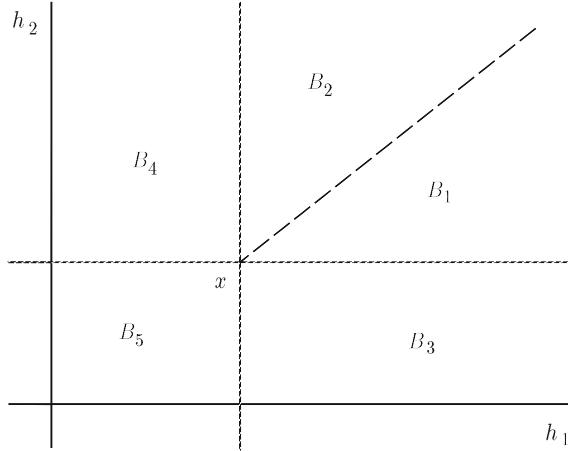


Fig. 1 Optimal basis regions of Example 1.

The optimal basis regions for the solution of this problem are illustrated in Figure 1. Here, the optimal bases are B_1 corresponding to (y_1^+, y_3) , B_2 corresponding to (y_2^+, y_3) , B_3 corresponding to (y_1^-, y_2^-) , B_4 corresponding to (y_1^-, y_2^+) , and B_5 corresponding to (y_1^-, y_2^-) with dual multipliers $\pi_1 = (1, 0)^T$, $\pi_2 = (0, 1)^T$, $\pi_3 = (1, -1)^T$, $\pi_4 = (-1, 1)^T$, and $\pi_5 = (-1, -1)^T$, respectively. Figure 1 shows the regions in which each of these bases is optimal.

We let $p_i = P(\Omega_i)$ for $i = 3, 4, 5$. To make the calculations somewhat simpler, we divide Ω_1 and Ω_2 into two sections each depending on x as $\Omega_1 = \Omega_{10}(x) + \Omega_{11}(x)$ and $\Omega_2 = \Omega_{20}(x) + \Omega_{21}(x)$ where $\Omega_{10}(x) = \{\omega | \omega \in \Omega_1, x_1 \leq h_1(\omega) \leq x_1 + \min(1-x_1, 1-x_2)\}$, $\Omega_{11}(x) = \{\omega | \omega \in \Omega_1, x_1 + \min(1-x_1, 1-x_2) < h_1(\omega) \leq 1\}$, $\Omega_{20}(x) = \{\omega | \omega \in \Omega_2, x_2 \leq h_2(\omega) \leq x_2 + \min(1-x_1, 1-x_2)\}$, and $\Omega_{21}(x) = \{\omega | \omega \in \Omega_2, x_2 + \min(1-x_1, 1-x_2) < h_2(\omega) \leq 1\}$ with corresponding integrals of \mathbf{h} over each of these regions given by $\bar{h}_{10}(x)$, $\bar{h}_{11}(x)$, $\bar{h}_{20}(x)$, and $\bar{h}_{21}(x)$ respectively. In this way, $\Omega_{10}(x)$ and $\Omega_{20}(x)$ are symmetric around the diagonal $x_1 = x_2$ with one of $\Omega_{11}(x)$ and $\Omega_{21}(x)$ corresponding to a rectangular region of positive probability if $1 \geq x_2 > x_1$ or $1 \geq x_1 > x_2$.

With these definitions, we can then write $\mathcal{Q}(x)$ for Example 1 as

$$\mathcal{Q}(x) = \sum_{i=1}^2 \sum_{j=0}^1 \pi_i^T (\bar{h}_{ij}(x) - Tx) + \sum_{i=3}^5 \pi_i^T (\bar{h}_i(x) - Tx). \quad (1.6)$$

Finding the value of \bar{h} for each region then yields the following expression (Exercise 2):

$$\begin{aligned}\mathcal{Q}(x) = & \frac{1}{2}(x_1 + x_1^2 + x_2 - 4x_1x_2 + x_1^2x_2 + x_2^2 + x_1x_2^2 + 2(1-x_2)^2 \max[0, -x_1 + x_2] \\ & + \max[0, x_1 - x_2](2(1-x_1)^2) + \frac{4}{3}(\min[1-x_1, 1-x_2])^3),\end{aligned}$$

for any $x \in [0, 1]^2$.

The regions Ω_i are polyhedral (Exercise 4) in general, which, as in Example 1, yields direct integration procedures when these regions are simple enough to have explicit integration formulas. Unfortunately, this is not often the case for the Ω_i regions that are common in stochastic programs with recourse. As Exercise 2 demonstrates, even in the simple cases of uniform distributions, the expectations over different regions depends on the relative values of the components of x and may require significant computation to find exactly.

In problems with probabilistic constraints, however, there are possibilities for creating deterministic equivalents when the data are, for example, normal as in Theorem 3.18. In general, however, efficient computation requires some form of approximation.

In the following sections, we explore several methods for approximating the value function and its subgradient in stochastic programming. The basic approaches are either approximations with known error bounds or approximations based on Monte Carlo procedures that may have associated confidence intervals. In the remainder of this chapter and Chapter 10, we explore bounding approaches, while in Chapter 9 we also consider methods based on sampling.

Exercises

1. The principle of Gaussian quadrature is to find points and weights on those points that yield the correct integral over all polynomials of a certain degree. For example, we can solve for points, ξ_1 , ξ_2 , and weights, p_1 , p_2 , so that we have a probability ($p_1 + p_2 = 1$) and distribution that matches the mean, ($p_1\xi_1 + p_2\xi_2 = \bar{\xi}$), the second moment, ($p_1\xi_1^2 + p_2\xi_2^2 = \bar{\xi}^{(2)}$), and the third moment, ($p_1\xi_1^3 + p_2\xi_2^3 = \bar{\xi}^{(3)}$). Solve this for a uniform distribution on $[0, 1]$ to yield the two points, 0.211 and 0.788, each with probability 0.5.
 - (a) Verify that this distribution matches the expectation of any polynomial up to degree three over $[0, 1]$.
 - (b) Consider a piecewise linear function, f , with two linear pieces and $0 \leq f(\xi) \leq 1$ for $0 \leq \xi \leq 1$. How large a relative error can the Gaussian quadrature points give? Can you use two other points that are better?
2. Derive the expression of $\mathcal{Q}(x)$ for Example 1 in (1.7) using (1.6).
3. Verify that $\mathcal{Q}(x)$ for Example 1 is convex on $[0, 1]^2$ using (1.7).
4. Show that each region Ω_i is polyhedral.

8.2 Discrete Bounding Approximations

The most common procedures in stochastic programming approximations are to find some relatively low cardinality discrete set of realizations that somehow represents a good approximation of the true underlying distribution or whatever is known about this distribution. The basic procedures are extensions of Jensen's inequality ([1906], generalization of the midpoint approximation) and an inequality due to Edmundson [1956] and Madansky [1959], the *Edmundson-Madansky inequality*, a generalization of the trapezoidal approximation. For convex functions in ξ , Jensen provides a lower bound while Edmundson-Madansky provides an upper bound. Significant refinements of these bounds appear in Huang, Ziemba, and Ben-Tal [1977], Kall and Stoyan [1982] and Frauendorfer [1988b].

We refer to a general integrand $g(x, \xi)$. Our goal is to bound $E(g(x)) = E_\xi[g(x, \xi)] = \int_{\Xi} g(x, \xi)P(d\xi)$. The basic ideas are to partition the support Ξ into a number of different regions (analogous to intervals in one-dimensional integration) and to apply bounds in each of those regions. We let the partition of Ξ be $\mathcal{S}^v = \{S^l, l = 1, \dots, v\}$. Define $\xi^l = E[\xi | S^l]$ and $p^l = P[\xi \in S^l]$. The basic lower bounding result is the following.

Theorem 1. Suppose that $g(x, \cdot)$ is convex for all $x \in D$. Then

$$E(g(x)) \geq \sum_{l=1}^v p^l g(x, \xi^l). \quad (2.1)$$

Proof: Write $E(g(x))$ as

$$\begin{aligned} E(g(x)) &= \sum_{l=1}^v \int_{S^l} g(x, \xi)P(d\xi) \\ &= \sum_{l=1}^v p^l E[g(x, \xi) | S^l] \quad \text{离散化} \\ &\geq \sum_{l=1}^v p^l g(x, E[\xi | S^l]), \end{aligned} \quad (2.2)$$

where the last inequality follows from Jensen's inequality that the expectation of a convex function of some argument is always greater than or equal to the function evaluated at the expectation of its argument, i.e., $E(g(\xi)) \geq g(E(\xi))$ (see Exercise 1). \square

This result applies directly to approximating $\mathcal{Q}(x)$ by $\mathcal{Q}^v(x) = \sum_{l=1}^v p^l Q(x, \xi^l)$. The approximating distribution P^v is the discrete distribution with *atoms*, i.e., points ξ^l of probability $p^l > 0$ for $l = 1, \dots, v$. By choosing \mathcal{S}^{v+1} so that each $S^l \in \mathcal{S}^{v+1}$ is completely contained in some $S'^l \in \mathcal{S}^v$, the approximations actually improve, i.e.,

$$E(g(x)) \geq E^{v+1}(g(x)) \geq E^v(g(x)). \quad (2.3)$$

Various methods can achieve convergence in distribution of the P^v to P . An example is given in Exercise 2.

In general, the goal of refining the partition from v to $v+1$ is to achieve as great an improvement as possible. We will describe the basic approaches; more details appear in Birge and Wets [1986], Frauendorfer and Kall [1988], and Birge and Wallace [1986]. Three basic decisions are to choose the cell, $S^{v^*} \in \mathcal{S}^v$, in which to make the partition, to choose the direction in which to split S^{v^*} , and to choose the point at which to make the split.

The reader should note that this section contains notation specific to bounding procedures. To keep the notation manageable, we reuse some from previous sections, including a and b for endpoints of rectangular regions and c for points within these intervals at which to subdivide the region. For ease of exposition, suppose that the sets S^l are all rectangular, defined by $[a_1^l, b_1^l] \times \cdots \times [a_N^l, b_N^l]$. The most basic refinement scheme for $l = v^*$ is to find i^* and $c_{i^*}^l$ so that $S^l(v)$ splits into $S^l(v+1) = [a_1^l, b_1^l] \times \cdots \times [a_{i^*}^l, c_{i^*}^l] \times [a_N^l, b_N^l]$ and $S^{v+1}(v+1) = [a_1^l, b_1^l] \times \cdots \times [c_{i^*}^l, b_{i^*}^l] \times [a_N^l, b_N^l]$.

If we also have an upper bound $UB(S^l) \geq E[g(x, \xi) \mid \xi \in S^l]$ for each cell S^l , then the most likely choice for S^{v^*} is the cell that maximizes $p_l(UB(S^l) - g(x, \xi^l))$, which bounds the error attributable to the approximation on S^l . Reducing this greatest partition error appears to offer the most hope in reducing the error on the $v+1$ approximation.

The direction choice is less clear. The general idea is to choose a direction in which the function g is “most nonlinear”. The use of subgradient (dual price) information for this process was discussed in Birge and Wets [1986]. Frauendorfer and Kall [1988] improved on this and reported good results by considering all 2^{m+1} pairs, (α_j, β_j) , of vertices of S^l , where $\alpha_j = (\gamma_1^j, \dots, a_i^l, \dots, \gamma_N^j)$ and $\beta_j = (\gamma_1^j, \dots, b_i^l, \dots, \gamma_N^j)$ with $\gamma_i^j = a_i^l$ or b_i^l . Given x , they assume a dual vector, π_{α_j} , at $Q(x, \alpha_j)$ and π_{β_j} at $Q(x, \beta_j)$. Because these represent subgradients of the recourse function $Q(x, \cdot)$, we have $Q(x, \beta_j) - (Q(x, \alpha_j) + \pi_{\alpha_j}^T(\beta_j - \alpha_j)) = \varepsilon_j^1 \geq 0$ and $Q(x, \alpha_j) - (Q(x, \beta_j) + \pi_{\beta_j}^T(\alpha_j - \beta_j)) = \varepsilon_j^2 \geq 0$. They then choose k^* that maximizes $\min\{\varepsilon_k^1, \varepsilon_k^2\}$ over k . They let i^* be i such that α_{k^*} and β_{k^*} differ in the i th coordinate. The position c_{i^*} is then chosen so that $Q(x, \beta^{k^*}) + \pi_{\beta_{k^*}}^T(c_{i^*} - b_{i^*}) = Q(x, \alpha^{k^*}) + \pi_{\alpha_{k^*}}^T(c_{i^*} - a_{i^*})$. (See Figure 2, where we use π for the subgradient at (a_1, b_2) and ρ for the subgradient at (a_1, a_2) .) The general idea is then to choose the direction that yields the maximum of the minimum of linearization errors in each direction.

Refinement schemes clearly depend on having upper bounds available. These bounds are generally based on convexity properties of g and the ability to obtain each ξ in terms of the extreme points. The fundamental result is the following theorem that also appears in Birge and Wets [1986]. In the following, we use P as the measure on Ω instead of Ξ because we wish to obtain a different measure derived from this domain. In context, this change should not cause confusion. We

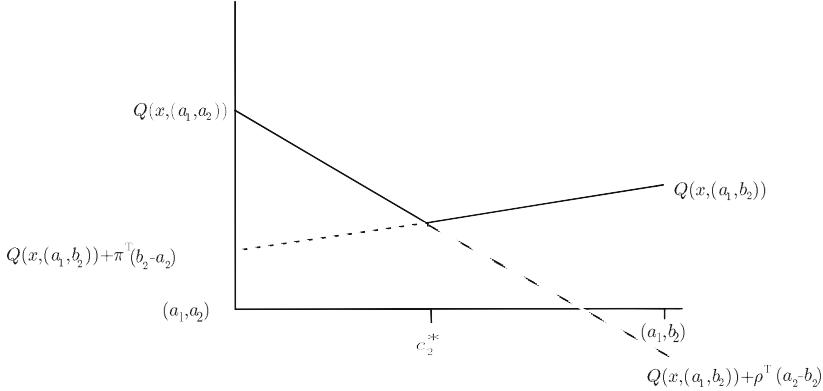


Fig. 2 Choosing the direction according to the maximum of the minimum linearization errors.

also let $\text{ext}\Xi$ be the set of extreme points of $\text{co}\Xi$ and \mathcal{E} is a Borel field of $\text{ext}\Xi$, in this case, the collections of all subsets of $\text{ext}\Xi$.

Theorem 2. Suppose that $\xi \mapsto g(x, \xi)$ is convex and Ξ is compact. For all $\xi \in \Xi$, let $\phi(\xi, \cdot)$ be a probability measure on $(\text{ext}\Xi, \mathcal{E})$, such that

$$\int_{e \in \text{ext}\Xi} e \phi(\xi, de) = \xi, \quad (2.4)$$

and $\omega \mapsto \phi(\xi(\omega), A)$ is measurable for all $A \in \mathcal{E}$. Then

$$E(g(x)) \leq \int_{e \in \text{ext}\Xi} g(x, e) \lambda(de), \quad (2.5)$$

where λ is the probability measure on \mathcal{E} defined by

$$\lambda(A) = \int_{\Omega} \phi(\xi(\omega), A) P(d\omega). \quad (2.6)$$

Proof: Because g is convex in ξ , for ϕ ,

$$g(x, \xi) \leq \int_{e \in \text{ext}\Xi} g(x, e) \phi(\xi, de). \quad (2.7)$$

Substituting $\xi(\omega)$ for ξ and integrating with respect to P , the result in (2.5) is obtained. \square

This result states that if we can choose the appropriate ϕ and find λ , we can produce an upper bound. The key is to make the calculation of λ as simple as possible. Of course, the cardinality of $\text{ext}\Xi$ may also play a role in the computability of the bound.

One way to reduce the cardinality of the supporting extreme points is simply to choose the extreme point that has the highest value as an upper bound. Let this upper bound be $UB^{\max}(x) = \sup_{e \in \text{ext } \Xi} g(x, e) \geq \int_{e \in \text{ext } \Xi} g(x, e) \lambda(de) \geq E(g(x))$ from Theorem 2, regardless of the particular λ . While UB^{\max} may only involve a single extreme point, it is often a poor bound (see the result from Exercise 3). Its calculation also often involves evaluating all the extreme points to maximize the convex function $g(x, \cdot)$.

In general, bounds built on the result in Theorem 2 construct the probability measure λ so that each extreme point e_j of Ξ has some weight, $p_j = \lambda(e_j)$. The following bounds, described in more detail in Birge and Wets [1986], find these weights in various cases. The first is general but involves some optimization. The second involves simplicial regions, and the third uses rectangular regions.

Because λ is constructed to be consistent with the distribution of ξ , we must have that

$$\begin{aligned} \int_{\Omega} \xi(\omega) P(d\omega) &= \int_{\Omega} \int_{e \in \text{ext } \Xi} e \phi(\xi(\omega), de) P(d\omega) \\ &= \int_{e \in \text{ext } \Xi} e \int_{\Omega} \phi(\xi(\omega), de) P(d\omega) \\ &= \int_{e \in \text{ext } \Xi} e \lambda(de). \end{aligned} \quad (2.8)$$

Hence, $\lambda \in \mathcal{P} = \{\mu \mid \mu \text{ is a probability measure on } \mathcal{E}, \text{ and } E_{\mu}[e] = \bar{\xi}\}$. The next upper bound, originally suggested by Madansky [1960] and extended by Gassmann and Ziembba [1986], builds on this idea by finding an upper bound through a linear program to maximize the objective expectation over all probability measures in \mathcal{P} . We write this bound as UB^{mean} , where

$$\begin{aligned} UB^{mean}(x) &= \max_{p_1, \dots, p_K} \sum_{k=1}^K p_k g(x, e_k) \\ \text{s. t. } &\sum_{k=1}^K p_k e_k = \bar{\xi}, \\ &\sum_{k=1}^K p_k = 1, \quad p_k \geq 0, \quad k = 1, \dots, K. \end{aligned} \quad (2.9)$$

As we shall see in Section 8.5, the probability measure that optimizes the linear program in (2.9) is the solution of a moment problem in which only the first moment is known. Another interpretation of this bound is that it represents the worst possible outcome if only the mean of the random variable is known. Optimizing with this bound, therefore, brings some form of risk avoidance if no other distribution information is available.

Assuming that the dimension of $\text{co } \Xi$ is N , Carathéodory's theorem states that $\bar{\xi}$ must be expressable as a convex combination of at most $N+1$ points in $\text{ext } \Xi$. Finding these $N+1$ points may, however, again involve computations for the values

at all extreme points. The number of extreme point representations may be much higher than $N+1$ if Ξ is, for example, rectangular, but lower if, for example, Ξ is a *simplex*, i.e., a convex combination of $N+1$ points, ξ^i , $i = 1, \dots, N+1$, such that $\xi^i - \xi^1$ are linearly independent for $i > 1$. The representation of interior points is, in fact, unique. Indeed, the p_j in this case are called the *barycentric coordinates* of $\bar{\xi}$.

Although Ξ may not be simplicial itself, it is often possible to extend $Q(x, \cdot)$ from Ξ to some simplex Σ including Ξ . The bound obtained with this approach is written UB^Σ . In this bound, the number of points used in the evaluation remains one more than the dimension of the affine hull of Ξ . Frauendorfer [1989, 1992] gives more details about this form of approximation and various methods for its refinement.

Often, Ξ is given as a rectangular region. In this case, the number of extreme points is 2^N . The number of simplices containing $\bar{\xi}$ may also be exponential in N . With relatively complete information about the correlations among random variables, however, bounds can be obtained that assign the same weight to each extreme point of Ξ (or a rectangular enclosing region), regardless of the value of x . This attribute is quite beneficial in algorithms where x may change frequently as an optimal solution is sought.

The basic bounds for rectangular regions follow Edmundson and Madansky, for which, the name *Edmundson-Madansky (E-M) bound* is used. They begin with the trapezoidal type of approximation on an interval. Here, if $\Xi = [a, b]$, we can easily construct $\phi(\xi, \cdot)$ in Theorem 2 as $\phi(\xi, a) = \pi(\xi)$ and $\phi(\xi, b) = 1 - \pi(\xi)$, where $\pi(\xi) = \frac{b-\xi}{b-a}$. Integrating over ω , we obtain

$$\begin{aligned}\lambda(a) &= \int_{\Omega} \phi(\xi(\omega), a) P(d\omega) \\ &= \int_{\Omega} \frac{b - \xi(\omega)}{b - a} P(d\omega) \\ &= \frac{b - \bar{\xi}}{b - a}.\end{aligned}\tag{2.10}$$

We then also have $\lambda(b) = \frac{\bar{\xi} - a}{b - a}$. The bound obtained is $UB^{EM}(x) = \lambda(a)g(x, a) + \lambda(b)g(x, b) \geq E(g(x))$. Observe in Figure 3 that this bound represents approximating the integrand $g(x, \cdot)$ with the values formed as convex combinations of extreme point values. This is the same procedure as in trapezoidal approximation for numerical integration except that the endpoint weights may change for nonuniform probability distributions.

The *E-M* bound on an interval extends easily to multiple dimensions, where $\Xi = [a_1, b_1] \times \dots \times [a_N, b_N]$ if either $g(x, \cdot)$ is separable in the components of ξ , in which case, the bound is applied in each component separately, or the components of ξ are stochastically independent. In this case, the bound is developed in each component $i = 1$ to N in order so that the full independent ξ_i bound contains the product of all combinations of each interval bound, i.e.,

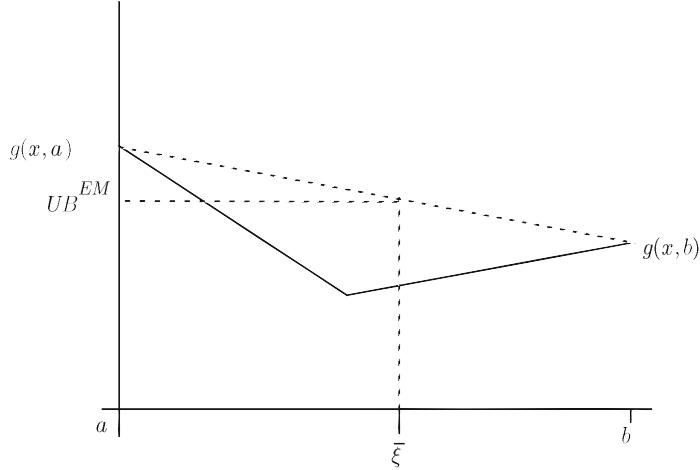


Fig. 3 Example of the Edmundson-Madansky bound on an interval.

$$UB^{EM-I}(x) = \sum_{e \in ext \Xi} \left(\prod_{i=1}^N \frac{|\bar{\xi}_i - e_i|}{b_i - a_i} \right) g(x, e), \quad (2.11)$$

where Ξ is again assumed polyhedral.

Example 1 (continued)

We return again to Example 1 and suppose an initial solution, $\bar{x} = (0.3, 0.3)^T$. From (1.7), $\mathcal{Q}(\bar{x}) = 0.466$. Our initial lower bound using the mean of the random vector is then the Jensen lower bound, $LB_1 = Q(\bar{x}, \xi = \bar{h} = (0.5, 0.5)^T) = 0.2$.

The upper bounds can be found using the values at the extreme points of the support of \mathbf{h} . These values are $Q(\bar{x}, (0, 0)^T) = 0.6$, $Q(\bar{x}, (0, 1)^T) = 1.0$, $Q(\bar{x}, (1, 0)^T) = 1.0$, and $Q(\bar{x}, (1, 1)^T) = 0.7$. For $UB_1^{max}(\bar{x})$, we must take the highest of these values; hence, $UB_1^{max}(\bar{x}) = 1.0$. For UB_1^{mean} , notice that $\bar{h} = (1/2)(1, 0)^T + (1/2)(0, 1)^T$; so, $UB_1^{mean}(\bar{x}) = UB_1^{max}(\bar{x}) = 1.0$. For UB_1^{EM} , each extreme point is weighted equally, so $UB_1^{EM}(\bar{x}) = (1/4)(1 + 1 + .7 + .6) = 0.825$. For the simplicial approximation, let $\Sigma = \text{co}\{(0, 0), (2, 0), (0, 2)\}$, which includes the support of \mathbf{h} . In this case, the weights on the extreme points are $\lambda(0, 0) = 0.5$ and $\lambda(2, 0) = \lambda(0, 2) = 0.25$. The resulting upper bound is $UB^\Sigma(\bar{x}) = 0.5(.6) + 0.25(2)(2) = 1.3$.

To refine the bounds, we consider the dual multipliers at each extreme point. At $(0, 0)$, they are $(-1, -1)$. At $(1, 0)$, they are $(1, -1)$. At $(0, 1)$, they are $(-1, 1)$. At $(1, 1)$, both bases B_1 and B_2 are optimal, so the multipliers are $(0, 1)$, $(1, 0)$, or any convex combination. The linearization along the line segment from

$(0,0)$ to $(1,0)$ is the minimum of $Q(\bar{x}, (1,0)^T) - Q(\bar{x}, (0,0)^T) + (-1, -1)^T(1,0) = 1 - (0.6 - 1) = 1.4$ and $Q(\bar{x}, (0,0)^T) - Q(\bar{x}, (1,0)^T) + (1, -1)^T(-1,0) = 0.6 - (1 - 1) = 0.6$. Hence, the minimum error on $(0,0)$ to $(1,0)$ is 0.6. Similarly, for $(0,0)$ to $(0,1)$, the error is 0.6. From $(1,0)$ to $(1,1)$, the minimum error is 0.3 if the $(0,1)$ subgradient is used at $(1,1)$; however, the minimum error on $(0,1)$ to $(1,1)$ is then $\min\{1 - (0.7 - 1), 0.7 - (1 - 1)\} = 0.7$. Thus, the maximum of these errors over each edge of the region is 0.7 for the edge $(0,1)$ to $(1,1)$.

To find the value of c_1^* to split the interval $[a_1 = 0, b_1 = 1]$, we need to find where $Q(\bar{x}, (0,1)^T) - c_1^* = Q(\bar{x}, (1,1)^T) + (c_1^* - 1)$ or where $1 - c_1^* = 0.7 - 1 + c_1^*$, i.e., where $c_1^* = 0.65$. We obtain two regions, $S_1 = [0, 0.65] \times [0, 1]$ and $S_2 = [0.65, 1] \times [0, 1]$, with $p_1 = 0.65$ and $p_2 = 0.35$.

The Jensen lower bound is now $LB_2 = 0.65(Q(\bar{x}, (0.325, 0.5)^T)) + (0.35)(Q(\bar{x}, (0.825, 0.5)^T)) = 0.65(0.2) + 0.35(0.525) = 0.31375$. The upper bounds are $UB_2^{max}(\bar{x}) = 0.65(1) + 0.35(1) = 1$, $UB_2^{mean}(\bar{x}) = 0.65(0.5)(1 + 0.65) + 0.35(0.5)(1 + 0.7) = 0.83375$, and $UB_2^{EM}(\bar{x}) = 0.65(0.25)(1 + 0.7 + 0.65 + 0.6) + 0.35(0.25)(0.7 + 0.7 + 1 + .65) = 0.74625$. (The simplicial bound is not given because we have split the region into rectangular parts.) Exercise 3 asks for these computations to continue until the lower and upper bounds are within 10% of each other.

Exercises

- For Example 1, $\bar{x} = (0.1, 0.7)^T$, compute $\mathcal{Q}(\bar{x})$, the Jensen lower bound, and the upper bounds, UB^{mean} , UB^{max} , UB^{EM} , and UB^Σ .
- Prove Jensen's inequality, $E(g(\xi)) \geq g(E(\xi))$, by taking an expectation of the points on a supporting hyperplane to $g(\xi)$ at $g(E(\xi))$.
- Follow the splitting rules for Example 1 until the Edmundson-Madansky upper and Jensen lower bounds are within 10% of each other. Compare UB^{EM} to UB^{max} on each step.

8.3 Using Bounds in Algorithms

The bounds in Section 8.2 can be used in algorithms in a variety of ways. We describe three basic procedures in this section: (1) uses of lower bounds in the L -shaped method with stopping criteria provided by upper bounds; (2) uses of upper bounds in generalized programming with stopping rules given by lower bounds; and (3) uses of the dual formulation in the separable convex hull function. The first two approaches are described in Birge [1983] while the last is taken from Birge and Wets [1989].

The L -shaped method as described in Chapter 5 is based on iteratively providing a lower bound on the recourse objective, $\mathcal{Q}(x)$. If a lower bound, $\mathcal{Q}^L(x)$, is used

in place of $\mathcal{Q}(x)$, then clearly for any supports, $E^L x + e^L$, if $\mathcal{Q}^L(x) \geq E^L x + e^L$, $\mathcal{Q}(x) \geq E^L x + e^L$. Thus, any cuts generated on a lower bounding approximation of $\mathcal{Q}(x)$ remain valid throughout a procedure that refines that lower bounding approximation. This observation leads to the following algorithm. We suppose that $\mathcal{Q}_j^L(x)$ and $\mathcal{Q}_j^U(x)$ are approximating lower and upper bounding approximations such that $\lim_{j \rightarrow \infty} \mathcal{Q}_j^L(x) = \mathcal{Q}(x)$ and $\lim_{j \rightarrow \infty} \mathcal{Q}_j^U(x) = \mathcal{Q}(x)$. We suppose that P_j^L is the j th lower bounding approximation measure so that $\mathcal{Q}_j^L(x) = \int_{\Omega} Q_j^L(x, \xi) P_j^L(d\omega)$. To simplify the algorithm in the following, we assume that all feasibility cuts are generated separately in (3.2) below before the sequential bounding procedure begins (which generally can be accomplished by first considering all extreme points of the domain of the random variables).

L-Shaped Method with Sequential Bounding Approximations

Step 0. Set $r = s = v = k = 0$.

Step 1. Set $v = v + 1$. Solve the linear program (3.1)–(3.3):

$$\begin{aligned} \min z &= c^T x + \theta \\ \text{s. t.} \quad Ax &= b, \end{aligned} \tag{3.1}$$

$$D_\ell x \geq d_\ell, \quad \ell = 1, \dots, r, \tag{3.2}$$

$$\begin{aligned} E_\ell x + \theta &\geq e_\ell, & \ell = 1, \dots, s, \\ x &\geq 0, & \theta \in \Re. \end{aligned} \tag{3.3}$$

Let (x^v, θ^v) be an optimal solution. If no constraint (3.3) is present, θ is set equal to $-\infty$ and is ignored in the computation.

Step 2. Find $\mathcal{Q}_j^L(x^v) = \int_{\Omega} Q_j^L(x^v, \xi) P_j^L(d\omega)$, the j th lower bounding approximation. Suppose $-(\pi^v(\xi))^T \mathbf{T} \in \partial_x \mathcal{Q}_j^L(x^v, \xi)$ (the simplex multipliers associated with the optimal solution of the recourse problem). Define

$$E_{s+1} = \int_{\Omega} (\pi^v(\xi))^T \mathbf{T} P_j^L(d\omega) \tag{3.4}$$

$$\text{and} \quad e_{s+1} = \int_{\Omega} (\pi^v(\xi))^T \mathbf{h} P_j^L(d\omega). \tag{3.5}$$

Let $w^v = e_{s+1} - E_{s+1} x^v = \mathcal{Q}_j^L(x^v)$. If $\theta^v \geq w^v$, x^v is optimal, relative to the lower bound; go to Step 4. Otherwise, set $s = s + 1$ and return to Step 1.

Step 3. Find $\mathcal{Q}_j^U(x^v) = \int_{\Omega} Q_j^U(x^v, \xi) P_j^U(d\omega)$, the j th upper bounding approximation. If $\theta^v \geq \mathcal{Q}_j^U(x^v)$, stop; x^v is optimal. Otherwise, refine the lower and upper bounding approximations from v to $v + 1$. Let $v = v + 1$. Go to Step 2.

This form of the L -shaped method follows the same steps as the standard L -shaped method, except that we add an extra check with the upper bound to determine the stopping conditions. We also describe the calculation of \mathcal{Q}_j^L somewhat generally to allow for more general types of approximating distributions and approximating recourse functions, $\mathcal{Q}_j^L(x^v, \xi)$.

Example 2

Consider Example 1 from Chapter 5, where:

$$\mathcal{Q}(x, \xi) = \begin{cases} \xi - x & \text{if } x \leq \xi, \\ x - \xi & \text{if } x > \xi, \end{cases} \quad (3.6)$$

$c^T x = 0$, and $0 \leq x \leq 10$. Instead of a discrete distribution on ξ , however, assume that ξ is uniformly distributed on $[0, 5]$. For the bounding approximation, we use the Jensen lower bound and Edmundson-Madansky upper bound for \mathcal{Q}^L and \mathcal{Q}^U , respectively. We use the refinement procedure to split the cell that contributes most to the difference between \mathcal{Q}^L and \mathcal{Q}^U . We split this cell at the intersection of the supports from the two extreme points of this cell (here, interval).

The sequence of iterations is as follows.

Iteration 1:

Here, $x^1 = 0$. Find $\mathcal{Q}_1^L(0) = Q(0, \bar{\xi} = 2.5) = 2.5$. $E_1 = -\partial_x \mathcal{Q}_1^L(0, 2.5) = -(-1)$ and $e_1 = -\partial_x \mathcal{Q}_1^L(0, 2.5)(h = 2.5) = -(-1)(2.5) = 2.5$. Add the cut:

$$\theta \geq 2.5 - x. \quad (3.7)$$

Iteration 2:

Here, $x^2 = 10$, $\theta = -7.5$, but $\mathcal{Q}_1^L(10) = Q(10, \bar{\xi} = 2.5) = 7.5$. At this point, the subgradient of $\mathcal{Q}_1^L(10)$ is 1. $E_2 = -\partial_x \mathcal{Q}_1^L(10, 2.5) = -1$, and $e_1 = -\partial_x \mathcal{Q}_1^L(0, 2.5)(h = 2.5) = -(1)(2.5) = -2.5$. Add the cut:

$$\theta \geq -2.5 + x. \quad (3.8)$$

Iteration 3:

Here, $x^3 = 2.5$, $\theta = 0$, $\mathcal{Q}_1^L(2.5) = Q(2.5, \bar{\xi} = 2.5) = 0$. Hence we meet the condition for optimality of the first lower bounding approximation. Now, go to Step 4 and consider the first upper bounding approximation with equal weights of 0.5 on $\xi = 0$ and $\xi = 5$. In this case, $\mathcal{Q}_1^U(2.5) = 0.5 * (Q(2.5, 0) + Q(2.5, 5)) = 2.5$. Thus, we must refine the approximation. Using the subgradient of -1 at $\xi = 0$ and 1 at $\xi = 5$, split at $c^* = 2.5$.

The new lower bounding approximation has equal weights of 0.5 on $\xi = 1.25$ and $\xi = 3.75$. In this case, $\mathcal{Q}_2^L(2.5) = 0.5 * (Q(2.5, 1.25) + Q(2.5, 3.75)) = 1.25$. Now, we add the cut $E_2 = 0.5(-\partial_x Q(2.5, 1.25) - \partial_x Q(2.5, 3.75)) = 0$ and $e_1 = 0.5(-\partial_x Q(2.5, 1.25)(1.25) - \partial_x Q(2.5, 3.75)(3.75)) = (0.5)(-1.25 + 3.75) = 1.25$. Thus, we add the cut:

$$\theta \geq 1.25. \quad (3.9)$$

Iteration 4:

Here, keep $x^4 = x^3 = 2.5$ (although other optima are possible) and $\theta = 1.25$. Again, $\mathcal{Q}_2^L(2.5) = 1.25$, so proceed to Step 4.

Checking the upper bound, we find that the upper bound places equal weights on the endpoints of each interval, $[0, 2.5]$ and $[2.5, 5]$. Thus, $\mathcal{Q}_2^U(2.5) = 0.5 * (Q(2.5, 2.5)) + (0.25) * (Q(2.5, 0) + Q(2.5, 5)) = 1.25$, and $\theta = \mathcal{Q}_2^U(2.5)$. Stop with an optimal solution.

The steps are illustrated in Figure 4. We show the true $\mathcal{Q}(x)$ as a solid line, with dashed lines representing the approximations (lower and upper). Note that the method may not have converged as quickly if we had chosen some point other than $x^4 = x^3 = 2.5$. The upper and lower bounds meet at this point, because we chose the division precisely at the link between the linear pieces of the recourse function $Q(x, \cdot)$.

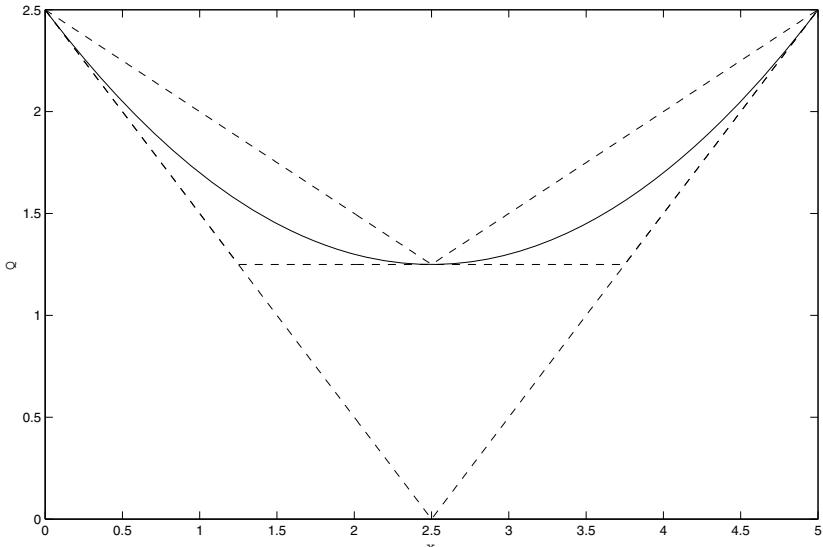


Fig. 4 Example of L -shaped method with sequential approximation.

Bounds with generalized programming

In generalized linear programming, the same types of procedures can be applied. The difference is that because the generalized programming method uses inner linearization instead of outer linearization, the bounds used should be upper bounds. We would thus substitute Ψ_j^U for Ψ in (5.6.6). The same steps are followed again with Ψ_j^U until optimality relative to Ψ_j^U is achieved. At this point, as in Step 4 of the L -shaped method with sequential bounding approximations, overall convergence is tested by solving (5.6.10) with a lower bounding Ψ_j^L in place of Ψ . If this value is again non-negative, then the procedure stops. If not, refinement is made until a new upper bounding column is generated or no solution of (5.6.10) is negative for a lower bounding approximation.

As stated in Chapter 5, generalized programming is most useful if the recourse function, $\Psi(\chi)$, is separable in the components of χ . The separable upper bounding procedure is a natural use for this approach. A separable lower bound can be obtained by using a supporting hyperplane. This leads to the Jensen lower bound.

This generalized programming approach applies most directly when a single basis separable approximation is used. With the convex hull operation, we would still have the problem of evaluating this function. This difficulty is, however, overcome by dualizing the problem. In this case, we suppose that the original problem using a set \mathcal{D} of bases is to find $x \in \mathbb{R}^{n_1}$, $\chi \in \mathbb{R}^{m_2}$ to

$$\begin{aligned} & \min c^T x + \text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi) \\ \text{s. t. } & Ax = b, \\ & Tx - \chi = 0, \\ & x \geq 0. \end{aligned} \tag{3.10}$$

The main result is the following theorem. Recall the conjugate function defined in Section 2.10.

Theorem 3. *A dual program to (3.10) is to find $\sigma \in \mathbb{R}^{m_1}$, $\pi \in \mathbb{R}^{m_2}$ to*

$$\begin{aligned} & \max \sigma^T b - \sup\{\Psi_D^*, D \in \mathcal{D}\}(-\pi) \\ \text{s. t. } & \sigma^T A + \pi^T T \leq c^T, \end{aligned} \tag{3.11}$$

where Ψ_D^* is the conjugate function and (3.10) and (3.11) have equal optimal values.

Proof: Let $\gamma(\chi) = \text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi)$. Then a dual to (3.10) (see, e.g., Geoffrion [1971], Rockafellar [1974]) is

$$\max_{\pi, \sigma} \left\{ \inf_{x \geq 0, \chi} [c^T x + \gamma(\chi) + \sigma^T(b - Ax) + \pi^T(\chi - Tx)] \right\},$$

which is equivalently

$$\begin{aligned} \max_{\pi, \sigma} \{ \inf_{x \geq 0, \chi} [(c^T - \sigma^T A - \pi^T T)x + \sigma^T(b) - (-\pi^T \chi - \gamma(\chi))] \} \\ = \max_{\sigma^T A + \pi^T T \leq c^T} \{ \sigma^T b - \gamma^*(-\pi) \}. \end{aligned} \quad (3.12)$$

Problem (3.12) immediately gives (3.11) because $(\text{co } \{\Psi_D, D \in \mathcal{D}\})(\chi)^*(-\pi) = \sup\{\Psi_D^*, D \in \mathcal{D}\}(-\pi)$ (Rockafellar [1969, Theorem 16.5]). \square

Problem (3.11) only involves finding the supremum of convex functions, which is again a convex function. The main difficulty is in finding expressions for the Ψ_D^* . These are, however, relatively straightforward to evaluate (Exercise 2). They can be used in a variety of optimization procedures, but the objective is nondifferentiable. In Birge and Wets [1989], this difficulty is overcome by making each Ψ_D^* a lower bound on some parameter that replaces $\sup\{\Psi_D^*, D \in \mathcal{D}\}$ in the objective.

The main refinement choice in the separable optimization procedure using (3.11) is to determine how to update the set \mathcal{D} . Choices of bases that are optimal for ξ and then $\xi \pm \delta e_i \sigma_i$ for increasing values of δ appear to give a rich set \mathcal{D} as in Birge and Wets [1989]. Any sense of *optimal* refinements or basis choice is, however, an open question.

Exercises

1. Consider Example 2 where we redefine Q as

$$Q(x, \xi) = \begin{cases} 2(\xi - x) & \text{if } x \leq \xi, \\ x - \xi & \text{if } x > \xi, \end{cases}$$

with ξ uniformly distributed on $[0, 5]$, $c^T x = 0$, and $0 \leq x \leq 10$. Follow the L -shaped sequential approximation method until achieving a solution with two significant digits of accuracy.

2. Find $\Psi_D^*(-\pi)$ and $\partial\Psi_D^*(-\pi)$. A useful set may be $\gamma_{D^i}(p) = \{y \mid P_{D^i}(y)^- \leq p \leq P_{D^i}(y)\}$.
3. Use the dualization procedure to solve a stochastic linear program with $c^T x = x$, $0 \leq x \leq 1$, and the recourse function in Example 1.

8.4 Bounds in Chance-Constrained Problems

Our procedures have so far concentrated on methods for recourse problems as we have throughout this book. In many cases, of course, probabilistic constraints may also be in the formulation or may be the critical part of the model. The basic results are aimed at finding some inequalities $\tilde{A}x \geq \tilde{h}$ (or, perhaps, nonlinear inequalities) that imply that $P\{\mathbf{Ax} \geq \mathbf{h}\} \geq \alpha$. In Section 3.2, we found some deterministic

equivalents for specific forms of the distribution, but these are not always available. In these cases, it is useful to have upper and lower bounds on $P\{\mathbf{A}x \geq \mathbf{h}\}$ for any x such that $\tilde{\mathbf{A}}x \leq \tilde{\mathbf{h}}$.

As an example, suppose a bank is trying to determine levels of exposure $x_j, j = 1, \dots, n$ in each of n loans which have a random value at time i (relative to the current date) of \mathbf{A}_{ij} . The bank may also have an uncertain liability value \mathbf{h}_i at each time i as well and wishes to ensure that the values of the loan assets exceed those of the liabilities at all times with high probability, i.e., $P\{\mathbf{A}x \geq \mathbf{h}\} \geq \alpha$. The bank wishes to avoid the problems of financial institutions who lost considerable amounts during the financial crisis of 2007-2010. Instead of assuming some specific distributions on the random variables, the bank prefers to find values for $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{h}}$ such that $\tilde{\mathbf{A}}x \geq \tilde{\mathbf{h}}$ will ensure $P\{\mathbf{A}x \geq \mathbf{h}\} \geq \alpha$ for a wide range of possible distributions and, therefore, seeks a set of bounds that depend on simple metrics. The types of bounds we consider here can then be used for this type of robust requirement.

The bounds for this purpose are generally of two types: bounds for a single inequality such as $P\{\mathbf{A}_i x \geq \mathbf{h}_i\}$ and bounds for the set of inequalities in terms of results in lower dimensions. In algorithms, (see Prékopa [1988]), it is often common to place the probabilistic constraint into the objective and to use a Lagrangian relaxation or parametric solution procedure.

For bounds with a single constraint, the basic results are extensions of Chebyshev's inequality and require only knowing (or bounding) the first two moments of the distribution. (See Hoeffding [1963] and Pintér [1989] for many of these results and additional details.) The basic Chebyshev inequality is (see, e.g., Feller [1971, Section V.7]) that if ξ has a finite second moment, then

$$P\{|\xi| \geq a\} \leq \frac{E[\xi^2]}{a^2}, \quad (4.1)$$

and for σ^2 , the variance of ξ ,

$$P\{|\xi - \bar{\xi}| \geq a\} \leq \frac{\sigma^2}{a^2}. \quad (4.2)$$

Another useful inequality is the one-sided inequality for $a > 0$ that

$$P\{\xi - \bar{\xi} \geq a\} \leq \frac{\sigma^2}{\sigma^2 + a^2}. \quad (4.3)$$

To apply (4.2) and (4.3) in the context of stochastic programming, we suppose that we can represent $\mathbf{A}_i x \geq \mathbf{h}_i$ as $\xi_0 + \xi^T x \geq r_0 + r^T x$, where $\mathbf{A}_{ij} = \xi_j - t_j$ and $\mathbf{h}_i = -\xi_0 + r_0$, to distinguish random elements from those that are not random and to allow us to set $\xi_j = 0$ for $j = 0, \dots, n$. If ξ has covariance matrix, C , then the variance of $\xi_0 + \xi^T x$ is $\hat{x}^T C \hat{x}$, where $\hat{x} = \begin{pmatrix} 1 \\ x \end{pmatrix}$. In this case, substituting $\hat{x}^T C \hat{x}$ for σ^2 and $r_0 + r^T x = \hat{r}^T \hat{x}$ for a in (4.3) yields for $\hat{r}^T \hat{x} > 0$:

$$P\{\mathbf{A}_i x \geq \mathbf{h}_i\} \leq \frac{\hat{x}^T C \hat{x}}{\hat{x}^T C \hat{x} + (\hat{r}^T \hat{x})^2}, \quad (4.4)$$

which implies that if x satisfies

$$\hat{x}^T C \hat{x} (1 - \alpha) \leq \alpha (\hat{r}^T \hat{x})^2, \quad (4.5)$$

then

$$P\{\mathbf{A}_i x \geq \mathbf{h}_i\} \leq \alpha. \quad (4.6)$$

Alternatively, if

$$P\{\mathbf{A}_i x \geq \mathbf{h}_i\} \geq \alpha, \quad (4.7)$$

then

$$\hat{x}^T C \hat{x} (1 - \alpha) \geq \alpha (\hat{r}^T \hat{x})^2. \quad (4.8)$$

Thus, adding constraint (4.8) in place of (4.7) in a stochastic program allows a large feasible region and in a minimization problem, would produce a lower bound on the objective value with constraint (4.7). For an upper bound, we could note that $P\{\mathbf{A}_i x \geq \mathbf{h}_i\} \geq \alpha$ is equivalent to $P\{\mathbf{A}_i x \leq \mathbf{h}_i\} \leq 1 - \alpha$ or $P\{\mathbf{h}_i - \mathbf{A}_i x \geq 0\} \leq 1 - \alpha$. We just replace the previous ξ and t with $-\xi$ and $-t$ and replace α with $(1 - \alpha)$ to obtain that if

$$\hat{x}^T C \hat{x} (\alpha) \leq (1 - \alpha) (\hat{r}^T \hat{x})^2, \quad (4.9)$$

then (4.7). Hence, replacing (4.7) with (4.9) yields a smaller region and an upper bound in a minimization problem.

In the context of the banking example discussed earlier, (4.9) provides a constraint that ensures the assets' value exceeds that of the liability with the prescribed probability α assuming that the covariance C and mean value \hat{r} are known. We make this example more precise in the following.

Example 3

For this example, suppose a typical portfolio that has $n = 125$ loans with an expected loss on each loan of 5% until the horizon so that $E[\mathbf{A}_{ij}] = t_j = 0.95$ with a common standard deviation of $\sigma = 0.025$. Suppose that the liability \mathbf{h}_i is a fixed value equal to 0.95 and that we want to ensure having the loan values exceed the liability with probability $\alpha = 0.99$. We can use (4.9) to determine $x_j = \frac{b}{125}$ for some $b > 0$ for an equally proportioned portfolio that meets the funding reliability requirement. If the future values of all of the loans are independent, then (4.9) is equivalent to:

$$(\alpha \sigma^2 - 0.95^2 (1 - \alpha)n)b^2 + 2(0.95)^2(1 - \alpha)nb - (1 - \alpha)(0.95)^2 \leq 0, \quad (4.10)$$

which then implies

$$b \geq 1.024, \quad (4.11)$$

(Exercise 1) which suggests that ensuring the expected asset value exceeds the liability by 2.4% would suffice in meeting the probabilistic constraint, regardless of the distribution if the means, variances, and covariances are all given as here.

In this case, the assumption about covariances (in this case, independence, such that all off-diagonal correlations are zero) can, however, be quite significant. Suppose instead of independence that all of the loans are linked to the same obligor (or borrower) and, therefore, that the correlations are all one. In that case, (4.11) becomes:

$$(\alpha\sigma^2 - 0.95^2(1-\alpha))n^2b^2 + 2(0.95)^2(1-\alpha)nb - (1-\alpha)(0.95)^2 \leq 0, \quad (4.12)$$

which then implies

$$b \geq 1.355, \quad (4.13)$$

(Exercise 2) requiring now a 35.5% greater expectation for the loans than the liability to have the same level of confidence as in the case of independence.

The extremes of zero and perfect correlation might be narrowed with additional information on the covariance matrix C . In that case, it may be possible to solve the *semi-definite program* (see, e.g., Vandenberghe and Boyd [1996]) to maximize $C \cdot \hat{X}$ (defined by $C \cdot \hat{X} = \sum_{i=1}^n \sum_{j=1}^n C_{ij} \hat{X}_{ij} = \hat{x}^T C \hat{x}$ if $\hat{X} = \hat{x} \hat{x}^T$) for C subject to $C \succeq 0$ (meaning that C is positive semi-definite) and other constraints representing available information on C . The resulting solution $C^*(\hat{x})$ can then be substituted for C in (4.9) to obtain a constraint that implies the reliability constraint for any covariance consistent with the available information.

Other information, such as ranges, can also be used to obtain sharper bounds. A particularly useful inequality (see, again, Feller [1971]) is that, for any function $u(\xi)$ such that $u(\xi) > \varepsilon > 0$, for all $\xi \geq t$,

$$P\{\xi \geq t\} \leq \frac{1}{\varepsilon} E[u(\xi)]. \quad (4.14)$$

In fact, using, $u(\xi) = (\xi + \frac{\sigma^2}{\alpha})^2$ yields (4.3) from (4.14). A difficulty in using bounds based on (4.3) is that the constraint in (4.8) or (4.9) may be quite difficult to include in an optimization problem. Various linearizations around certain values of x of this constraint can be used in place of (4.8) or (4.9). Other approaches, as in Pintér [1989] and Nemirovski and Shapiro [2006], are based on the expectations of exponential functions of ξ_i (i.e., its moment-generating function) that can in turn be bounded using the Jensen inequality and other convexity properties.

Given these approaches or deterministic equivalents for a single inequality as in Section 3.2, we wish to find approximations for multiple inequalities, $P\{\mathbf{Ax} \leq \mathbf{h}\}$. With relatively few inequalities and special distributions, such as the multivariate gamma described in Szántai [1986], deterministic equivalents can again be found. The general cases are, however, most often treated with approximations based on Boole-Bonferroni inequalities. A thorough description is found in Prékopa [1988].

We suppose that $\mathbf{A} \in \Re^{m \times n}$ and that $\mathbf{h} \in \Re^m$. The Boole-Bonferroni inequality bounds are based on evaluating $P\{\mathbf{A}_i x \leq \mathbf{h}_i\}$ and $P\{\mathbf{A}_i x \leq \mathbf{h}_i, \mathbf{A}_j x \leq \mathbf{h}_j\}$ for each

i and j and using these values to bound the complete expression $P\{\mathbf{A}x \leq \mathbf{h}\}$. To distinguish among the rows of \mathbf{A} , we let $\mathbf{A}_{ij} = \xi_j^i - t_j^i$ and $\mathbf{h}_i = -\xi_0^i + t_0^i$. A main result is then the following.

Theorem 4. *Given these assumptions,*

$$P\{\mathbf{A}x \leq \mathbf{h}\} = 1 - \left(a - \frac{2b}{m} \right) + \lambda \left[\frac{(c-1)a}{c+1} - \frac{2(-m+c(c+1))b}{m(c(c+1))} \right], \quad (4.15)$$

with

$$\begin{aligned} a &= \sum_{1 \leq i \leq m+1} P(\boldsymbol{\eta}_i > s_i(x)), \\ b &= \sum_{1 \leq i < j \leq m+1} P(\boldsymbol{\eta}_i > s_i(x), \boldsymbol{\eta}_j > s_j(x)), \\ c &= \lfloor \frac{2b}{a} \rfloor, \\ 0 \leq \lambda \leq 1, \quad \boldsymbol{\eta}_i &= (\xi^i)^T \hat{x}, \quad s_i(x) = (r^i)^T \hat{x}. \end{aligned}$$

Proof: Denote the event $\eta_i \leq s_i(x)$ by E_i . Then

$$P(\mathbf{A}x \leq \mathbf{h}) = P(E_1 \dots E_m) = 1 - P(\hat{E}_1 + \dots + \hat{E}_m), \quad (4.16)$$

where \hat{S} for a set S indicates the complement of S , i.e., the set of elements not in S .

By the inequality of Dawson and Sankoff [1967] ((7) of Prékopa [1988]),

$$P(\hat{E}_1 + \dots + \hat{E}_m) \geq \frac{2}{c+1} a - \frac{2}{c(c+1)} b, \quad (4.17)$$

where

$$\begin{aligned} a &= \sum_{1 \leq i \leq m} P(\hat{E}_i) = \sum_{1 \leq i \leq m} P(\boldsymbol{\eta}_i > s_i(x)), \\ b &= \sum_{1 \leq i < j \leq m} P(\hat{E}_i \cdot \hat{E}_j) = \sum_{1 \leq i < j \leq m} P(\boldsymbol{\eta}_i > s_i(x), \boldsymbol{\eta}_j > s_j(x)), \\ c &= \lfloor \frac{2b}{a} \rfloor. \end{aligned}$$

Similarly, by the inequality of Sathe, Pradhan, and Shah [1980] ((8) of Prékopa [1988]),

$$P(\hat{E}_1 + \dots + \hat{E}_m) \leq a - \frac{2}{m} b. \quad (4.18)$$

Combining (4.16)–(4.18), we obtain (4.15). \square

We may use (4.15) to approximate $P\{\mathbf{A}x \leq \mathbf{h}\}$ by assigning λ in $[0, 1]$, e.g., 0.5, or by using (4.15) for bounds with $\lambda = 0$ or 1 (see Exercise 6). With the marginal distribution of η_i and the joint distribution of η_i and η_j , we can again use bounds on the variances of these random variables to calculate additional bounds from (4.15). Of course, with normally distributed random variables, we may again obtain the η_i to be normally distributed or may obtain such limiting distributions (see, e.g., Salinetti [1983]). In this case, besides the exact results in Section 3.2, we should mention the specializations of Gassmann [1988] and Deák [1980]. They also combine these inequalities with Monte Carlo simulation schemes (see, e.g., Rubinstein [1981]). In general, the inequalities from (4.15) can reduce the variance of Monte Carlo schemes. For this approach and the bivariate gamma, we again refer to Szántai [1986].

Before closing this section, we should also mention that approximating probabilities is quite useful in recourse problems because the gradient of the linear recourse function with fixed q and T is simply a weighted probability of given bases' optimality. From Theorem 3.11, if x is in the interior of K_2 , then

$$\begin{aligned}\partial \mathcal{Q}(x) &= E_{\xi}[-\pi(\xi)^T T] \\ &= \sum_{j=1}^J -\pi^j T P \left\{ (\pi^j)^T (\mathbf{h} - Tx) \geq \pi^T (\mathbf{h} - Tx), \forall \pi^T W \leq q \right\},\end{aligned}\quad (4.19)$$

where $\{\pi^1, \dots, \pi^J\}$ is the set of extreme values of $\{\pi \mid \pi^T W \leq q\}$. Because $(\pi^j)^T = (W^j)^{-1} q^T$ is optimal, if and only if $(W^j)^{-1}(\mathbf{h} - Tx) \geq 0$, the result reduces to finding the probability that $(W^j)^{-1}(\mathbf{h} - Tx) \geq 0$. This observation can be useful in guiding algorithms based on subgradient information. This idea is explored in Birge and Qi [1995].

Other model forms also lead to bounds of this type that can in some cases be stronger because of the structure of \mathbf{A} . A particular case is when \mathbf{A} represents a network. In this case, bounds on project network completion times can be found in Maddox and Birge [1991] with other generalizations using semi-definite programming in Bertsimas, Natarajan, and Teo [2004] and Bertsimas and Popescu [2004]. These bounds, as well as those given earlier, can be derived from solutions of a generalized moment problem. That is one of the main topics of the generalizations in the next section.

Exercises

1. Show how (4.10) and (4.11) follow from (4.9).
2. Show how (4.12) and (4.13) follow from (4.9).
3. Under what conditions is (4.9) a convex constraint on \hat{x} ?
4. Derive (4.14).

5. Define u in (4.14) as $u(\xi) = c\sigma^2 - (\xi - \frac{u+t}{2})^2$, where it is known, however, that $\xi \leq U = \beta a$, a.s., for some finite β . For given β and a , can you find c such that (4.14) gives a better bound with this u than with the u used to obtain (4.3)?
6. Consider Example 3 with multiple ($m = 3$) periods such that each \mathbf{A}_{ij} is conditionally an independent Bernoulli random variable such that $P\{\mathbf{A}_{ij} = 1 | \mathbf{A}_{(i-1)j} = 1\} = 0.95$, $P\{\mathbf{A}_{ij} = 0 | \mathbf{A}_{(i-1)j} = 1\} = 0.05$, and $P\{\mathbf{A}_{ij} = 0 | \mathbf{A}_{(i-1)j} = 0\} = 1$. Suppose also $\mathbf{h} = [0.95, 0.95, 0.95]^T$, $\alpha = 0.99$, and the goal again is to find b so that $x_j = \frac{b}{125}, j = 1, \dots, n$ satisfies $P\{\mathbf{Ax} \geq \mathbf{h}\} \geq \alpha$. Use (4.15) to obtain a constraint that implies $P\{\mathbf{Ax} \geq \mathbf{h}\} \geq \alpha$ and find the smallest b satisfying this constraint. What happens in the case where the random variables within each period could be perfectly correlated?
7. Suppose $\xi_i, i = 1, 2, 3$, are jointly multivariate normally distributed with zero means and variance-covariance matrix

$$C = \begin{pmatrix} 1 & 0.25 & -0.25 \\ 0.25 & 1 & -0.5 \\ -0.25 & -0.5 & 1 \end{pmatrix}.$$

Use Theorem 4 to bound $P\{\xi_i \leq 1, i = 1, 2, 3\}$. What is the exact result? (Hint: Try a transformation to independent normal random variables.)

8.5 Generalized Bounds

a. Extensions of basic bounds

When the components of ξ are correlated, a bound is still tractable (see Frauendorfer [1988b]), although somewhat more difficult to evaluate. In this subsection, we give the necessary generalizations. The notation here is particularly cumbersome, although the results are straightforward.

For the general results, we define:

$$\eta(e, \xi_i) = \begin{cases} (\xi_i - a_i) & \text{if } e_i = a_i, \\ (b_i - \xi_i) & \text{if } e_i = b_i. \end{cases} \quad (5.1)$$

Then we have (Exercise 1) that

$$\phi(\xi, e) = \prod_{i=1}^N \frac{\eta(e, \xi_i)}{(b_i - a_i)}. \quad (5.2)$$

The $\lambda(e)$ values can be found by integrating over ω . This may involve all products of the ξ_i components. Defining $\mathcal{M} = \{M \mid M \subset \{1, \dots, N\}\}$, and $\rho_M =$

$E[\prod_{i \in M} \xi_i] - \prod_{i \in M} \bar{\xi}_i$, we obtain the general E-M extension:

$$\begin{aligned} UB^{EM-D}(x) &= UB^{EM-I}(x) \\ &+ \sum_{e \in ext \Xi} \frac{1}{\prod_{i=1}^N (b_i - a_i)} \left\{ \sum_{M \in \mathcal{M}} \left[\prod_{i \notin M} (-1)^{\frac{e_i - a_i}{b_i - a_i}} \left(a_i \left(\frac{e_i - a_i}{b_i - a_i} \right) \right. \right. \right. \\ &\quad \left. \left. \left. + b_i \left(\frac{b_i - e_i}{b_i - a_i} \right) \right) \times \prod_{i \in M} (-1)^{1 - \frac{e_i - a_i}{b_i - a_i}} \right] \rho_M \right\} g(x, e). \quad (5.3) \end{aligned}$$

Notice, in (5.3), that if the components of ξ are independent, then $\rho_M = 0$ for all M and $UB^{EM-D}(x) = UB^{EM-I}(x)$, as expected.

Each of these upper bounds is a solution of a corresponding moment problem in which the highest expected function value is found over all probability distributions with the given moment information. The upper bounds derived so far all used first moment information plus some information about correlations. In Subsection 8.5c, we will explore the possibilities for higher moments and methods for constructing bounds with this additional information.

For different support regions, Ξ , we can combine the bounds or use enclosing regions as we mentioned for simplicial approximations. To use the bounds in a convergent method, the partitioning scheme in Theorem 1 is again employed. Instead of applying the bounds on Ξ in its entirety, they are applied on each S^l . The dimension of these cells may, however, make computations quite cumbersome, especially if the S^l have an exponentially increasing number of extreme points in the dimension. For this reason, algorithms primarily concentrate on a lower bounding approximation for most computations and only use the upper bound to check optimality and stopping conditions.

So far, we have only considered convex $g(x, \cdot)$. In the recourse problem, $Q(x, \xi(\omega))$ is generally convex in $h(\omega)$ and $T(\omega)$ but concave in $q(\omega)$. In this general case, the Jensen-type bounds provide an upper bound on \mathcal{Q} in terms of q while the extreme point bounds provide lower bounds in q . We can combine these results with the convex function results to obtain overall bounds by, for example, determining $UB(x) = \int_{\Omega} UB(x, \mathbf{q}) P(d\omega)$ where $UB(x, \mathbf{q}) = UB_{\mathbf{h}, \mathbf{T}}(Q(x, \xi))$, where the last upper bound is taken with respect to the \mathbf{h} and \mathbf{T} with q fixed. The difficulty of evaluating $\int_{\Omega} UB(x, \mathbf{q}) P(d\omega)$ may determine the success of this effort. In the case of \mathbf{q} independent of \mathbf{h} and \mathbf{T} , it is simple. In other cases, linear upper bounding hulls may be constructed to allow relatively straightforward computation (Frauendorfer [1988a]) or extensions of the approach in UB^{mean} may be used (Edirisinghe [1991]).

For the procedure in Frauendorfer [1988a], assume that Ξ is compact and rectangular with $q \in \Xi_1 = [c_1, d_1] \times \cdots \times [c_{n_2}, d_{n_2}]$ and $(h, T)^T \in \Xi_2 = [a_1, b_1] \times \cdots \times [a_{N-n_2}, b_{N-n_2}]$. For convenience here, we consider T as a single vector of all components in order, T_1, \dots, T_{m_2} . We also delete transposes on vectors when they are used as function arguments.

Let the extreme points of the support of \mathbf{q} be e_l , $l = 1, \dots, L$, and the extreme points of the support of (\mathbf{h}, \mathbf{T}) be e_k , $k = 1, \dots, K$. In this case, because $Q(x, \cdot)$ is convex in (h, T) , for any e_l , we can take any support $\pi(e_l)$ such that $\pi(e_l)^T W \leq e_l$ and obtain a lower bound on $Q(x, (e_l, h, T))$ as

$$\pi(e_l)^T (h - Tx) \leq Q(x, (e_l, h, T)). \quad (5.4)$$

We can also let $\phi(q, e_l) = \prod_{i=1}^{n_2} \frac{\eta(e_l, q_i)}{(d_i - c_i)}$, where η is as defined earlier with c replacing a and d replacing b . Because for any (h, T) , $Q(x, (q, h, T))$ is concave in q , we have that

$$\begin{aligned} Q(x, (q, h, T)) &\geq \sum_{l=1}^L \phi(q, e_l) Q(x, (e_l, h, T)) \\ &\geq \sum_{l=1}^L \phi(q, e_l) \pi(e_l)^T (h - Tx), \end{aligned} \quad (5.5)$$

where we note that $\pi(e_l)$ need not depend on (h, T) . A bound is obtained by integrating over $(h(\omega), T(\omega))$ in (5.5), so that

$$\mathcal{Q}(x) \geq \sum_{l=1}^L \int_{\Omega} \prod_{i=1}^{n_2} \frac{\eta(e_l, \mathbf{q}_i)}{(d_i - c_i)} \pi(e_l)^T (\mathbf{h} - \mathbf{T}x) P(d\omega). \quad (5.6)$$

Note the terms in (5.6) just involve products of the components of \mathbf{q} and each component of \mathbf{h} or $\mathbf{T}x$ singly. Following Frauendorfer [1988a], we let $\mathcal{L} = \{\Lambda \mid \Lambda \subset \{1, \dots, n_2\}\}$ and define

$$\begin{aligned} c_{\Lambda}(e_l) &= \frac{1}{\prod_{i=1}^{n_2} (d_i - c_i)} \left[\prod_{i \notin \Lambda} (-1)^{\frac{e_{l,i} - c_i}{d_i - c_i}} \left(c_i \frac{e_{l,i} - c_i}{d_i - c_i} + d_i \frac{d_i - e_{l,i}}{d_i - c_i} \right) \right] \\ &\quad \times \left[\prod_{i \in \Lambda} (-1)^{1 - \frac{e_{l,i} - c_i}{d_i - c_i}} \right], \end{aligned} \quad (5.7)$$

$$m_{\Lambda} = \int_{\Omega} \prod_{i \in \Lambda} \mathbf{q}_i P(d\omega), \quad (5.8)$$

$$\text{and } m_{j, \Lambda} = \int_{\Omega} \mathbf{h}_j \prod_{i \in \Lambda} \mathbf{q}_i P(d\omega), \quad (5.9)$$

where $j = 1, \dots, m_2$. We may also include stochastic components of \mathbf{T} in place of \mathbf{h}_j in (5.9). For simplicity, however, we only consider \mathbf{h} stochastic next.

Assuming that $\sum_{\Lambda \in \mathcal{L}} c_{\Lambda}(e_l) m_{\Lambda} > 0$ for all $l = 1, \dots, L$, the integration in (5.6) yields a lower bound. With the definitions in (5.7)–(5.9), we can define a general dependent lower bound, $LB^{q,h}(x)$, as

$$LB^{q,h}(x)$$

$$\begin{aligned}
&= \sum_{l=1}^L \left(\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda \right) \left[\sum_{j=1}^{m_2} \pi(e_l, j) \left(\frac{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_{j,\Lambda}}{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda} - (Tx)_j \right) \right] \\
&= \sum_{l=1}^L \left(\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda \right) Q \left(x, e_l, \frac{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_{j,\Lambda}}{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda} \right) \\
&\leq \mathcal{Q}(x),
\end{aligned} \tag{5.10}$$

where $\pi(e_l)$ is chosen so that

$$\begin{aligned}
&Q \left(x, e_l, \frac{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_{j,\Lambda}}{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda} \right) \\
&= \left[\sum_{j=1}^{m_2} \pi(e_l, j) \left(\frac{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_{j,\Lambda}}{\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda} - (Tx)_j \right) \right].
\end{aligned}$$

When $\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_\Lambda = 0$, we also have $\sum_{\Lambda \in \mathcal{L}} c_\Lambda(e_l) m_{j,\Lambda} = 0$ (Exercise 5) making the l th component of the bound zero in that case. A completely analogous upper bound is also available then.

Dependency can be removed if the random variables, \mathbf{h} , can be written as linear transformations of independent random variables. Here, the independent case needs only to be slightly altered. A discussion appears in Birge and Wallace [1986].

The difficulty with the upper bounds for convex $g(x, \cdot)$ and the other bounds with concave components is that they minimally require function evaluations at the extreme points of the support of the random vectors. They also may require joint moment information that is not available. These factors make bounds based on extreme points unattractive for practical computation with more than a small number of random elements. As we saw earlier, in the case of simplicial support, we can reduce the effort to only being linear in the dimension of the support, but the bounds generally become imprecise.

Another problem with the upper bounds described so far in this chapter is that they require bounded support. In Subsection 8.5c., we will describe generalizations to eliminate this requirement for Edmundson-Madansky types of bounds. In the next subsection, we consider other bounds that do not have this limitation. They are based on exploiting separable structure in the problem. The goal in this case is to avoid exponential growth in effort as the number of random variables increases. The bounds of Section 8.3 are, however, still quite useful for low dimensions.

b. Bounds based on separable functions

As we observed earlier, simple recourse problems are especially attractive because they only require simple integrals to evaluate. The basic idea in this section is to construct approximating functions that are separable and, therefore, easy to integrate. This idea can be extended to separate low-dimension approximations, which can then be combined with the bounds in Section 8.3. These ideas also generalize to multistage approximations, such as approximate dynamic programming considered in Chapter 10.

In the simple recourse problem (Section 3.1d.), we noticed that $\Psi(\chi)$ can be written as

$$\Psi(\chi) = \sum_{i=1}^{m_2} \Psi_i(\chi_i), \quad (5.11)$$

in the case when only \mathbf{h} is random in the recourse problem. We again consider this case and build approximations on it. These results appear in Birge and Wets [1986, 1989], Birge and Wallace [1988], and, for network problems, Wallace [1987].

The basic simple recourse approximation is to consider an optimal response to changes in each component of \mathbf{h} separately and to combine those responses into an approximating function. For the i th component of \mathbf{h} , this response is the pair of optimal solutions, $y^{i,+}, y^{i,-}$, to:

$$\begin{aligned} & \min q^T y \\ & \text{s. t. } Wy = \pm e_i, y \geq 0, \end{aligned} \quad (5.12)$$

where e_i is the i th coordinate direction, $y^{i,+}$ corresponds to a right-hand side of e_i , and $y^{i,-}$ corresponds to a right-hand side of $-e_i$. Thus, for any value h_i of \mathbf{h}_i , the approximating response of $y^{i,+}(h_i - \chi_i)$ if $h_i \geq \chi_i$ and $y^{i,-}(\chi_i - h_i)$ if $h_i < \chi_i$. We have thus used the positive homogeneity of $\psi(\chi, h + \chi)$.

Using $y^{i,+}$ and $y^{i,-}$, we then obtain the approximate simple recourse functions:

$$\Psi_{I(i)}(\chi_i, h_i) = \begin{cases} q^T y^{i,+}(h_i - \chi_i) & \text{if } h_i \geq \chi_i, \\ q^T y^{i,-}(\chi_i - h_i) & \text{if } h_i < \chi_i, \end{cases} \quad (5.13)$$

which are integrated to form

$$\Psi_{I(i)}(\chi_i) = \int_{\mathbf{h}_i} \Psi_{I(i)}(\chi_i, h_i) P_i(d\mathbf{h}_i), \quad (5.14)$$

where we let P_i be the marginal probability measure of \mathbf{h}_i . Note that the calculation in (5.14) only requires the conditional expectation of \mathbf{h}_i on each interval $(-\infty, \chi_i]$ and (χ_i, ∞) and the expectation of these intervals.

The $\Psi_{I(i)}$ functions combine to form

$$\Psi_I(\chi) = \sum_{i=1}^{m_2} \Psi_{I(i)}(\chi_i), \quad (5.15)$$

which is a simple recourse function. The next theorem states the main result of this section.

Theorem 5. *The function $\Psi_I(\chi)$ constructed in (5.13)–(5.15) represents an upper bound on the recourse function $\Psi(\chi)$, i.e.,*

$$\Psi(\chi) \leq \Psi_I(\chi), \quad (5.16)$$

for all χ .

Proof: Consider the solution $\mathbf{y}_I = \sum_{i=1}^{m_2} [y^{i,+}(\mathbf{h}_i - \chi_i)^+ + y^{i,-}(-)(\mathbf{h}_i - \chi_i)^-]$. Note that \mathbf{y}_I is feasible in the recourse problem for \mathbf{h} . Thus

$$\begin{aligned} \Psi(\chi) &= \int_{\Omega} \psi(\chi, \mathbf{h}) P(d\omega) \\ &\leq \int_{\Omega} q^T \mathbf{y}_I P(d\omega) = \sum_{i=1}^{m_2} \Psi_I(\chi_i) = \Psi_I(\chi). \end{aligned} \quad (5.17)$$

□

The result in Theorem 5 is straightforward but useful. In particular, we can construct other approximations that use different representations of a solution to the recourse problem with right-hand side $\mathbf{h} - \chi$. A particularly useful type of this approximation is to consider a set of vectors, $V = \{v_1, \dots, v_V\}$, such that any vector in \Re^{m_2} can be written as a non-negative linear combination of the vectors in V . This defines V as a *positive linear basis* of \Re^{m_2} . For such V , we suppose that $y^{V,i}$ solves:

$$\begin{aligned} \min q^T y \\ \text{s. t. } Wy = v_i, \quad y \geq 0. \end{aligned} \quad (5.18)$$

We can then represent any $h - \chi$ in terms of non-negative combinations of the v_i or W times the corresponding non-negative combination of the $y^{V,i}$. Thus, we construct a feasible solution that responds separately to the components of V .

If V is a simplex, the construction of $h - \chi$ from V corresponds to a barycentric coordinate system. Bounds based on this idea are explored in Dulá [1991]. Another option is to let V be the set of positive and negative components of a basis $D = [d_1 | \dots | d_{m_2}]$ of \Re^{m_2} , or, $V = \{d_1, \dots, d_{m_2}, -d_1, \dots, -d_{m_2}\}$. This yields solutions, $y^{D,i,+}$, to (5.18) when $v_i = d_i$ and $y^{D,i,-}$ when $v_i = -d_i$. To use these in approximating a recourse problem solution with right-hand side $h - \chi$, we want the values of ζ such that $D\zeta = h - \chi$ or $\zeta = D^{-1}(h - \chi)$. Then the weight on d_i is ζ_i if $\zeta_i \geq 0$ and the weight on $-d_i$ is $-\zeta_i$ if $\zeta_i < 0$. We thus construct simple recourse-type functions,

$$\psi_{D^i}(\zeta_i) = \begin{cases} q^T y^{D,i,+}(\zeta_i) & \text{if } \zeta_i \geq 0, \\ q^T y^{D,i,-}(-\zeta_i) & \text{if } \zeta_i < 0, \end{cases} \quad (5.19)$$

which are integrated to form

$$\Psi_{D^i}(\chi) = \int_{\zeta_i} \psi_{D^i}(\zeta_i) P_{D^i}(d\zeta_i), \quad (5.20)$$

where P_{D^i} is the marginal probability measure of ζ_i . Again, these are added to create a new upper bound,

$$\Psi_D(\chi) = \sum_{i=1}^{m_2} \Psi_{D^i}(\chi) \geq \Psi(\chi). \quad (5.21)$$

Now, computation of Ψ_D relies on the ability to find the distribution of ζ_i . In special cases, such as when \mathbf{h} is normally distributed, then ζ , the affine transformation of a normal vector is also normally distributed so that the marginal ζ_i can be easily calculated. In other cases, full distributional information of \mathbf{h} may not be known. In this case, first or higher moments of ζ_i can be calculated and bounds such as those in Section 8.2 or those based on the moment problem in Subsection 8.5c., can be used. In either case, the calculation of Ψ_D reduces to evaluating or bounding the expectation of a function of a single random variable.

Of course, if a set of bases, \mathcal{D} , is available, then the best bound within this set can be used. In fact, the convex hull of all approximations, Ψ_D , for $D \in \mathcal{D}$, is also a bound. We write this function as:

$$\begin{aligned} \text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi) &= \inf \left\{ \sum_{i=1}^K \lambda^i \Psi_{D^i}(\chi^i) \mid \sum_{i=1}^K \lambda^i \chi^i = \chi, \right. \\ &\quad \left. \sum_{i=1}^j \lambda^i = 1, \lambda^i \geq 0, i = 1, \dots, K \right\}, \end{aligned} \quad (5.22)$$

where $\mathcal{D} = \{D^1, \dots, D^j\}$. This definition yields the following.

Theorem 6. For any set \mathcal{D} of linear bases of \Re^{m_2} ,

$$\Psi(\chi) \leq \text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi). \quad (5.23)$$

Proof: From earlier,

$$\Psi(\chi^i) \leq \Psi_{D^i}(\chi^i) \quad (5.24)$$

for each $i = 1, \dots, K$ and choice of χ^i . By convexity of Ψ , $\Psi(\chi) \leq \sum_{i=1}^j \lambda^i \Psi(\chi^i)$ where

$$\sum_{i=1}^K \lambda^i \chi^i = \chi, \quad \sum_{i=1}^j \lambda^i = 1, \quad \lambda^i \geq 0, \quad i = 1, \dots, K. \quad (5.25)$$

Combining (5.24) and (5.25) with the definition in (5.22) yields (5.23). \square

From Theorem 6, we continue to add bases D^i to \mathcal{D} to improve the bound on $\Psi(\chi)$. Even if $\mathcal{D}(W)$, the set of all bases in W are included; however, the bound is not exact. In this case, $\text{co}\{\psi_D(D^{-1}(\mathbf{h} - \chi)) \mid D \in \mathcal{D}(W)\} = \psi(\chi, \mathbf{h})$ because $\psi(\chi, \mathbf{h}) = q^T \mathbf{y}^* = q^T(D^*)^{-1}(\mathbf{h} - \chi)$ for some $D^* \in \mathcal{D}(W)$. However,

$$\begin{aligned}\Psi(\chi) &= \int \text{co}\{\psi_D(D^{-1}(\mathbf{h} - \chi)) \mid D \in \mathcal{D}(W)\} P(d\mathbf{h}) \\ &\leq \text{co}\left\{\int \psi_D(D^{-1}(\mathbf{h} - \chi)) P(d\mathbf{h}) \mid D \in \mathcal{D}(W)\right\} \\ &= \text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi),\end{aligned}\tag{5.26}$$

where the inequality is generally strict except for unusual cases (such as Ψ linear in χ).

As we shall see in an example later, the main intention of this approximation is to provide a means to find the optimal x value. Thus, the most important consideration is whether the subgradients of $\text{co}\{\Psi_D, D \in \mathcal{D}\}(\chi)$ are approximately the same as those for $\Psi(\chi)$. In this case, the approximation appears to perform quite well (see Birge and Wets [1989]).

Example 1 (continued)

Let us consider Example 1 again, as in Section 8.2. The optimal bases and their regions of optimality were given there. In this case, we let $D^1 = B^1$, $D^2 = B^2$, and $D^3 = B^3$. Note that this last approximation is derived for B^4 and B^5 because they correspond to the same positive linear basis as $[B_3, -B_3]$. At $\chi = (0.3, 0.3)^T$, we can evaluate each of the bounds, Ψ_{D^i} . For $i = 1$, we have $(D^1)^{-1} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$, so that $\zeta_1^1 = \mathbf{h}_1 - \mathbf{h}_2$ and $\zeta_2^1 = \mathbf{h}_2 - \chi_2 = \mathbf{h}_2 - 0.3$. In this case, $y^{D^1,1,+} = (y_1^+, y_1^-, y_2^+, y_2^-, y_3)^T = (1, 0, 0, 0, 0)^T$, $y^{D^1,1,-} = (0, 1, 0, 0, 0)^T$, $y^{D^1,2,+} = (0, 0, 0, 0, 1)^T$, and $y^{D^1,2,-} = (0, 1, 0, 1, 0)^T$. Integrating out each ζ_i^1 , we obtain $\Psi_{D^1}(0.3, 0.3) = 0.668$. Symmetrically, $\Psi_{D^2}(0.3, 0.3) = 0.668$. For $\Psi_{D^3}(0.3, 0.3)$, we note that each component is simply the probability that $\mathbf{h}_i \leq 0.3$ times the conditional expectation of $\mathbf{h}_i - 0.3$ given $\mathbf{h}_i \leq 0.3$ plus the probability that $\mathbf{h}_i > 0.3$ times the conditional expectation of $\mathbf{h}_i - 0.3$ given $\mathbf{h}_i > 0.3$. Thus, $\Psi_{D^3}(0.3, 0.3) = 2[(0.3)(0.15) + (0.7)(0.35)] = 0.580$.

Comparing the best of these bounds with those in the previous chapters leads to a more accurate approximation. We should note, however, that this approach requires more distributional information.

Taking convex hulls can produce even better bounds. The convex hull operation is, however, a nonconvex optimization problem. The dual gives some computational advantage. To give an idea of the advantage of the convex hull, however, consider Figure 5, where the graphs of Ψ_{D^i} are displayed with that of Ψ as functions of χ_1

for $\chi_2 = 0.1$. Note how the convex hull of the graphs of the approximations appears to have similar subgradients to that of Ψ . This observation appears to hold quite generally, as indicated by the computational tests in Birge and Wets [1989].

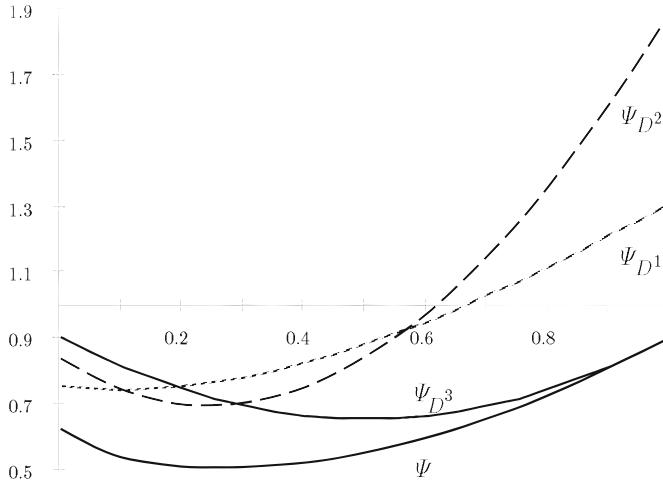


Fig. 5 Graphs of Ψ (solid line) and the approximations, Ψ_{D^i} (dashed lines).

The separable bounds in Ψ_{D^i} can also be enhanced by, for example, including fixed values (due to known entries in h) in the right-hand sides of (5.18). Other possibilities are to combine the component approximations on an interval instead of assuming that they may apply for all positive multiples of the v_i . In this case, the solution for some interval of v_i multiples can serve as a constraint for determining solutions for the next v_{i+1} . This procedure is carried out in Birge and Wallace [1988]. It appears especially useful for problems with bounded random variables and networks (Wallace [1987]).

To improve on these bounds and obtain some form of convergence requires relaxation of complete separability. For example, pairs of random variables can be considered together. In this way, more precise bounds can be found. Determination of these terms is, however, problem-specific. In general, the structure of the problem must be used to obtain the most efficient improvements on the basic separable approximation bounds.

So far, we have presented bounds for the recourse function with a fixed χ value. In the next subsection, we consider how to combine these approximations into solution algorithms where x varies from iteration to iteration. In the case of the separable bounds, this implementation results from a dualization that turns the difficult convex hull operation into a simpler supremum operation.

c. General-moment bounds

Many other bounds are possible in addition to those presented so far. A general form for many of these bounds is found through the solution of an abstract linear program, called a *generalized moment problem*. This problem provides the lowest or highest expected probabilities or objective values that are possible given certain distribution information that can be written as generalized moments. In this subsection, we present this basic framework, some results using second moments, and generalizations to nonlinear functions. Concepts from measure theory appear again in this development.

To obtain bounds that hold for all distributions with certain properties, we can find $P \in \mathcal{P}$, a set of probability measures on (Ξ, \mathcal{B}^N) , to extremize a moment problem. We let \mathcal{B}^N be the Borel field of \mathfrak{R}^N where $\mathfrak{R}^N \supset \Xi$. We use probability measures defined directly on \mathcal{B}^N to simplify the following discussion. We wish to find:

$$P \in \mathcal{P} \text{ a set of probability measures on } (\Xi, \mathcal{B}^N)$$

$$\begin{aligned} \text{s. t. } & \int_{\Xi} v_i(\xi) P(d\xi) \leq \alpha_i, \quad i = 1, \dots, s, \\ & \int_{\Xi} v_i(\xi) P(d\xi) = \beta_i, \quad i = s+1, \dots, M, \end{aligned}$$

$$\text{to maximize } \int_{\Xi} g(\xi) P(d\xi), \tag{5.27}$$

where M is finite and the v_i are bounded, continuous functions. A solution of (5.27) obtains an upper bound on the expectation of g with respect to any probability measure satisfying the conditions given earlier. We could equally well have posed this to find a lower bound.

Problem (5.27) is a *generalized moment problem* (Krein and Nudel'man [1977]). When the v_i are powers of ξ , the constraints restrict the moments of ξ with respect to P . In this context, (5.27) determines an upper bound when only limited moment information on a distribution is available.

Problem (5.27) can also be interpreted as an *abstract linear program*, i.e., a linear program defined over an abstract space, because the objective and constraints are linear functions of the probability measure. The solution is then an extreme point in the infinite-dimensional space of probability measures. The following theorem, proven in Karr [1983, Theorem 2.1], gives the explicit solution properties. We state it without proof because our main interests here are in the results and not the particular form of these solutions. Readers with statistics backgrounds may compare the result with the Neyman-Pearson lemma and the proof of the optimality conditions as in Dantzig and Wald [1951]. For details on the weak * topology that appears in the theorem, we refer the reader to Royden [1968].

Theorem 7. Suppose Ξ is compact. Then the set of feasible measures in (5.27), \mathcal{P} , is convex and compact (with respect to the weak* topology), and \mathcal{P} is the closure of the convex hull of the extreme points of \mathcal{P} . If g is continuous relative to Ξ , then an optimum (maximum or minimum) of $\int_{\Xi} g(x, \xi) P(d\xi)$ is attained at an extreme point of \mathcal{P} . The extremal measures of \mathcal{P} are those measures that have finite support, $\{\xi_1, \dots, \xi_L\}$, with $L \leq M + 1$, such that the vectors

$$\begin{pmatrix} v_1(\xi_1) \\ v_2(\xi_1) \\ \vdots \\ v_M(\xi_1) \\ 1 \end{pmatrix}, \quad \dots, \quad \begin{pmatrix} v_1(\xi_L) \\ v_2(\xi_L) \\ \vdots \\ v_M(\xi_L) \\ 1 \end{pmatrix} \quad (5.28)$$

are linearly independent.

Kemperman [1968] showed that the supremum is attained under more general continuity assumptions and provides conditions for \mathcal{P} to be nonempty. Dupačová (formerly Žáčková) [1976, 1977, 1966] pioneered the use of the moment problem as a bounding procedure for stochastic programs in her work on a minimax approach to stochastic programming. She showed that (5.27) attains the Edmundson-Madansky bound (and the Jensen bound if the objective is minimized) when the only constraint in (5.27) is $v_1 = \xi$, i.e., the constraints fix the first moment of the probability measure. She also provided some properties of the solution with an additional second-moment constraint ($v_2(x) = \xi^2$) for a specific objective function g . Frauendorfer's [1988b] results can be viewed as solutions of (5.27) when the constraints satisfy all of the joint moment conditions.

To solve (5.27) generally, we consider a generalized linear programming procedure.

Generalized Linear Programming Procedure for the Generalized Moment Problem (GLP)

Step 0. Initialization. Identify a set of $L \leq M + 1$ linearly independent vectors as in (5.28) that satisfy the constraints in (5.27). (Note that a phase one-objective (Dantzig [1963]) may be used if such a starting solution is not immediately available. For $N = 1$, the Gaussian quadrature points may be used.) Let $r = L$, $v = 1$; go to 1.

Step 1. Master problem solution. Find $p_1 \geq 0, \dots, p_r \geq 0$ such that

$$\sum_{l=1}^r p_l = 1, \\ \sum_{l=1}^r v_l(\xi_l) p_l \leq \beta_i, \quad l = 1, \dots, s,$$

$$\sum_{l=1}^r v_l(\xi_l) p_l = \beta_i , \quad l = s+1, \dots, M ,$$

and $z = \sum_{l=1}^r g(\xi_l) p_l$ is maximized. (5.29)

Let $\{p_1^j, \dots, p_r^j\}$ attain the optimum in (5.29), and let $\{\sigma^j, \pi_1^j, \dots, \pi_M^j\}$ be the associated *dual multipliers* such that

$$\begin{aligned} \sigma^j + \sum_{i=1}^M \pi_i^j v_i(\xi_l) &= g(\xi_l) , & \text{if } p_l^j > 0 , \\ \sigma^j + \sum_{i=1}^M \pi_i^j v_i(\xi_l) &\geq g(\xi_l) , & \text{if } p_l^j = 0 , \\ \pi_i^j &\geq 0 , & i = 1, \dots, s . \end{aligned} \quad (5.30)$$

Step 2. Subproblem solution. Find ξ^{r+1} that maximizes

$$\gamma(\xi, \sigma^j, \pi^j) = g(\xi) - \sigma^j - \sum_{i=1}^M \pi_i^j v_i(\xi) . \quad (5.31)$$

If $\gamma(\xi^{r+1}, \sigma^j, \pi^j) > 0$, let $r = r + 1$, $v = v + 1$ and go to Step 1. Otherwise, stop; $\{p_1^j, \dots, p_r^j\}$ are the optimal probabilities associated with $\{\xi_1, \dots, \xi_r\}$ in a solution to (5.27).

As we saw in Chapter 3, the generalized programming approach is useful in problems with a potentially large number of variables. This approach is used in Er-molieva, Gaivoronski, and Nedeva [1985] to solve a class of problems (5.27). The difficulty in GLP is in the solution of the subproblem (5.31), which generally involves a nonconvex function. Birge and Wets [1986] describe how to solve (5.31) with constrained first and second moments, if convexity properties of γ can be identified. Cipra [1985] describes other methods for this problem based on discretizations and random selections of candidate points, x_i . Dulá [1991] gives results when g is sublinear and has simplicial level sets. Kall [1991] gives the results for sub-linear, polyhedral functions with known generators. Edirisinghe [1996] also finds bounds using second moment information that is somewhat looser than the generalized moment solution.

Kall's result is useful when the optimal recourse problem multipliers are known, so that

$$Q(x, \xi) = \max_{i=1, \dots, K} \pi_i^T (\mathbf{h} - Tx) , \quad (5.32)$$

where we again assume that $\xi = \mathbf{h}$ or that T and q are known. Kall's result pertains to having known means for all \mathbf{h}_i and a limit ρ on the *total second moment*, defined as

$$\rho = \int_{\Xi} \|\xi\|^2 P(d\xi). \quad (5.33)$$

The moment problem becomes:

$$\begin{aligned} & \sup_{P \in \mathcal{P}} \int_{\Omega} Q(x, \xi) P(d\xi) \\ & \text{s. t. } \int_{\Xi} \xi P(d\xi) = \bar{h} \quad \text{and} \quad (5.33), \end{aligned} \quad (5.34)$$

where \mathcal{P} is a set of probability measures with support, Ξ .

Kall shows that the solution of (5.34) with Q defined as in (5.32) is equivalent to the following finite-dimensional optimization problem:

$$\inf_{y \in \mathbb{R}^m} \left\{ \max_{i=1, \dots, K} \left(\sqrt{\rho - 2(\bar{h})^T T x + \|T x\|^2} \right) \|\pi_i - y\| + (\bar{h} - T x)^T y \right\}. \quad (5.35)$$

Dulá obtained similar results for strictly simplicial Q . Note that when $\bar{h} = T x$, this reduces to a form of location problem to minimize the maximum weighted distance from π_i to y . The solution to (5.34) may involve calculations with each of these recourse problem solutions, but the resulting distribution P that solves (5.34) still has only $m_2 + 2$ points of support. These are found by solving for the Karush-Kuhn-Tucker conditions for problem (5.34), where the y values correspond to multipliers for the mean value constraints.

Other bounds are also possible for different types of objective functions. In particular, we consider functions built around separable properties. The use of the generalized programming formulation is limited in multiple dimensions because of the difficulty in solving subproblem (5.32). These computational disadvantages for large values of N suggest that a looser but more computationally efficient upper bound on the value of (5.27) may be more useful than solving (5.27) exactly for large N .

If a separable function, $\eta(x) = \sum_{i=1}^N \eta_i(x(i))$, is available, it offers an obvious advantage by only requiring single integrals, as we stated earlier. Here, we would also like to show that these bounds can be extended to nonlinear recourse functions. We suppose that the recourse function becomes some general $g(\xi(\omega))$, where

$$g(\xi) = \inf_y \{q(y) \mid g(y) \leq \xi\}. \quad (5.36)$$

In this case, we would like to find $\eta(\xi) = \sum_{i=1}^N \eta_i(\xi(i)) \geq g(\xi)$ where each $\eta_i(\xi(i))$ is a convex function. Methods for constructing these functions to bound the optimal value of a linear program with random right-hand sides were discussed in Subsection 8.5b. We next give the results for the general problem in (5.36).

Lemma 8. *If g is defined as in (5.36), then g is a convex function of ξ .*

Proof: Let y_1 solve the optimization problem in (5.36) for ξ_1 and let y_2 solve the corresponding problem for ξ_2 . Consider $\xi = \lambda \xi_1 + (1 - \lambda) \xi_2$. In this case,

$g(\lambda y_1 + (1 - \lambda)y_2) \leq \lambda g(y_1) + (1 - \lambda)g(y_2) \leq \lambda \xi_1 + (1 - \lambda)\xi_2$. So $g(\lambda \xi_1 + (1 - \lambda)\xi_2) \leq q(\lambda y_1 + (1 - \lambda)y_2) \leq \lambda g(\xi_1) + (1 - \lambda)g(\xi_2)$, giving the result. \square

Let

$$\eta_i(\xi(i)) \equiv \frac{1}{N} g(N\xi(i)e_i), \quad (5.37)$$

which is the optimal value of a parametric mathematical program. The following theorem shows that these values supply the separable bound required. Related bounds are possible by defining η_i with other right scalar multiples, $g\lambda_i(\xi(i)e_i)$ (see Rockafellar [1969] for general properties), where $\sum_{i=1}^N \lambda_i = 1$. The following proof below is easily extended to these cases and to translations of the constraints and explicit variable bounds.

Theorem 9. *The function $\eta(\xi) = \sum_{i=1}^N \eta_i(\xi(i)) \geq g(\xi)$, where g is defined as in (5.36).*

Proof: In this case, let $y_i(\xi(i))$ solve (5.36), where $\xi(\omega) = N\xi(i)e_i$. Then, $g\left(\frac{\sum_{i=1}^N y_i(\xi(i))}{N}\right) \leq \sum_{i=1}^N \left(\frac{1}{N}\right) [g(y_i(\xi(i)))] \leq \sum_{i=1}^N \left(\frac{1}{N}\right) N\xi(i)e_i = \xi$. Next, let y^* solve (5.36) for ξ in the right-hand side of the constraints. By feasibility of $\sum_{i=1}^N \frac{y_i(\xi(i))}{N}$, $g(\xi) = g(y^*) \leq q\left(\sum_{i=1}^N \frac{y_i(\xi(i))}{N}\right) \leq \sum_{i=1}^N \left(\frac{1}{N}\right) q(y_i(\xi_i)) = \sum_{i=1}^N \eta_i(\xi(i)) = \eta(\xi)$. \square

This result demonstrates that a parametric optimization of (5.36) in $i = 1, \dots, N$ yields an upper bound on $g(\xi)$ for any ξ . The bound may be tight, as in some examples for stochastic linear programs as given in Subsection 8.5b.

Generalizations of the stochastic linear program bound as in Subsection 8.5b. can also be given for the general bound in Theorem 9. For example, we may apply a linear transformation T to ξ to obtain $u = T\xi$. The constraints become $g(y) \leq G^{-1}(u)$. To use any bound of the general type in Theorem 9 to bound $\int_{\Re^N} g(\xi) dg(\xi)$ requires a bound on $\int_{\Re} \eta_i(\xi(i)) dF_i(\xi_i)$ or $\int_{\Re} \mu_i(u(i)) dF_{u_i}(u(i))$, where F_i is the marginal distribution on ξ_i and F_{u_i} is the marginal distribution on $u(i)$. Because it may be difficult to find the distribution of \mathbf{u} , the generalized moment problem can be solved to obtain bounds on each integral in \Re . Generalized linear programming may solve this problem but can be inefficient. To simplify this process, in Birge and Dulá [1991], it is shown that a large class of functions requires only two points of support in the bounding distribution. A single line search can determine these points and give a bound on f over all distributions with bounded first and second moments for the marginals.

We develop bounds following Birge and Dulá [1991] on $\int \eta_i(x(i)) dF_i(x(i))$ by referring to g as a function on \Re ($N = 1$). We then consider the moment problem (5.27) with $s = 0$, and $M = 2$ and where the constraints correspond to known first and second moments. In other words, we wish to find:

$$U = \sup_{Q \in \mathcal{P}} \int_{\Xi} g(\xi) P(d\xi)$$

$$\int_{\Xi} \xi P(d\xi) = \bar{\xi},$$

$$\int_{\Xi} \xi^2 P(dx) = \bar{\xi}^{(2)}, \quad (5.38)$$

where $P \in \mathcal{P}$ is the set of probability measures on (Ξ, \mathcal{B}^1) , the first moment of the true distribution is $\bar{\xi}$, and the second moment is $\bar{\xi}^{(2)}$.

A generalization of Carathéodory's theorem (Valentine [1964]) for the convex hull of connected sets tells us that y^* can be expressed as a convex combination of at most three extreme points of C , giving us a special case of Theorem 9. Therefore, an optimal solution to (5.38) can be written, $\{\xi^*, p^*\}$, where the points of support, $\xi^* = \{\xi_1^*, \xi_2^*, \xi_3^*\}$ have probabilities, $p^* = \{p_1^*, p_2^*, p_3^*\}$. An optimal solution may, however, have two points of support. A function that has this property for a given instance of (5.27) is called a *two-point support* function. We will give sufficient conditions for a function to have this two-point support property. This property then allows a simplified solution of (5.38). It is given in the next theorem which is proven in Birge and Dulá [1991].

Theorem 10. *If g is convex with derivative g' defined as a convex function on $[a, c]$ and as a concave function on $(c, b]$ for $\Xi = [a, b]$ and $a \leq c \leq b$, then there exists an optimal solution to (5.38) with at most two support points, $\{\xi_1, \xi_2\}$, with positive probabilities, $\{p_1, p_2\}$.*

A corollary of Theorem 10 is that any function g that has a convex or concave derivative has the two-point support property. The class of functions that meets the criteria of Theorem 10 contains many useful examples, such as:

1. Polynomials defined over ranges with at most one third-derivative sign change.
2. Exponential functions of the form, $c_0 e^{c_1 \xi}$, $c_0 \geq 0$.
3. Logarithmic functions of the form, $\log_j(c\xi)$, for any $j \geq 0$.
4. Certain hyperbolic functions such as $\sinh(c\xi)$, $c, \xi \geq 0$, $\cosh(cx)$.
5. Certain trigonometric and inverse trigonometric functions such as $\tan^{-1}(c\xi)$, $c, \xi \geq 0$.

In fact, Theorem 10 can be applied to provide an upper bound on the expectation of any convex function with known third derivative when the distribution function has a known third moment, $\bar{\xi}^{(3)}$. Suppose $a > 0$ (if not, then this argument can be applied on $[a, 0]$ and $[0, b]$); then let $g(\xi) = \beta \xi^3 + g(\xi)$. The function g is still convex on $[0, b]$ for $\beta \geq 0$. By defining $\beta \geq (-1/6) \min(0, \inf_{\xi \in [a, b]} f'''(\xi))$, g' is convex on $[a, b]$, and an upper bound, $UB(g)$, on $E_g(\xi)$ has a two-point support. The expectation of g is then bounded by

$$Eg(\xi) \leq UB(g) - \beta \bar{\xi}^{(3)}. \quad (5.39)$$

The conditions in Theorem 10 are only sufficient for a two-point support function. They are not necessary (see Exercise 8). Note also that not all functions are two-point support functions (although bounds using (5.34) are available). A function requiring three support points, for example, is $g(\xi) = (1/2) - \sqrt{(1/4) - (\xi - (1/2))^2}$ (Exercise 9).

Given that a function is a two-point support function, the points $\{\xi_1, \xi_2\}$ can be found using a line search to find a maximum.

For the special case of piecewise linear functions, the points, ξ_1, ξ_2 , can be found analytically. In this case, suppose that $g(\xi) = \psi^{SR}(h, \chi)$, the simple recourse function defined by:

$$\psi^{SR}(h, \chi) = \begin{cases} q^-(\chi - h) & \text{if } h \leq \chi, \\ q^+(h - \chi) & \text{if } h > \chi. \end{cases} \quad (5.40)$$

Consider the nonintersecting intervals, $A = (0, \bar{\xi}^{(2)} / (2\bar{\xi}))$, $B = [\bar{\xi}^{(2)} / (2\bar{\xi}), 1]$, and $C = ((1 - \bar{\xi}^{(2)}) / (2(1 - \bar{\xi})), 1)$. The points of support for this semilinear, convex function defined on $[0, 1]$ are

$$\{\xi_1^*, \xi_2^*\} = \begin{cases} \{0, \bar{\xi}^{(2)} / \bar{\xi}\} & \text{if } \chi \in A, \\ \{\chi - d, \chi + d\} & \text{if } \chi \in B, \\ \{(\bar{\xi} - \bar{\xi}^{(2)}) / (1 - \bar{\xi}), 1\} & \text{if } \chi \in C, \end{cases} \quad (5.41)$$

where $d = \sqrt{\chi^2 - 2\chi\bar{\xi} + \bar{\xi}^{(2)}}$. This result can be obviously extended to other finite intervals. Infinite intervals can also be solved analytically for these semilinear, convex functions. For $X = [0, \infty)$, the results are as in (5.41) with $B = [\bar{\xi}^{(2)} / (2\bar{\xi}), \infty)$ and $C = \emptyset$. For the interval $(-\infty, \infty)$, the points of support are those for interval B in (5.41). We note that special cases for these supports of semilinear, convex functions were considered in Jagganathan [1977] and Scarf [1958].

Other bounds are also possible using the generalized moment problem framework. One possible approach is to use piecewise linear constraints on the quadratic functions defining second-moment constraints as in (5.38). This approach is described in Birge and Wets [1987] which also considers unbounded regions that lead to measures that are limiting solutions to (5.27) but that may not actually be probability measures but are instead nonnegative measures with weights on extreme directions of Ξ . An example is given in Exercise 12.

To see how these bounds are constructed for unbounded regions, weights can be placed on extreme recession directions, r^j , $j = 1, \dots, J$, such that $\xi + \beta r^j \in \Xi$ for all $\xi \in \Xi$ and r^j not decomposable into non-negative multiples of other recession directions. Then, if the recourse function Q has a recession function, $\text{rc } Q(x, r^j) \geq \frac{Q(x, \xi + \beta r^j) - Q(x, \xi)}{\beta}$ for all $\beta > 0$, then $Q(x, \xi) \leq \sum_{k=1, \dots, K} \lambda^k Q(x, e^k) + \sum_{j=1, \dots, J} \mu^j \text{rc } Q(x, r^j)$, when $\xi = \sum_{k=1, \dots, K} \lambda^k e^k + \sum_{j=1, \dots, J} \mu^j r^j$, $\sum_{k=1, \dots, K} \lambda^k = 1$, $\lambda^k, \mu^j \geq 0$. Now, an analogous result to Theorem 1 can be constructed where $\lambda^k = \int_{\Xi} \lambda(\xi, e^k) P(d\xi)$ and $\mu^j = \int_{\Xi} \mu(\xi, r^j) P(d\xi)$ are constructed from measures $\lambda(\xi, \cdot)$ and $\mu(\xi, \cdot)$ such that $\xi = \sum_{k=1, \dots, K} e^k \lambda(\xi, e^k) + \sum_{j=1, \dots, J} r^j \mu^j(\xi, r^j)$ for all $\xi \in \Xi$.

With piecewise linear functions, $v_i(\xi) = \beta_{il} \xi + \beta_{il}$ on Ξ^l , $l = 1, \dots, L$, $P[\Xi^l] = p^l$,

$$\int_{\Xi} v_i(\xi) P(d\xi) = \sum_{l=1}^L \sum_{e \in \text{ext } \Xi_l} \beta_{il} e \lambda^l(e) + \sum_{r \in r \subset \Xi_l} \beta_{il} r \mu_l(r) - \beta_{il} p^l, \quad (5.42)$$

where $\lambda^l(e)$ is a weight on the extreme point e in Ξ^l and $\mu^l(r)$ is a weight on extreme direction r of Ξ^l . From (5.42), we can use a piecewise linear v_i to bound nonlinear v from below. If

$$\int_{\Xi} v(\xi) P(d\xi) \leq \bar{v}, \quad (5.43)$$

then

$$\int_{\Xi} v_i(\xi) P(d\xi) \leq \bar{v}. \quad (5.44)$$

Thus, we can use (5.44) in place of (5.43) to obtain an upper bound on a moment problem. The advantage of (5.44) is that we need only use the extreme values of the Ξ^l from (5.42) in (5.44).

Other types of bounds are also possible that depend on different types of functions, such as lower piecewise linear functions (see Marti [1975] or Birge and Wets [1986]). Stochastic dominance of probability distributions can also be used to construct bounds. This approach tends to be difficult in higher dimensions (see Birge and Wets [1986, Section 7]) but has useful properties for accounting for general risk preferences (see Dentcheva and Ruszczyński [2010]). Another alternative is to identify optimization procedures that improve among all possible distributions (see, e.g., Marti [1988]). Still other procedures are possible using conjugate function information directly such as in Birge and Teboulle [1989], which considers nonlinear functions that are otherwise not easily evaluated.

We have not yet considered approximations based on sampling ideas. Many possibilities exist in this area as well. We will describe these bounds and algorithms based on them in Chapter 9.

Exercises

1. Verify the derivation of $\eta(\xi, \cdot)$ in (5.2).
2. Derive the result in (5.3).
3. Consider the sugar beet recourse function, \mathcal{Q}_3 , in Section 1.1. Suppose that the selling price above 6000 is actually a random variable, \mathbf{q} , that has mean 10 and is distributed on $[5, 15]$. Suppose also that $E[\mathbf{q} \mathbf{r}_3] = 250$. Use (5.9) to derive a lower bound on $\mathcal{Q}_3(300)$.
4. Verify the result of the integration in (5.5) given in (5.9).
5. Verify that $\sum_{\Lambda \in \mathcal{L}} c_{\Lambda}(e_l) m_{\Lambda} = 0$ implies $\sum_{\Lambda \in \mathcal{L}} c_{\Lambda}(e_l) m_{j,\Lambda} = 0$ and that, if both are nonzero, then $\frac{\sum_{\Lambda \in \mathcal{L}} c_{\Lambda}(e_l) m_{j,\Lambda}}{\sum_{\Lambda \in \mathcal{L}} c_{\Lambda}(e_l) m_{\Lambda}}$ is in the closure of the support of \mathbf{h} .

6. Find the functions Ψ_{D^i} as functions of χ for each i as in the example. Also find the optimal value function Ψ in terms of χ . Graph these functions as functions of χ_2 for values of $\chi_1 = \frac{j}{10}, j = 0, \dots, 9$. Compare the convex hulls of the approximations with the graph of Ψ .
7. Using the data for Example 1, solve (5.34) to determine an upper bound with the total second-moment constraint.
8. Construct a two-point support function that does not meet the conditions in Theorem 3.
9. A European *call option* is a type of financial derivative contract that provides the right (but not the obligation) to purchase an asset at a fixed price K at a future time T . If the asset's price at time t is given by a price process S_t , then, under a complete and perfect assumption, the value (or *premium*) of the call option at time 0 is given as $C_0 = e^{-r_f T} E_Q[(S_T - K)^+]$, where r_f is the riskfree rate (earned by a riskless zero-coupon bond that pays no interest until time T when it matures and pays a face value of one with certainty) and Q is a measure over ω known as the *equivalent martingale measure*. While it is common to assume S_T under Q has a log-normal distribution, the distribution is often unknown. The bounds in this section can be used if only partial information is given. Suppose that only the mean and variance of S_T under Q is known. Show that the call function satisfies the conditions for a two-point support and find the maximum price that is consistent with these moments as a function of the mean, variance, and K . (Note that $S_T \geq 0$ as well.) (Lo [1987].)
10. Show that $g(\xi) = (1/2) - \sqrt{(1/4) - (\xi - (1/2))^2}$ requires three support points to obtain the best upper bound with mean of 0.5 and variance of 1/6 on $\Xi = [0, 1]$.
11. Find the Edmundson-Madansky and two-moment bounds for ξ uniform on $\Xi = [0, 1]$ and the following functions: $e^{-\xi}$, ξ^3 , $\sin(\pi(\xi + 1))$.
12. Use the results in Theorems 9 and 10 to bound the following nonlinear recourse function with the form in (5.38). We suppose in this case that

$$\begin{aligned} g(\xi_1, \xi_2) &= \min (\xi_1 - 1)^2 + (\xi_2 - 2)^2 \\ \text{s. t. } &\quad \xi_1^2 + \xi_2^2 - 1 \leq \xi_1, \\ &\quad (\xi_1 - 1)^2 + \xi_2^2 - 1 \leq \xi_2. \end{aligned}$$

13. Suppose that it is known that the ξ_i are non-negative, that $\bar{\xi}_i = 1$, and that $\bar{\xi}_i^{(2)} = 1.25$. In this case, we would like an upper bound on the expected performance $E(g(\xi))$. We construct a bound by first finding $\eta_i(\xi_i)$ as in (5.37). This problem may correspond to determining a performance characteristic of a part machined by two circular motions centered at $(0, 0)$ and $(1, 0)$, respectively. Here, the performance characteristic is proportional to the distance from the finished part to another object at $(2, 1)$. The square of the radii of the tool motions is $\xi_i + 1$, where ξ_i is a non-negative random variable associated with the machines' precision.

14. As an example of using (5.41), consider Example 1, but assume that Ξ is the entire non-negative orthant and that each ξ_i is exponentially distributed with mean 0.5. Use a piecewise linear lower bound on the individual second moments that is zero for $0 \leq \xi_i \leq 0.5$, and $2\xi_i - 1$ for $\xi_i \geq 0.5$. Solve the moment problem using these regions to obtain an upper bound for all expected recourse functions with the same means and variances as the exponential. Also, solve the moment problem with only mean constraints and compare the results.

8.6 General Convergence Properties

For the following bounding discussions, we use a general function notation because these results hold quite broadly. The discussion in this section follows Birge and Qi [1995], which gives a variety of results on convergence of probability measures. Other references are Birge and Wets [1986] and King and Wets [1991]. This section is fundamental for theoretical properties of convergence of approximations.

We consider the *expectational functional* $E(g(\cdot)) = E\{g(\cdot, \xi)\}$, where ξ is a random vector with support $\Xi \subseteq \Re^N$ and g is an extended real-valued function on $\Re^n \times \Xi$. Here,

$$E(g(x)) = \int g(x, \xi) P(d\xi), \quad (6.1)$$

where P is a probability measure defined on \Re^n .

We assume that $E(g(\cdot))$ (which represents the recourse function \mathcal{Q}) is difficult to evaluate because of the complications involved in g and the dimension of Ξ . The basic goal in most approximations is to approximate (6.1) by

$$E^v(g(x)) = \int g(x, \xi) P^v(d\xi), \quad (6.2)$$

where $\{P^v, v = 1, \dots\}$ is a sequence of probability measures converging in distribution to the probability measure P . By *convergence in distribution*, we mean that $\int g(\xi) P^v(d\xi) \rightarrow \int g(\xi) P(d\xi)$ for all bounded continuous g on Ξ . For more general information on convergence of distribution functions, we refer to Billingsley [1968].

In the following, we use E^0 and P^0 instead of E and P for convenience. If $C \subseteq \Re^n$ is a closed convex set, then Ψ_C^* is the support function of C , defined by $\Psi^*(g | C) = \sup\{\langle x, g \rangle : x \in C\}$. A sequence of closed convex sets $\{C_v : v = 1, \dots\}$ in \Re^n is said to converge to a closed convex set C in \Re^n if for any $g \in \Re^n$,

$$\lim_{v \rightarrow +\infty} \Psi^*(g | C_v) = \Psi^*(g | C).$$

One may easily prove the following proposition that is stated without proof.

Proposition 11. *Suppose that C and C^v , for $v = 1, \dots$, are closed convex sets in \Re^n . The following two statements are equivalent:*

- (a) C_v converges to C as $v \rightarrow +\infty$;
 (b) a point $x \in C$ if and only if there are $x^v \in C^v$ such that $x_v \rightarrow x$. \square

This notion of set convergence is important in the study of convergence of functions. We say that a sequence of functions, $\{g^v; v = 1, \dots\}$, *epi-converges* to function, g , if and only if the epigraphs, $\text{epi } g^v = \{(x, \beta) \mid \beta \geq g^v(x)\}$, of the functions converge as sets to the epigraph of g , $\text{epi } g = \{(x, \beta) \mid \beta \geq g(x)\}$. Epi-convergence has many important properties, which are explored in detail in Wets [1980a] and Attouch and Wets [1981]. A chief property (Exercise 1) is that any limit point of minima of g^v is a minimum of g .

In the following, we restrict our attention to convex integrands g although extensions to nonconvex functions are also possible as in Birge and Qi [1995]. In this case, one can use the generalized subdifferential in the sense of Clarke [1983] or other definitions as in Michel and Penot [1984] or Mordukhovich [1988]. The next theorem appears in Birge and Wets [1986] with some extensions in Birge and Qi [1995]. Other results of this type appear in Kall [1987].

Theorem 12. *Suppose that*

- (i) $\{P^v, v = 1, \dots\}$ converges in distribution to P ;
 (ii) $g(x, \cdot)$ is continuous on Ξ for each $x \in D$, where

$$D = \{x : E(g(x)) < +\infty\} = \{x : g(x, \xi) < +\infty, \text{ a. s.}\};$$

- (iii) $g(\cdot, \xi)$ is locally Lipschitz on D with Lipschitz constant independent of ξ ;
 (iv) for any $x \in D$ and $\varepsilon > 0$, there exists a compact set S_ε and v_ε such that for all $v \geq v_\varepsilon$,

$$\int_{\Xi \setminus S_\varepsilon} |g(x, \xi)| P^v(d\xi) < \varepsilon,$$

and with $V_x = \{\xi : g(x, \xi) = +\infty\}$, $P(V_x) > 0$ if and only if $P^v(V_x) > 0$ for $v = 0, 1, \dots$.

Then

- (a) $E^v(g(\cdot))$ epi- and pointwise converges to $E(g(\cdot))$; if $x, x^v \in D$ for $v = 1, 2, \dots$ and $x^v \rightarrow x$, then

$$\lim_{v \rightarrow \infty} E^v(g(x^v)) = E(g(x));$$

- (b) $E^v(g(\cdot))$, where $v = 0, 1, \dots$, is locally Lipschitz on D ; furthermore, for each $x \in D$, $\{\partial E^v(g(x)) : v = 0, 1, \dots\}$ is bounded;
 (c) if $x^v \in D$ minimizes $E^v(g(x))$ for each v and x is a limiting point of $\{x^v\}$, then x minimizes $E(g(x))$.

Proof: First, we establish pointwise convergence of the expectation functionals. Suppose $x \in D$ and consider S_ε as in the hypothesis. Let $M_\varepsilon = \sup_{\xi \in S_\varepsilon} |g(x, \xi)|$, which is finite for g continuous and S_ε compact. Construct a bounded and continuous function,

$$g^\varepsilon(\xi) = \begin{cases} g(x, \xi) & \text{if } |g(x, \xi)| \leq M_\varepsilon, \\ M_\varepsilon & \text{if } |g(x, \xi)| > M_\varepsilon, \\ -M_\varepsilon & \text{if } |g(x, \xi)| < -M_\varepsilon. \end{cases}$$

By convergence in distribution, $\beta_\varepsilon^v \rightarrow \beta_\varepsilon$, for $\beta_\varepsilon^v = \int_{\Xi} g_\varepsilon(\xi) P^v(d\xi)$ and $\beta_\varepsilon = \int_{\Xi} g_\varepsilon(\xi) P(d\xi)$. Let $\beta^v = \int_{\Xi} g(x, \xi) P^v(d\xi)$. Noting that for $v > v_\varepsilon$, $\int_{\Xi \setminus S_\varepsilon} g_\varepsilon(\xi) P^v(d\xi) < \varepsilon$,

$$|\beta^v - \beta_\varepsilon^v| < 2\varepsilon. \quad (6.3)$$

We also have that

$$|\beta - \beta_\varepsilon| < 2\varepsilon. \quad (6.4)$$

From the convergence of the β^v , there exists some \bar{v}_ε such that for all $v \geq \bar{v}_\varepsilon$,

$$|\beta_\varepsilon^v - \beta_\varepsilon| < 2\varepsilon. \quad (6.5)$$

Combining (6.3), (6.4), and (6.5) for any $v > \max\{\bar{v}_\varepsilon, v_\varepsilon\}$,

$$|\beta - \beta^v| < 6\varepsilon,$$

which establishes that $E^v(g(x)) \rightarrow E(g(x))$ for any $x \in D$.

To establish epi-convergence, from (b) of Proposition 11, we need to show that if $x \in D$ and $h \geq E(g(x))$, then there exists $x^v \in D$ and $h^v \geq E^v(g(x^v))$ such that $(x^v, h^v) \rightarrow (x, h)$, and, if $x^v \in D$ and $h^v \geq E^v(g(x^v))$ such that $(x^v, h^v) \rightarrow (x, h)$, then $x \in D$ and $h \geq E(g(x))$. The former follows by letting $x^v = x$ and $h^v = E^v(g(x)) + (h - E(g(x)))$ and using pointwise convergence. The latter follows from pointwise convergence and continuity because $v = \lim_v h^v \geq \lim_v E^v(g(x^v)) = \lim_v [(E^v(g(x^v)) - E^v(g(x)) + (E^v(g(x)) - E(g(x))) + E(g(x))] = E(g(x))$.

For (b), again let $x, x^v \in D$, $x^v \rightarrow x$. For any $x \in D$, y , and z close to x , $v = 0, 1, \dots$,

$$\begin{aligned} |E^v(g(y)) - E^v(g(z))| &\leq \int |g(y, \xi) - g(z, \xi)| P^v(d\xi) \\ &\leq \int L_x \|y - z\| P^v(d\xi) \\ &= L_x \|y - z\|, \end{aligned}$$

where L_x is the Lipschitz constant of $g(\cdot, \xi)$ near x , which is independent of ξ by (iii). By (ii) and (iii), x is in the interior of the domain of $E^v(g(x))$. Hence, (see Theorem 23.4 in Rockafellar [1969]), the subdifferential $\partial E^v(g(x))$ is a nonempty, compact convex set, for each v . The two-norms of subgradients in these subdifferentials are bounded by L_x .

By (b), $E^v(g(x))$ are lower semicontinuous functions. By (a), $E^v(g(x))$ epi-converges to $E(g(x))$. We get the conclusion of (c) from the statement in Exercise 1. This completes the proof. \square

This result also extends directly to nonconvex functions, as we mentioned earlier. In terms of stochastic programming computations, the most useful result may be (c), which implies convergence of optima for approximating distributions. Actually achieving optimality for each approximation may be time-consuming. One might, therefore, be interested in achieving convergence of subdifferentials. This may allow suboptimization for each approximating distribution.

In the case of closed convexity, Wets showed in Theorem 3 of Wets [1980a] that if $g, g^v : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $v = 1, 2, \dots$, are closed convex functions and $\{g^v\}$ epi-converges to g , then the graphs of the subdifferentials of g^v converge to the graph of the subdifferential of g , i.e., for any convergent sequence $\{(x^v, u^v) : u^v \in \partial g^v(x^v)\}$ with (x, u) as its limit, one has $u \in \partial g(x)$; for any (x, u) with $u \in \partial g(x)$, there exists at least one such sequence $\{(x^v, u^v) : u^v \in \partial g^v(x^v)\}$ converging to it.

However, in general, it is not true that

$$\partial g(x) = \lim_{v \rightarrow \infty} \partial g^v(x) \quad (6.6)$$

even if $x \in \text{int}(\text{dom}(g))$ (See Exercise 2). However, if g is G -differentiable at x , (6.6) is true. This is the following result from Birge and Qi [1995].

Theorem 13. Suppose that $g, g^v : \mathbb{R}^n \rightarrow \mathbb{R} \cup \{+\infty\}$, $v = 1, 2, \dots$, are closed convex functions and $\{g^v\}$ epi-converges to g . Suppose further that g is G -differentiable at x . Then

$$\nabla g(x) = \lim_{v \rightarrow \infty} \partial g^v(x). \quad (6.7)$$

In fact, for any $x \in \text{int}(\text{dom}(g))$, there exists v_x such that for any $v \geq v_x$, $\partial g^v(x)$ is nonempty, and $\{\partial g^v(x) : v \geq v_x\}$ is bounded. Thus, for any $x \in \text{int}(\text{dom}(g))$, the right hand side of (6.7) is nonempty and always contained in the left-hand side of (6.7). But equality does not necessarily hold by our example. We also state the following result (Corollary 2.5 of Birge and Qi [1995]).

Corollary 14. Suppose the conditions of Theorem 12 and that $g(\cdot, \xi)$ is convex for each $\xi \in \Xi$. Then for $D = \text{dom}(\mathbb{E}(g(\cdot)))$, in addition to results (a)–(c) in Theorem 12,

(d) there is a Lebesgue zero-measure set $D_1 \subseteq D$ such that $\mathbb{E}(g(x))$ is G -differentiable on $D \setminus D_1$, $\mathbb{E}(g(x))$ is not G -differentiable on D_1 , and for each $x \in D \setminus D_1$

$$\lim_{v \rightarrow \infty} \partial \mathbb{E}^v(g(x)) = \nabla \mathbb{E}(g(x));$$

(e) for each $x \in D$,

$$\partial \mathbb{E}(g(x)) = \left\{ \lim_{v \rightarrow \infty} u^v : u^v \in \partial \mathbb{E}^v(g(x^v)), x^v \rightarrow x \right\}.$$

Proof: By closed convexity of $g(\cdot, \xi)$, $E^v(g(x))$ are also closed convex for all v . Now (d) follows from Theorem 13 and the differentiability property of convex functions, and (e) follows from Theorem 3 of Wets [1980a]. \square

Many other results are possible using Theorem 13 and results on epi-convergence. As an example, we consider convergence of sampled problem minima following King and Wets [1991]. Let P^v be an empirical measure derived from an independent series of random observations $\{\xi^1, \dots, \xi^v\}$ each with common distribution P . Then, for all x ,

$$E^v(g(x)) = \frac{1}{v} \sum_{i=1}^v g(x, \xi_i).$$

Let (Ξ, \mathcal{A}, P) be a probability space completed with respect to P . A closed-valued multifunction G mapping Ξ to \Re^n is called *measurable* if for all closed subsets $C \subseteq \Re^n$, one has

$$G^{-1}(C) := \{\xi \in \Xi : G(\xi) \cap C \neq \emptyset\} \in \mathcal{A}.$$

In the following, “with probability one” refers to the sampling probability measure on $\{\xi^1, \dots, \xi^v, \dots\}$ that is consistent with P (see King and Wets [1991] for details). Applying Theorem 2.3 of King and Wets [1991] and Corollary 14, we have the following.

Corollary 15. Suppose for each $\xi \in \Xi$, $g(\cdot, \xi)$ is closed convex and the epigraphical multifunction $\xi \mapsto \text{epi } g(\cdot, \xi)$ is measurable. Let $E^v(g(x))$ be calculated by (6.2). If there exists $\bar{x} \in \text{dom}(E^v(g(x)))$ and a measurable selection $\bar{u}(\xi) \in \partial g(\bar{x}, \xi)$ with $\int \|\bar{u}(\xi)\| P(d\xi)$ finite, then the conclusions of Corollary 14 hold with probability one. \square

King and Wets [1991] applied their results to the two-stage stochastic program with fixed recourse repeated here as

$$\begin{aligned} & \min c^T x + \int Q(x, \xi) P(d\xi) \\ & \text{s. t. } Ax = b, \\ & \quad x \geq 0, \end{aligned} \tag{6.8}$$

where $x \in \Re^n$ and

$$Q(x, \xi) = \inf\{q(\xi)^T y \mid Wy = h(\xi) - T(\xi)x, y \in \Re_+^{n_2}\}. \tag{6.9}$$

It is a fixed recourse problem because W is deterministic. Combining their Theorem 3.1 with our Corollary 14, we have the following.

Corollary 16. Suppose that the stochastic program (6.8) has fixed recourse (6.9) and that for all i, j, k , the random variables $\mathbf{q}_i \zeta_j$ and $\mathbf{q}_i \mathbf{T}_{jk}$ have finite first moments. If there exists a feasible point \bar{x} of (6.9) with the objective function of (6.9) finite, then the conclusions of Corollary 14 hold with probability one for

$$g(x, \xi) = c^T x + Q(x, \xi) + \delta(x),$$

where $\delta(x) = 0$ if $Ax = b$, $x \geq 0$, $\delta(x) = +\infty$ otherwise. \square

By Theorem 3.1 of King and Wets [1991], one may solve the approximation problem

$$\begin{aligned} \min c^T x + \frac{1}{v} \sum_{i=1}^v Q(x, \xi_i) \\ \text{s. t. } Ax = b, \\ x \geq 0, \end{aligned} \tag{6.10}$$

instead of solving (6.8). If the solution of (6.10) converges as v tends to infinity, then the limiting point is a solution of (6.8). Alternatively, by Corollary 16, one may directly solve (6.8) with a nonlinear programming method and use

$$c^T x + \frac{1}{v} \sum_{i=1}^v Q(x, \xi_i) \quad \text{and} \quad c + \frac{1}{v} \sum_{i=1}^v \partial_x Q(x, \xi_i)$$

as approximate objective function values and subdifferentials of (6.8) with $v = v(k)$ at the k th step. Notice that $-u^T T(\xi_i) \in \partial_x Q(x, \xi_i)$ if and only if u is an optimal dual solution of (6.9) with $\xi = \xi_i$. In this way, one may directly solve the original problem using the subgradients $-u^T T(\xi_i)$ and the probability that each is optimal (equivalently that the corresponding basis is primal feasible). The calculation is therefore reduced to obtaining the probability of satisfying a system of linear inequalities, which can be approximated well (see Prékopa [1988] and Section 8.4). This procedure may allow computation without calculating the actual objective value, which may involve a more difficult multiple integral.

These results give some general idea about the uses of approximations in stochastic programming. We can also introduce approximating functions, g^v , such that g^v converges to g pointwise in D . Similar convergence results are also obtained there. The general rule is that approximating distribution functions that converge in distribution (even with probability one) to the true distribution function lead to convergence of optima and, for differentiable points, convergence of subgradients.

Exercises

1. Prove that if g^v epi-converges to g and x^* is a limit point of $\{x^v\}$, where $x^v \in \operatorname{argmin} g^v = \{x \mid g^v(x) \leq \inf g^v\}$, then $x^* \in \operatorname{argmin} g$.
2. Construct an example where g^v epi-converges to g but $\partial g(x) \neq \lim_v \partial g^v(x)$.
3. Consider the basic bounding method in Section 8.2. Suppose that Ξ is compact and that for any $\epsilon > 0$, there exists some v_ϵ such that for all $v \geq v_\epsilon$,

$\text{diam } S_l \leq \varepsilon$ for all $S_l \in \mathcal{S}^v$. Show that this implies that P^v converges to P in distribution.

Chapter 9

Monte Carlo Methods

Each function value in a stochastic program can involve a multidimensional integral in extremely high dimensions. Because Monte Carlo simulation appears to offer the best possibilities for higher dimensions (see, e.g., Deák [1988] and Asmussen and Glynn [2007]), it seems to be the natural choice for use in stochastic programs. In this chapter, we describe some of the basic approaches built on sampling methods. The key feature is the use of statistical estimates to obtain confidence intervals on results. Some of the material uses probability measure theory which is necessary to develop the analytical results.

To build on our earlier emphasis on decomposition algorithms, Section 9.1 begins this discussion with a description of the basic sampling approximation, the *sample-average approximation*, and then approaches uses of this system with the *L*-shaped method. We first consider possibilities for estimating the cuts in this method using a large number of samples for each cut. Section 9.2 then considers the *stochastic decomposition method* (Higle and Sen [1991b]) that forms many cuts with few additional samples on each iteration. Section 9.3 considers methods based on the stochastic quasi-gradient, which can be viewed as a generalization of the steepest descent method. These approaches have a wide variety of applications that extend beyond stochastic programming. In Section 9.4, we consider extensions of Monte Carlo methods to include analytical evaluations exploiting problem structure in probabilistic constraint estimation and empirical sample information for methods that may use updated information in dynamic problems. Section 9.5 describes basic theoretical results for the statistical analysis of stochastic programs and, in particular, for the sample-average approximation. We describe asymptotic properties and large-deviation bounds for optimal values and solutions to those problems.

9.1 Sample Average Approximation and Importance Sampling in the *L*-Shaped Method

The most direct sampling approach to the two-stage stochastic program is to replace the recourse function, $\mathcal{Q}(x)$, by a Monte Carlo estimate,

$$\mathcal{Q}^v(x) = \sum_{k=1}^v \frac{\mathcal{Q}(x, \xi^k)}{v}, \quad (1.1)$$

where ξ^1, \dots, ξ^v are random samples of the random vector ξ . This then yields the *sample average approximation* (SAA) problem for the general two-stage problem as:

$$\min_{x \in X} f^1(x) + \sum_{k=1}^v \frac{\mathcal{Q}(x, \xi^k)}{v}, \quad (1.2)$$

where X represents the feasibility set as, for example, in the nonlinear program in (3.4.1). For a stochastic linear program, we can then write (1.2) as:

$$\begin{aligned} \min & c^T x + \frac{1}{v} \sum_{k=1}^v q_k^T y_k \\ \text{s. t. } & Ax = b, \\ & T_k x + W y_k = h_k, \\ & x \geq 0, y_k \geq 0. \end{aligned} \quad (1.3)$$

As we show in Section 9.5, by increasing the sample size v , solutions to (1.3) converge to an optimal solution of the two-stage stochastic program (3.1.2). A disadvantage of solving (1.3) completely for each v using any algorithm is that some effort might be wasted on optimizing when the approximation is not accurate. An approach to avoid these problems is to use sampling within another algorithm without complete optimization. In this section, we describe this process for the *L*-shaped method, which often works well for discrete distributions. To ensure that the process makes efficient use of the sample information, we first describe a version using importance sampling to reduce variance in deriving each cut based on a large sample (see Dantzig and Glynn [1990]). In the following section, we consider an approach that uses a single sample stream to derive many cuts that eventually drop away as iteration numbers increase (Higle and Sen [1991b]).

The general approach is to sample \mathcal{Q} to construct cuts in the *L*-shaped method to obtain an approximate solution to (3.1.2). Using a crude Monte Carlo sample of ξ , however, may result in high variance for the sample values $\mathcal{Q}(x, \xi^k)$, slowing convergence or leading to biased results. Instead, to reduce the variance of the sample values, we use the *importance sampling* (see, e.g., Rubinstein [1981] and Deák [1990]) variance-reduction technique to concentrate samples where they provide the most information.

If we use a crude Monte Carlo estimate, ξ^1, \dots, ξ^v , then, given an iterate x^s , the result is a recourse function estimate, $\mathcal{Q}^v(x^s) = \frac{1}{v} \sum_{i=1}^v Q(x^s, \xi^i)$, and a corresponding estimate of the gradient, $\nabla \mathcal{Q}(x^s)$, as $\bar{\pi}_s^v = \frac{1}{v} \sum_{i=1}^v \pi_s^i$ where $\pi_s^i \in \partial Q(x^s, \xi^i)$. Now, for Q convex in x , one obtains

$$Q(x, \xi^i) \geq Q(x^s, \xi^i) + (\pi_s^i)^T (x - x^s) \quad (1.4)$$

for all x . We also have that

$$\mathcal{Q}^v(x) = \left(\frac{1}{v} \right) \left(\sum_{i=1}^v Q(x, \xi^i) \right) \geq \mathcal{Q}^v(x^s) + (\bar{\pi}_s^v)^T (x - x^s) = LB_s^v(x), \quad (1.5)$$

where, by the central limit theorem, \sqrt{v} times the right-hand side is asymptotically normally distributed with a mean value,

$$\sqrt{v}(\mathcal{Q}(x^s) + \nabla \mathcal{Q}(x^s)^T (x - x^s)), \quad (1.6)$$

which is a lower bound on $\sqrt{v}\mathcal{Q}(x)$, and a variance, $\rho^s(x)$.

Note that the cut placed on $\mathcal{Q}(x)$ as the right-hand side of (1.5) is a support of \mathcal{Q} with some error,

$$\mathcal{Q}(x) \geq \mathcal{Q}^v(x^s) + (\bar{\pi}_s^v)^T (x - x^s) - \varepsilon_s(x), \quad (1.7)$$

where $\varepsilon_s(x)$ is an error term with zero mean and variance equal to $\frac{1}{v} \rho^s(x)$. Of course, the error term is not known. At iteration s , the L -shaped method involves the solution of:

$$\begin{aligned} & \min c^T x + \theta \\ \text{s. t. } & Ax = b, \\ & D_l x \geq d_l, \quad l = 1, \dots, r, \\ & E_l x + \theta \geq e_l, \quad l = 1, \dots, s, \\ & x \geq 0, \end{aligned} \quad (1.8)$$

where D_l, d_l is a feasibility cut as in (5.1.7)–(5.1.8), $E_l = -\bar{\pi}_l$, and $e_l = \mathcal{Q}^v(x^l) + (\bar{\pi}_l)^T (-x^l)$, where we count iterations only when a finite $\mathcal{Q}^v(x^s)$ is found. Note that the generation of feasibility cuts occurs whenever ξ^i is sampled and $Q(x^l, \xi^i)$ is ∞ .

We suppose that (1.8) is solved to yield x^{s+1} and θ^{s+1} , where

$$\theta^{s+1} = \max_l \{e_l - E_l x^{s+1}\}, \quad (1.9)$$

where each $e_l - E_l x^{s+1}$ can be viewed as a sample from a normally distributed random variable with mean at most $\mathcal{Q}(x^{s+1})$ and variance at most $\frac{1}{v} (\sigma^{\max}(x^{s+1}))^2 = \frac{1}{v} (\max_l \rho^l(x^{s+1}))$. Note that θ^{s+1} is a maximum of these random variables so, if the samples are taken independently on each iteration s , the solution of (1.8)

has a bias that may skew results for large s . Confidence intervals can, however, be developed based on certain assumptions about the functions and the supports. Alternatively, the same sample set, ξ^1, \dots, ξ^v can be used on each iteration so that the L -shaped method iterations solve (1.2) for the given sample with the theory of sample average approximations providing convergence results (see Section 9.5).

If the variances of the sample estimates are sufficiently small, one can stop with a high confidence solution. Other approaches may also be used. Infanger [1991] makes several assumptions that can lead to tight confidence intervals on the optimal value and allow solutions of large problems (see, e.g., Dantzig and Infanger [1991]). Variances and any form of confidence interval may, however, be quite large when crude Monte Carlo samples are used as indicated earlier. Importance sampling can, however, reduce the variance substantially (see Dantzig and Glynn [1990]).

In importance sampling, the goal is to replace a sample using the distribution of ξ with one that uses an alternative distribution that places more weight in the areas of importance. To see this, suppose that ξ has a density $f(\xi)$ over Ξ so that we are trying to find:

$$\mathcal{Q}(x) = \int_{\Xi} Q(x, \xi) f(\xi) d\xi . \quad (1.10)$$

The crude Monte Carlo technique generates each sample ξ^i according to the distribution given by density f .

In importance sampling, a new probability density $g(\xi)$ is introduced that is somewhat similar to $Q(x, \xi)$ and such that $g(\xi) > 0$ whenever $Q(x, \xi) f(\xi) \neq 0$. We then generate samples ξ^i according to this distribution while writing the integral as:

$$\mathcal{Q}(x) = \int_{\Xi} \frac{Q(x, \xi) f(\xi)}{g(\xi)} g(\xi) d\xi . \quad (1.11)$$

In this case, we generate random samples of $\frac{Q(x, \xi) f(\xi)}{g(\xi)}$ from the distribution with density $g(\xi)$. Note that if $g(\xi) = \frac{Q(x, \xi)}{f(\xi) \mathcal{Q}(x)}$, then every sample ξ_{imp}^i under importance sampling yields an importance sampling expectation, $\mathcal{Q}_{imp}^i(x) = \mathcal{Q}(x)$.

Of course, if we could generate samples from the density $\frac{Q(x, \xi)}{f(\xi) \mathcal{Q}(x)}$, we would already know $\mathcal{Q}(x)$. We can, however, use approximations such as the sublinear approximations in Section 8.5 that may be close to $\mathcal{Q}(x)$ and should result in lower variances for \mathcal{Q}_{imp}^v over \mathcal{Q}^v . This approximation is the approach suggested in Infanger [1991].

In the sublinear approximation approach, the approximating density $g(\xi)$ is chosen as

$$g(\xi) = \sum_{i=1}^{m_2} \psi_{I(i)}(T_i x, h_i) f(\xi) / \Psi_I(T x) , \quad (1.12)$$

where g may also depend on x . Using this construction, much lower variances can result in comparison to the crude Monte Carlo approach. One complication is, however, in generating a random sample from the density in (1.12). The general techniques for generating such random vectors is to generate sequentially from

the marginal distributions conditionally, first choosing ξ_1 with the first marginal, $g_1(\xi_1) = \int_{\xi_2, \dots, \xi_N} g(\xi) d\xi$. Then, sequentially, ξ_i is chosen with density, $g_i(\xi_i | \xi_1, \dots, \xi_{i-1})$. Remember that in each case, a random sample with density $g_i(\xi_i)$ on an interval Ξ_i of \mathfrak{X} can be found by choosing from a uniform random sample u from $[0, 1]$ and then taking ξ such that $G(\xi) = u$ where $G(x) = \int_{-\infty}^x g_i(\xi_i) d\xi_i$.

Example 1

Consider Example 1 of Section 8.2 with $x_1 = x_2 = x$. We consider both the crude Monte Carlo approach and the importance sampling using the sublinear approximation for $g(\xi)$. In this case, $g(\xi)$ is actually chosen to depend on x as $g_x(\xi)$ defined by:

$$g_x(\xi) = \frac{|x - \xi_1| + |x - \xi_2|}{E_\xi [|x - \xi_1| + |x - \xi_2|]} . \quad (1.13)$$

For comparison, we first consider the L -shaped method with ξ^i chosen by crude Monte Carlo from the original uniform density on $[0, 1] \times [0, 1]$ and by the importance sampling method with distribution $g_x(\xi)$ in (1.13). The results appear in Figure 1 for the solution x^s at each iteration s of the crude Monte Carlo and importance sampling L -shaped method with $v = 500$ on each L -shaped iteration. The figures show up to 101 L -shaped iterations, which involve more than 50,000 recourse problem solutions.

In Figure 1, the crude Monte Carlo iteration values x appear as $x(\text{crude})$ while the importance sampling iterations appear as $x(\text{imp})$. We also include the optimal solution $x^* = \sqrt{2} - 1$ on the graph. Note that $x(\text{imp})$ is very close to x^* from just over 40 iterations while $x(\text{crude})$ does not appear to approach this accuracy within 100 iterations. Note that $x(\text{imp})$ begins to deteriorate after 80 iterations as the accumulation of cuts increases the probability that some cuts are actually above $\mathcal{Q}(x)$. If each cut is generated independently, this adds a bias to the results since the expectation of the outer linearization is the expectation of the maximum of a set of random approximations, which is greater than the maximum of the expectations of those cuts in an exact procedure. This problem is reduced but not eliminated with importance sampling. As a remedy, a fixed set of samples can be used to obtain convergence for that sample set and then checked for convergence using sequential sampling procedures as discussed in Section 9.5.

The advantage of importance sampling can also be seen in Figure 2, which compares the optimal value $\mathcal{Q}(x^*)$ with sample values, $\mathcal{Q}^v(x^v)$, with crude Monte Carlo denoted as $\mathcal{Q}(\text{crude})$ and $\mathcal{Q}_{\text{imp}}^v(x^v)$ with importance sampling denoted as $\mathcal{Q}(\text{imp})$. Note that the crude Monte Carlo values have a much wider variance, in fact, double the variance of the importance sampling results. Also note that in both sampling methods, the estimates have a mean close to the optimal value after 40 iterations.

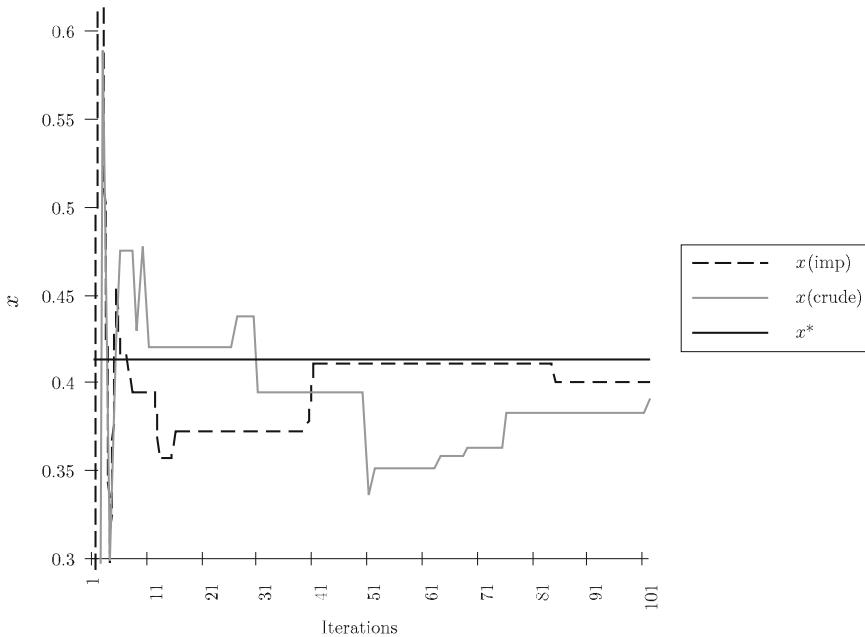


Fig. 1 Solutions for crude Monte Carlo and importance sampling.

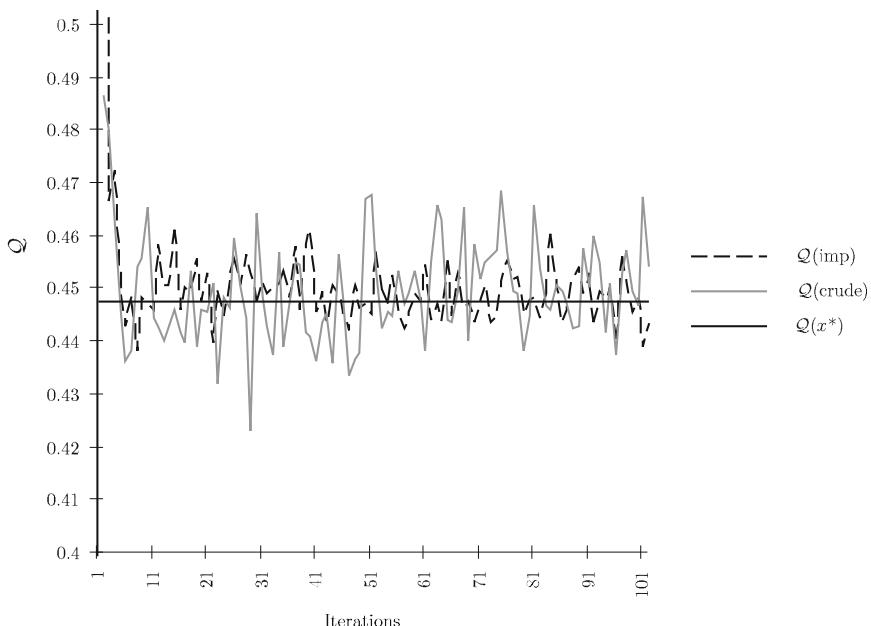


Fig. 2 Objective values for crude Monte Carlo and importance sampling.

The results in Figures 1 and 2 indicate that sampled cuts in the L -shaped method can produce fairly accurate results but that convergence to optimal values may require large numbers of samples for each cut even for small problems and may yield inaccurate results with independent samples for each cut due to the bias issue. This difficulty is particularly an issue if initial cuts are generated with small numbers of samples since these may be particularly inaccurate and limit convergence unless they are removed in favor of more accurate cuts. A procedure to avoid this problem is gradually to remove initial cuts as the algorithm progresses. This is the intent of the approach in the next section.

Exercises

1. Show how to sample from the density $g_x(\xi)$ as the sum of the absolute values $|x - \xi_i|$, $i = 1, 2$ for Example 1.
2. Consider Example 1 in Section 5.1 with ξ uniformly distributed on $[1, 5]$. Apply the crude Monte Carlo L -shaped method to this problem for 100 iterations with 100 samples per cut. What would the result be with importance sampling in this case?
3. Apply both the crude Monte Carlo and importance sampling approaches to Example 1 with both x_1 and x_2 decision variables. First, use 100 samples for each cut for 100 iterations and then compare to results with an increase to 500 samples per cut.

9.2 Stochastic Decomposition

An alternative approach to using cuts produced with multiple samples in the L -shaped method is to use cuts constructed from small but increasing numbers of samples. This approach from Higle and Sen [1991b] generates many cuts with small numbers of additional samples on each cut and adjusts these cuts to drop away as the algorithm continues processing. The method is called *stochastic decomposition*. We will give a basic development here and refer to Higle and Sen [1996] for more details. For simplicity, we assume complete recourse, a known (probability one) lower bound on $Q(x, \xi)$ (e.g., 0), and a bounded set of dual solutions to the recourse problem (3.1.1). We also assume that K_1 and Ξ are compact.

With these assumptions, the basic stochastic decomposition method generates iterates, x^k , and observations, ξ^k . We can state the basic stochastic decomposition method in the following way.

Basic Stochastic Decomposition Method

Step 1. Set $v = 0$, $\xi^0 = \bar{\xi}$, and let x^1 solve

$$\min_{Ax=b, x \geq 0} \{c^T x + Q(x, \xi^0)\}. \quad (2.1)$$

Step 2. Let $v = v + 1$ and let ξ^v be an independent sample generated from ξ . Find $\mathcal{Q}^v(x^v) = \frac{1}{v} \sum_{s=1}^v Q(x^v, \xi^s) = \frac{1}{v} \sum_{s=1}^v (\pi_s^v)^T (\xi^s - Tx^v)$. Let $E_v = \frac{1}{v} \sum_{s=1}^v (\pi_s^v)^T T$ and $e_v = \frac{1}{v} \sum_{s=1}^v (\pi_s^v)^T \xi^s$.

Step 3. Update all previous cuts by $E_s \leftarrow \frac{v-1}{v} E_s$ and $e_s \leftarrow \frac{v-1}{v} e_s$ for $s = 1, \dots, v-1$.

Step 4. Solve the L -shaped master problem as in (1.8) to obtain x^{v+1} . Go to Step 2.

This method differs slightly from the basic method in Higle and Sen [1991b] in our assuming π_s^v to be optimal dual solutions in each iteration. Higle and Sen allow a restricted set of dual optima that may decrease the solution effort (with perhaps fewer effective cuts).

The main convergence result is contained in the following theorem.

Theorem 1. *Assuming complete recourse, $Q(x, \xi) \geq 0$, bounded dual solutions to (3.1.1), K_1 and Ξ compact, there exists a subsequence, $\{x^{v_j}\}$, of the iterates of the basic stochastic decomposition method such that every limit point of $\{x^{v_j}\}$ solves the recourse problem (3.1.1) with probability one.*

Proof: We follow the proof of Theorem 4 in Higle and Sen [1991b]. We use their Theorem 3 (Exercise 1), which gives the existence of a subsequence of $\{x^v\}$ such that

$$\lim_{v \rightarrow \infty} \theta^v - \max_{l=1, \dots, v} (e_{v-1}^l - E_{v-1}^l x^v) = 0. \quad (2.2)$$

Suppose $\{x^{v_j}\}$ is a subsequence of the subsequence achieving (2.2) such that $\lim_j x^{v_j} = \hat{x}$ where $A\hat{x} = b$, $x \geq 0$. This occurs for some subsequence by compactness. From x^* optimal,

$$c^T x^* + \mathcal{Q}(x^*) \leq c^T \hat{x} + \mathcal{Q}(\hat{x}). \quad (2.3)$$

Note that because $Q(x, \xi) \geq 0$ for all $\xi \in \Xi$ and $Q(x, \xi^i) \geq \pi^T (h^i - Tx)$ for any $\pi^T W \leq q$ and any sample ξ^i , for any $1 \leq s \leq v$,

$$\sum_{i=1}^v Q(x, \xi^i) \geq \sum_{i=1}^s \pi^T (h^i - Tx), \quad (2.4)$$

where π is any feasible multiplier in the recourse problem for ξ^i . From (2.4), it follows that $\frac{1}{v} \sum_{i=1}^v Q(x, \xi^i) \geq e_l^v - E_l^v x$ for all l and v , where E_l^v and e_l^v are the components of Cut l on Iteration v . Therefore,

$$c^T x + \max_{l=1,\dots,v} (e_l^v - E_l^v x) \leq c^T x + \frac{1}{v} \sum_{i=1}^v Q(x, \xi^i). \quad (2.5)$$

As v increases, $\frac{1}{v} \sum_{i=1}^v Q(x, \xi^i) \rightarrow \mathcal{Q}(x)$, so

$$\limsup_v [c^T x^* + \max_{l=1,\dots,v} (e_l^v - E_l^v x^*)] \leq c^T x^* + \mathcal{Q}(x^*), \quad (2.6)$$

with probability one. We can also show that (Exercise 2)

$$\lim_j c^T x^{v_j} + \max_{l=1,\dots,v} (e_l^v - E_l^v x^{v_j}) = c^T \hat{x} + \mathcal{Q}(\hat{x}), \quad (2.7)$$

with probability one. Thus, (2.6), (2.7), and the fact that x^{v_j} minimizes $c^T x + \max_{l=1,\dots,v-1} (e_l^{v-1} - E_l^{v-1} x)$ over feasible x yield

$$\begin{aligned} c^T x^* + \mathcal{Q}(x^*) &\leq c^T \hat{x} + \mathcal{Q}(\hat{x}) \\ &\leq \limsup_v [c^T x^* + \max_{l=1,\dots,v} (e_l^v - E_l^v x^*)] \\ &\leq c^T x^* + \mathcal{Q}(x^*), \end{aligned} \quad (2.8)$$

which proves the result. \square

One difficulty in this basic method is that convergence to an optimum may only occur on a subsequence. To remedy this, Higle and Sen suggest retaining an incumbent solution that changes whenever the objective function falls below the best known value so far. The incumbent is updated each time a sufficient decrease in the v th iteration objective value is obtained. They also show that the sequence of incumbents contains a subsequence with optimal limit points, and then show how this subsequence can be identified. Various approaches may be used for practical stopping conditions, such as the statistical verification tests for optimality conditions in Higle and Sen [1991a].

Example 1 (continued)

We again consider Example 1 from Section 8.2. The basic stochastic decomposition method results appear in Figures 3 and 4. In Figure 3, both the basic result x^v and the incumbent solution, x^v (incumbent), which is adjusted whenever a solution after the first 100 iterations improves the previous best estimate by 1%. Figure 3 also gives the optimal solution, x^* . The total number of iterations yields about 50,000 subproblem solutions, which is approximately equal to the total number of iterations in Figures 1 and 2. Note that the raw solutions x^v oscillate rapidly, while the incumbent solutions settle close to x^* quite quickly after their initiation at $v = 100$.

The objective value estimates, θ^v , \mathcal{Q}^v , and $\mathcal{Q}^v(x^v)$ (inc) for the incumbent, and the optimal objective value, $\mathcal{Q}(x^*)$, appear in Figure 4. Note that the θ^v values from the master problem have wide oscillations. The $\mathcal{Q}^v(x^v)$ values have lower but significant variation. The incumbent objective values, however, show low variation that begins to approach the optimum.

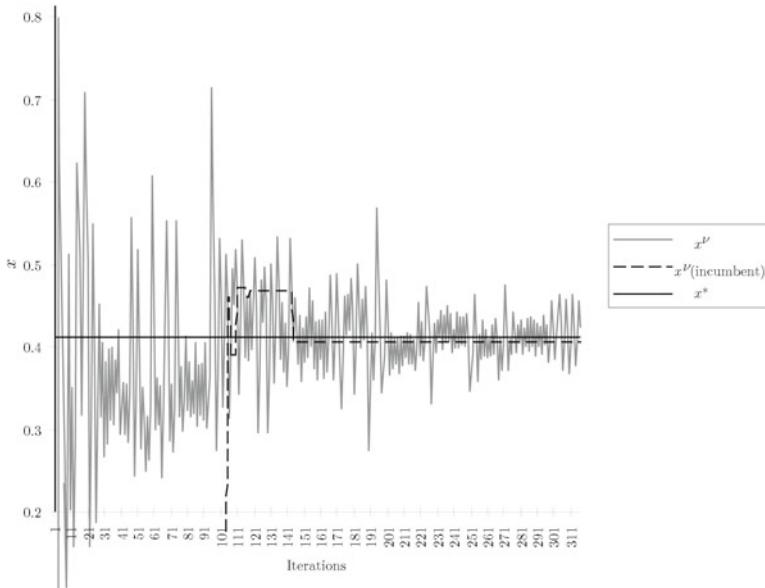


Fig. 3 Solutions for the stochastic decomposition method.

Exercises

1. Prove Theorem 1. Show first that eventually the objective value of (1.8) for x^{v_n} at iteration v_n is the same as the objective value of (1.8) for x^{v_n} at iteration v_{n-1} .
2. Prove that there exists a subsequence of iterates $\{x^v\}$ in the basic stochastic decomposition method with the assumptions so that (2.2) holds.
3. Suppose a subsequence of iterates $x^{v_j} \rightarrow \hat{x}$ in the basic stochastic decomposition method. Prove that (2.7) holds.
4. Apply the basic stochastic decomposition method to Example 1 in Section 5.1 with ξ uniformly distributed on $[1, 5]$. Record the sequence of iterations until 10 consecutive iterations are within 1% of the optimal objective value.

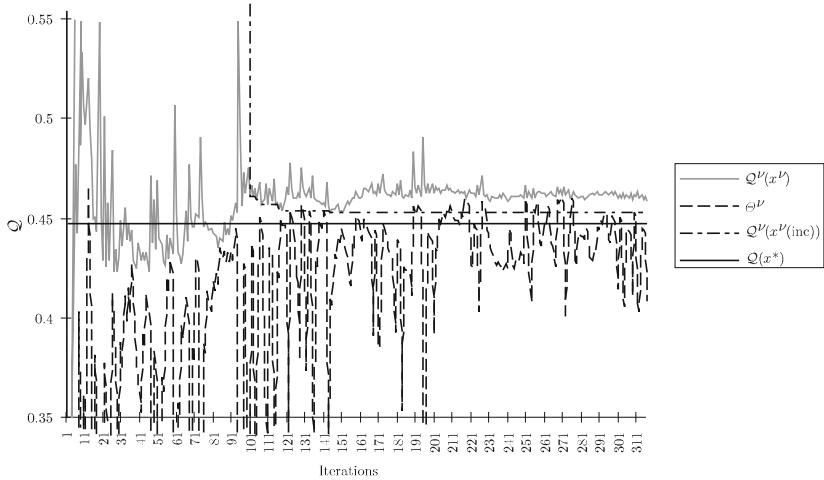


Fig. 4 Objective values for the stochastic decomposition method.

9.3 Stochastic Quasi-Gradient Methods

Stochastic quasi-gradient methods represent one of the first computational developments in stochastic programming. They apply to a broad class of problems and represent extensions of stochastic approximation methods (see, e.g., Dupač [1965] and Kushner [1971]). Our treatment will be brief because the emphasis in this book is on methods that exploit the structure of deterministic equivalent or approximation problems. Ermoliev [1988] provides a more complete survey of these methods.

Stochastic quasi-gradient methods (SQG) apply to a general mathematical program of the form:

$$\begin{aligned} \min_{x \in X \subset \Re^n} & g_0(x) \\ \text{s. t. } & g_i(x) \leq 0, \quad i = 1, \dots, m, \end{aligned} \tag{3.1}$$

where we assume that each g_i is convex. We suppose that an initial point, $x^0 \in X$, is given. The method generates a sequence of points, $\{x^\nu\}$, that converges to an optimal solution of (3.1).

Given a history at time ν , (x^0, \dots, x^ν) , the method selects function estimates, $\eta_i(\nu)$, and subgradient estimates, $\beta_i(\nu)$, such that

$$E[\eta_i(\nu) | (x^0, \dots, x^\nu)] = g_i(x^\nu) + a_i(\nu) \tag{3.2}$$

and

$$E[\beta_i(\nu) | (x^0, \dots, x^\nu)] + b_i(\nu) \subset \partial g_i(x^\nu), \tag{3.3}$$

where $a_i(v)$, $b_i(v)$ may depend on (x^0, \dots, x^v) but must satisfy

$$a_i(v) \rightarrow 0 \quad \text{and} \quad \|b_i(v)\| \rightarrow 0. \quad (3.4)$$

When $b_i(v) \neq 0$, $\beta_i(v)$ is called a *stochastic quasi-gradient*. Otherwise, $\beta_i(v)$ is a *stochastic subgradient*.

We first consider the method when all constraints are deterministic and represented in X . Thus, Problem 3.1 becomes

$$\min_{x \in X \subset \Re^n} g_0(x). \quad (3.5)$$

The method requires a projection onto X represented by

$$\prod_X(y) = \arg \min_x \{ \|x - y\|^2 \mid x \in X \}.$$

In the basic method, a sequence of step sizes $\{\rho^v\}$ is given. The stochastic quasi-gradient method defines a stochastic sequence of iterates, $\{\mathbf{x}^v\}$, by

$$\mathbf{x}^{v+1} = \prod_X[\mathbf{x}^v - \rho^v \beta_0(v)], \quad (3.6)$$

where we interpret the projection as operating separately on each element $\omega \in \Omega$, so that $x^{v+1}(\omega) = \prod_X[(x^v(\omega) - \rho^v \beta_0(\omega)(v))]$.

To place all these results into the two-stage recourse problem as in (1.1.2), let $X = \{x \mid Ax = b, x \geq 0\}$, $g_0(x) = \int g^0(x, \xi) P(d\xi)$ where

$$g^0(x, \xi) = \inf_y \{ \mathbf{q}^T \mathbf{y} \mid W\mathbf{y} = \mathbf{h} - \mathbf{T}x, \mathbf{y} \geq 0 \}.$$

Thus, we can use $\beta_0^i(x)$ such that $\beta_0^i(x)^T(\mathbf{h}^i - \mathbf{T}^i x) = g^0(x, \xi^i)$, $W^T \beta_0^i(x) \leq \mathbf{q}^i$ for a sample ξ^i composed of the components, \mathbf{h}^i , \mathbf{T}^i , and \mathbf{q}^i . The stochastic quasi-gradient method takes a step in this direction and then projects back onto X . In the following example and the exercises, we explore the use of this approach.

For these examples, we use an estimate of the objective value by taking a moving average of the last 500 samples, $\mathcal{Q}^{v-ave}(x^v) = \sum_{i=0}^{499} Q(x^{v-i}, \xi^{v-i}) / 500$. Changes in this estimate (or the lack thereof) can be used to evaluate the convergence of stochastic quasi-gradient methods. Gaivoronski [1988] discusses various practical approaches in this regard.

Example 1 (continued)

We consider the same example and apply the stochastic quasi-gradient method. On each step v , a random sample ξ^v is taken with $\beta_0(v) \in \partial Q(x^v, \xi^v)$. For $X = \{x \mid 0 \leq x \leq 1\}$, the projection operation yields $x^{v+1} = \min(1, \max(x^v + \rho^v \beta_0(v), 0))$.

Figures 5 and 6 show these iterations for solutions x^v and objective estimates, $\mathcal{Q}^{v\text{-ave}}$ for every multiple of 500 iterations up to 50,000 so that total numbers of recourse problem solutions are the same as in Figures 1 to 4.

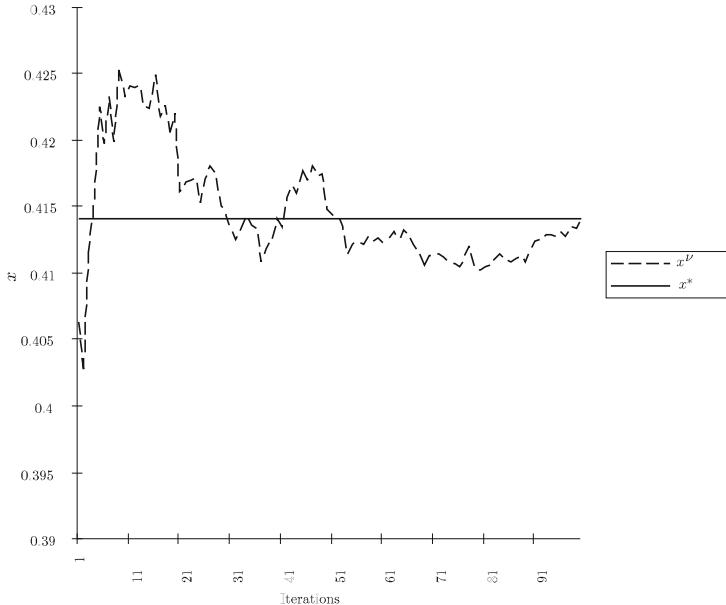


Fig. 5 Solutions for the stochastic quasi-gradient method.

Note that the iterations in Figure 5 appear to approach x^* much more quickly than the results in Figures 1 to 4. They also seem to show lower variances in the objective estimates in Figure 6, although these results are not converging to zero variance because the sample length 500 is not changing. To achieve convergence or greater confidence in a solution, the number of samples in the estimate must increase.

While the results in Figures 5 and 6 indicate that stochastic quasi-gradient methods may be more effective than the decomposition methods, we should note that this example is quite low in dimension. For higher dimensions, the results are often quite different. In general, stochastic quasi-gradient methods exhibit similar behavior to subgradient optimization methods that often have slow convergence properties in higher dimensions. They are, nonetheless, easy to implement and can give good results, especially in small problems.

In the rest of this section, we discuss the theory behind the stochastic quasi-gradient method convergence. The exercises consider examples for using SQG.

The basic method in (3.6) traces back to the unconstrained methods of Robbins-Monro [1951]. The main device in demonstrating convergence of $\{\mathbf{x}^v\}$ to a point

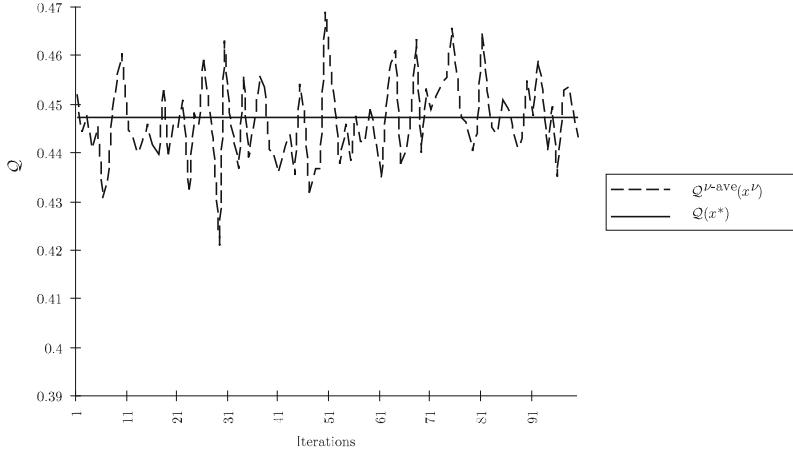


Fig. 6 Objective values for the stochastic quasi-gradient method.

in X^* is the use of a *stochastic quasi-Feyer sequence* (see Ermolieva [1969]), a sequence of random vectors, $\{\mathbf{w}^v\}$, defined in (Ω, Σ, P) such that for a set $W \subset \mathfrak{R}^n$, $E[\|\mathbf{w}^0\|^2] < +\infty$, and any $w \in W$,

$$E\{\|w - \mathbf{w}^{v+1}\|^2 | \mathbf{w}^0, \dots, \mathbf{w}^v\} \leq \|w - \mathbf{w}^v\|^2 + \gamma_v, \\ v = 0, 1, \dots, \quad \gamma_v \geq 0, \quad \sum_{v=0}^{\infty} E[\gamma_v] < +\infty. \quad (3.7)$$

The following result shown in Ermolieva [1976] is the basis for the convergence results.

Theorem 2. *If $\{\mathbf{w}^v\}$ is a stochastic quasi-Feyer sequence for a set W , then*

- (a) $\{\|w - \mathbf{w}^{v+1}\|^2\}$ converges with probability one for any $w \in W$, and $E[\|w - \mathbf{w}^v\|^2] < c < +\infty$;
- (b) the set of limit points of $\{w^v(\omega)\}$ is nonempty for almost all $\omega \in \Omega$;
- (c) if $\bar{w}^1(\omega)$ and $\bar{w}^2(\omega)$ are two distinct limit points of $\{w^v(\omega)\}$ such that $\bar{w}^1(\omega) \notin W$, $\bar{w}^2(\omega) \notin W$, then $W \subset H$, a hyperplane such that $\eta = \{w | \alpha^T w = \alpha_0\}$, $\|\bar{w}^1(\omega) - \Pi_{\eta}(\bar{w}^1(\omega))\| = \|\bar{w}^2(\omega) - \Pi_{\eta}(\bar{w}^2(\omega))\|$, where Π_{η} denotes projections onto η .

With this result, we can obtain the most basic convergence result, given below without proof.

Theorem 3. *Given the following:*

(a) $g_0(x)$ is convex and continuous,

(b) X is a convex compact set,

(c) the parameters, ρ_v , and $\gamma^0(v) = \inf_{x^* \in X^*} \beta_0(v)^T (\mathbf{x}^v - x^*)$, satisfy with probability one,

$$\begin{aligned} \rho^v > 0, \quad \sum_{v=0}^{\infty} \rho^v &= +\infty, \\ \sum_{v=0}^{\infty} E[\rho^v | \gamma^0(v) | + (\rho^v)^2 \|\beta_0(v)\|^2] &< \infty, \end{aligned} \quad (3.8)$$

then, with probability one, for any $\bar{x}(\omega) = \lim_{V_i} x^{V_i}(\omega)$, $\bar{x}(\omega) \in X^*$.

The general method can be amplified in a variety of ways. Condition (c) can be relaxed to remove the finiteness of $\sum_{v=0}^{\infty} \rho_v^2$ when $\gamma(v) = 0$ for all v , but the convergence is for $\frac{\sum_v \mathbf{x}^v \rho^v}{\sum_v \rho^v}$ (see Uriasiev [1988]).

Two important aspects of stochastic quasi-gradient implementations are the determinations of step sizes and stopping rules. Various adaptive step sizes are considered by Mirzoachmedov and Uriasiev [1983]. For stopping rules, we refer to Pflug [1988], where details appear. The results describe the use of stopping times $\{\tau_\epsilon\}$, to yield uniform asymptotic level α confidence regions, defined by

$$\liminf_{\epsilon \rightarrow 0} P_{x^0}\{\|\mathbf{x}^{\tau_\epsilon} - x^*\| \leq \epsilon\} \geq 1 - \alpha. \quad (3.9)$$

Deterministic step size rules do not, unfortunately, produce such uniform confidence intervals. Instead, Pflug shows that an oscillation test stopping rule does obtain such confidence regions. In this rule, a test is performed to check whether the iterates are oscillating without objective improvement. The key is building consistent estimates of the objective Hessian at x^* and the covariance matrix of objective errors. For other issues concerning implementation, we refer to Gaivoronski [1988].

The use of sample subgradients can also produce results for efficient computation with some confidence level. Nesterov and Vial [2008] give one of these results for a stochastic subgradient method in which the values of multiple algorithm paths are combined to achieve efficient computation. Dyer, Kannan, and Stougie [2002] and Shmoys and Swamy [2006] both use stochastic subgradients in a different way to define regions of improvement over which the ellipsoid optimization method can be applied. Combining this approach and a search step can for many stochastic linear programs in fact find a solution within ϵ of optimal with effort on the order of $\frac{1}{\epsilon}$ and a factor depending on the problem input size (Shmoys and Swamy [2006]).

Exercises

1. Consider Example 1 in Section 5.1. Find the projection of a point onto $X = \{x \mid 0 \leq x \leq 10\}$. Solve this problem using the stochastic quasi-gradient method until 20 consecutive iterations are within 1% of the optimal solution.
2. Consider Example 1, where both x_1 and x_2 can be chosen instead of $x = x_1 = x_2$. Follow the stochastic quasi-gradient method again until 20 consecutive iterations are within 1% of the optimal solution.
3. Prove Theorem 2.
4. Consider Example 1 in Section 5.1. Find the projection of a point onto $X = \{x \mid 0 \leq x \leq 10\}$. Solve this problem using the stochastic quasi-gradient method until three consecutive iterations are within 1% of the optimal solution.

9.4 Sampling Methods for Probabilistic Constraints and Quantiles

Monte Carlo sampling methods can also be quite useful for general types of probabilistic constraints, such as:

$$\Pr\{g_i(x, \xi) \leq 0, i = 1, \dots, m\} \geq \alpha, \quad (4.1)$$

where the functions $g_i, i = 1, \dots, m$ are all convex in $x \in \mathbb{R}^n$. In this case, a simple procedure is to select a random independent sample of v realizations, $\{\xi^1, \dots, \xi^v\}$, of ξ and, with a linear objective, to solve the sample problem:

$$\begin{aligned} & \min c^T x \\ & \text{s. t. } g_i(x, \xi^k) \leq 0, i = 1, \dots, m; k = 1, \dots, v. \end{aligned} \quad (4.2)$$

As shown in Calafiore and Campi [2005], we have the following:

Theorem 4. *If $v \geq \frac{n}{\epsilon\beta} - 1$ for any $\epsilon \in (0, 1 - \alpha]$ and $\beta \in (0, 1)$, then with probability at least $1 - \beta$, the solution x^v to (4.2) also satisfies the probabilistic constraint (4.1).*

Other results for general conditions on x , such as in de Farias and Van Roy [2005] can also be useful in this context. To see how these approximations can work, consider Example 8.3 for a single period where each A_{ij} is a Bernoulli random variable that each loan has a value of one at maturity if not in default and has value zero otherwise. To include correlation, we suppose the Merton [1974] model where default occurs for loan j if (the natural logarithm of) its underlying asset value satisfies the inequality, $\sqrt{\rho}\xi_0 + \sqrt{1-\rho}\xi_j \leq d_j$ for some default point d_j , where ρ is the correlation between any pair of underlying asset values, and ξ_0 and ξ_j are independent and normally distributed random variables with zero mean with unit

variance. In terms of the probabilistic constraint, $\mathbf{A}_j = \mathbf{1}_{\{\sqrt{\rho}\xi_0 + \sqrt{1-\rho}\xi_j > d_j\}}$ where $\mathbf{1}$ is the indicator with value one on the given set. The probability of a default for loan $j (= 1, \dots, n)$ is $p = p_j = 1 - E[\mathbf{A}_j] = \Phi(d_j)$, where Φ is the standard normal cumulative distribution function. (In this development, which follows Vasicek [1987, 1991, 2002], the time to maturity is normalized to one and the information about the drift and volatility of the underlying asset values are subsumed in d_j .)

We can evaluate the probability of satisfying the constraint with liability level h as

$$P\{\mathbf{Ax} \geq h\} = P\left\{\sum_{j=1}^n \mathbf{A}_j \geq \frac{hn}{b}\right\}, \quad (4.3)$$

and then use the probability mass function of $\sum_{j=1}^n \mathbf{A}_j$ given by (Exercise 1):

$$\begin{aligned} P\left\{\sum_{j=1}^n \mathbf{A}_j = n - k\right\} &= \binom{n}{k} \int_{-\infty}^{\infty} \left(\Phi\left(\frac{1}{\sqrt{1-\rho}}(\Phi^{-1}(p) - \sqrt{\rho}s)\right)\right)^k \\ &\quad \left(\Phi\left(\frac{1}{\sqrt{1-\rho}}(\Phi^{-1}(p) - \sqrt{\rho}s)\right)\right)^{n-k} d\Phi(s). \end{aligned} \quad (4.4)$$

Using (4.4) to solve:

$$\min b \text{ s. t. } P\{\mathbf{Ax} \geq h\} \geq \alpha, x = \frac{b}{n}, \quad (4.5)$$

may not be practical for large n (e.g., when $n = 125$ as in the example). Instead, we can use the sample approximation in (4.2). Exercise 2 asks for this computation for typical default probabilities and correlations.

The sampling approximation in (4.2) can be relaxed with some allowable fraction of constraint violations to achieve more precise approximations. Luedtke and Ahmed [2008] use Hoeffding's inequality to achieve bounds from this sampling approach. Other approximations for probabilistic constraints appear in Deák [1980], Gassmann [1988], Szántai [1986], and elsewhere. We briefly describe Szántai's method here. The basic idea is to use Bonferroni-type inequalities to write the probability of a set with many constraints in terms of sums and differences of integrals of subsets of the constraints, as we described in Section 8.5. In sampling procedures, these alternative estimates allow for significant variance reduction.

For Szántai's approach, suppose we wish to find

$$p = P[A = A_1 \cap \dots \cap A_m] = \int_A dF(\xi). \quad (4.6)$$

Szántai takes three estimates of p :

1. \hat{p}^1 —a direct Monte Carlo sample;
2. \hat{p}^2 —finding the first-order Bonferroni terms, $1 - \sum_{i=1}^m P(\hat{A}_i)$, directly and sampling from higher-order terms;

3. \hat{p}^3 — Calculating the first- and second-order terms explicitly, $1 - \sum_{i=1}^m P(\hat{A}_i) + \sum_{i < j} P(\hat{A}_i \cap \hat{A}_j)$, and sampling from higher order terms.

Sampling from all higher order terms may be difficult, but Szántai shows that the effort may be reduced at each sample ξ^j to finding $\hat{n}(j)$ defined as the number of constraints violated by ξ^j , i.e., $\hat{n}(j) = \sum_{i=1}^N 1_{\{\xi^j \notin A_i\}}$. With this quantity defined, we can define *unbiased estimates*, i.e., estimates whose expectations have no error, using the following:

$$\gamma^1 = \frac{1}{v} \sum_{j=1}^v \max\{0, 1 - \hat{n}(j)\}, \quad (4.7)$$

$$\gamma^2 = \frac{1}{v} \sum_{j=1}^v \max\{0, \hat{n}(j) - 1\}, \quad (4.8)$$

$$\text{and} \quad \gamma^3 = \frac{1}{v} \sum_{j=1}^v \frac{\max\{0, \hat{n}(j) - 1\} \hat{n}(j) - 2}{2}. \quad (4.9)$$

These quantities are then used to form unbiased estimates:

$$\hat{p}^1 = \gamma^1, \quad (4.10)$$

$$\hat{p}^2 = 1 - \sum_{i=1}^m P[\hat{A}_i] + \gamma^2, \quad (4.11)$$

$$\hat{p}^3 = 1 - \sum_{i=1}^m P[\hat{A}_i] + \sum_{i < j} \sum P[\hat{A}_i \cap \hat{A}_j] - \gamma^3. \quad (4.12)$$

These three estimators are combined to form

$$\hat{p}^4 = \lambda_1 \hat{p}^1 + \lambda_2 \hat{p}^2 + (1 - \lambda_1 - \lambda_2) \hat{p}^3, \quad (4.13)$$

where the weights λ_1 and λ_2 are chosen to minimize the variance of \hat{p}^4 . They are calculated using the sample covariance matrix of $(\gamma^1, \gamma^2, \gamma^3)$, which we denote as $C = [c_{ij}]$. In this case, $\lambda_1 = \frac{\mu_1}{\mu_1 + \mu_2 + \mu_3}$, $\lambda_2 = \frac{\mu_2}{\mu_1 + \mu_2 + \mu_3}$, where

$$\mu_1 = c_{12}(c_{33} - c_{23}) + c_{22}(c_{13} - c_{33}) + c_{23}(c_{23} - c_{13}), \quad (4.14)$$

$$\mu_2 = c_{11}(c_{23} - c_{33}) + c_{12}(c_{33} - c_{13}) + c_{13}(c_{123} - c_{23}), \quad (4.15)$$

$$\text{and} \quad \mu_3 = c_{11}(c_{23} - c_{22}) + c_{12}(c_{12} - c_{23}) + c_{13}(c_{22} - c_{12}). \quad (4.16)$$

The result is that \hat{p}^4 can have significantly lower variance than standard Monte Carlo. In fact, Szántai obtains efficiencies (variance ratios) of 100 and higher, implying that the same error can be obtained with \hat{p}^4 in 1% of the number of samples for using \hat{p}^1 alone.

This approach combines analytical techniques with simulation to produce lower variance. Another approach is to use empirical sample information. This is the area studied in Jaggaranathan [1985], where some sample information can be used in a Bayesian framework to determine probabilities of underlying distributions. These may be used for probabilistic constraints, for recourse functions, or for both.

As an example, consider the basic two-stage model in (3.1.1), where the distribution function of ξ is $F(\xi, \eta)$, where η is a k -vector of unknown parameters with prior distribution function, $G(\cdot)$. Given an observation, $\hat{\xi}^l = (\xi^1, \dots, \xi^l)$, we can define a posterior distribution, $G_l(\cdot | \hat{\xi}^l)$. Using this, we may obtain an improved solution.

Without sample information, we would have the solution to (3.1.1) as

$$R(G) = \min_{x \in K_1} \left\{ c^T x + \int_{\eta} \int_{\xi} Q(x, \xi) F(\xi, \eta) G(d\eta) \right\}. \quad (4.17)$$

However, using $\hat{\xi}^l$, which we assume has a conditional distribution given by $W(\hat{\xi}^l, \eta)$ for some value η of η , we obtain a value with sample information as

$$\begin{aligned} R^l(G) = \int_{\eta} & \left[\min_{x \in K_1} \left\{ c^T x \right. \right. \\ & \left. \left. + \int_{\xi} \int_{\xi} Q(x, \xi) F(d\xi, \eta) G_l(d\eta | \hat{\xi}^l) \right\} \right] W(d\hat{\xi}^l, \eta) G(d\eta). \end{aligned} \quad (4.18)$$

The difference $R^l(G) - R(G)$ is the *expected value of sample information*. This represents the additional expected value from observing the sample information. This type of analysis can also be extended to problems with probabilistic constraints.

A different use of sample information is for dynamic problems that may change over time. In these cases, future characteristics, such as product demand, may not be known with certainty but they can be predicted roughly using past experience. These problems were examined by Cipra [1991], who also considered the possibility that more recent information might be more valuable than older information.

For example, consider the news vendor problem in Section 1.1. Suppose the demand occurs as ξ^t for periods $t = 1, \dots, H$. At time H , suppose that $\xi^H = (\xi_1, \dots, \xi_H)$ have been observed. The news vendor wishes to place an order based on these observations. One solution might be to use a discount factor, $\beta \in (0, 1)$, to choose $x(H)$ to

$$\min_{x \geq 0} \left(\sum_{i=0}^{H-1} \beta^i ((a-s)x + (s-r)(x - \xi_{H-i})^+) \right). \quad (4.19)$$

The solution of this problem is straightforward (Exercise 5). Alternative perspectives on the value of empirical observations can also be introduced, as could Bayesian approaches as in (4.18). For another view of decisions made over time, refer to Jaggaranathan [1991].

Exercises

1. Derive (4.4) (Vasiček [1987]).
2. Use a sufficient number of samples v from Theorem 4 for the sample approximation, (4.2), to ensure that the probabilistic constraint in (4.5) is satisfied with a confidence level of $1 - \beta = 0.99$ with target reliability level, $\alpha = 0.95$, $h = 0.95$, $n = 125$, $p = 0.01$ for the probability of default on any single loan, and correlation coefficient $\rho = 0.5$. (Since b is the only decision parameter to consider when all the loans are symmetric, you can assume the dimension to use in computing v is one.) Find the minimum b^* for 100 different sample problems. Verify the result in Theorem 4 empirically by constructing a sample of 10,000 sample portfolios and solving (4.5) for this large sample. What would you expect to happen if the problem is interpreted as making 125 separate decisions x_j on the initial size of loan j ?
3. In Exercise 2, you should have noticed that Theorem 4 provides an estimate of b^* that appears quite conservative. An alternative approximation is to use a limiting distribution when a portfolio contains many loans. Suppose F_n is the cumulative distribution of the fractional loss of a portfolio of n loans so that $F_n(\delta) = \sum_{k=1}^{\lfloor \delta n \rfloor} P\{\sum_{j=1}^n A_j = n - k\}$. Substituting y for $\Phi(\frac{1}{\sqrt{1-\rho}}(\Phi^{-1}(p) - \sqrt{\rho}s))$ and $dG(y) = d\Phi(s)$, $F_n(\delta)$ can be written as:

$$F_n(\delta) = \sum_{k=1}^{\lfloor \delta n \rfloor} \binom{n}{k} \int_0^1 y^k (1-y)^{n-k} dG(y). \quad (4.20)$$

Take the limit $n \rightarrow \infty$ in (4.20) to show:

$$F_\infty(\delta) = G(\delta), \quad (4.21)$$

and find G (Vasiček [1991]). Compare this approximation to your simulation results in Exercise 2.

4. Show that \hat{p}^i are unbiased estimators of the probability p in (4.6).
5. Suppose that γ_i , $i = 1, 2, 3$, are independent standard gamma random variables with parameters, η_i , $i = 1, 2, 3$. Let $\mathbf{x}_i = \gamma_1 + \gamma_{i+1}$ for $i = 1, 2$. Give a one dimensional integral that represents $P[\mathbf{x}_i \leq w_i, i = 1, 2]$ using cumulative gamma distribution functions in the integrand.
6. The result in Exercise 2 allows calculations of \hat{p}^2 . For example, suppose that \mathbf{y}_i , $i = 1, 2, 3, 4$ in Exercise 2 and $\mathbf{x}_i = \mathbf{y}_1 + \mathbf{y}_{i+1}$ for $i = 1, 2, 3$. Find \hat{p}_4 for $p = P[\mathbf{x}_i \leq z_i, i = 1, 2, 3]$ when $z_i = 6$, $i = 1, 2, 3$, and $\eta_i = 3$, $i = 1, 2, 3, 4$. Also, find sample variances for increasing sample sizes and compare to the sample variances for \hat{p}_1 .
7. Suppose that ξ is known to take on a finite number K of possible values but the probabilities η^i of these values are not known but have a Dirichlet prior distribution. Show how to find $R(G)$ and $R^l(G)$ in this case.

8. Find the solution to (4.19). (Hint: Order the observed demands.)

9.5 General Results for Sample Average Approximation and Sequential Sampling

We will give a brief overview of general sampling results. For this analysis, we consider a stochastic program in the following basic form:

$$\inf_{x \in X} \int_{\Xi} g(x, \xi) P(d\xi), \quad (5.1)$$

where $X \subset \Re^n$ and ξ is now defined on the probability space (Ξ, \mathcal{B}, P) so that we can work directly with ξ instead of through ω . Suppose that (5.1) has an optimal solution, x^* , and value, z^* .

A direct sampling approach to solving (5.1) is to consider an approximate problem derived by taking v samples from ξ . The discrete distribution with these samples could be P^v , which would allow us to apply the results in Chapter 9 to obtain convergence of the v problem optimal solutions to the optimal solution in (5.1). It can be even more valuable to describe distributional properties of these solutions so that we can construct confidence intervals in place of the (probability one) bounds found in Chapter 8.

We therefore wish to consider a sample $\{\xi^i\}$ of independent observations of ξ that are used in the general sample average approximation (SAA) problem:

$$z^v = \inf_{x \in X} \frac{1}{v} \sum_{i=1}^v g(x, \xi^i). \quad (5.2)$$

Suppose that \mathbf{x}^v is the random vector of solutions to (5.2) with independent random samples, ξ^i , $i = 1, \dots, v$. The general question considered in King and Rockafellar [1993] is to find a distribution \mathbf{u} such that $\sqrt{v}(\mathbf{x}^v - x^*)$ converges to \mathbf{u} in distribution. Properties of \mathbf{u} can then be used to derive confidence intervals for x^* from an observation of \mathbf{x}^v .

We give the main result without proof. The interested reader can refer to King and Rockafellar [1993] and, for the statistical origin, Huber [1967].

Theorem 5. Suppose that $g(\cdot, \xi)$ is convex and twice continuously differentiable, X is a convex polyhedron, $\nabla g : \Xi \times \Re^n \mapsto \Re^n$:

- i. is measurable for all $x \in X$;
- ii. satisfies the Lipschitz condition that there exists some $a : \Xi \mapsto \Re$, $\int_{\Xi} |a(\xi)|^2 P(d\xi) < \infty$, $|\nabla g(x_1, \xi) - \nabla g(x_2, \xi)| \leq a(\xi)|x_1 - x_2|$, for all $x_1, x_2 \in X$;

iii. satisfies that there exists $x \in X$ such that $\int_{\Xi} |g(x, \xi)|^2 P(d\xi) < \infty$; and, for $G^* = \int \nabla^2 g(x^*, \xi) P(d\xi)$,

$$\text{iv. } (x_1 - x_2)^T G^* (x_1 - x_2) > 0, \quad \forall x_1 \neq x_2, x_1, x_2 \in X.$$

Then the solution \mathbf{x}^v to (5.2) satisfies:

$$\sqrt{v}(\mathbf{x}^v - x^*) \mapsto \mathbf{u}, \quad (5.3)$$

where \mathbf{u} is the solution to:

$$\begin{aligned} & \min \frac{1}{2} u^T G^* u + \mathbf{c}^T u \\ & \text{s. t. } A_i u_i \leq 0, i \in I(x^*), u^T \nabla \bar{g}^* = 0, \end{aligned} \quad (5.4)$$

$X = \{x \mid Ax \leq b\}$, (x^*, π^*) solve $\nabla \int_{\Xi} g(x^*, \xi) P(d\xi) + (\pi^*)^T A = 0$, $\pi^* \geq 0$, $Ax^* \leq b$, $I(x^*) = \{i \mid A_i x^* = b_i\}$, $\nabla \bar{g}^* = \int \nabla g(x^*, \xi) P(d\xi)$, and \mathbf{c} is distributed normally $N(0, \Sigma^*)$ with $\Sigma^* = \int (\nabla g(x^*, \xi) - \nabla \bar{g}^*)(\nabla g(x^*, \xi) - \nabla \bar{g}^*)^T P(d\xi)$.

Example 2

Suppose that $X = [a, \infty)$, ξ is normally distributed $N(0, 1)$, and $g(x, \xi) = (x - \xi)^2$. Problem (5.1) then becomes:

$$\inf_{x \geq a} \int_{\Xi} \frac{(x - \xi)^2}{\sqrt{2\pi}} e^{-\frac{\xi^2}{2}} d\xi, \quad (5.5)$$

where we substituted for P the standard normal density with mean zero and unit standard deviation.

Because the expectation in (5.5) is just $x^2 + 1$, for $a \geq 0$, the clear solution is $x^* = a$. For $a < 0$, $x^* = 0$. In this case, $\nabla g(x^*, \xi) = 2(x^* - \xi)$, $G^* = 2$, $A = [-1]$, and $\nabla \bar{g}^* = 2x^*$. The variance of \mathbf{c} is $\Sigma^* = E_{\xi}[(2\xi)^2] = 4$. The asymptotic distribution \mathbf{u} then solves:

$$\begin{aligned} & \min u^2 + \mathbf{c}^T u \\ & \text{s. t. } u \geq 0 \quad \text{if } x^* = a, u(2x^*) = 0. \end{aligned} \quad (5.6)$$

For $a > 0$, the solution of (5.6) is $u^* = 0$ so that asymptotically $\sqrt{v}(\mathbf{x}^v - x^*) \mapsto 0$ in distribution. If $a = 0$, then note that because $\mathbf{c}/2$ is $N(0, 1)$, the overall result is that asymptotically the estimate, $\sqrt{v}\mathbf{x}^v$, for (5.5) approaches a distribution with a probability mass of 0.50 at 0 and the density of the normal distribution, $N(0, 1)$, over $(0, \infty)$. Exercise 1 asks the reader to find the asymptotic distribution for $a < 0$. In each case, the actual distribution of \mathbf{x}^v can be found and compared to the asymptotic result (see Exercise 2).

Many other results along these lines are possible (see, e.g., Dupačová and Wets [1988]). They often concern the stability of the solutions with respect to the underlying probability distribution. For example, one might only have observations of some random parameter but may not know the parameter's distribution. This type of analysis appears in Dupačová [1984], Römisch and Schultz [1991a], and the survey in Dupačová [1990].

Another useful result is to have asymptotic properties of the optimal approximation value. For this, suppose that z^* is the optimal value of (5.1) and z^v is the random optimal value of (5.2). We use properties of g and ξ^i so that each $g(x, \xi^i)$ is an independent and identically distributed observation of $g(x, \xi)$, and $g(x, \xi)$ has finite variance, $\text{Var}(g(x)) = \int_{\Xi} |g(x, \xi)|^2 P(d\xi) - (\text{E}g(x))^2$. We can thus apply the central limit theorem to state that $\sqrt{v}[(\frac{1}{v}) \sum_{i=1}^v g(x, \xi^i) - \int_{\Xi} g(x, \xi) P(d\xi)]$ converges to a random variable with distribution, $N(0, \text{Var}(g(x)))$. Moreover, with the condition in Theorem 5, the random function on x defined by $\sqrt{v}[(\frac{1}{v}) \sum_{i=1}^v g(x, \xi^i) - \int_{\Xi} g(x, \xi) P(d\xi)]$ is continuous. We can then derive the following result of Shapiro [1991, Theorem 3.3].

Theorem 6. Suppose that X is compact and g satisfies the following conditions:

- i. $g(x, \cdot)$ is measurable for all $x \in X$;
- ii. there exists some $a : \Xi \mapsto \mathbb{R}$, $\int_{\Xi} |a(\xi)|^2 P(d\xi) < \infty$, $|g(x_1, \xi) - g(x_2, \xi)| \leq a(\xi)|x_1 - x_2|$, for all $x_1, x_2 \in X$;
- iii. for some $x^0 \in X$, $\int g(x^0, \xi) P(d\xi) < \infty$;

and $\text{E}g(x)$ has a unique minimizer $x^0 \in X$. Then $\sqrt{v}[z^v - z^*]$ converges in distribution to a normal $N(0, \text{Var}g(x^0))$.

Further results along these lines are possible using the specific structure of g for the recourse problem as in (3.1.1). For example, if K_1 is bounded and \mathcal{Q} has a strong convexity property, Römisch and Schultz [1991b] show that the distance between the optimizing sets in (5.1) and (5.2) can be bounded.

Given the results in Theorems 5 and 6 and some bounds on the variances and covariances, one can construct asymptotic confidence intervals for solutions using (5.2). We discuss this use in a sequential sampling method below. In addition, note that all previous discrete methods can be applied to (5.2) to obtain solutions as v increases. Various procedures can be used to increment v and solving the resulting approximation (5.2) using a previous solution.

Stronger results than Theorem 6 are possible when the minimum in (5.1) is a *sharp minimum* in the following sense:

$$\text{E}g(x, \xi) \leq \text{E}g(x^*, \xi) + k\|x - x^*\|, \quad (5.7)$$

for some $k > 0$ for all $x \in X$. In this case, with probability one, $x^v = x^*$ for all v sufficiently large, i.e., the convergence is exact, and, for two-stage stochastic linear programs with relatively complete recourse, the rate of convergence is exponentially fast, i.e., the probability of not converging in v iterations is bounded by $\alpha e^{-\beta v}$ for some constants $\alpha > 0$ and $\beta > 0$ (Shapiro and Homem-de-Mello [2000]). Similar

convergence results in general cases are possible using large-deviation theory such that the probability of error in the objective value and in the solution (under certain conditions) is greater than any tolerance decreases exponentially fast in the number of samples. We state these results in the following theorem (see Theorems 3.1 and 3.2 in Dai, Chen, and Birge [2000] for the proof).

Theorem 7. Assume that there exist $a > 0$, $\theta_0 > 0$, $\eta(\cdot) : \mathbb{R}^m \rightarrow \mathbb{R}^1$ such that

$$|g(x, \xi)| \leq a\eta(\xi), \quad \mathbb{E}[e^{\theta\eta(\xi)}] < \infty$$

for all $x \in X$ and for all $0 \leq \theta \leq \theta_0$; then, for any $\varepsilon > 0$, there are $\alpha > 0, \beta > 0$ such that

$$\mathbb{P}[\mathbb{E}[\mathbf{z}^\nu - z^*]] \geq \varepsilon \leq \alpha e^{-\beta\nu}, \quad (5.8)$$

for all $\nu > 0$, and, if x^* is unique,

$$\mathbb{P}[||x^\nu - x^*|| \geq \varepsilon] \leq \alpha e^{-\beta\nu} \quad (5.9)$$

for all $\nu \geq 1$.

Theorem 7 provides the possibility of some stopping criteria for sampling methods to achieve approximate optimality with some confidence. Exercise 4 asks for estimates of the parameters α and β for Example 1. In some cases (with a quadratic objective) discussed in Dai, Chen, and Birge [2000], these parameters can be found explicitly (although these analytical values of the parameters often result in loose bounds). These results provide asymptotic results that may be used in an algorithm to obtain convergence within some tolerance of the optimal solution value with a given level of confidence. A key aspect of these procedures is that they need to include increasingly large sample sizes to ensure that the algorithm does not terminate prematurely. We give a basic algorithm from Bayraksan and Morton [2009] that follows earlier results in Morton [1998] and Mak, Morton, and Wood [1999].

We wish to use convergence for the two-stage, sample-average linear program with ν samples. In this case, x^* may not be unique. In that case, we let the set of optima be X^* and let $x_{\min}^*(x) = \operatorname{argmin}_{x' \in X^*} \operatorname{Var}[g(x, \xi) - g(x', \xi)]$. For the two-stage model, we have $g(x, \xi) = c^T x + \{\min \mathbf{q}^T \mathbf{y} | \mathbf{W}\mathbf{y} = \mathbf{h} - \mathbf{T}x, \mathbf{y} \geq 0\}$, which means that we can write the sample-average approximation problem (5.2) as follows:

$$\begin{aligned} z^N &= \min c^T x + \frac{1}{\nu} \sum_{i=1}^{\nu} q_i^T y_i \\ \text{s. t. } &Ax = b, \\ &T_i x + W y_i = h_i, \\ &x \geq 0, y \geq 0, \end{aligned} \quad (5.10)$$

with optimal solution $(x^\nu, y_1^\nu, \dots, y_\nu^\nu)$. We would like to estimate the gap to optimality,

$$\Delta(x^\nu) = \mathbb{E}[g(x, \xi)] - z^*, \quad (5.11)$$

and the smallest variance of the objective difference among the optimal solutions,

$$\sigma^2(x) = \text{Var}[g(x, \xi) - g(x_{\min}^*(x), \xi)]. \quad (5.12)$$

We then define an optimality gap estimator and its sample variance estimator as follows:

$$G_v(x) = \frac{1}{v} \sum_{i=1}^v (g(x, \xi^i) - g(x^v, \xi^i)) \quad (5.13)$$

$$s_v^2(x) = \frac{1}{v-1} \sum_{i=1}^v [(g(x, \xi^i) - g(x^v, \xi^i)) - G_v(x)]^2. \quad (5.14)$$

The goal in sequential sampling is to obtain, for any given confidence level $\alpha \in (0, 1)$ and tolerance $\varepsilon > 0$, x^v after v samples such that

$$\liminf_{l \downarrow l'} P(E[g(x^v, \xi) - g(x^*, \xi)] \leq lb + \varepsilon') \geq 1 - \alpha \quad (5.15)$$

for some parameters $l > l' > 0$, $b > 0$ and $\varepsilon > \varepsilon'$. For defining an algorithm, we use additional parameters: k_f , the frequency of re-sampling, and $p > 0$, which is used in determining a minimum sample size for k iterations as follows:

$$v_k \geq \left(\frac{1}{l-l'} \right)^2 \left(\max \left(2 \ln \left(\sum_{j=1}^{\infty} j^{-p \ln j} / \sqrt{2\pi\alpha} \right), 1 \right) + 2p \ln^2 k \right) \quad (5.16)$$

(following Bayraksan and Morton [2009] who use $p \approx 2 \times 10^{-1}$, $\varepsilon \approx 2 \times 10^{-7}$, $\varepsilon' \approx 10^{-7}$, $l \approx 0.045$, $l' \approx 0.015$). There are also two sequences of sample numbers: v_k for checking optimality and μ_k (e.g., $\mu_k = 2v_k$) for choosing the next candidate. The candidate solution is x^{μ_k} that solves (5.10) with $v = \mu_k$ samples.

Sequential Sampling Method (SSM)

Step 0. Initialize with $k = 1$, v_1 from (5.16).

Step 1. Generate μ_k samples to obtain $\hat{x}^k = x^{\mu_k}$. (These can start with the previous μ_{k-1} samples to use the previous solution as a starting point, but they are independent from the v_k samples for gap estimates.)

Step 2. Generate v_k samples (IID) to form $G_k = G_{v_k}(\hat{x}^k)$ and $s_k^2 = s_{v_k}^2(\hat{x}^k)$.

Step 3. If $s_k > b$ or $G_k > l'b + \varepsilon'$, then: set $k = k + 1$, find new v_k , re-sample if k is a multiple of k_f , and return to Step 1. Else, $x^v = \hat{x}^k$ (with, for $\mu_k = 2v_k$, $3 \sum_{i=1}^k v_i$ total samples, including re-samples).

The convergence result for this method is contained in the following theorem, where P refers to the probability measure over the sampling distribution.

Theorem 8. For a two-stage stochastic linear program (3.1.1) with relatively complete recourse, almost surely finite second-stage value, and compact non-empty feasible region X ,

1. for $\varepsilon > \varepsilon' > 0$, $p > 0$, and $0 < \alpha < 1$ fixed values, if the method stops at iteration v , then

$$\liminf_{l \downarrow l'} P(\Delta_v(x^v) \leq ls_v(x^v) + \varepsilon) \geq 1 - \alpha; \text{ and,} \quad (5.17)$$

2. for fixed $\varepsilon' > 0$ and $l > l' > 0$ where the sequential sampling method stops at iteration v , $P(v < \infty) = 1$.

Proof. The proof follows Bayraksan and Morton [2009], Theorem 3 and Proposition 4, with additional observations about characteristics of the estimates for sample-average approximations of two-stage stochastic linear programs (see, e.g, Römisch [2003]). \square

As a final note, we should mention that analogous procedures can be built around *quasi-random* sequences that seek to fill a region of integration with approximately uniformly spaced points. The result is that errors are asymptotically about of the order $\log(v)/v$ instead of $1/\sqrt{v}$ (see Niederreiter [1978]). The difficulty is in the estimation of the constant term but quasi-Monte Carlo appears to work quite well in practice (see Fox [1986] and Birge [1994]). In terms of expected performance over broad function classes, quasi-Monte Carlo performs with the same order of complexity (Woźniakowski [1991]). For the methods used in this chapter, we may substitute quasi-random sequences for pseudo-random sequences for practical implementations. Other generalizations known as *sparse grid* (from Smolyak [1960]) can also be used to obtain efficient characterizations of the integrals in stochastic programs (see Chen and Mehrotra [2007]).

The sample average approximation method has also been used for stochastic integer programs. An SAA problem for an SIP is similar to the program (1.2), with integer requirements in the first- and/or the second-stage programs. An SAA with a moderate sample size can be solved using classical deterministic MIP techniques. The process can be repeated with different samples to obtain candidate solutions along with statistical estimates of their optimality gaps. These various candidate solutions cannot be combined (or averaged) as they would produce non-integer solutions. Instead, a new and independent large sample is created to form an estimated objective function. The various first-stage candidate solutions are evaluated using this estimated objective solution. These evaluations still require several second-stage optimizations. The computational burden remains low as the first-stage is given. At the end, the best candidate first-stage solution is selected. A detailed computational study of the application of the SAA method to solve three classes of stochastic routing problems can be found in Verweij et al. [2003].

Exercises

1. For Example 2, find the asymptotic result from Theorem 5 for $\sqrt{v}(\mathbf{x}^v - \mathbf{x}^*)$ for $a < 0$.
2. For Example 2, derive the actual distribution of $\sqrt{v}(\mathbf{x}^v - \mathbf{x}^*)$ for a feasible region $x \geq a$ in each case of a , $a < 0$, $a = 0$ and $a > 0$. Find the limits of these distributions and verify the result from Theorem 5.
3. Consider a news vendor problem as in Section 1.1. Suppose this problem is solved using a sampling approach. The sampled problem with continuous cumulative distribution function F^v has a solution at $(F^v)^{-1}\left(\frac{s-a}{s-r}\right) = x^v$. Find the distribution of this quantile and show how to construct a confidence interval around x^* .
4. Consider Example 1. First, verify the assumptions in Theorem 7. Solve 100 samples each for $v = 10 + 10i$ for $i = 0, 1, \dots, 10$. Use these observations to estimate values for α and β in Theorem 7 for $\varepsilon = 0.03$.
5. Implement the sequential sampling method for the continuous distribution version of the two-stage stochastic linear program for the farming example in Section 1.1. Start with the parameter recommendations above. Vary them to observe the impact of the parameters on the convergence behavior.

Chapter 10

Multistage Approximations

Most decision problems involve effects that carry over from one time to another. Sometimes, as in the power expansion problem of Section 1.3, random effects can be confined to a single period so that recourse is block separable. In other cases, however, this separation is not possible. For example, power may be stored by pumping water into the reservoir of a hydroelectric station when demand is low. In this way, decisions in one period are influenced by decisions in previous periods.

Problems with this type of linkage among periods are the subject of this chapter. We again wish to derive approximations that can be used to bound the error involved in any decision based on the approximate problem solution. In Chapter 9, we saw that the number of random variables can lead to rapidly growing problems. In this chapter, we have the additional effect that the number of periods leads to exponential increases in problem size even if the number of realizations in each period remains constant (see Figure 3.4).

We can again construct bounds based on the properties of the multistage recourse functions. These analogues of the basic Jensen and Edmundson-Madansky bounds are given in Section 10.1. They correspond to fixing values at means or extreme values of the support of the random vectors in each period.

Keeping the number of periods fixed may not lead to sufficient reductions in problem size, especially if no time is clearly the end of the problem. This case would mean facing either an uncertain or an infinite horizon decision problem. These problems can also be approximated by aggregating several periods together. Section 10.2 describes this procedure to obtain both upper and lower bounds.

Sampling methods that apply generally to multistage methods are described in Section 10.3. Section 10.4 then describes methods based on decomposition approaches.

The bounds of Sections 10.1 and 10.2 and the sampling methods used in Sections 10.3 and 10.4 can be viewed as discretization procedures. We can also construct separable bounds that do not require discretization as in Chapter 8. These bounds correspond to separable responses to any changes in the problem and are part of a general approach to value-function approximation known as *approximate dynamic programming*. They are described in Section 10.5. In multistage problems,

specific problem forms and structures can also lead to substantial savings. These structures are particularly valuable for approximations of the value function. We also describe such special cases for network revenue management, production, and vehicle allocation in this concluding section.

10.1 Extensions of the Jensen and Edmundson-Madansky Inequalities

The basic Jensen and Edmundson-Madansky inequalities can be extended to multiple periods directly. The principle is to use Jensen's inequality (or a feasible dual solution) to derive the lower bound and construct a feasible primal solution using extreme points to construct the upper bound. To present these results, we consider the linear case first, although extensions to nonlinear, convex problems are directly possible. We use concepts from measure theory in the following discussion. Readers without this background may skip to the declarations to find the major results for actual implementations.

The multistage stochastic linear program is to find $\mathbf{x} = (x^1, \mathbf{x}^2, \dots, \mathbf{x}^H)$ (where we suppress transposes as earlier when they can be implied from the context) in the following:

$$\begin{aligned} & \min c^1 x^1 + E_{\Omega}[\mathbf{c}^2 \mathbf{x}^2 + \dots + \mathbf{c}^H \mathbf{x}^H] \\ \text{s. t. } & W^1 x^1 = h^1, \\ & \mathbf{T}^{t-1} \mathbf{x}^{t-1} + W^t \mathbf{x}^t = \mathbf{h}^t, \quad t = 2, \dots, H, \text{ a.s.}, \\ & \mathbf{x}^t - E_{\Omega^t}[\mathbf{x}^t] = 0, \quad t = 2, \dots, H, \text{ a.s.}, \\ & \mathbf{x}^t \geq 0, \quad t = 1, \dots, H, \text{ a.s.}, \end{aligned} \tag{1.1}$$

where we have used explicit nonanticipativity constraints as in (3.5.11). We have also assumed that the recourse within each period W^t is known and not random.

The basic Jensen bound again follows by assuming a partition of Ω , the support vector of all random components. Here, we write Ω as $\Omega = \Omega_1 \times \dots \times \Omega_H$. We suppose that $\Omega^t = \{\omega^t = (\omega_1, \dots, \omega_t) \mid \omega_i \in \Omega_i, i = 1, \dots, t\}$. In this way, we can characterize all events up to time t by measurability with respect to the Borel field defined by Ω^t , Σ^t . We assume that Ω^t is partitioned as $\Omega^t = S_1^t \cup \dots \cup S_{v^t}^t$ and that $S_i^t = \bigcup_{j \in \mathcal{D}^{t+1}(i)} \{\omega^t \mid (\omega^t, \omega_{t+1}) \in S_j^{t+1}\}$ so that the partitions are consistent from one period to another. We construct measurable decisions at time t if they are constant over each $S_{t,j}$.

Next, assume that $p_i^t = P[S_i^t]$, $\mathbf{c}^t = c^t$, and that $E_{S_i^t}[(\mathbf{h}^t, \mathbf{T}^t)] = (\bar{h}_i^t, \bar{T}_i^t)$ for all t and i . The problem then is to find:

$$\min c^1 x^1 + \sum_{i=2}^{v^2} p_i^2 c^2 x_i^2 + \dots + \sum_{i=1}^{v^H} p_i^H c^H x_i^H$$

$$\begin{aligned} \text{s. t.} \quad & W^1 x^1 = h^1, \\ & \bar{T}_i^{t-1} x_i^{t-1} + W^t x_j^t = \bar{h}_j^t, \quad t = 2, \dots, H, \quad i = 1, \dots, v^{t-1}, \\ & \quad j \in \mathcal{D}^{t+1}(i), \\ & x_i^t \geq 0, \quad i = 1, \dots, v^t, \quad t = 1, \dots, H. \end{aligned} \tag{1.2}$$

The first result is that (1.2) provides a lower bound on the optimal solution in (1.1) provided the expectations of $(\bar{h}_i^t, \bar{T}_i^t)$ are independent of the past. If not, then the conditional expectation form in (1.2) may not actually achieve a bound.

Theorem 1. *Given that $E_{S_i^t}[(\mathbf{h}^t, \mathbf{T}^t)] = (\bar{h}_i^t, \bar{T}_i^t) = E_{S_j^t}[(\mathbf{h}^t, \mathbf{T}^t)]$ for all S_i^t and S_j^t that have a common outcome at time t , i.e., such that $(\omega^{t-1}, \omega_t) \in S_j^t$ if and only if there exist some $(\hat{\omega}^{t-1}, \omega_t) \in S_j^t$. The optimal value of (1.2) with the definitions given earlier provides a lower bound on the optimal value of (1.1).*

Proof: Suppose an optimal solution x^* to (1.2) with optimal dual variables π_i^{t*} corresponding to constraints in (1.2) with right-hand sides, \bar{h}_i^t . By dual feasibility in (1.2),

$$p_i^t c^t \geq \pi_i^{t*} W^t + \sum_{j \in \mathcal{D}^{t+1}(i)} \pi_j^{t+1*} \bar{T}_j^{t+1}, \tag{1.3}$$

for every (t, i) . Let $\pi^t(\omega) = \sum_{i=1}^{v^t} 1_{\{\omega^t \in S_i^t\}} [\pi_i^{t*} / p_i^t]$. We also have

$$\rho^t(\omega) = - \sum_{i=1}^{v^t} 1_{\{\omega^t \in S_i^t\}} [\pi_i^{t*} T_i^t(\omega) / p_i^t] + \sum_{i' | i' \in \mathcal{D}^{t-1}(\mathcal{A}^{t-1}(i))} [\pi_{i'}^{t*} \bar{T}_i^t / p_{i-1, \mathcal{A}^{t-1}(i)}].$$

Note how the ρ variables represent nonanticipativity. The condition for dual feasibility from the multistage version of Theorem 3.13 (see Exercise 1) is that

$$c^t(\omega) - \pi^t(\omega) W^t - \pi^{t+1}(\omega) T^{t+1}(\omega) - \rho^{t+1}(\omega) \geq 0, \text{ a.s.}, \tag{1.4}$$

and

$$E_{\Sigma^t}[\rho^{t+1}(\omega)] = 0. \tag{1.5}$$

Substituting in the right-hand side of (1.4) yields:

$$\begin{aligned} c^t - (\pi_i^{t*} / p_i^t) W^t - [\pi_j^{t+1*} / p_j^{t+1}] T_j^{t+1}(\omega) \\ + \left[[\pi_j^{t+1*} / p_j^{t+1}] T_j^{t+1}(\omega) - \sum_{j | j \in \mathcal{D}^t(i)} [\pi_j^{t+1*} \bar{T}_j^{t+1} / p_i^t] \right] \end{aligned} \tag{1.6}$$

for each S_i^t and $j \in \mathcal{D}^t(i)$, which is non-negative from (1.3). Also, by their definition and the assumption that integration of $T_j^{t+1}(\omega)$ over varying S_i^t does not change its conditional outcome,

$$E_{\Sigma_t}[\rho_{t+1}(\omega)] = \sum_{j \in \mathcal{D}^t(i)} \left[(\pi_j^{t+1*}/p_j^{t+1}) \bar{T}_j^{t+1} p_j^{t+1} - \sum_{j|j \in \mathcal{D}^t(i)} p_i^t (\pi_j^{t+1*} \bar{T}_j^{t+1} / p_i^t) \right] = 0 \quad (1.7)$$

yielding (1.5). Hence, we have constructed a dual feasible solution whose objective value is a lower bound on the objective value of (1.1) by the multistage version of Theorem 3.13. Because this value is the same as the optimal value in (1.2), we obtain the result. \square

Thus, lower bounds can be constructed in the same way for multistage problems as for two-stage problems, provided the data have serial independence. Such independence is not necessary if only right-hand sides vary because the dual feasibility is not affected in that case. The key procedure is in developing a dual feasible solution (lower bounding support function). Upper bounds can follow as before by constructing primal feasible solutions. These bounds can also be used in conjunction with the lower bounds to obtain bounds when objective coefficients (c^t) are also random.

To develop the upper bounds, the basic result is an extension of Theorem 8.2. We assume the following general form in which the decision variables x are explicit functions of the random outcome parameters, ξ :

$$\inf_{\mathbf{x} \in \mathcal{N}} E_{\Xi} \left[\sum_{t=0}^T f^t(x^t(\xi), x_{t+1}(\xi), \xi_{t+1}) \right], \quad (1.8)$$

where we use the convention for the general nonlinear objective formulation that subscript t corresponds to decisions or parameters within period t while superscript t refers to all periods from 1 to t . The random vector $\xi = (\xi_1, \dots, \xi_H)$ has an associated probability space, (Ξ, Σ, P) , \mathcal{N} is the space of nonanticipative decisions, f^t is convex, and ξ_{t+1} is measurable with respect to Σ_{t+1} and $\xi_{t+1} \in \Xi_{t+1}$, which is compact, convex, and has extreme points, $\text{ext } \Xi_{t+1}$, with Borel field, \mathcal{E}_{t+1} . In this representation, \mathbf{x} *nonanticipative* means that $x^t(\xi(\omega))$ is Σ_t -measurable for all t . It could also be described in terms of measurability with respect to Σ^t , the Borel field defined by the history process $\xi^t = (\xi_1, \dots, \xi_t)$.

Suppose that $e = (e_1, \dots, e_H)^T$ where each $e_t \in \text{ext } \Xi_t$. The set of all such extreme points is written $\text{ext } \Xi$. Suppose $x' = (x'_1, \dots, x'_H)$, where $x'_t : \text{ext } \Xi_t \rightarrow \mathbb{R}^{n_t}$. We say that x' is *extreme point nonanticipative*, or $x' \in \mathcal{N}'$, if x'_t is measurable with respect to the Borel field, \mathcal{E}_t , on $\text{ext } \Xi$, defined by (e_1, \dots, e_t) , where $e_j \in \text{ext } \Xi_j$ (for $t = 1$, this will be with respect to $\{\emptyset, \text{ext } \Xi\}$). With these definitions, we obtain the following result.

Theorem 2. Suppose that $\xi \mapsto f^t(x^t, x_{t+1}, \xi_{t+1})$ is convex for $t = 0, \dots, H$, Ξ_t is compact, convex, and has extreme points, $\text{ext } \Xi_t$. For all $\xi_t \in \Xi_t$, let $\phi(\xi, \cdot)$ be a probability measure on $(\text{ext } \Xi, \mathcal{E})$ where \mathcal{E} is the Borel field of $\text{ext } \Xi$, such that

$$\int_{e \in \text{ext } \Xi} e \phi(\xi, de) = \xi, \quad (1.9)$$

and $\xi \mapsto \phi(\xi, A)$ is measurable with respect to Σ_t for all $A \in \mathcal{E}^t$. Then there exists, $\mathbf{x} \in \mathcal{N}$, such that $x_t(\xi) = \int_{e \in \text{ext } \Xi} x'_t(e) \phi(\xi, de)$,

$$\mathbb{E} \left[\sum_{t=0}^T f^t(\mathbf{x}^t, \mathbf{x}_{t+1}, \xi_{t+1}) \right] \leq \int_{e \in \text{ext } \Xi} \sum_{t=0}^T f^t((x')^t, x'_{t+1}, e_{t+1}) \lambda(de), \quad (1.10)$$

where x' is extreme point nonanticipative and λ is the probability measure on \mathcal{E} defined by

$$\lambda(A) = \int_{\Xi} v(\xi, A) P(d\xi). \quad (1.11)$$

Proof: We must first show that \mathbf{x} as defined in the theorem is nonanticipative, or that $x_t(\xi)$ is Σ_t -measurable. This follows because $x'_t(e)$ is \mathcal{E}_t -measurable, and, for any $A \in \mathcal{E}_t$, $\phi(\xi, A)$ is Σ_t -measurable. Because each f^t is convex, for any ξ ,

$$\begin{aligned} & f^t(x^t(\xi), x_{t+1}(\xi), \xi_{t+1}) \\ &= f^t \left(\int_{e \in \text{ext } \Xi} (x')^t(e) \phi(\xi, de), \int_{e \in \text{ext } \Xi} x'_{t+1}(e) \phi(\xi, de), \int_{e \in \text{ext } \Xi} e_{t+1} \phi(\xi, de) \right) \\ &\leq \int_{e \in \text{ext } \Xi} f^t((x')^t(e), x'_{t+1}(e), e_{t+1}) \phi(\xi, de). \end{aligned} \quad (1.12)$$

Integrating with respect to P , the result in (1.10) is obtained. \square

As in Chapter 8, we implement the result in Theorem 2 by finding an appropriate ϕ and then solving the following approximation problem.

$$\inf_{x \in \mathcal{N}'} \int_{\text{ext } \Xi} \left[\sum_{t=0}^H f^t(x^t(e), x_{t+1}(e), e_{t+1}) \right] \lambda(de) \quad (1.13)$$

to find an upper bound on the value in (1.8). One can also refine these bounds by taking partitions of Ξ .

The simplest type of bound from Theorem 2 is the extension of the Edmundson-Madansky bound on rectangular regions with independent components. For this bound, we assume that all components, $\xi_t(i)$, are stochastically independent and distributed on $[a_t(i), b_t(i)]$. In this case, we can define

$$v^{EM-I}(\xi, e) = \prod_{t=1}^H \prod_{i=1}^{m_t} \frac{|\xi_t(i) - e_t(i)|}{(b_t(i) - a_t(i))}, \quad (1.14)$$

so that

$$\lambda^{EM-I}(e) = \prod_{t=1}^H \prod_{i=1}^{m_t} \frac{|\bar{\xi}_t(i) - e_t(i)|}{(b_t(i) - a_t(i))}. \quad (1.15)$$

It is easy to check that this v meets the nonanticipative measurability requirements. Problem (1.13) now can be written as:

$$\inf_x \left[\sum_{t=0}^H \left[\sum_{i_1=1}^{I_1} \cdots \sum_{i_{t+1}=1}^{I_{t+1}} \left[\sum_{i_{t+2}=1}^{I_{t+2}} + \cdots + \sum_{i_{H+1}=1}^{I_{H+1}} \lambda(e_{i_1}, \dots, e_{i_{H+1}}) \right] f^t(x^t(i_1, \dots, i_t), x_{t+1}(i_1, \dots, i_{t+1}), e_{i_{t+1}}) \right] \cdots \right], \quad (1.16)$$

where $x^t(i_1, \dots, i_t)$ corresponds to the t th-period decision depending on the outcomes in extreme point combination e_{i_s} from each period $s = 1, \dots, H$. This places the nonanticipativity back into the problem implicitly.

Example 1

To see how this bound might be implemented, consider Example 1 in Section 6.1. Suppose that demand is uniformly and independently distributed on $[1, 3]$ in each period. In this case, we obtain a decision vector (x_s^t, w_s^t, y_s^t) in period t for scenario $s = 2^{i_1} + i_2$ for i_1 and i_2 in $\{1, 2\}$. Problem (7.1.7) is, therefore, the upper bounding problem (1.16) for this uniform distribution case, yielding an upper bound of 6.25. In this case, the lower bound using the expected demand value of two in each period is three. In Exercise 2, you are asked to refine these bounds until they are within 25% of each other.

Other extreme point combinations are clearly also possible in multiperiod problems as they are in single-period problems. Extensions to dependent random variables and f^t concave in some arguments can also be made.

The bounds given in this section so far apply only to fixed numbers of periods. When periods are combined, we call the resulting problem an *aggregated problem*. These problems are described in the next section.

Exercises

1. Consider the multistage stochastic linear program in the form of (1.1). Prove the multistage version of Theorem 3.13.
2. Refine the extreme point (Edmundson-Madansky) and conditional expectation (Jensen) bounds on partitions for Example 1 from Section 6.1 until the upper bound is within 25% of the lower bound.

10.2 Bounds Based on Aggregation

The main motivation for aggregation bounds is to deal with problems with many (perhaps an infinite number of) periods by combining periods to obtain a simpler

approximate problem with fewer periods. The basic procedures in this chapter appear in Birge [1985a] and Birge [1984]. They follow the general aggregation results in Zipkin [1980a, 1980b]. Similar methods, especially for dealing with infinite horizon problems, appear in Grinold ([1976, 1983, 1986]). Generalizations appear in Wright [1994] and Kuhn [2008].

To derive both upper and lower bounds in this framework, we consider a special form for the multistage problem in (3.4.1). We allow feasibility by adding a penalty variable y^t that can achieve feasibility in each period. This notion of model robustness is quite common, although the penalty parameter q may be quite high. The form of the multistage stochastic linear program in this case is:

$$\begin{aligned} \min z = & c^T x^1 + E_{\xi} \left[\sum_{t=2}^H \rho^{t-1} (c^T x^t(\xi^2, \dots, \xi^t) + q^T y^t(\xi^2, \dots, \xi^t)) \right] \\ \text{s. t. } & Wx^1 \geq h^1, \\ & Tx^{t-1}(\xi^2, \dots, \xi^{t-1}) + Wx^t(\xi^2, \dots, \xi^t) + y^t(\xi^2, \dots, \xi^t) \geq \xi^t, \quad t = 2, \dots, H, \\ & x^1 \geq 0; \quad x^t(\omega) \geq 0, \text{ a.s., } \quad t = 2, \dots, H, \\ & y^t(\omega) \geq 0, \text{ a.s., } \quad t = 2, \dots, H, \end{aligned} \tag{2.1}$$

where superscript t again represents the variables or parameters for period t (i.e., not the full history), c is a known vector in \Re^{n_1} , h^1 is a known vector in \Re^{m_1} , $\xi^t(\omega) = h^t(\omega)$ is a random m -vector defined on (Ω, Σ^t, P) (where $\Sigma^t \subset \Sigma^{t+1}$) for all $t = 2, \dots, H$, and T and W are known $m \times n$ matrices. We also suppose that Ξ^t is the support of ξ^t . The parameter ρ is a discount factor.

Note that in (2.1), we assume that the parameters T , W , c , and q are all constant across time (with objective coefficients varying only with the discount factor). This assumption is basically made to simplify the following presentation. Varying parameters are possible with little additional work.

The key observation for these bounds is that an optimal solution in (2.1) is no lower than

$$\pi^1 h^1 + E_{\xi} \left[\sum_{t=2}^H (\pi^t(\xi^2, \dots, \xi^t))^T \xi^t \right] \tag{2.2}$$

for any $(\pi^1, \dots, \pi^t(\xi^2, \dots, \xi^t), \dots, \pi^T(\xi^2, \dots, \xi^T)) \geq 0$ a.s. that satisfies

$$\begin{aligned} & (\pi^1)^T W + E_{\xi} [\pi^2(\xi^2)]^T T \leq c^T, \\ & \pi^t(\xi^2, \dots, \xi^t)^T W + E_{\xi|(\xi^2, \dots, \xi^t)} [\pi^{t+1}(\xi^2, \dots, \xi^{t+1})]^T T \leq \rho^{t-1} c^T, \\ & \quad t = 2, \dots, H-1, \\ & \pi(\xi^2, \dots, \xi^H)^T W \leq \rho^{H-1} c^T, \\ & \pi(\xi^2, \dots, \xi^H)^T W \leq \rho^{H-1} q^T. \end{aligned} \tag{2.3}$$

You are asked to show that (2.2) subject to (2.3) provides a bound in Exercise 1.

The basic idea behind the aggregation bounds is that we can either construct either solutions (\mathbf{x}, \mathbf{y}) that are feasible in (2.1) or solutions $\boldsymbol{\pi}$ that are feasible in (2.3). As before, the former provide upper bounds, while the latter provide lower bounds.

The other assumption we make is that some set of finite upper bounds exists in \mathbf{x}^t so that for any \mathbf{x}^* optimal in (2.1):

$$\mathbf{x}^{t*}(\xi^2, \dots, \xi^t) \leq u^t(\xi^2, \dots, \xi^t). \quad (2.4)$$

In most problems, some form of bound satisfying (2.4) can be found. The tightness of this bound may, however, significantly affect the bounding results.

The basic bound is first to assume that the Jensen type of conditional expectation bound has been applied in each period. We illustrate this with a single partition, although finer partitions are possible. We also collapse everything into a two-period problem. Less aggregated models are constructed in the same way. Note in the following that H is quite arbitrary and, assuming finite sums, could even be infinite.

The problem is formed by defining aggregate variables, \hat{X}^1 , \hat{X}^2 , and \hat{Y}^2 , and parameters,

$$\begin{aligned} \hat{W} &= \left(\sum_{t=2}^H \rho^{t-2} \right) W + \left(\sum_{t=2}^H \rho^{t-2} \right) T, & \hat{I} &= \left(\sum_{t=2}^H \rho^{t-2} I \right), \\ \hat{c} &= \left(\sum_{t=2}^H \rho^{t-1} \right) c, & \hat{q} &= \left(\sum_{t=2}^H \rho^{t-1} \right) q, & \hat{\xi} &= \left(\sum_{t=2}^H \rho^{t-2} \xi^t \right). \end{aligned}$$

The resulting aggregate approximation problem is:

$$\begin{aligned} \min & c^T \hat{X}^1 + \hat{c}^T \hat{X}^2 + \hat{q}^T \hat{Y}^2 \\ \text{s. t.} & W \hat{X}^1 \geq h^1, \\ & T \hat{X}^1 + \hat{W} \hat{X}^2 + \hat{T} \hat{Y}^2 \geq \hat{\xi}, \\ & \hat{X}^1, \hat{X}^2, \hat{Y}^2 \geq 0. \end{aligned} \quad (2.5)$$

Suppose (2.5) has an optimal solution $(X^{1,*}, X^{2,*}, Y^{2,*})$ with multipliers, Π^* . These solutions are not directly feasible in (2.1) or (2.3), but feasible solutions can be easily constructed from them. To do so, we need only let $\hat{x}^1 = X^{1,*}$, $\hat{x}^t(\xi^2, \dots, \xi^t) = X^{2,*}$, and $\hat{y}^t(\xi^2, \dots, \xi^t) = Y^{2,*}$ for all t and ξ . We also let $\hat{\pi}_1 = \Pi_1^*$ and $\hat{\pi}^t(\xi^2, \dots, \xi^t) = \rho^{t-2} \Pi_2^*$ for all t and ξ . In this way, the value of (2.5) is the same as

$$\hat{z} = c^T \hat{x}^1 + E_\Xi \left[\sum_{t=2}^H \rho^{t-1} (c^T \hat{x}^t(\xi^2, \dots, \xi^t) + q^T \hat{y}^t(\xi^2, \dots, \xi^t)) \right], \quad (2.6)$$

which forms the basis for our bounds. The result is contained in the following theorem.

Theorem 3. Let z^* be a finite optimal value for (2.1). Then

$$\hat{z} + \varepsilon^+ \geq z^* \geq \hat{z} - \varepsilon^- , \quad (2.7)$$

where

$$\begin{aligned} \varepsilon^- = & - \sum_{t=2}^H \sum_{j=1}^n \left[\int_{\Xi} [\min \{ \rho^{t-1} c_j - \rho^{t-2} \Pi_2^* W_{.j} \right. \\ & \left. - \rho^{t-1} \Pi_2^* T_{.j}, 0 \} u^t(j)(\xi)] P(d\xi) \right] \end{aligned}$$

and

$$\begin{aligned} \varepsilon^+ = & \sum_{t=2}^H \sum_{j=1}^n \left[\int_{\Xi} [\max \{ -W_{.j} X^{2,*} - T_{.j} X^{2,*} \right. \\ & \left. - Y^{2,*}(j) + \xi^t, 0 \} \rho^{t-1} q(j)] P(d\xi) \right] . \end{aligned}$$

The proof of this theorem is Exercise 2. The basic idea is to write out z^* in terms of $(\mathbf{x}^*, \mathbf{y}^*)$ and to add on $\hat{\pi}^t(\xi)^T (\xi^t - Wx^{t*}(\xi) - y^{t*} - Tx^{t-1*}(\xi))$ terms, which are all nonpositive. This yields ε^- . The upper bound comes from showing that $\{\hat{x}^t(\xi), \hat{y}^t(\xi) + \max\{-W_{.j} X^{2,*} - T_{.j} X^{2,*} - Y^{2,*}(j) + \xi^t, 0\}\}$ is always feasible in (2.1).

These bounds can be quite useful, but the penalty and variable bound assumptions may not be apparent in many problems. Sometimes bounds on groups of variables are possible and can be useful. In other cases, properties of the constraint matrices can be exploited to obtain other bounds similar to those in Theorem 3. Several of these ideas are presented in Birge [1985a].

Example 2

In production/inventory problems, these values are especially easy to find, as in Birge [1984]. Consider a basic problem of the form

$$\begin{aligned} \min z = & E_{\xi} \left[\sum_{t=1}^H \rho^{t-1} (-c^t x^t(\xi) + q^t y_+^t(\xi) + r^t s^t(\xi)) \right] \\ \text{s. t.} \quad & \mathbf{x}^t - \mathbf{s}^t \leq k^t, \text{ a.s.,} \quad \mathbf{w}^{t-1} + \mathbf{x}^t - \mathbf{w}^t = 0, \text{ a.s.,} \quad (2.8) \\ & \mathbf{w}^t \geq b^t, \text{ a.s.,} \quad \mathbf{y}_+^{t-1} + \mathbf{x}^t - \mathbf{y}_+^t + \mathbf{y}_-^t = \xi^t, \text{ a.s.,} \\ & \mathbf{y}_+^{t-1}, \mathbf{y}_-^t, \mathbf{x}^t, \mathbf{s}^t, \mathbf{w}^t \geq 0, \text{ a.s.,} \quad t = 1, \dots, H ; \\ & \mathbf{y}_+^t, \mathbf{y}_-^t, \mathbf{x}^t, \mathbf{s}^t, \mathbf{w}^t, \text{ all } \Sigma^t \text{ measurable} \quad t = 1, \dots, H , \end{aligned}$$

where x^t represents total production, s^t represents overtime production, w^t is cumulative production, y_+^t is inventory, y_-^t is lost sales (i.e., no backordering), b^t is a lower bound to achieve a service reliability criterion (see Bitran and Yanasse [1984]), c^t is the unit margin, q^t , and r^t are cost parameters, and ξ^t is the random demand.

For problems with the form in (2.8), it is possible to find bounds on all primal and dual variables for an optimal solution. These bounds can then be used with Theorem 3. Exercises 3, 4, and 5 explore the aggregation bounds in this context more fully.

Exercises

1. Verify that a non-negative π satisfying the conditions in (2.3) provides a bound on (2.1)'s optimal value through (2.2).
2. Prove Theorem 3.
3. Find bounds on all optimal variable values in (2.8) as functions of the parameters and previous realizations.
4. Using the bounds in (2.3), construct bounds based on Theorem 3 for a problem as in (2.8) with four periods, uniform demand on $[8000, 10,000]$, $b^t = t(9500)$, $c^t = 19$, $r^t = 4$, $k = 9000$, for $t = 1, 2, 3, 4$, and $q^t = 9.5$ for $t = 1, 2, 3$, $q^4 = 30$ (to account for unsold products at the end of the horizon), and $\rho = 0.9$.
5. It is not necessary to take expectations before aggregating periods. Using the example in (2.8), construct bounds with a two-period problem that uses a weighted sum of future demands in the first period. What type of stochastic program is this?

10.3 Scenario Generation and Distribution Fitting

Sampling methods are a common approach for multistage stochastic programs, just as they are for two-stage models. Due to the exponential increase in the number of possible scenarios as the horizon length increases, multistage scenario generation approaches place a greater emphasis on reducing the number of required samples. The result is that the sampling procedure often involves considerable effort to ensure that the samples provide similar solution characteristics to a true underlying model. Main concerns are that the sample distribution has similar moments to the underlying distribution, that the sample distribution is not too distant from the underlying in terms of the probability of any event, and that the solution of the model using the sample distribution is consistent with practical limitations, such as the absence of *arbitrage*. Under mild conditions, these criteria can ensure that the sampling model

solution converges asymptotically to a solution of the model with the underlying distribution.

In the following, we assume that an underlying distribution is known, although, as elsewhere in this book, this can be interpreted in the Bayesian sense that the underlying distribution represents the prior belief of the decision maker. For the development here, we assume the structure of the multistage stochastic linear program in (3.4.1), although extensions to nonlinear models are straightforward. The random parameters in period t are $\xi^t = \xi^t(\omega)$. A basic sampling method would be to take \mathcal{K}_1 independent and identically distributed draws, $\xi_1^1, \dots, \xi_{\mathcal{K}_1}^1$, from ξ^1 and then recursively to draw \mathcal{K}_t samples from ξ^t conditional on $\xi_{k_1}^1, \dots, \xi_{k_{t-1}}^{t-1}$ where $1 \leq k_s \leq \Pi_{i=1}^s \mathcal{K}_i$, $s = 1, \dots, t-1$ for each of the $\mathcal{K}^{t-1} = \Pi_{i=1}^{t-1} \mathcal{K}_i$ possible scenarios in the sampled decision tree through period $t-1$. When ξ^t is serially independent (i.e., the distribution is the same for all realizations of the history process at time $t-1$ for all t), the same ξ^t samples may be used along any branch of the tree, but, in stochastic programming, we assume that optimal decisions may be *path-dependent* and, therefore, that the exponential increase in the size of the tree is necessary to capture all possible future actions.

To keep the sizes of decision trees manageable for computation, stochastic programming models generally limit the size of the sample tree so that \mathcal{K}_t is relatively small (and may be decreasing in t). To help ensure that the solution of the sample problem suffers as little as possible from small-sample bias, sample scenario generation in multistage models often aims to ensure that the sample distribution shares important characteristics, such as moments and quantiles, with the underlying distribution of ξ .

To see how multistage sampling works in practice, we consider the investment model from Section 1.2, where instead of the two possible values in each period, we suppose that the returns ξ^t are lognormally distributed where $\log \xi^t \sim N(\mu, \Sigma)$, a bivariate normally distributed random vector with mean $\mu = \begin{pmatrix} 0.141 \\ 0.122 \end{pmatrix}$ and variance/covariance matrix $\Sigma = 10^{-3} \begin{pmatrix} 6.740 & 0.291 \\ 0.291 & 0.0784 \end{pmatrix}$. This distribution gives the same mean and variance for each component of ξ^t as in Section 1.2, but, instead of being perfectly correlated, the correlation between the stock and bond is 0.4. In particular, the mean return of each asset i is $\bar{\xi}_i = e^{\mu_i + \frac{1}{2}\sigma_{ii}}$, written as

$$\bar{\xi} = \begin{pmatrix} 1.155 \\ 1.130 \end{pmatrix}, \quad (3.1)$$

and the covariances are $E[(\xi(i) - \bar{\xi}(i))(\xi(j) - \bar{\xi}(j))] = e^{\mu_i + \mu_j + \frac{\sigma_{ij} + \sigma_{jj}}{2}} (e^{\sigma_{ij}} - 1)$, which we write collectively as the matrix V , where

$$V = 10^{-3} \begin{pmatrix} 9.027 & 0.380 \\ 0.380 & 0.100 \end{pmatrix}, \quad (3.2)$$

To create samples of ξ^t for the stochastic program, we first start by taking a random sample of \mathcal{K}_1 values, using, for example, independent standard normal draws $z^1, \dots, z^{\mathcal{K}_1}$ where each component z_j^k , $j = 1, 2$, $k = 1, \dots, \mathcal{K}_1$, is an independent standard normal draw as well. We then have an initial set of samples $\hat{\xi}^k = e^{\mu + \Sigma^{0.5} z^k}$, where the exponential operator is interpreted as operating separately on each component of $\mu + \Sigma^{0.5} z^k$ and where $\Sigma^{0.5}$ is the Cholesky factor of Σ (i.e., the upper triangular matrix such that $\Sigma = (\Sigma^{0.5})^T \Sigma^{0.5}$). Here is a possible sample with $\mathcal{K}_1 = 6$ ¹: The mean of this sample is $\bar{\xi} = (1.236, 1.131)^T$ and the covariance of

Table 1 Original sample values.

$\hat{\xi}(1)$	$\hat{\xi}(2)$
1.113	1.124
1.195	1.136
1.185	1.129
1.236	1.130
1.234	1.129
1.452	1.137

the sample is $\hat{V} = 10^{-3} \begin{pmatrix} 11.01 & 0.343 \\ 0.343 & 0.020 \end{pmatrix}$, which may differ enough from $\bar{\xi}$ and V to bias the stochastic program results. To correct for this problem, as long as \mathcal{K}_1 is sufficiently large that \hat{V} has full rank, we can update the sample as follows to produce a sample with mean $\bar{\xi}$ and covariance V :

$$\tilde{\xi} = \bar{\xi} + V^{0.5}(\hat{V}^{-0.5}(\hat{\xi} - \bar{\xi})), \quad (3.3)$$

which results in the values in Table 10.3, which now has mean $\bar{\xi}$ and covariance V .

Table 2 Adjusted sample values.

$\tilde{\xi}_1$	$\tilde{\xi}_2$
1.044	1.116
1.118	1.148
1.109	1.127
1.155	1.127
1.153	1.125
1.351	1.137

These samples can then be used again to generate $\mathcal{K}_2 = 6$ samples for period 2 (assuming serial independence). For a three period model, this results in $\mathcal{K}_1 \mathcal{K}_2 = 36$ total scenarios. Including a third set of realizations as in Section 1.2 would yield

¹ Much larger samples are often used in practice for \mathcal{K}_1 , but, since $\prod_{t=1}^H \mathcal{K}_t$ grows quickly for larger values of H , sample sizes for larger values of t are often small.

$6^3 = 216$ scenarios, but often fewer scenarios are used in later periods. (In Exercise 2, we use $\mathcal{K}_3 = 2$ for 72 total scenarios.)

This procedure of modifying a random sample to match moments of an assumed underlying distribution is called *adjusted random sample* generation. Results in Kouwenberg [2001] suggest that this procedure can improve outcomes relative to using random samples alone. Exercises 2 and 3 explore this issue for the financial planning example.

In addition to fitting the mean and second moments, improved scenario trees may result from fitting higher moments, such as through fits of skewness and kurtosis. Høyland and Wallace [2001] describe how to use an optimization procedure to fit these moments, which may include extreme values to represent tail risk and inter-period moments to represent serial dependence. In experiments in Kouwenberg [2001], the use of additional moment information provides minor improvement over adjusting only for first and second moments.

In longer horizon problems, an initial sampling procedure often still yields scenario trees that are too large for efficient direct computation. To simplify these trees further, scenarios may be collapsed while retaining as much moment information as possible (e.g., Cariño, et al. [1994]). Other alternatives in reducing scenario trees are to ensure that the reduced tree stays as close as possible in a distribution metric to the original (possibly sample-based) scenario tree (see Dupačová, Gröwe, and Römisch [2003]). Alternatively, a tree can be constructed directly that minimizes the distance in the distribution metric to the original underlying distribution (Pflug [2001]) or the tree can be adjusted (to be smaller or larger) in the process of solution by examining the expected value of perfect information at each node of the tree to collapse branches with small *EVPI* and to expand branches with large *EVPI* (Dempster [2006]).

An important consideration for generating scenarios in financial applications is, unless conditions are known not to be in equilibrium, for the scenario trees not to admit arbitrage in which trading among different assets could earn positive returns in all scenarios without any initial investment. Arbitrage most often occurs in models when derivative securities are included that depend on the same underlying security, but their prices are not consistent with the set of scenarios.

Example 3

As an example, we again consider the financial planning in Section 1.2 but with the original two branches in each period and where short-selling (negative positions) of the stock and bond are allowed. We now add an additional asset as a *call option* that gives the holder of the option the right (but not the obligation) to buy the stock at 1.15 times its original price at the end of the first period. In this way, the call option has the following contingent payoff, \mathbf{C}^1 , for each unit of stock value at time 0, such that:

$$\mathbf{C}^1 = \begin{cases} 0.10 & \text{if } \xi^1(1) = 1.25, \\ 0 & \text{if } \xi^1(1) = 1.10. \end{cases} \quad (3.4)$$

Suppose that the model includes a price for each unit of this call option of $C^0 = 0.02$ of the value of one unit of the stock. This would mean that the return value $\xi_1(3)$ corresponding to the call option asset follows:

$$\xi^1(3) = \begin{cases} 5 & \text{if } \xi^1(1) = 1.25, \\ 0 & \text{if } \xi^1(1) = 1.10. \end{cases} \quad (3.5)$$

Now, an initial investment strategy can include the following $(x^1(1), x^1(2), x^1(3)) = (-18\frac{2}{3}, 17\frac{2}{3}, 1)\alpha$, for any $\alpha \geq 0$ since this requires no additional wealth. The wealth at the end of the first period is then

$$\xi^1(3) = \begin{cases} 1.8067\alpha = (-(18\frac{2}{3})1.25 + (17\frac{2}{3})1.14 + 5)\alpha & \text{if } \xi^1(1) = 1.25, \\ 0 & \text{if } \xi^1(1) = 1.10, \end{cases} \quad (3.6)$$

which, as $\alpha \rightarrow \infty$, leads to infinite wealth in the state where stocks increase in value by 25%. The problem in this case is that $C^0 = 0.02$ is inconsistently low or $\xi^1(3) = 5$ in the high-stock-value case is too high. Note that if instead $\xi^1(3) = 3.1933$ in the high-stock-value scenario (corresponding to $C^0 = 0.031524$), then the wealth is zero under both scenarios. This is the *no-arbitrage condition* that the future value of a net initial investment of zero cannot be non-negative in all states and strictly positive in some states (with probability greater than zero).

Consistent equilibrium prices (in a market with zero transaction costs and allowable short sales) should satisfy the no-arbitrage condition; otherwise, investors would exploit the price differences to create unlimited riskless profits. To maintain this condition requires precise agreement of prices within a model. Transaction costs, which are present to some degree in practice (for example, in the bid-ask spread), allow for a range of consistent prices. Other restrictions, such as no-short-sale constraints, can eliminate unbounded solutions in the model, but inconsistent prices, even without pure arbitrage, can lead to solutions that are far from the optimal choice for a model with consistent prices. In the financial planning example considered here, for example, the optimal initial investment (Exercise 5) choices are given in Table 3. The solution with the consistent high-stock-value return of $\xi(3) = 3.1933$ results in a balanced initial portfolio, while the solution of the model using $\xi(3) = 5$ for the high-stock-value scenarios places almost the entire portfolio into the call option. Such wide swings can occur with small changes in the model data from values consistent with equilibrium prices (see Exercise 5). Ensuring consistent prices can then be a critical part of proper model generation.

The process we used to eliminate arbitrage can be simplified by using the equivalent martingale measure or risk-neutral measure, i.e., a probability distribution that weights scenarios based on their state prices to reflect a premium for non-diversifiable risk such that the value of all financial market assets equals the expected value under this distribution of all future payoffs discounted by the risk-free rate

(see Harrison and Kreps [1979]). Klaassen [1998] describes how this process applies for stochastic programming scenario trees, including important considerations for maintaining consistency while aggregating states and periods as in Section 10.2. Various methods can be used to represent the equivalent martingale measure by ensuring consistency in the expectation and fitting parameters to be consistent with market prices. Alternatively, in some cases, it is possible instead to modify the constraints and to use the natural probability measure (again ensuring consistency) (see Birge [2000]).

Theoretical results for obtaining convergence of solutions from a sample problem to that of the original problem are also possible for multistage problems as they are for two-stage problems, but including adjustments such as matching moments makes the analysis more difficult and the theoretical bounds on convergence are often worse than what is actually observed. The basic multistage results are direct extensions of the two-period results. As shown in Shapiro [2003], under suitable conditions (e.g., finite expectations, bounded sets of optimal solutions, and a pointwise Strong Law of Large Numbers holding for the sample values, $\mathcal{Q}^{t,\mathcal{K}_t}(x^t) \rightarrow \mathcal{Q}^t(x^t)$, a.e.), then, as $\mathcal{K}_t \rightarrow \infty, t = 1, \dots, H$,

- the sample average approximation value, $z^{\mathcal{K}^H} \rightarrow z^*$, the true optimal value;
- the distance between first-stage optimal solution sets decreases to zero with probability one;
- if the support of the true distribution is finite, then the first-stage optimal solution set is a nonempty face of the true optimal solution set with probability one.

For a special class of problems with non-negative objective values and non-negative constraint matrices (except possibly in the first and last stage), Swamy and Shmoys [2005] show that, for any tolerance $\varepsilon > 0$, the required number of samples in a multistage sample average approximation to achieve a high probability of a solution within a $1 + \varepsilon$ multiple of the optimal value is polynomial in $\frac{1}{\varepsilon}$ and a parameter that depends on cost growth across time.

Table 3 Initial values of x_1^* for different returns on a call option.

$\xi(3) =$	3.1933	5.0
Asset		
<i>Stock</i>	16.82	0.0
<i>Bond</i>	16.54	2.86
<i>Call</i>	21.64	52.14

Exercises

1. Show that the assumption that $\log \xi^t \sim N(\mu, \Sigma)$, $\mu = \begin{pmatrix} 0.141 \\ 0.122 \end{pmatrix}$ and $\Sigma = 10^{-3} \begin{pmatrix} 6.740 & 0.291 \\ 0.291 & 0.0784 \end{pmatrix}$ matches the mean and variance of the stock and bond returns for the financial planning example in Section 1.2 and that the correlation between the two assets is 0.4.
2. Solve the financial planning example with a 72-scenario event tree corresponding to two periods with returns given by $\hat{\xi}$ in Table 1 and one period with the original two return realizations given in Chapter 1. Let the first period solution be \hat{x}^1 . Solve also for the 72-scenario event tree given by $\tilde{\xi}$ in Table 2 and let the first period solution be \tilde{x}^1 . To test for their relative performance of these solution, perform a simulation with 1000 runs, where the initial allocations are \hat{x}^1 and \tilde{x}^1 respectively and the random returns are ξ_k^t for stage t drawn from the underlying lognormal distribution. For each run $k = 1, \dots, 1000$, for the second-period allocation, re-solve a two-stage model with input wealth $(\xi_k^1)^T \hat{x}^1$ and $(\xi_k^1)^T \tilde{x}^1$ respectively for the two alternatives and then obtain solutions on the remaining (36-node) sample trees as \hat{x}^{k2} and \tilde{x}^{k2} ; then, use the second-period return ξ_k^2 , and repeat for the third and final periods to obtain sample objective values \hat{z}_k and \tilde{z}_k . Compare the distributions of \hat{z} and \tilde{z} for these samples by plotting their percentiles. What does this suggest about the use of adjusted samples?
3. Repeat Exercise 2 by randomly drawing ten additional random samples $\hat{\xi}$ and adjusting to fit the mean and covariance in $\hat{\xi}$ (so that now the tree has $2 \cdot 16^2 = 512$ scenarios). (Warning: this requires fast subproblem optimization.)
4. Suppose that instead of a call option to buy the stock at 15% above its current value, the option is buy the stock at 10% above its current value. If this is included in the financial planning example with two branches per period, what should the initial call price or premium C_0 be for this option to avoid arbitrage possibilities?
5. Solve the 8-scenario, 3-period financial planning example with the addition of a call option. First, solve with the consistent high-stock-increase return on the call option of 3.1933 and then with a high-stock-increase return of 5. Verify the solutions x^{1*} that are given in Table 3. Re-solve with $\xi^1(3) = 3.20$ in the high-stock-return scenario. What is the value of initial investments x^{1*} now?

10.4 Multistage Sampling and Decomposition Methods

In this section, we consider algorithms that incorporate sampling into decomposition methods for multistage stochastic programs with explicit confidence intervals on the convergence of the sample problem value to an optimal solution value. For the

exposition here, we consider multistage stochastic linear programs with relatively complete recourse and a finite optimal objective value.

Assume that the stochastic elements are defined over a discrete probability space $(\Xi, \sigma(\Xi), P)$, where $\Xi = \Xi^2 \otimes \dots \otimes \Xi^H$ is the support of the random data in stages two through H , with $\Xi^t = \{\xi_i^t = (h^t(\xi_i^t), c^t(\xi_i^t), T_{:,1}^{t-1}(\xi_i^t), \dots, T_{:,n-1}^{t-1}(\xi_i^t), i = 1, \dots, M^t)\}$. Further, assume that the random parameters are serially independent. Thus, the probability of a particular stage t realization ξ_i^t is constant from all possible $(t-1)$ -stage scenarios.

For the following, we describe the strategy of *abridged nested decomposition* (AND) (Donohue and Birge [2006]), which is an extension of the sampling strategy of *stochastic dual dynamic programming* (SDDP) in Pereira and Pinto [1991]. Both algorithms use sampling to generate an upper bound on the expected value (over an H -stage planning horizon) of a given first stage solution and to use decomposition to generate a lower bound. The algorithm terminates when the two bounds are sufficiently close. As in the nested decomposition algorithm, each iteration of SDDP and AND algorithm begins by solving the first stage subproblem, after which, K H -stage scenarios are sampled. Let x_k^t and ξ_k^t denote the stage t solution vector and the stage t random parameter realization, respectively, in sampled scenario k . A forward pass through a sampled version of the scenario tree solves the nested decomposition subproblem (6.1.1–1.5) for stages $t = 2, \dots, H$ and scenarios $k = 1, \dots, K$.

The algorithm uses an upper bound estimate on z^* based on individual scenario objective values, z_k , where

$$z_k = c^1 x_k^1 + \sum_{t=2}^H c^t(\xi_k^t) x_k^t, \quad (4.1)$$

where x_k^1 is the same for all values of k . The z_k values are combined to form an estimate with K samples as:

$$\hat{z}_K = \frac{1}{K} \sum_{k=1}^K z_k, \quad (4.2)$$

with standard deviation of the estimate given by,

$$\sigma_{z_K} = \sqrt{\left(\frac{1}{K^2} \sum_{k=1}^K (\hat{z}_K - z_k)^2 \right)}. \quad (4.3)$$

Using these values, a confidence interval on the upper bound estimate can be constructed.

After the forward pass is completed, the method follows a backward pass as in the nested decomposition algorithm, but, without considering all branches of the

tree. The essential difference between AND and SDDP is that, in AND, instead of considering the full sample-path tree, a set of *branching solutions*, B^t , are used to generate new cuts in the backward pass. The branching solutions are quite flexible under the assumption of serial independence since the cuts generated for any values of x^t yield valid cuts. These solutions may correspond to solutions along the previous sample-paths, combinations of solutions, or some other set of possible state values. In the backward pass, all child scenarios of each branching solutions are solved to ensure that the solutions of each subproblem (6.1.1–1.5) obtain a valid lower bound on $\mathcal{Q}^{t+1}(x^t)$ for each x^t in B^t .

The backward pass progresses for periods $t = H - 1, \dots, 1$ generating a new optimality cut for each branching solution in B^t . Once a new optimality cut has been added to the first-stage subproblem, the backward pass completes, followed again by a new generation of a new set of sample paths and the forward pass to construct an upper bound estimate.

Finite convergence of this algorithm follows from the finite convergence of the nested decomposition algorithm, since the scenarios from which the optimality cuts are generated are re-sampled each iteration (see Donohue [1996] and the detailed proof in Philpott and Guan [2008]). Since the accuracy of the optimal solution depends on the accuracy of the estimated upper bound, the performance of the algorithm depends on the number of scenarios sampled in each iteration.

The Abridged Nested Decomposition Algorithm

Step 0. For $t = 1, \dots, H - 1$, set $s^t = 0$, and add the constraint $\theta^t = 0$ to the stage t subproblem. Choose initial values for $|F^t|$ (forward branching values) and $|B^t|$ for $t = 2, \dots, N - 1$. Go to Step 1.

Step 1. Solve the first stage problem. Let \tilde{x}^1 be the current optimal solution and $\tilde{\theta}^1$ be the current expected recourse approximation value. Let \tilde{z}^1 be the current optimal objective value. Let \tilde{x}^1 be the first stage branching value. Go to Step 2.

Step 2. Forward Pass.

For $t = 2, \dots, H - 1$,

For $j = 1, \dots, |B^{t-1}|$,

For $k = 1, \dots, |F^t|$,

Solve the stage t subproblem (6.1.1–1.5) with input value $x_j^{t-1} \in B^{t-1}$ and sample realization $\xi_k^t \in F^t$.

Select $|B^t|$ branching values x^t from subproblem solutions.

Go to Step 3.

Step 3. Backward Pass.

For $t = N, \dots, 2$,

For $j = 1, \dots, |B^{t-1}|$,

For $i = 1, \dots, M^t$,

Solve stage t subproblem (6.1.1–1.5) with input value $x_j^{t-1} \in B^{t-1}$ for scenario ξ_i^t . Let $(\pi_{i,m}^t, \sigma_{i,m}^t)$ denote the optimal dual vector values.

Compute

$$E^{t-1} = \sum_{i=1}^{M^t} p_k^t \pi_{i,m}^t T_i^{t-1}, \quad e^{t-1} = \sum_{i=1}^{M^t} p_k^t (\pi_{i,m}^t h_i^t + \sigma_{i,m}^t e_i^t)$$

The new cut is then: $E^{t-1}x^{t-1} + \theta^{t-1} \geq e^{t-1}$.

If the constraint $\theta^{t-1} = 0$ appears in the stage $t-1$ subproblem, then remove it. Increment s^{t-1} by one and add the new cut to the stage $t-1$ subproblem. If $t = 2$, then the updated first stage expected recourse function upper bound is: $\bar{\theta}^1 = e^1 - E^1\tilde{x}^1$. If $\bar{\theta}^1$ is within a relative tolerance of $\bar{\theta}^1$, then go to Step 4. Otherwise, go to Step 1.

Step 4. Sampling Step.

Let $x_k^1 = \tilde{x}^1$, for $k = 1, \dots, K$.

For $k = 1, \dots, K$,

Generate H -stage sample scenario, $(\xi_k^2, \dots, \xi_k^H)$.

For $t = 2, \dots, H$,

Given stage $t-1$ solution x_k^{t-1} and realization ξ_k^t , solve the stage t subproblem (6.1.1–1.5). Let x_k^t denote the optimal solution.

Using Equations (4.1), (4.2), and (4.3), obtain a confidence interval on the expected objective value of the current first stage solution. If $c^1\tilde{x}^1 + \tilde{\theta}^1$ is in the confidence interval, stop with \tilde{x}^1 as the optimal solution. Else, increase F^t and B^t for stage $t = 2, \dots, N$ and go to Step 1.

To ensure that the algorithm terminates with a valid confidence interval on z^* , a procedure such as the sequential sampling method in Section 8.5 should be used. For this algorithm to be effective, the branching values in B^t also must be chosen carefully. As shown in Donohue and Birge [2006], however, any convex combination of feasible values at time t has a feasible completion in period $t+1$. This observation allows for consolidation in the branching step. Various fixed rules can be used for selecting branches or branching solution values can be chosen randomly. This strategy gives an unbiased sample of stage t solution values, which may have advantages. We note that this general approach can also be extended to problems with infinite horizons (see Exercise 2).

Exercises

1. Generate 50 random samples from the distribution given in Section 10.3 for the three-period financial planning example from Section 1.2. Implement AND on this problem using the following strategies starting with $|B^t| = 3$ and $|F^t| = 6$, increasing each by one whenever required, and terminating whenever $\hat{z}_K \leq c^1\tilde{x}^1 + \tilde{\theta}^1 + 2\sigma_{z_K}(x^1)$.
 - (a) Choose B^t randomly from the set of period t solutions.

- (b) Choose B^t initially corresponding to solutions with the maximum, median, and minimum wealth in each period. If B^t increases, choose additional branching solutions randomly from the set of solutions.
2. For an infinite-horizon problem with stationary data (i.e., ξ^t has the same distribution ξ for all t), the goal is to find a function Ψ^∞ such that $\Psi^\infty = T(\Psi^\infty)$, where T is the dynamic programming operator defined by

$$T(\Psi^\infty(h_0 - T_0 x_0, c_0)) = \min_{x|Wx=h_0-T_0x_0} c_0^T x + \beta E[\Psi^\infty(\mathbf{h} - \mathbf{T}x, \mathbf{c})], \quad (4.4)$$

where $0 < \beta < 1$ is a fixed discount factor. Given a linear lower bound $\Psi^0(y, z) = e_0 + E_0 \begin{pmatrix} y \\ z \end{pmatrix} \leq \Psi^\infty(y, z)$, for any y and z , describe a sampling-based outer-linearization method to find Ψ^∞ . (Birge and Zhao [2007]).

10.5 Approximate Dynamic Programming and Special Cases

The approaches discussed in the previous sections have focused on sampling and state or tree aggregation to obtain tractable formulations. Another alternative is to use approximations of the value function \mathcal{Q}^t constructed in other ways. The outer linearization approach in the AND method is one possible value function approximation. In this section, we discuss other value-function approximations that collectively are often called *approximate dynamic programming* (ADP) or *neuro-dynamic programming* (see, e.g., Bertsekas [2007], Bertsekas and Tsitsiklis [1995], and Powell [2007]). As noted earlier, other approximations may include approximations of the actions (or policy) (as, for example, a parameterized function of the state variables), but the discussion here focuses on value-function approximations.

The general approach in ADP is to replace the value function $\mathcal{Q}^{t+1}(x^t)$, or the subproblem (scenario-conditional) value functions, $\mathcal{Q}^{t+1}(x^t, \xi^t)$, with an approximation that does not require full optimization of the sub-tree corresponding to ξ^t given x^t . In general, the functions are constructed recursively over time, possibly with some iteration to update the approximations,

A common approach is to construct an approximation $\hat{\mathcal{Q}}^{t+1}(x^t, \xi^t)$ as a linear combination of known *basis functions* $\Phi^t(\cdot, \cdot) = (\phi_1^t(\cdot, \cdot), \dots, \phi_M^t(\cdot, \cdot))$ that are fitted with weights, λ^t , so that

$$\hat{\mathcal{Q}}^{t+1}(x^t, \xi^t) = \Phi^t(x^t, \xi^t)\lambda^t. \quad (5.1)$$

The ϕ^t functions can be chosen quite generally to provide close approximation for a wide range of possible value functions. The λ^t values can be chosen with a backward recursion to simulate x^t and ξ^t values at samples (x_k^t, ξ_k^t) for $k = 1, \dots, K$ and then to choose λ^t to fit (e.g., using regression) $\Phi^t(x^t, \xi^t)\lambda^t$ to the values (for a multistage stochastic linear program):

$$\begin{aligned}\tilde{Q}^{t+1}(x_k^t, \xi_k^t) &= \min c_k^{t+1} x^{t+1} + E[\Phi^{t+1}(x^{t+1}, \xi^{t+1}) \lambda^{t+1} | \xi_k^t] \\ \text{s. t. } W^{t+1} x^{t+1} &= h_k^t - T_k^t x_k^t, \\ x^{t+1} &\geq 0.\end{aligned}\quad (5.2)$$

For the integration of Φ^{t+1} , if the integral is easily calculated (as in the separable approximations below), then this can be evaluated directly; otherwise, additional samples of ξ^{t+1} can be used to find an approximate value. For specific forms of the Φ functions, independent samples of paths can be used without requiring that the tree structure be maintained in each period with effort just increasing in a number K of paths instead of $\prod_{t=1}^H \mathcal{K}_t$ as in tree-generation methods. Suppose, for example, a multistage stochastic linear program such that each Φ^{t+1} is an affine function of $x^t = h^t - T^t x^t$ (which is most applicable when only h^t and T^t are random). We consider a set of K sample paths, ξ_1, \dots, ξ_K . The approximate value at period t of sample k in (5.2) can then be written with explicit dependence on the λ values as:

$$\begin{aligned}\tilde{Q}^{t+1}(x_k^t, \xi_k^t, \lambda^{t+1}) &= \min c_k^{t+1} x^{t+1} + (\lambda^{t+1})^T (\bar{h}^{t+1} - \bar{T}^{t+1} x^{t+1}) + \lambda_0^{t+1} \\ \text{s. t. } W^{t+1} x^{t+1} &= h_k^t - T_k^t x_k^t, \\ x^{t+1} &\geq 0,\end{aligned}\quad (5.3)$$

where \bar{h}^{t+1} and \bar{T}^{t+1} are understood as conditional expectations of \mathbf{h}^t and \mathbf{T}^t given ξ_k^t and x_k^t respectively and λ_0^{t+1} is the scalar value in the affine approximation. For a dual solution to (5.3), π_k^t , $\tilde{Q}^{t+1}(x_k^t, \xi_k^t, \lambda^{t+1}) = (h_k^t - T_k^t x_k^t)^T \pi_k^t(\lambda) + (\lambda^{t+1})^T \bar{h}^{t+1} + \lambda_0^{t+1}$. We can then define the linear approximation with λ to be consistent with these dual values in each period t :

$$(\lambda^t)^T (\bar{h}^t - \bar{T}^t x^{t+1}) + \lambda_0^t = \frac{1}{K} \sum_{k=1}^K (h_k^t - T_k^t x_k^t)^T \pi_k^t(\lambda) + (\lambda^{t+1})^T \bar{h}^{t+1} + \lambda_0^{t+1},$$

which then yields a dual bounding problem with additional constraints to ensure consistent future period values in (5.2) and (5.3) to find $\tilde{z}_L^K =$

$$\begin{aligned}\max_{\pi} h^1 \pi^1 + \frac{1}{K} \sum_{t=1}^H \sum_{k=1}^K h_k^t \pi_k^t \\ \text{s. t. } (W^t)^T \pi_k^t + \frac{1}{K} \sum_{l=1}^K (T_l^{t+1})^T \pi_l^{t+1} \leq q^t, t = 1, \dots, H-1; k = 1, \dots, K; \\ (W^H)^T \pi_k^H \leq q^H; k = 1, \dots, K;\end{aligned}\quad (5.4)$$

with optimal value $\tilde{\pi}$. Since π is a dual feasible solution of (3.4.1), this process produces a lower bound estimate on the optimal value z^* of (3.4.1) such that $E[\tilde{z}_L^K] \leq z^*$ (Exercise 1). In fact, any feasible solution of (5.4) provides a lower bound on z^* . The approximation comes on any path k from restricting the subsequent period multipliers π_l^{t+1} to be the same across all paths instead of depending explicitly on each path (or, in the primal view, on each solution x_k^t). Relaxations of

this restriction are possible by for example allowing some conditioning in the values of π_l^{t+1} used in the constraints with each π_k^t . In general, the method can also be viewed as a version of nested decomposition in which only a single cut is added in each period.

Upper bound estimates are available directly using

$$\tilde{z}_U^K = c^1 x^1 + \frac{1}{K} \sum_{t=2}^H \sum_{k=1}^K c^t x_k^t, \quad (5.5)$$

such that $E[\tilde{z}_U^K] \geq z^*$. Increasing the number of samples does not necessarily bring the lower and upper bound estimates together, but the ability to improve the lower bounding estimate through some use of conditional information in π suggests a possible approach to convergence. In any event, this method for estimates has substantially reduced complexity from full-tree generation methods and can be quite effective in practice, as we discuss below for problems in network revenue management.

a. Network revenue management

A typical application where ADP can be applied is in *network revenue management*, which represents decisions on allocating capacity to different products (e.g., fare classes and itineraries) that use common resources (e.g., seats on a flight, rooms in a hotel on a given night, or cars of a given class on a given day). The decision vector includes x^t and y^t at time t where x^t is an $n+m$ -vector of n product reservation acceptances in the current period and m cumulative resource commitments and y^t an n -vector of penalized acceptances (due to insufficient demand) which is used to allow for relatively complete recourse. The demand is given by \mathbf{d}^t , an n -vector of current period demand. The full problem (where y variables are included for completeness only) is to find $z^* =$

$$\begin{aligned} & \min c^1 x^1 + E \left[\sum_{t=1}^T c^t \mathbf{x}^t - c^t \mathbf{y}^t \right] \\ \text{s. t. } & W^1 x_{1,\dots,n}^1 + x_{n+1,\dots,n+m}^1 = h^1; \\ & W^t \mathbf{x}_{1,\dots,n}^t + \mathbf{x}_{n+1,\dots,n+m}^t = \mathbf{x}_{n+1,\dots,n+m}^{t-1}, t = 2, \dots, T; \\ & \mathbf{x}_{1,\dots,n}^t - \mathbf{y}^t \leq \mathbf{d}^t, t = 1, \dots, T; \\ & \mathbf{x}^t, \mathbf{y}^t \geq 0, t = 1, \dots, T, \text{ a.s.}; \\ & \mathbf{x}^t, \mathbf{y}^t \text{ nonanticipative, } t = 1, \dots, T, \text{ a.s.}; \end{aligned} \quad (5.6)$$

where we can assume for simplicity that $W = W^t, t = 1, \dots, H$, the resource-usage matrix, in each period is the same. A common approximation to (5.6) is the *bid-price* linear program (see Williamson [1992] and Talluri and van Ryzin [2004])

which solves the aggregated expected value problem as in (2.5) as: $\hat{z} =$

$$\begin{aligned} \min \quad & C^1 \hat{X}^1 \\ \text{s. t.} \quad & A(H\hat{X}_{1,\dots,n}^1) + \hat{X}_{n+1,\dots,n+m}^1 = h^1, \\ & H\hat{X}_{1,\dots,n}^1 \leq \sum_{t=1}^H \bar{d}^t; \\ & \hat{X}_{1,\dots,n}^1 \geq 0, \end{aligned} \tag{5.7}$$

where note that $H\hat{X}^1$ can be replaced by a different variable X' as is commonly given. In comparison to (2.5), we have collapsed everything into the first period (or have an empty initial period). We omitted the \hat{Y} variables which would be zero in an optimal solution. From (5.6), we obtain a feasible dual solution to (5.6) so that $\varepsilon^- = 0$ in Theorem 3 and $z^* \geq \hat{z}$ (Exercise 2). For an upper bound, we could use the solution $x^t = X^1$ in each period t (and then compute penalties in ε^+ whenever $x^t > \mathbf{d}^t$) or we can define x^t recursively as $x^1 = \min\{d^1, H\hat{X}^1\}$ and then $\mathbf{x}^t = \min\{H\hat{X}^1 - \mathbf{x}_{t+1}, \mathbf{d}^t\}$, and $\mathbf{x}^H = H\hat{X}^1 - \mathbf{x}_{H-1}$ to obtain a sharper bound, which amounts to using $H\hat{X}^1$ as a static *booking limit* vector (Exercise 3).

An upper bound can also be obtained (as done in practice) by using the optimal dual multipliers $\hat{\Pi} = (\hat{\Pi}_1, \hat{\Pi}_2)$ of (5.7) to determine whether to accept a reservation or not. In this process, if $c_i^t - A_{:,i}^T \hat{\Pi}_1 \leq 0$, then a reservation for product i is accepted if there is sufficient demand and available capacity. This is the notion of bid-prices in which the $-\hat{\Pi}_1$ values are prices on the resources bid against the revenue of each product. Generally, new versions of (5.7) are re-solved in each period with updated information to obtain new prices to determine acceptance.

Still another possible disaggregation is to use $\frac{\hat{X}_i^1}{\sum_{t=1}^H \bar{d}_i^t}$ as the probability of accepting a reservation for product i and again to define the values sequential in time with repeated solution of the updated version of (5.7). This approach is described in Jasin and Kumar [2010], who obtain an a priori bound on the loss in value from this approximate policy and then show how to choose re-solving times such that asymptotically as the system size grows, the relative loss in performance from using the approximation goes to zero.

Another interpretation of (5.7) is in its dual, in which case, it represents an aggregation of the linear ADP formulation in (5.4), which then implies that the lower bound in (5.7) is not as sharp as would be obtained using (5.4) (Exercise 4). This is the observation in Adelman [2007], which also presents a method to obtain an approximate solution with bounded accuracy for a linearization of the full problem.

b. Vehicle allocation problems

Vehicle allocation problems provide a different structure that allows specific bound construction. These problems can be represented as multistage network problems

with only arc capacities random. A formulation would then be the same as (1.1). The matrices W^t correspond to flows leaving nodes in period t while T^t corresponds to flow entering nodes in period $t+1$. The only exception is in the last period for which W^H just gathers flow into ending nodes. For simplicity, this model assumes that all flow requires one period to move between nodes.

The $\mathbf{x}^t(ij)$ decisions are then flows from i in period t to j in period $t+1$. The randomness involves the demand from i to j in period t . We assume that $\mathbf{x}^t(ij) = \mathbf{x}^{t,f}(ij) + \mathbf{x}^{t,e}(ij)$, where $\mathbf{x}^{t,f}(ij)$ represents *full* loads (or vehicles) and $\mathbf{x}^{t,e}(ij)$ represents *empty* vehicles (assuming that fractional vehicle loads are feasible). For demand of $\xi^t(ij)$, we would have $\mathbf{x}^{t,f}(ij) \leq \xi^t(ij)$. The costs $c^{t,f}(ij)$ and $c^{t,e}(ij)$ then correspond to the unit values of moving full and empty vehicles from i to j at t . The result is that vehicles are conserved in (5.8). The decisions generally depend on the locations of vehicles at any point in time.

Frantzeskakis and Powell [1993] consider several alternative approximations of (5.8). First, one could solve the expected value problem to obtain \bar{x}^t values. These corresponding decisions can be used regardless of realized demand (as, e.g., in Bi-tran and Yanasse [1984]). Then the x^t values could be split into full and empty parts, $\mathbf{x}^t = \bar{\mathbf{x}}^t$, $\mathbf{x}^{t,f}(ij) = \max\{\bar{x}^t(ij), \xi^t(ij)\}$, according to realized demand to produce both upper and lower bounds. This could be viewed as a generalization of a simple recourse strategy; hence Powell and Frantzeskakis refer to it as the *simple recourse* strategy.

Another approach is simply to solve the mean value problem, but only actually to send a vehicle from i to j at t if there is sufficient demand. In this way, $\mathbf{x}^{t,f}(ij) = \max\{\bar{x}^t(ij), \xi^t(ij)\}$, but $\mathbf{x}^t(ij) = \mathbf{x}^{t,f}(ij)$ whenever $i \neq j$. This strategy is called *null recourse*.

A further strategy is called *nodal recourse*, in which a set of decisions or a policy, $\delta^t(i)$, is defined for each node i at all times t . This policy would be a list of options for flow from i at t . The list would be a ranking of full loads (i.e., preferred nodes, $j_1(i), \dots, j_k(i)$) if capacity is available followed by an alternative for any remaining empty vehicles.

This preference structure can be constructed using a separable approximation from period $t+1$ to H . In period H , we can begin by assigning some salvage/final value $-c^H(i)$ to vehicles on the arcs corresponding to travel from one node to itself.

At period $H-1$, the value of sending a full load from i to j is simply $-c^{H-1,f}(ij) - c^H(j)$. Including empty loads in the obvious way and ordering in decreasing orders for each p determines the strategy at $H-1$. Now, given the distributions of ξ^{H-1} , these values yield an expected value function for vehicles at i at t . The argument of this function is a new (state) variable, $y^{H-1}(i)$. With the function defined, similar decisions on expected values of loads from i to j can be made in period $H-2$. A dynamic programming recursion would be to find $\mathcal{Q}^t(\mathbf{y}^t) = E_{\xi^t}[Q^t(\mathbf{y}^t, \xi^t)]$ where:

$$Q^t(\mathbf{y}^t, \xi^t) = \min_{\mathbf{x}^t, \mathbf{y}^t} c^t \mathbf{x}^t + \mathcal{Q}^{t+1}(\mathbf{y}^{t+1})$$

$$\begin{aligned} \text{s. t.} \quad & W^t \mathbf{x}^t = \mathbf{y}^t, \\ & T^t \mathbf{x}^t - \mathbf{y}^{t+1} = 0, \\ & \xi^t \geq \mathbf{x}^t \geq 0. \end{aligned} \tag{5.8}$$

If $\mathcal{Q}^{t+1}(\mathbf{y}^{t+1})$ is linear with coefficients, $\bar{Q}^{t+1}(i)$ in each component i of \mathbf{y}^{t+1} as it is for $t = H - 1$, then the optimal solution to (5.8) is given by the increasing ordering of $c^{t,f}(ij) + \hat{Q}^{t+1}(j)$ with each successive $x^{t,f}(ij)$ used up to the minimum of $y^t(i)$ and $\xi^t(ij)$ according to this realization of ξ^t . The key is then to construct a linear approximation to $\mathcal{Q}^{t+1}(\mathbf{y}^{t+1})$.

With a linearization, the entire strategy can be simply carried back to the first period. As in other ADP methods, this represents a feasible but not optimal strategy because it avoids calculating the full nonlinear value function. One way to compute the linearization is to assume an input value $\hat{y}^t(i)$ and to find the probability of each option multiplied by the expected linearized value of that option. Using this to determine the recourse value at each stage can lead to a lower bound at each stage and overall when the first-period problem is solved (see Exercise 4). An upper bounding linearization is also possible. This is analogous to the Edmundson-Madansky approach (Exercise 5).

Frantzeskakis and Powell [1993] mention that extensions of nodal recourse can apply to general network problems. These procedures are similar to the separable bounding procedures presented next. They again rely on building responses to random variation that depend separately on the random components and that are also feasible.

c. Piecewise-linear separable bounds

Another approach to ADP is to extend the basic separable bounds presented in Section 8.5b. to multistage problems. The main idea is to use the two-stage method repeatedly to approximate the objective function by separable functions (and not just single affine functions as in (5.2)). For linear problems, this leads to sublinear or piecewise linear functions as in Section 8.5b. Functions without recession directions (e.g., quadratic functions) would require some type of nonlinear (e.g., quadratic) function that should again be easily integrable, requiring, for example, limited moment information (second moments for quadratic functions). We consider the linear case (following Birge [1989]).

The goal is to construct a problem that is separable in the components of the random vector. In each period t , a decision, x^t , is made subject to the constraints, $W^t x^t = \xi^t - T^{t-1} x^{t-1}$, $x^t \geq 0$, where ξ^t is the realization of random constraints and x^{t-1} was the decision in period $t - 1$. The objective contribution from this decision is $c^t x^t$. We can view this decision as a response to the input, $\eta^t = \xi^t - T^{t-1} x^{t-1}$. The period t decision, x^t , then becomes a function of this input, so $x^t(\omega)$ becomes $x^t(\eta^t)$. Problem (2.2) becomes

$$\begin{aligned}
& \min c^1 x^1 + E[c^2 x^2(\boldsymbol{\eta}^2) + \cdots + c^H x^H(\boldsymbol{\eta}^H)] \\
\text{s. t. } & W^1 x^1 = h^1, \\
& W^t x^t(\boldsymbol{\eta}^t) = \boldsymbol{\eta}^t, \quad t = 2, \dots, H, \text{ a.s.,} \\
& \boldsymbol{\eta}^t = \boldsymbol{\xi}^t - T^{t-1} x^{t-1}(\boldsymbol{\eta}^{t-1}), \quad t = 2, \dots, H, \text{ a.s.,} \\
& x^t(\boldsymbol{\eta}) \geq 0, \quad t = 1, \dots, H.
\end{aligned}$$

The optimization problem is to determine the correct response to $\boldsymbol{\eta}^t$. The two-stage method given in Section 8.5b. gives a response that is separable in the components of $\boldsymbol{\xi} = \boldsymbol{\eta}^2$. In multiple stages, $\boldsymbol{\xi}$ is replaced by $\boldsymbol{\eta}^t$ for period t . The response must consider future actions and costs; so, it is no longer simply optimization of the second-period problem.

The dimension of $\boldsymbol{\eta} = (\boldsymbol{\eta}^2, \dots, \boldsymbol{\eta}^H)$ makes direct solution difficult in general. An upper bound is, however, obtained for any feasible response, i.e., decision vectors, $x^t(\boldsymbol{\eta}^t)$, that satisfy $W^t x^t(\boldsymbol{\eta}^t) = \boldsymbol{\eta}^t$, $x^t(\boldsymbol{\eta}^t) \geq 0$, a.s., where $\boldsymbol{\eta}^t = \boldsymbol{\xi}^t - T^{t-1} x^{t-1}(\boldsymbol{\eta}^{t-1})$ for all t . The two-stage method can be used to obtain feasible responses that are separable in the components of $\boldsymbol{\eta}^t$, i.e., where $x^t(\boldsymbol{\eta}^t) = \sum_i x_i^t(\boldsymbol{\eta}_i^t)$.

One choice is to let $x_i^t(\boldsymbol{\eta}_i^t)$ solve

$$\min c^t x^t \quad \text{s. t. } W^t x^t = \boldsymbol{\eta}_i^t e_i, \quad x^t \geq \beta, \quad (5.9)$$

where e_i is the i th unit vector and β depends on choices for the other x_i^t . Program (5.9) is a parametric linear program in $\boldsymbol{\eta}_i^t$. It is particularly easy to solve if $\beta = 0$. In this case, $x_i^t(\boldsymbol{\eta}_i^t)$ is linear for positive and negative $\boldsymbol{\eta}_i^t$. We suppose this case and let the optimal solutions be $x_i^{t,\pm}$ when $\boldsymbol{\eta}_i^t = \pm 1$.

A solution can be obtained if we can find the distribution of the $\boldsymbol{\eta}_i^t$ given responses determined by solutions of (5.9). The resulting problem to solve is

$$\begin{aligned}
(SL) \quad & \min c^1 x^1 + \sum_{t=2}^H \sum_{i=1}^{m_t} \int \psi_i^t(\boldsymbol{\eta}_i^t) P(d\boldsymbol{\eta}_i^t) \\
\text{s. t. } & W^1 x^1 = h^1, \quad x^1 \geq 0,
\end{aligned}$$

where $\psi_i^t(\boldsymbol{\eta}_i^t) = c^t x_i^{t,+} \boldsymbol{\eta}_i^t$ if $\boldsymbol{\eta}_i^t \geq 0$, and $\psi_i^t(\boldsymbol{\eta}_i^t) = c^t x_i^{t,-} (-\boldsymbol{\eta}_i^t)$ if $\boldsymbol{\eta}_i^t \leq 0$. Assuming that the distribution of $\boldsymbol{\eta}^t$ is known in this approximation, we can find $\boldsymbol{\eta}^{t+1}$. Initially, $\boldsymbol{\eta}^2 = \boldsymbol{\xi}^2 - T^1 x^1$, which has the same distributional form as $\boldsymbol{\xi}^2$. In general, $\boldsymbol{\eta}_j^{t+1}$ is given by:

$$\boldsymbol{\eta}_j^{t+1} = \boldsymbol{\xi}_j^{t+1} - T_j^t \cdot \left[\sum_{i=1}^{m_t} (x_i^{t,+} \mathbf{1}_{\boldsymbol{\eta}_i^t \geq 0} + x_i^{t,-} \mathbf{1}_{\boldsymbol{\eta}_i^t < 0})(|\boldsymbol{\eta}_i^t|) \right]. \quad (5.10)$$

Note that the values in (5.10) are linear functions of $\boldsymbol{\eta}^t$ on the regions where $\boldsymbol{\eta}^t$ has constant sign. We can, therefore, construct $\boldsymbol{\eta}^{t+1}$ as a function of $\boldsymbol{\eta}^t$ by overlaying these linear transformations of random variables. For normally distributed data, this may be possible because the transformation does not affect the distribution class. For

other distributions, it is more difficult. Even in the normal case, however, we have different distribution parameters for all possible sign combinations of all random variables in previous period inputs. Exponential growth of the calculations in the number of periods is not avoided.

Because the approximation given earlier may be difficult to compute even with normal distributions, it may be necessary to approximate the distribution of η^{t+1} . We can use bounds on $P\{\eta_i^t \geq 0\}$ and on the moments conditional on $\eta_i^t \geq 0$ or < 0 . Given these values, moment problems can be solved to calculate corresponding values for η^{t+1} and to bound ψ_i^t (see Birge and Wets [1989]). Any other bounds on the input $(T^t x^t)$ from period t to period $t+1$ can also be used to obtain crude bounds on the ψ values. Also, note that certain problems, such as networks, may have few nonzeros in the T^t terms and close-to-simple recourse structure. The random input vector η^{t+1} may be easily calculable for these problems.

Another looser but more implementable bound can be obtained by forcing a feasible and separable response in all future periods depending on a single random variable in the current period. This eliminates the problem of characterizing the distribution of inputs to all periods. It does, however, force a dependency in future periods that may increase the bound.

To develop this response function, let $X^t(\pm i)$ be an optimal solution, (x^t, \dots, x^H) , ($t > 1$), to:

$$\begin{aligned} \min c^t x^t + \dots + c^H x^H \\ \text{s. t.} \quad W^t x^t = \pm e_i, \\ T^t x^t + W^{t+1} x^{t+1} = 0, \\ \dots \quad \vdots \\ W^H x^H = 0, \\ x^\tau \geq 0, \quad \tau = t, \dots, H. \end{aligned} \tag{5.11}$$

Now define

$$z_i^t(\hat{\xi}_i^t) = \int_{\xi_i^t - \hat{\xi}_i^t > 0} C^t X_i^{t+}(\xi_i^t - \hat{\xi}_i^t) P(d\xi^t) + \int_{\xi_i^t - \hat{\xi}_i^t \leq 0} C^t X_i^{t-}(-\xi_i^t + \hat{\xi}_i^t) P(d\xi^t), \tag{5.12}$$

where $C^t = (c^t, \dots, c^H)$. An *upper bound* on the objective value of (5.9) is obtained by solving the separable nonlinear program:

$$\begin{aligned} \min c^1 x^1 + \dots + c^H x^H + \sum_{t=2}^H \sum_{i=1}^{m_t} z_i^t(\hat{\xi}_i^t) \\ \text{s. t.} \quad W^1 x^1 = h^1, \\ T^t x^t + W^{t+1} x^{t+1} - \hat{\xi}_i^{t+1} = 0, \quad t = 1, \dots, H-1, \\ x^t \geq 0, \quad t = 1, \dots, H, \quad \hat{\xi} \in \Xi, \end{aligned} \tag{5.13}$$

where Ξ is the support set of the random variables. Note that if we drop the nonlinear term in the objective and replace $\hat{\xi}_i$ in the constraints with a fixed valued of $E[\xi_i]$, then we can obtain a *lower bound* on the optimal objective value in (5.9) (see Birge and Wets [1986]). We should note that in some cases, we may not have a solution to (5.11) for $\pm e_i$ but may only have a solution for $+e_i$, e.g. In this case, $\hat{\xi}_i^{t+1}$ could be constrained to be less than the minimum possible value of ξ_i^t .

In (5.13), we are solving to determine a *centering point*, $\hat{\xi}^t$, that obtains minimum cost if we assume the response to any variation from $\hat{\xi}^t$ is a solution of (5.11). By allowing some variation of the choice of centering point, a “best” approximation of this type is found. The value of (5.13) is an upper bound because the composition of the x^t solutions from (5.13) and the X^t values used in the z terms yield a feasible solution for all ξ .

This procedure may also be implemented as responses to several scenarios. In this case, the random vectors are partitioned as in Section 10.1. The partitions may also be part of the higher-level optimization problem so that in some way a “best” partition can be found. The points used within the partitions may be chosen as expected values, in which case the solution without penalty terms is again a lower bound on the optimal objective value. For an upper bound, this vector may be allowed to take on any value in the partition.

The use of multiple scenarios enters directly into the progressive hedging approach of Rockafellar and Wets (see Section 5.3). This can be used to solve the top-level problem and to approach a solution that is optimal for a given set of partitions and the piecewise linear penalty structure presented here. Computations are then restricted to optimizing separable nonlinear functions subject to linear constraints. Implementations can be based on previous procedures (such as decomposition).

The basic framework for the upper bounding procedures given earlier is to construct a feasible solution that is easily integrated. Other procedures for constructing such feasible responses are possible. For example, Wallace and Yan [1993] suppose two types of restrictions of the set of solutions to obtain bounds. The first is to suppose only a subset of variables is used within a period, as, for example, with the penalty terms used for aggregation bounds in Section 10.2. The other approach is to suppose that all realizations from period to period must meet some common constraint on values passed between periods. This procedure effectively divides the multistage problem into a sequence of two-stage problems. It appears to work well on problems with many stages.

d. Nonlinear bounds and a production planning example

As noted earlier, many multistage stochastic program approximations can take advantage of the specific problem structure. For Example 2 in Section 10.2, we considered a basic production problem that allows the construction of bounds on optimal primal and dual variables that can then be used in constructing optimal objective value bounds as in (2.7). Other bounds and approximations using similar production

problem structures are also possible. We explore some of those bounds developed by Ashford [1984], following Beale, Forrest, and Taylor [1980], and Bitran and Yanasse [1984], and Bitran and Sarkar [1988]. These bounds can be viewed as extensions of the aggregation-type bounds in Section 10.2.

The first type of extension of the production problem we consider is the model used in Ashford [1984] which is a slight generalization of (2.8). It is also an extension of similar work by Beale, Forrest, and Taylor [1980] on a production problem similar to (2.8). The model is to

$$\begin{aligned} \min z &= E_{\xi} \left[\sum_{t=1}^T (-c^t x^t(\xi) - q^t y^t(\xi)) \right] \\ \text{s. t. } & T^{t-1} \mathbf{y}^{t-1} + W^t \mathbf{x}^t - \mathbf{y}^t \leq \xi^t, \text{ a.s., } t = 1, \dots, H, \\ & \mathbf{y}^t \geq \mathbf{l}^t, \quad u^t \geq \mathbf{x}^t \geq \mathbf{0}, \text{ a.s., } t = 1, \dots, H, \end{aligned} \quad (5.14)$$

where x^t represents production and related variables and y^t represents the state (e.g., inventory) after realizing demands, ξ^t . Both variables are bounded, although y^t may only have trivial bounds. One upper bound directly analogous to that in Theorem 3 can be constructed using this structure (see Exercise 1).

A lower bound on the optimal value of (5.14) can be obtained simply by substituting expected values for the random elements in (5.14). Ashford also presents an improved lower bound, however, that forms the basis for an approximation procedure. This bound consists of solving a *reduced problem*:

$$\begin{aligned} \min z^{RED}(G_1, \dots, G_H) &= \sum_{t=1}^T (-c^t x^t - q^t y^t) \\ \text{s. t. } & T^{t-1} \mathbf{y}^{t-1} + W^t \mathbf{x}^t - \mathbf{w}^t = \bar{\xi}^t, \quad t = 1, \dots, H, \\ & -y^t - w^t \leq -f^t(w^t - l^t), \quad t = 1, \dots, H, \\ & u^t \geq \mathbf{x}^t \geq \mathbf{0}, \text{ a.s., } t = 1, \dots, H, \end{aligned} \quad (5.15)$$

where the G^t are m_t -vectors of given distribution functions, G_{it} , $i = 1, \dots, m_t$, and $f^t = (f_{1t}, \dots, f_{m_t t})$, with

$$f_{it}(\eta_i) = \int_{-\infty}^{-\eta_i} (\eta_i + \zeta) dG_{it}(\zeta), \quad (5.16)$$

for $i = 1, \dots, m_t$.

The bound in (5.15) is chosen by first determining the distribution function, G_{it} . If G_t^* is the vector of distribution functions of $ZT^{t-1} \mathbf{y}^{t-1,*} + W^t \mathbf{x}^{t,*} - \xi^t$ for an optimal solution $(\mathbf{y}^*, \mathbf{x}^*)$ of (5.14), then the following theorem holds.

Theorem 4. *The solution $z^{RED}(G_1^*, \dots, G_H^*)$ provides a lower bound on the optimal solution z^* in (5.14) and $z^{RED}(G_1^*, \dots, G_H^*) \geq z(\bar{\xi})$, the solution of the expected value problem, i.e., (5.14) with all random variables replaced by their expectations.*

Proof: Exercise 2. \square

It is possible to make the approximation in (5.15) into a deterministic equivalent of (5.14) if appropriate penalties are placed on the violation of bound constraints on x^t , but the calculation of this and of the bound given by Theorem 1 requires information about the optimal solutions which is not known. Another bound is, however, obtainable by substituting $G^\xi(t)$, the distribution function vector, corresponding to $(\xi^t - \bar{\xi}^t)$ (see Exercise 3). This represents the beginning of an approximation when the ξ^t vectors are normally distributed. The approximation successively estimates parameters of a normal approximation of the distribution of $T^{t-1}\mathbf{y}^{t-1,*} + W^t\mathbf{x}^{t,*} - \xi^t$ from t to $t+1$. This procedure continues until little improvement occurs in this updating procedure. Computational results with this procedure show significant savings over dynamic programming calculations.

This process can be viewed as a form of dynamic programming approximation using the input to each period's decisions as the quantity, $T^{t-1}\mathbf{y}^{t-1,*} + W^t\mathbf{x}^{t,*} - \xi^t$. In this way, it is also similar to the response method given above. An alternative approach is to build approximations of the value function from period to period. One application to problems with uncertainties in the W^t matrix in (5.14) appears in Beale, Dantzig, and Watson [1986]. The bounds developed by Bitran et al. follow these production examples closely. The model is again of the form in (2.8).

e. Extensions

Other structures can also yield bounds in specific cases. For PERT networks (see, e.g., Taha [1992]), for example, a typical problem is to balance the benefits of early completion against the possible penalty costs of exceeding a due date or promise date. In these problems, a natural separation occurs that allows calculation despite the interconnected structure of paths and possibly correlated times. Klein Haneveld [1986] considers bounds on expected tardiness penalties with mean constraints. Maddox and Birge extend this analysis to bounds with second moment information (Birge and Maddox [1995, 1996]) and to bounding probabilities of tardiness (Maddox and Birge [1991]).

The basic principle throughout this and previous chapters on approximations is to use convexity of objective and constraints. Relax the problem and substitute expectations properly to obtain a lower bound. Restrict the problem and maintain a feasible solution (as perhaps a combination of extremal solutions) to obtain an upper bound. Many more bounding approximations are possible based on these fundamental observations.

Exercises

1. Show that the ADP estimate satisfies the inequality, $E[\bar{z}^K] \leq z^*$, for the multi-stage stochastic linear program with randomness only in \mathbf{h}^t and \mathbf{T}^t .

2. Show that $\hat{z} \leq z^*$ for the bid-price linear program (5.7).
3. Show that the alternative booking limit disaggregate solution provides a sharper upper bound than the bound using Theorem 3.
4. Show that (5.4) provides a sharper lower bound on (5.6) than the bid-price linear program (5.7).
5. Consider a network revenue management model with $A = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix}$, $c^t = [-200 - 150 - 100]^T$, and $\mathbf{d}_i^t = 1$ for $i = 1, 2, 3$ with probability 0.5, 0.3, and 0.4 respectively with $b^0 = [1510]^T$. Let $H = 20$.
 - (a) Solve the bid-price linear program (5.7) and the ADP linear approximation (5.4) with 100 random sample paths to obtain lower bound estimates on z^* .
 - (b) Construct upper bounds using (i) Theorem 3 for the bid-price linear program; (ii) the modified booking limit upper bound.
 - (c) Construct a simulation to test the use of: (i) re-solving the bid-price linear program in each period; (ii) re-solving the ADP linear approximation (5.4); (iii) using the probability interpretation in the re-solving step as in Jasin and Kumar [2010].
6. Use the separable function approach and (5.12) to construct an upper bound on Example 1 with uniform demand distributions.
7. Let A'^+ be the matrix composed of the positive elements of W^t in (5.14) (with zeros elsewhere). Use this to construct a bound on any feasible dual variable value with $\beta^t = \sum_{\tau=t}^H (\prod_{s=t}^{\tau-1} (A^{s+})^T) q^\tau$, where $\prod_{s=t}^{t-1} (A^{s+})^T = I$. Combine this with Theorem 3 to obtain an upper bound on the optimal objective value using the solution to the mean value problem.
8. Prove Theorem 4.
9. Show that $z^{RED}(G_1^\xi, \dots, G_H^\xi) \leq z^*$.
10. To construct a lower bound for nodal recourse, assume a projected value, $\hat{y}^t(i)$ of $\mathbf{y}^t(i)$ (as, e.g., an average of incoming and outgoing loads). Find an expression (in terms of the demand distributions on the ranked full load alternatives) for the expected value (assuming linearized future costs) of an additional vehicle beyond $\hat{y}^t(i)$. Show that this procedure gives a lower bound on (5.8) when $t = 1$.
11. Show how an upper bounding linearization can be constructed for (5.8) using a linearization of $\mathcal{Q}^{t+1}(\mathbf{y}^{t+1})$. (Note: You can assume a constant number of total vehicles.)
12. Consider a three-period example with five total vehicles, three nodes (cities), and salvage values, $c^3(1) = -2$, $c^3(2) = -1$, and $c^3(3) = -4$. Currently, two vehicles are at A , two vehicles are at B , and one vehicle is at C . Suppose demand in each period is uniform on the integers from zero to $\xi^{\max}(ij)$, where $\xi^{\max}(ij)$ has the following values:

To $j = 1 2 3$

From $i =$

1	0	2	3
2	2	0	2
3	3	3	0.

Suppose the costs (negative of profits) on each route for a full truck are

To $j = 1 2 3$

From $i =$

1	0	-1	-2
2	-1	0	-3
3	-2	-3	0.

Empty load costs are

To $j = 1 2 3$

From $i =$

1	0	1	2
2	1	0	3
3	2	3	0.

Use the lower and upper bounding procedures in Exercises 4 and 5 to construct upper and lower bounds on (5.8) for these data.

Appendix A

Sample Distribution Functions

This appendix gives the basic distributions used in the text. We provide their means and variances. Tables of numerical data for these distributions are easily available on the web. One such website is <http://stattrek.com/>.

A.1 Discrete Random Variables

Uniform: $U[1, n]$

$$P(\xi = i) = \frac{1}{n}, \quad i = 1, \dots, n, \quad n \geq 1,$$

with $E[\xi] = \frac{n+1}{2}$ and $\text{Var}[\xi] = \frac{n^2-1}{12}$.

Binomial: $Bi(n, p)$

$$P(\xi = i) = \binom{n}{i} p^i (1-p)^{n-i}, \quad i = 0, 1, \dots, n; \quad 0 < p < 1,$$

with $E[\xi] = np$ and $\text{Var}[\xi] = np(1-p)$.

Poisson: $P(\lambda)$

$$P(\xi = i) = e^{-\lambda} \frac{\lambda^i}{i!}, \quad \lambda > 0, \quad i = 0, 1, \dots,$$

with $E[\xi] = \lambda$ and $\text{Var}[\xi] = \lambda$.

A.2 Continuous Random Variables

Uniform: $U[0, a]$

$$f(\xi) = \frac{1}{a}, \quad 0 \leq \xi \leq a, \quad a > 0,$$

with $E[\xi] = \frac{a}{2}$ and $\text{Var}[\xi] = \frac{a^2}{12}$.

Exponential: $\exp(\lambda)$

$$f(\xi) = \lambda e^{-\lambda \xi}, \quad 0 \leq \xi, \quad \lambda > 0,$$

with $E[\xi] = \frac{1}{\lambda}$ and $\text{Var}[\xi] = \left(\frac{1}{\lambda}\right)^2$.

Normal: $N(\mu, \sigma^2)$

$$f(\xi) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(\xi-\mu)^2}{2\sigma^2}}, \quad \sigma > 0,$$

with $E[\xi] = \mu$ and $\text{Var}[\xi] = \sigma^2$.

Gamma: $G(\alpha, \beta)$

$$f(\xi) = \frac{1}{\beta^2 \Gamma(\alpha)} \xi^{\alpha-1} e^{-\frac{\xi}{\beta}}, \quad \alpha > 0, \quad \beta > 0,$$

where $\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x} dx$, $\alpha > 0$, $E[\xi] = \alpha\beta$ and $\text{Var}[\xi] = \alpha\beta^2$.

References

1. P.G. Abrahamson, “A Nested Decomposition Approach for Solving Staircase Linear Programs,” Ph.D. Dissertation, Stanford University (Stanford, CA, 1983).
2. D. Adelman, “Dynamic bid prices in revenue management,” *Operations Research* 55 (2007) pp. 647–661.
3. S. Ahmed, “Convexity and decomposition of mean-risk stochastic programs,” *Mathematical Programming Series A* 106 (2006) pp. 433–446.
4. S. Ahmed, M. Tawarmalani, and N. V. Sahinidis, “A finite branch and bound algorithm for two-stage stochastic integer programs,” *Mathematical Programming* 100 (2004) pp.355-377.
5. E.D. Andersen, “The homogeneous and self-dual model and algorithm for linear optimization,” MOSEK Technical report: TR-1-2009, Copenhagen, DK, 2009.
6. S.A. Andreou, “A capital budgeting model for product-mix flexibility,” *Journal of Manufacturing and Operations Management* 3 (1990) pp. 5–23.
7. K.M. Anstreicher, “A combined Phase I–Phase II projective algorithm for linear programming,” *Mathematical Programming* 43 (1989) pp. 209–223.
8. K.A. Ariyawansa and D.D. Hudson, “Performance of a benchmark parallel implementation of the Van Slyke and Wets algorithm for two-stage stochastic programs on the Sequent/Balance,” *Concurrency Practice and Experience* 3 (1991) pp. 109–128.
9. P. Artzner, F. Delbaen, J-M. Eber and D. Heath, “Coherent measures of risk,” *Mathematical Finance* 9 (1999) pp. 203-228.
10. R. Ashford, “Bounds and an approximate solution method for multistage stochastic production problems,” Warwick Papers in Industry, Business and Administration, No. 15, University of Warwick, Coventry, UK (1984).
11. S. Asmussen and P. Glynn, *Stochastic Simulation: Algorithms and Analysis*, Springer, New York, 2007.
12. H. Attouch and R.J-B Wets, “Approximation and convergence in nonlinear optimization” in: O.L. Mangasarian, R.R. Meyer and S.M. Robinson, Eds., *Nonlinear programming*, 4 (Academic Press, New York–London, 1981) pp. 367–394.
13. M. Avriel and A.C. Williams, “The value of information and stochastic programming,” *Operations Research* 18 (1970) pp. 947–954.
14. O. Bahn, J.-L. Goffin, O. du Merle, and J.-Ph. Vial, “A cutting plane method from analytic centers for stochastic programming,” *Mathematical Programming*, 69 (1995) pp. 45–73.
15. G. Bayraksan and D.P. Morton, “A sequential sampling procedure for stochastic programming,” Working Paper, University of Arizona, July, 2009.
16. M.S. Bazaraa and C.M. Shetty, *Nonlinear Programming: Theory and Algorithms* (John Wiley, Inc., New York, NY, 1979).
17. M.S. Bazaraa, J.J. Jarvis, and H.D. Sherali, *Linear Programming and Network Flows* (John Wiley, Inc., New York, NY, 1990).

18. E.M.L. Beale, "On minimizing a convex function subject to linear inequalities," *J. Royal Statistical Society, Series B* 17 (1955) pp. 173–184.
19. E.M.L. Beale, "The use of quadratic programming in stochastic linear programming," Rand Report P-2404-1, The Rand Corporation (1961).
20. E.M.L. Beale, J.J.H. Forrest, and C.J. Taylor, "Multi-time-period stochastic programming" in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980) pp. 387–402.
21. E.M.L. Beale, G.B. Dantzig, and R.D. Watson, "A first order approach to a class of multi-time-period stochastic programming problems," *Mathematical Programming Study* 27 (1986) pp. 103–117.
22. R. Bellman, *Dynamic Programming* (Princeton University Press, Princeton, NJ, 1957).
23. Ben-Tal, A., Boyd, S., Nemirovski, A., Extending the Scope of Robust Optimization: Comprehensive Robust Counterparts of Uncertain Problems, *Mathematical Programming* 107:1-2 (2006), 63–89.
24. Ben-Tal, A. and Arkadi Nemirovski, A. (2002). Robust optimization methodology and applications, *Mathematical Programming*, Series B 92, 453–480.
25. A. Ben-Tal and M. Teboulle, "Expected utility, penalty functions, and duality in stochastic nonlinear programming," *Management Science* 32 (1986) pp. 1445–1466.
26. J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik* 4 (1962) pp. 238–252.
27. B. Bereanu, "Some numerical methods in stochastic linear programming under risk and uncertainty" in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980) pp. 169–205.
28. J.O. Berger, *Statistical Decision Theory and Bayesian Analysis* (Springer-Verlag, New York, NY, 1985).
29. O. Berman, R.C. Larson, and S.S. Chiu, "Optimal server location on a network operating as a M/G/1 queue," *Operations Research* 33 (1985) pp. 746–770.
30. D.P. Bertsekas, *Dynamic Programming and Optimal Control, Volume II*, Third Edition (Athena Scientific, Boston, 2007).
31. D.P. Bertsekas and J.N. Tsitsiklis, *Neuro-Dynamic Programming* (Athena Scientific, Boston, 1995).
32. D. Bertsimas, D.A. Iancu, and P.A. Parrilo, "Optimality of affine policies in multistage robust optimization," *Mathematics of Operations Research* 35 (2010) pp. 363–394.
33. D. Bertsimas, P. Jaillet, and A. Odoni, "A priori optimization," *Operations Research* 38 (1990) pp. 1019–1033.
34. D. Bertsimas, K. Natarajan, and C-P. Teo, "Probabilistic combinatorial optimization: Moments, semidefinite programming and asymptotic bounds," *SIAM J. of Optimization* 15 (2004) pp. 185–209.
35. D. Bertsimas and I. Popescu, "Optimal inequalities in probability: A convex programming approach," *SIAM Journal of Optimization*, 15 (2004) pp. 780–804.
36. D. Bertsimas and M. Sim, "Tractable approximations to robust conic optimization problems," *Mathematical Programming* 107 (2006) pp. 5–36.
37. D. Bienstock and J.F. Shapiro, "Optimizing resource acquisition decisions by stochastic programming," *Management Science* 34 (1988) pp. 215–229.
38. P. Billingsley, *Convergence of Probability Measures* (John Wiley, Inc., New York, NY, 1968).
39. J.R. Birge, "Solution Methods for Stochastic Dynamic Linear Programs," Ph.D. Dissertation and Technical Report SOL 80-29, Systems Optimization Laboratory, Stanford University (Stanford, CA, 1980).
40. J.R. Birge, "The value of the stochastic solution in stochastic linear programs with fixed recourse," *Mathematical Programming* 24 (1982) pp. 314–325.
41. J.R. Birge, "Using sequential approximations in the L-shaped and generalized programming algorithms for stochastic linear programs," Technical Report 83-12, Department of Industrial and Operations Engineering, University of Michigan (Ann Arbor, MI, 1983); available at <http://hdl.handle.net/2027.42/3642>.

42. J.R. Birge, "Aggregation in stochastic production problems," *Proceedings of the 11th IFIP Conference on System Modelling and Optimization* (Springer-Verlag, New York, 1984).
43. J.R. Birge, "Aggregation in stochastic linear programming," *Mathematical Programming* 31 (1985a) pp. 25–41.
44. J.R. Birge, "Decomposition and partitioning methods for multi-stage stochastic linear programs," *Operations Research* 33 (1985b) pp. 989–1007.
45. J.R. Birge, "Exhaustible recourse models with uncertain returns from exploration investment" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988a) pp. 481–488.
46. J.R. Birge, "The relationship between the L-shaped method and dual basis factorization for stochastic linear programming" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988b) pp. 267–272.
47. J.R. Birge, "Multistage stochastic planning models using piecewise linear response functions" in: G. Dantzig and P. Glynn, Eds., *Resource Planning under Uncertainty for Electric Power Systems* (NSF, 1989).
48. J.R. Birge, "Quasi-Monte Carlo methods for option evaluation," Technical Report, Department of Industrial and Operations Engineering , University of Michigan (Ann Arbor, MI, 1994); available at <http://hdl.handle.net/2027.42/3632>.
49. J.R. Birge, "Option methods for incorporating risk into linear capacity planning models," *Manufacturing and Service Operations Management* 2 (2000), pp. 189–194.
50. J.R. Birge and M.A.H. Dempster, "Optimality conditions for match-up strategies in stochastic scheduling and related dynamic stochastic optimization problems," Technical Report 92-58, Department of Industrial and Operations Engineering, University of Michigan (Ann Arbor, MI, 1992); available at <http://hdl.handle.net/2027.42/3645>.
51. J.R. Birge, C.J. Donohue, D.F. Holmes, and O.G. Svitiski, "A parallel implementation of the nested decomposition algorithm for multistage stochastic linear programs," *Mathematical Programming* 75 (1996) pp. 327–352.
52. J.R. Birge and J. Dulá, "Bounding separable recourse functions with limited distribution information," *Annals of Operations Research* 30 (1991) pp. 277–298.
53. J.R. Birge, R.M. Freund, and R.J. Vanderbei, "Prior reduced fill-in in the solution of equations in interior point algorithms," *Operations Research Letters* 11 (1992) pp. 195–198.
54. J.R. Birge and D.F. Holmes, "Efficient solution of two-stage stochastic linear programs using interior point methods," *Computational Optimization and Applications* 1 (1992) pp. 245–276.
55. J.R. Birge and F.V. Louveaux, "A multicut algorithm for two-stage stochastic linear programs," *European Journal of Operations Research* 34 (1988) pp. 384–392.
56. J.R. Birge and M.J. Maddox, "Bounds on expected project tardiness," *Operations Research* 43 (1995) pp. 838–850.
57. J.R. Birge and M.J. Maddox, "Using second moment information in stochastic scheduling" in: G. Yin and Q. Zhang, Eds., *Recent Advances in Control and Manufacturing Systems* (Springer-Verlag, New York, NY, 1996) pp. 99–120.
58. J.R. Birge and L. Qi, "Computing block-angular Karmarkar projections with applications to stochastic programming," *Management Science* 34 (1988) pp. 1472–1479.
59. J.R. Birge and L. Qi, "Semiregularity and generalized subdifferentials with applications to optimization," *Mathematics of Operations Research* 18 (1993) pp. 982–1006.
60. J.R. Birge and L. Qi, "Subdifferential convergence in stochastic programs," *SIAM J. Optimization* 5 (1995) pp. 436–453.
61. J.R. Birge and C.H. Rosa, "Parallel decomposition of large-scale stochastic nonlinear programs," *Annals of Operations Research* 64 (1996), pp. 39–65.
62. J.R. Birge and M. Teboulle, "Upper bounds on the expected value of a convex function using subgradient and conjugate function information," *Mathematics of Operations Research* 14 (1989) pp. 745–759.
63. J.R. Birge and S.W. Wallace, "Refining bounds for stochastic linear programs with linearly transformed independent random variables," *Operations Research Letters* 5 (1986) pp. 73–77.

64. J.R. Birge and S.W. Wallace, "A separable piecewise linear upper bound for stochastic linear programs," *SIAM Journal on Control and Optimization* 26 (1988) pp. 725–739.
65. J.R. Birge and R.J-B Wets, "Approximations and error bounds in stochastic programming" in: Y. Tong, Ed., *Inequalities in Statistics and Probability* (IMS Lecture Notes—Monograph Series, 1984) pp. 178–186.
66. J.R. Birge and R.J-B Wets, "Designing approximation schemes for stochastic optimization problems, in particular, for stochastic programs with recourse," *Mathematical Programming Study* 27 (1986) pp. 54–102.
67. J.R. Birge and R.J-B Wets, "Computing bounds for stochastic programming problems by means of a generalized moment problem," *Mathematics of Operations Research* 12 (1987) pp. 49–162.
68. J.R. Birge and R.J-B Wets, "Sublinear upper bounds for stochastic programs with recourse," *Mathematical Programming* 43 (1989) pp. 131–149.
69. J.R. Birge and G. Zhao, "Successive linear approximation solution of infinite horizon dynamic stochastic programs," *SIAM Journal on Optimization* 18 (2007) pp. 1165–1186.
70. G.R. Bitran and D. Sarkar, "On upper bounds of sequential stochastic production planning problems," *European Journal of Operational Research* 34 (1988) pp. 191–207.
71. G.R. Bitran and H. Yanasse, "Deterministic approximations to stochastic production problems," *Operations Research* 32 (1984) pp. 999–1018.
72. C.E. Blair and R.G. Jeroslow, "The value function of an integer program," *Mathematical Programming* 23 (1982) pp. 237–273.
73. F. Black and M. Scholes, "The pricing of options and corporate liabilities," *Journal of Political Economy* 81 (1973) pp. 737–654.
74. D. Blackwell, "Discounted dynamic programming," *Annals of Mathematical Statistics* 36 (1965) pp. 226–235.
75. C. Borell, "Convex set functions in d -spaces," *Periodica Mathematica Hungarica* 6 (1975) pp. 111–136.
76. S.L. Brumelle and J.I. McGill, "Airline seat allocation with multiple nested fare classes," *Operations Research* 41 (1993) pp. 127–137.
77. G. Calafiore and M.C. Campi, "Uncertain convex programs: randomized solutions and confidence levels," *Mathematical Programming* 102(2005) pp. 25–46.
78. D.R. Cariño, T. Kent, D.H. Myers, S. Stacy, M. Sylvanus, A.L. Turner, K. Watanabe, and W.T. Ziemba, "The Russel- Yasuda Kasai model: An asset/liability model for a Japanese insurance company using multistage stochastic programming," *Interfaces* 24 (1994) pp. 29–49.
79. C.C. Carøe and J. Tind, "L-shaped decomposition of two-stage stochastic programs with integer recourse," *Mathematical Programming* 83 (1998) pp. 451–464.
80. T. Carpenter, I. Lustig, and J. Mulvey, "Formulating stochastic programs for interior point methods," *Operations Research* 39 (1991) pp. 757–770.
81. H.P. Chao, "Exhaustible resource models: the value of information," *Operations Research* 29 (1981) pp. 903–923.
82. A. Charnes and W.W. Cooper, "Chance-constrained programming," *Management Science* 5 (1959) pp. 73–79.
83. A. Charnes and W.W. Cooper, "Deterministic equivalents for optimizing and satisficing under chance constraints," *Operations Research* 11 (1963) pp. 18–39.
84. A. Charnes and W.W. Cooper, "Response to 'Decision problems under risk and chance constrained programming: dilemmas in the transition,'" *Management Science* 29 (1983) pp. 750–753.
85. A. Charnes, W.W. Cooper, and G.H. Symonds, "Cost horizons and certainty equivalents: an approach to stochastic programming of heating oil," *Management Science* 6 (1958) pp. 235–263.
86. M. Chen and S. Mehrotra, "Epi-convergent scenario generation method for stochastic problems via sparse grid," Technical Report 8, Northwestern University, December 2007 (Stochastic Programming E-print Series,<http://edoc.hu-berlin.de/docviews/abstract.php?lang=ger&id=28882>).

87. I.C. Choi, C.L. Monma, and D.F. Shanno, "Further development of a primal-dual interior point method," *ORSA Journal on Computing* 2 (1990) pp. 304–311.
88. K. L. Chung, *A Course in Probability Theory* (Academic Press, New York, NY, 1974).
89. V. Chvátal, *Linear Programming* (Freeman, New York/San Francisco, CA, 1980).
90. T. Cipra, "Moment problem with given covariance structure in stochastic programming," *Ekonom.-Mat. Obzor* 21 (1985) pp. 66–77.
91. T. Cipra, "Stochastic programming with random processes," *Annals of Operations Research* 30 (1991) pp. 95–105.
92. F. Clarke, *Optimization and Nonsmooth Analysis* (John Wiley, Inc., New York, NY, 1983).
93. A.R. Conn, N.I.M. Gould, and P.L. Toint, *Trust-Region Methods* (SIAM/MPS, Philadelphia, PA, 2000).
94. J. Cox and S. Ross, "The valuation of options for alternative stochastic processing," *Journal of Financial Economics* 3 (1976) pp. 145–166.
95. L. Dai, C. Chen, and J.R. Birge, "Convergence Properties of Two-Stage Stochastic Programming," *Journal Of Optimization Theory And Applications* 106 (2000) pp. 489–509.
96. G.B. Dantzig, "Linear programming under uncertainty," *Management Science* 1 (1955) pp. 197–206.
97. G.B. Dantzig, *Linear Programming and Extensions* (Princeton University Press, Princeton, NJ, 1963).
98. G.B. Dantzig and P. Glynn, "Parallel processors for planning under uncertainty," *Annals of Operations Research* 22 (1990) pp. 1–21.
99. G.B. Dantzig and G. Infanger, "Large-scale stochastic linear programs—Importance sampling and Benders decomposition" in: C. Brezinski and U. Kulisch, Eds., *Computational and applied mathematics, I (Dublin, 1991)* (North-Holland, Amsterdam, 1991) pp. 111–120.
100. G.B. Dantzig and A. Madansky, "On the solution of two-stage linear programs under uncertainty," Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, (University of California Press, Berkeley, CA, 1961).
101. G.B. Dantzig and A. Wald, "On the fundamental lemma of Neyman and Pearson," *The Annals of Mathematical Statistics* 22 (1951) pp. 87–93.
102. G.B. Dantzig and P. Wolfe, "The decomposition principle for linear programs," *Operations Research* 8 (1960) pp. 101–111.
103. D. Dawson and A. Sankoff, "An inequality for probabilities," *Proceedings of the American Mathematical Society* 18 (1967) pp. 504–507.
104. I. Deák, "Three-digit accurate multiple normal probabilities," *Numerische Mathematik* 35 (1980) pp. 369–380.
105. I. Deák, "Multidimensional integration and stochastic programming," in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 187–200.
106. I. Deák, *Random Number Generators and Simulation* (Akadémiai Kiadó, Budapest, 1990).
107. D.P. de Farias and B. Van Roy, "On constraint sampling in the linear programming approach to approximate dynamic programming," *Mathematics of Operations Research* 29 (2004) pp. 462–478.
108. M.H. DeGroot, *Optimal Statistical Decisions* (McGraw-Hill, New York, NY, 1970).
109. M.A.H. Dempster, "Introduction to Stochastic Programming" in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980) pp. 3–59.
110. M.A.H. Dempster, "The expected value of perfect information in the optimal evolution of stochastic problems" in: M. Arato, D. Vermes, and A.V. Balakrishnan, Eds., *Stochastic Differential Systems* (Lecture Notes in Information and Control, Vol. 36, 1981) pp. 25–40.
111. M.A.H. Dempster, "On stochastic programming II: dynamic problems under risk," *Stochastics* 25 (1988) pp. 15–42.
112. M.A.H. Dempster, "Sequential importance sampling algorithms for dynamic stochastic programming," *Journal of Mathematical Sciences* 133 (2006), pp. 1422–1444.
113. M.A.H. Dempster and A. Papagaki-Papoulias, "Computational experience with an approximate method for the distribution problem" in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980) pp. 223–243.

114. V.F. Demyanov and L.V. Vasiliev, *Nedifferentiiruemaya optimizatsiya (Nondifferentiable optimization)* (Nauka, Moscow, 1981).
115. D. Dentcheva and A. Ruszczyński, “Robust stochastic dominance and its application to risk-averse optimization,” *Mathematical Programming, Series B* 123 (2010) pp. 85–100.
116. C.J. Donohue, “Stochastic Network Programming And The Dynamic Vehicle Allocation Problem,” Ph.D. Dissertation, University of Michigan (Ann Arbor, MI, 1996).
117. Christopher J. Donohue and John R. Birge, “The Abridged Nested Decomposition Method for Multistage Stochastic Programs,” *Algorithmic Operations Research* 1 (2006) pp. 20–30.
118. J.H. Dulá, “An upper bound on the expectation of simplicial functions of multivariate random variables,” *Mathematical Programming* 55 (1991) pp. 69–80.
119. V. Dupač, “A dynamic stochastic approximation method,” *Annals of Mathematical Statistics* 6 (1965) pp. 1695–1702.
120. J. Dupačová, “Minimax stochastic programs with nonconvex nonseparable penalty functions” in: A. Prékopa, Ed., *Progress in Operations Research* (Janos Bolyai Math. Soc., 1976) pp. 303–316.
121. J. Dupačová, “The minimax approach to stochastic linear programming and the moment problem,” *Ekonom.-Mat. Obzor* 13 (1977) pp. 297–307.
122. J. Dupačová, “Stability in stochastic programming with recourse-contaminated distributions,” *Mathematical Programming Study* 28 (1984) pp. 72–83.
123. J. Dupačová, “Stability and sensitivity analysis for stochastic programming,” *Annals of Operations Research* 27 (1990) pp. 115–142.
124. J. Dupačová, N. Gröwe-Kuska and W. Römisch, “Scenario reduction in stochastic programming: An approach using probability metrics,” *Mathematical Programming, Ser. A* 95 (2003) pp. 493–511.
125. J. Dupačová and R.J-B Wets, “Asymptotic behavior of statistical estimators and of optimal solutions of stochastic optimization problems,” *Annals of Statistics* 16 (1988) pp. 1517–1549.
126. S. Dye, L. Stougie, and A. Tomasdard, “The stochastic single resource service-provision problem,” *Naval Research Logistics* 50 (2003) pp. 869887.
127. M. Dyer, R. Kannan, and L. Stougie, “A simple randomised algorithm for convex optimisation,” SPORReport 2002-05, Dept. of Mathematics and Computer Science, Eindhoven Technical University, Eindhoven, 2002.
128. M. Dyer and L. Stougie, “Computational complexity of stochastic programming problems,” *Mathematical Programming, Ser. A* 106 (2006) pp. 423–432.
129. B.C. Eaves and W.I. Zangwill, “Generalized cutting plane algorithms,” *SIAM J. Control* 9 (1971) pp. 529–542.
130. N.C.P. Edirisinghe, “Essays on Bounding Stochastic Programming Problems,” Ph.D. Dissertation, The University of British Columbia (Vancouver, BC, 1991).
131. N.C.P. Edirisinghe, “New second-order bounds on the expectation of saddle functions with applications to stochastic linear programming,” *Operations Research* 44 (1996) pp. 909–922.
132. H.P. Edmundson, “Bounds on the expectation of a convex function of a random variable,” RAND Corporation Paper 982, Santa Monica, CA (1956).
133. M. Eisner and P. Olsen, “Duality for stochastic programming interpreted as l.p. in L_p -space,” *SIAM Journal of Applied Mathematics* 28 (1975) pp. 779–792.
134. G.D. Eppen, R.K. Martin, and L. Schrage, “A scenario approach to capacity planning,” *Operations Research* 37 (1989) pp. 517–527.
135. Epstein, L. and S. Zin, “Substitution, risk aversion and the temporal behavior of consumption and asset returns: A theoretical framework,” *Econometrica* 57 (1989), pp. 937–969.
136. Y. Ermoliev, “On the stochastic quasigradient method and quasi-Feyer sequences,” *Kibernetika* 5 (2) (1969) pp. 73–83 (in Russian; also published in English as *Cybernetics* 5 (1969) pp. 208–220).
137. Y. Ermoliev, *Methods of Stochastic Programming* (Nauka, Moscow (in Russian) 1976).
138. Y. Ermoliev, “Stochastic quasigradient methods and their applications to systems optimization,” *Stochastics* 9 (1983) pp. 1–36.
139. Y. Ermoliev, “Stochastic quasigradient methods” in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 141–186.

140. Y. Ermoliev, A. Gaivoronski, and C. Nedeva, "Stochastic optimization problems with partially known distribution functions," *SIAM Journal on Control and Optimization* 23 (1985) pp. 377–394.
141. Y. Ermoliev and R. Wets, "Introduction" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988).
142. L.F. Escudero, P.V. Kamesam, A.J. King, and R.J-B Wets, "Production planning via scenario modeling," *Annals of Operations Research* 43 (1993) pp. 311–335.
143. W. Feller, *An Introduction to Probability Theory and Its Applications* (John Wiley, Inc., New York, NY, 1971).
144. A. Ferguson and G.B. Dantzig, "The allocation of aircraft to routes: an example of linear programming under uncertain demands," *Management Science* 3 (1956) pp. 45–73.
145. S.D. Flåm, "Nonanticipativity in stochastic programming," *Journal of Optimization Theory and Applications* 46 (1985) pp. 23–30.
146. S.D. Flåm, "Asymptotically stable solutions to stochastic problems of Bolza" in: F. Archetti, G. Di Pillo, and M Lucertini, Eds., *Stochastic Programming* (Lecture Notes in Information and Control 76, 1986) pp. 184–193.
147. A.D. Flaxman, A. Frieze, and M. Krivelevich, "On the random 2-stage minimum spanning tree," *Random Structures and Algorithms* 28 (2006) pp. 24–36.
148. A. Flaxman, A.T. Kalai, and H.B. McMahan, "Online convex optimization in the bandit setting: gradient descent without a gradient," In *Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2005, Vancouver, British Columbia, Canada, January 23-25, 2005* (SIAM, Philadelphia, PA, 2005) pp. 385–394.
149. W. Fleming and R. Rischel, *Deterministic and Stochastic Control* (Springer-Verlag, New York, NY, 1975).
150. R. Fourer, "A simplex algorithm for piecewise-linear programming. I: derivation and proof," *Mathematical Programming* 33 (1985) pp. 204–233.
151. R. Fourer, "A simplex algorithm for piecewise-linear programming. II: finiteness, feasibility, and degeneracy," *Mathematical Programming* 41 (1988) pp. 281–315.
152. R. Fourer, D.M. Gay, and B.W. Kernighan, *AMPL: A Modeling Language for Mathematical Programming* (Scientific Press, South San Francisco, CA, 1993).
153. B. Fox, "Implementation and relative efficiency of quasirandom sequence generators," *ACM Transactions on Mathematical Software* 12 (1986) pp. 362–376.
154. L. Frantzeskakis and W. Powell, "A successive linear approximation procedure for stochastic, dynamic vehicle allocation problems," *Transportation Science* 24 (1990) pp. 40–57.
155. L.F. Frantzeskakis and W.B. Powell, "Bounding procedures for multistage stochastic dynamic networks," *Networks* 23 (1993) pp. 575–595.
156. K. Frauendorfer, "Solving SLP recourse problems: The case of stochastic technology matrix, RHS, and objective," *Proceedings of 13th IFIP Conference on System Modelling and Optimization* (Springer-Verlag, Berlin, 1988a).
157. K. Frauendorfer, "Solving S.L.P. recourse problems with arbitrary multivariate distributions – the dependent case," *Mathematics of Operations Research* 13 (1988b) pp. 377–394.
158. K. Frauendorfer, "A simplicial approximation scheme for convex two-stage stochastic programming problems," Manuskrifte, Institut für Operations Research, University of Zurich (Zurich, 1989).
159. K. Frauendorfer, *Stochastic Two-Stage Programming* (Lecture Notes in Economics and Mathematical Systems 392, 1992).
160. K. Frauendorfer and P. Kall, "A solution method for SLP recourse problems with arbitrary multivariate distributions—the independent case," *Problems in Control and Information Theory* 17 (1988) pp. 177–205.
161. A.A. Gaivoronski, "Implementation of stochastic quasigradient methods" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 313–352.
162. J. Galambos, *The Asymptotic Theory of Extreme Order Statistics* (John Wiley, Inc., New York, 1978).

163. S.J. Gartska, "An economic interpretation of stochastic programs," *Mathematical Programming* 18 (1980) pp. 62–67.
164. S.J. Gartska and D. Rutenberg, "Computation in discrete stochastic programs with recourse," *Operations Research* 21 (1973) pp. 112–122.
165. S.J. Gartska and R.J-B Wets, "On decision rules in stochastic programming," *Mathematical Programming* 7 (1974) pp. 117–143.
166. H.I. Gassmann, "Conditional probability and conditional expectation of a random vector" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 237–254.
167. H.I. Gassmann, "Optimal harvest of a forest in the presence of uncertainty," *Canadian Journal of Forest Research* 19 (1989) pp. 1267–1274.
168. H.I. Gassmann, "MSLiP: a computer code for the multistage stochastic linear programming problem," *Mathematical Programming* 47 (1990) pp. 407–423.
169. H.I. Gassmann and W.T. Ziemba, "A tight upper bound for the expectation of a convex function of a multivariate random variable," *Mathematical Programming Study* 27 (1986) pp. 39–53.
170. D.M. Gay, "A variant of Karmarkar's linear programming algorithm for problems in standard form," *Mathematical Programming* 37 (1987) pp. 81–90.
171. M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing," *European Journal of Operational Research* 88 (1996) pp. 3–12.
172. M. Gendreau, G. Laporte, and R. Séguin, "An exact algorithm for the vehicle routing problem with stochastic demands and customers," *Transportation Science* 29 (1995) pp. 143–155.
173. A.M. Geoffrion, "Elements of large-scale mathematical programming," *Management Science* 16 (1970) pp. 652–675.
174. A.M. Geoffrion, "Duality in nonlinear programming: a simplified applications-oriented development," *SIAM Rev.* 13 (1971) pp. 1–37.
175. I. Gilboa and D. Schmeidler, "Maxmin expected utility with non-unique prior," *Journal of Mathematical Economics* 18 (1989) pp. 141–153.
176. C.R. Glassey, "Nested decomposition and multistage linear programs," *Management Science* 20 (1973) pp. 282–292.
177. J. Gondzio and A. Grothey, "Exploiting structure in parallel implementation of interior point methods for optimization," *Computational Management Science* 6 (2009) pp. 135–160.
178. R.C. Grinold, "A new approach to multistage stochastic linear programs," *Mathematical Programming Study* 6 (1976) pp. 19–29.
179. R.C. Grinold, "Model building techniques for the correction of end effects in multistage convex programs," *Operations Research* 31 (1983) pp. 407–431.
180. R.C. Grinold, "Infinite horizon stochastic programs," *SIAM Journal on Control and Optimization* 24 (1986) pp. 1246–1260.
181. A. Gupta, M. Pál, R. Ravi, and A. Sinha, "Boosted sampling: Approximation algorithms for stochastic optimization problems," in: L. Babai, Ed., *Proc. 36th Annual ACM Symp. Theory Comput., Chicago, IL, USA, June 13–16, 2004* (ACM Press, New York, 2004) pp. 417–425.
182. A. Gupta, M. Pál, R. Ravi, and A. Sinha, "What about Wednesday? Approximation algorithms for multistage stochastic optimization," in C. Chekuri, K. Jansen, J.D.P. Rolim, and L. Trevisan, Eds., *Approximation, Randomization and Combinatorial Optimization, Algorithms and Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2005, and 9th International Workshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22–24, 2005, Proceedings, Lecture Notes in Computer Science* 3624 (Springer, Berlin, 2005) pp. 86–98.
183. A. Gupta, R. Ravi, and A. Sinha, "LP rounding approximation algorithms for stochastic network design," *Mathematics of Operations Research* 32 (2007) pp. 345–364.
184. L.P. Hansen and T. Sargent, "Discounted linear exponential quadratic gaussian control," *IEEE Transactions on Automatic Control* 40 (1995) pp. 968–971.
185. J.M. Harrison, *Brownian Motion and Stochastic Flow Systems* (John Wiley, Inc., New York, NY, 1985).

186. J.M. Harrison and D.M. Kreps, "Martingales and arbitrage in multiperiod securities markets," *Journal of Economic Theory* 20 (1979) pp. 381–408.
187. J.M. Harrison and L.M. Wein, "Scheduling networks of queues: Heavy traffic analysis of a two-station closed network," *Operations Research* 38 (1990) pp. 1052–1064.
188. D. Haugland and S.W. Wallace, "Solving many linear programs that differ only in the right-hand side," *European Journal of Operational Research* 37 (1988) pp. 318–324.
189. E. Hazan, A. Kalai, S. Kale, and A. Agarwal, "Logarithmic regret algorithms for online convex optimization," in: G. Lugosi and H-U. Simon, Eds., *Learning Theory, 19th Annual Conference on Learning Theory, COLT 2006, Pittsburgh, PA, USA, June 22-25, 2006, Proceedings. Lecture Notes in Computer Science* 4005 (Springer, Berlin, 2006) pp. 499–513.
190. R. Hemmecke and R. Schultz, "Decomposition of test sets in stochastic integer programming," *Mathematical Programming* 94 (2003) pp. 323–341.
191. D.P. Heyman and M.J. Sobel, *Stochastic Models in Operations Research, Volume II, Stochastic Optimization* (McGraw-Hill, New York, NY, 1984).
192. J. Higle and S. Sen, "Statistical verification of optimality conditions for stochastic programs with recourse," *Annals of Operations Research* 30 (1991a) pp. 215–240.
193. J. Higle and S. Sen, "Stochastic decomposition: an algorithm for two stage linear programs with recourse," *Mathematics of Operations Research* 16 (1991b) pp. 650–669.
194. J.L. Higle and S. Sen, *Stochastic Decomposition: A Statistical Method for Large Scale Stochastic Linear Programming* (Kluwer Academic Publisher, Dordrecht, 1996).
195. J.-B. Hiriart-Urruty, "Conditions nécessaires d'optimalité pour un programme stochastique avec recours," *SIAM Journal on Control and Optimization* 16 (1978) pp. 317–329.
196. C. Hjörung and J. Holt, "New optimality cuts for a single vehicle stochastic routing problem," *Annals of Operations Research* 86 (1999), pp. 569–584.
197. J.K. Ho and A.S. Manne, "Nested decomposition for dynamic models," *Mathematical Programming* 6 (1974) pp. 121–140.
198. W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American Statistical Association* 58 (1963) pp. 13–30.
199. A. Hogan, J. Morris, and H. Thompson, "Decision problems under risk and chance constrained programming: dilemmas in the transition," *Management Science* 27 (1981) pp. 698–716.
200. A. Hogan, J. Morris, and H. Thompson, "Reply to Professors Charnes and Cooper concerning their response to 'Decision problems under risk and chance constrained programming: dilemmas in the transition,'" *Management Science* 30 (1984) pp. 258–259.
201. R.A. Howard, *Dynamic Programming and Markov Processes* (MIT Press, Cambridge, MA, 1960).
202. K. Høyland and S.W. Wallace, "Generating Scenario Trees for Multistage Decision Problems," *Management Science* 47 (2001) pp. 295–307.
203. C.C. Huang, W.T. Ziemba, and A. Ben-Tal, "Bounds on the expectation of a convex function of a random variable: with applications to stochastic programming," *Operations Research* 25 (1977) pp. 315–325.
204. P.J. Huber, "The behavior of maximum likelihood estimates under nonstandard conditions," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, (University of California, Berkeley, CA, 1967).
205. P.J. Huber, *Robust Statistics*, John Wiley, 1981.
206. J.C. Hull, *Options, Futures and Other Derivatives*, third edition, (Prentice-Hall, Upper Saddle River, NJ, 1997).
207. G. Infanger, "Monte Carlo (importance) sampling within a Benders decomposition algorithm for stochastic linear programs; Extended version: including results of large-scale problems," Technical Report SOL 91-6, Systems Optimization Laboratory, Stanford University (Stanford, CA, 1991).
208. G. Infanger, *Planning under Uncertainty: Solving Large-Scale Stochastic Linear Programs* (Boyd and Fraser, Danvers, MA, 1994).
209. R. Jagganathan, "A minimax procedure for a class of linear programs under uncertainty," *Operations Research* 25 (1977) pp. 173–177.

210. R. Jagganathan, "Use of sample information in stochastic recourse and chance-constrained programming models," *Management Science* 31 (1985) pp. 96–108.
211. R. Jagganathan, "Linear programming with stochastic processes as parameters as applied to production planning," *Annals of Operations Research* 30 (1991) pp. 107–114.
212. P. Jaillet, "A priori solution of a traveling salesman problem in which a random subset of the customers are visited," *Operations Research* 36 (1988) pp. 929–936.
213. R.A. Jarrow and A. Rudd, *Option Pricing* (Irwin, Homewood, IL, 1983).
214. S. Jasin and S. Kumar, "A re-solving heuristic with bounded revenue loss for network revenue management with customer choice," Working Paper, Stanford University (Stanford, CA, 2010).
215. J.L. Jensen, "Sur les fonctions convexes et les inégalités entre les valeurs moyennes," *Acta Math.* 30 (1906) pp. 175–193.
216. P. Kall, *Stochastic Linear Programming* (Springer-Verlag, Berlin, 1976).
217. P. Kall, "Computational methods for solving two-stage stochastic linear programming problems," *Journal of Applied Mathematics and Physics* 30 (1979) pp. 261–271.
218. P. Kall, "Stochastic programs with recourse: an upper bound and the related moment problem," *Zeitschrift für Operations Research* 31 (1987) pp. A119–A141.
219. P. Kall, "An upper bound for stochastic linear programming using first and total second moments," *Annals of Operations Research* 30 (1991) pp. 267–276.
220. P. Kall and J. Mayer, "SLP-IOR: an interactive model management system for stochastic linear programs," *Mathematical Programming* 75 (1996) pp. 221–240.
221. P. Kall and D. Stoyan, "Solving stochastic programming problems with recourse including error bounds," *Math. Operationsforsch. Statist. Ser. Optim.* 13 (1982) pp. 431–447.
222. P. Kall and S.W. Wallace, *Stochastic Programming* (John Wiley and Sons, Chichester, UK, 1994).
223. J.G. Kallberg, R.W. White, and W.T. Ziemba, "Short term financial planning under uncertainty," *Management Science* 28 (1982) pp. 670–682.
224. J.G. Kallberg and W.T. Ziemba, "Comparison of alternative utility functions in portfolio selection problems," *Management Science* 29 (1983) pp. 1257–1276.
225. M. Kallio and E. Porteus, "Decomposition of arborescent linear programs," *Mathematical Programming* 13 (1977) pp. 348–356.
226. R.E. Kalman, *Topics in Mathematical System Theory* (McGraw-Hill, New York, NY, 1969).
227. E. Kao and M. Queyranne, "Budgeting costs of nursing in a hospital," *Management Science* 31 (1985) pp. 608–621.
228. N. Karmarkar, "A new polynomial-time algorithm for linear programming," *Combinatorica* 4 (1984) pp. 373–395.
229. A. Karr, "Extreme points of certain sets of probability measure, with applications," *Mathematics of Operations Research* 8 (1983) pp. 74–85.
230. J. Kemperman, "The general moment problem, a geometric approach," *Annals of Mathematical Statistics* 39 (1968) pp. 93–122.
231. A.I. Kibzun and Y.S. Kan, *Stochastic Programming Problems with Probability and Quantile Functions* (John Wiley Inc., Chichester, UK, 1996).
232. A.I. Kibzun and V.Yu. Kurbakovskiy, "Guaranteeing approach to solving quantile optimization problems," *Annals of Operations Research* 30 (1991) pp. 81–93.
233. A. King, "Finite generation method" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988a) pp. 295–312.
234. A. King, "Stochastic programming problems: Examples from the literature" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988b) pp. 543–567.
235. A. King and R.T. Rockafellar, "Asymptotic theory for solutions in generalized M-estimation and stochastic programming," *Mathematics of Operations Research* 18 (1993) pp. 148–162.
236. A.J. King and R.J-B Wets, "Epiconsistency of convex stochastic programs," *Stochastics and Stochastics Reports* 34 (1991) pp. 83–92.
237. K.C. Kiwiel, "An aggregate subgradient method for nonsmooth convex minimization," *Mathematical Programming* 27 (1983) pp. 320–341.

238. P. Klaassen, "Financial asset-pricing theory and stochastic programming models for asset/liability management: a synthesis," *Management Science* 44 (1998) pp. 31–48.
239. W.K. Klein Haneveld, *Duality in Stochastic Linear and Dynamic Programming* (Lecture Notes in Economics and Mathematical Systems 274, Springer-Verlag, Berlin, 1985).
240. W.K. Klein Haneveld, "Robustness against dependence in PERT: an application of duality and distributions with known marginals," *Mathematical Programming Study* 27 (1986) pp. 153–182.
241. N. Kong, A.J. Schaefer, and B.K. Hunsaker, "Two-stage integer programs with stochastic right-hand sides - A superadditive dual approach," *Mathematical Programming* 108 (2006) pp. 275–296.
242. R. Kouwenberg, "Scenario generation and stochastic programming models for asset-liability management," *European Journal of Operations Research* 134 (2001) pp. 279–292.
243. M.G. Krein and A.A. Nudel'man, *The Markov Moment Problem and Extremal Problems* (Translations of Mathematical Monographs 50, 1977).
244. D.M. Kreps and E.L. Porteus, "Temporal von Neumann-Morgenstern and Induced Preferences," *Journal Of Economic Theory* 20(1979) pp. 81–10.
245. Daniel Kuhn, "Aggregation and Discretization in Multistage Stochastic Programming," *Mathematical Programming A* 113 (2008) pp. 61–94.
246. H. Kushner, *Introduction to Stochastic Control* (Holt, New York, NY, 1971).
247. M. Kusy and W.T. Ziemba, "A bank asset and liability management model," *Operations Research* 34 (1986) pp. 356–376.
248. B.J. Lageweg, J.K. Lenstra, A.H.G. Rinnooy Kan, and L. Stougie, "Stochastic integer programming by dynamic programming" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 403–412.
249. G. Laporte and F.V. Louveaux, "The integer L-shaped method for stochastic integer programs with complete recourse," *Operations Research Letters* 13 (1993) pp. 133–142.
250. G. Laporte, F.V. Louveaux, and H. Mercure, "Models and exact solutions for a class of stochastic location-routing problems," *European Journal of Operational Research* 39 (1989) pp. 71–78.
251. G. Laporte, F.V. Louveaux, and H. Mercure, "An exact solution for the a priori optimization of the probabilistic traveling salesman problem," *Operations Research* 42 (1994) pp. 543–549.
252. G. Laporte, F.V. Louveaux, and L. Van Hamme, "Exact solution to a location problem with stochastic demands," *Transportation Science* 28 (1994) pp. 95–103.
253. G. Laporte, F.V. Louveaux and L. Van hamme, "An integer L-shaped algorithm for the capacitated vehicle routing problem with stochastic demands," *Operations Research* 50 (2002) pp. 415–423.
254. L. Lasdon, *Optimization Theory for Large Systems* (Macmillan, New York, NY, 1970).
255. C. Lemaréchal, "Bundle methods in nonsmooth optimization" in: *Nonsmooth optimization (Proc. IIASA Workshop)* (Pergamon, Oxford-Elmsford, New York, NY, 1978) pp. 79–102.
256. J. Linderoth and S. Wright, "Decomposition algorithms for stochastic programming on a computational grid," *Computational Optimization and its Applications* 24 (2003) pp. 207–250.
257. A.W. Lo, "Semi-parametric upper bounds for option prices and expected payoffs," *Journal of Financial Economics* 19 (1987) pp. 373–387.
258. F.V. Louveaux, "Piecewise convex programs," *Mathematical Programming* 15 (1978) pp. 53–62.
259. F.V. Louveaux, "A solution method for multistage stochastic programs with recourse with application to an energy investment problem," *Operations Research* 28 (1980) pp. 889–902.
260. F.V. Louveaux, "Multistage stochastic programs with block-separable recourse," *Mathematical Programming Study* 28 (1986) pp. 48–62.
261. F.V. Louveaux and D. Peeters, "A dual-based procedure for stochastic facility location," *Operations Research* 40 (1992) pp. 564–573.

262. F. Louveaux and R. Schultz, "Stochastic integer programming," in: A. Ruszczyński and A. Shapiro, Eds., *Handbooks in Operations Research and Management Science 10*, (Elsevier, Amsterdam, 2003) pp. 213–266.
263. F.V. Louveaux and Y. Smeers, "Optimal investments for electricity generation: a stochastic model and a test-problem" in: *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 33–64.
264. F.V. Louveaux and Y. Smeers, "Stochastic optimization for the introduction of a new energy technology," *Stochastics (to appear)* (2011).
265. F.V. Louveaux and M. van der Vlerk, "Stochastic programming with simple integer recourse," *Mathematical Programming* 61 (1993) pp. 301–325.
266. J. Luedtke and S. Ahmed, "A sample approximation approach for optimization with probabilistic constraints," *SIAM Journal on Optimization* 19 (2008) pp. 674–699.
267. A. Madansky, "Bounds on the expectation of a convex function of a multivariate random variable," *Annals of Mathematical Statistics* 30 (1959) pp. 743–746.
268. A. Madansky, "Inequalities for stochastic linear programming problems," *Management Science* 6 (1960) pp. 197–204.
269. M. Maddox and J.R. Birge, "Bounds on the distribution of tardiness in a PERT network," Technical Report, Department of Industrial and Operations Engineering, University of Michigan (Ann Arbor, MI, 1991).
270. W. Mak, D.P. Morton, and R.K. Wood, "Monte Carlo bounding techniques for determining solution quality in stochastic programs," *Operations Research Letters* 24 (1999) pp. 47–56.
271. O. Mangasarian and J.B. Rosen, "Inequalities for stochastic nonlinear programming problems," *Operations Research* 12 (1964) pp. 143–154.
272. A.S. Manne, "Waiting for the breeder" in: *Review of Economic Studies Symposium* (1974) pp. 47–65.
273. A.S. Manne and R. Richels, *Buying Greenhouse Insurance—The Economic Costs of Carbon Dioxide Emission Limits* (MIT Press, Cambridge, MA, 1992).
274. H.M. Markowitz, *Portfolio Selection; Efficient Diversification of Investments* (John Wiley, Inc., New York, NY, 1959).
275. K. Marti, "Approximationen von Entscheidungsproblemen mit linearer Ergebnisfunktion und positiv homogener, subadditiver Verlusfunktion," *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete* 31 (1975) pp. 203–233.
276. K. Marti, *Descent Directions and Efficient Solutions in Discretely Distributed Stochastic Programs*, (Lecture Notes in Economics and Mathematical Systems 299, Springer-Verlag, Berlin, 1988).
277. R.K. Martin, *Large Scale Linear and Integer Optimization: A Unified Approach* (Kluwer Academic, Boston, 1999).
278. L. McKenzie, "Turnpike theory," *Econometrica* 44 (1976) pp. 841–864.
279. R.C. Merton, "On the pricing of corporate debt: the risk structure of interest rates," *The Journal of Finance* 29 (1974) pp. 449–470 (Papers and Proceedings of the Thirty-Second Annual Meeting of the American Finance Association, New York, New York, December 28–30, 1973).
280. P. Michel and J.-P. Penot, "Calcul sous-différentiel pour des fonctions lipschitziennes et non lipschitziennes," *Comptes Rendus des Séances de l'Académie des Sciences Paris. Serie 1. Mathématique* 298 (1984) pp. 269–272.
281. J. Miller and H. Wagner, "Chance-constrained programming with joint chance constraints," *Operations Research* 12 (1965) pp. 930–945.
282. G.J. Minty, "On the maximal domain of a 'monotone' function," *Michigan Mathematics Journal* 8 (1961) pp. 135–137.
283. F. Mirzachmedov and S. Uriasiev, "Adaptive step-size control for stochastic optimization algorithm," *Zhurnal vycisl. mat. i mat. fiz.* 6 (1983) pp. 1314–1325 (in Russian).
284. B. Mordukhovich, "Approximation methods and extremum conditions in nonsmooth control systems," *Soviet Mathematics Doklady* 36 (1988) pp. 164–168.
285. D.P. Morton, "An enhanced decomposition algorithm for multistage stochastic hydroelectric scheduling," *Annals of Operations Research* 64 (1996) pp. 211–235.

286. D.P. Morton, "Stopping rules for a class of sampling-based stochastic programming algorithms," *Operations Research* 46 (1998) pp. 710–718.
287. J.M. Mulvey and A. Ruszczyński, "A new scenario decomposition method for large scale stochastic optimization," *Operations Research* 43 (1995) pp. 477–490.
288. J.M. Mulvey and H. Vladimirou, "Stochastic network optimization models for investment planning," *Annals of Operations Research* 20 (1989) pp. 187–217.
289. J.M. Mulvey and H. Vladimirou, "Applying the progressive hedging algorithm to stochastic generalized networks," *Annals of Operations Research* 31 (1991a) pp. 399–424.
290. J.M. Mulvey and H. Vladimirou, "Solving multistage stochastic networks: an application of scenario aggregation," *Networks* 21 (1991b) pp. 619–643.
291. J.M. Mulvey and H. Vladimirou, "Stochastic network programming for financial planning problems," *Management Science* 38 (1992) pp. 1642–1664.
292. K.G. Murty, "Linear programming under uncertainty: a basic property of the optimal solution," *Z. Wahrscheinlichkeitstheorie und Verw. Gebiete* 10 (1968) pp. 284–288.
293. K.G. Murty, *Linear Programming* (John Wiley, Inc., New York, NY, 1983).
294. J.L. Nazareth and R.J-B Wets, "Algorithms for stochastic programs: the case of nonstochastic tenders," *Mathematical Programming Study* 28 (1986) pp. 1–28.
295. G.L. Nemhauser and L.A. Wolsey, *Integer and Combinatorial Optimization* (Wiley-Interscience, New York, NY, 1988).
296. A. Nemirovski and A. Shapiro, "Convex approximations of chance constrained programs," *SIAM Journal on Optimization* 17 (2006) 969–996.
297. Yu. Nesterov and J.-Ph. Vial, "Confidence level solutions for stochastic programming," *Automatica* 44 (2008), 1559–1568.
298. H. Niederreiter, "Quasi-Monte Carlo methods and pseudorandom numbers," *Bulletin of the American Mathematical Society* 84 (1978) pp. 957–1041.
299. S.S. Nielsen and S.A. Zenios, "A massively parallel algorithm for nonlinear stochastic network problems," *Operations Research* 41 (1993a) pp. 319–337.
300. S.S. Nielsen and S.A. Zenios, "Proximal minimizations with D -functions and the massively parallel solution of linear stochastic network programs," *International Journal of Supercomputing and Applications* 7 (1993b) pp. 349–364.
301. M.-C. Noël and Y. Smeers, "Nested decomposition of multistage nonlinear programs with recourse," *Mathematical Programming* 37 (1987) pp. 131–152.
302. V.I. Norkin, Y.M. Ermoliev, and A. Ruszczyński, "On optimal allocation of indivisibles under uncertainty," *Operations Research* 46 (1998) pp. 381–395.
303. V.I. Norkin, G.Ch. Pflug, and A. Ruszczyński, "A branch and bound method for stochastic global optimization," *Mathematical Programming* 83 (1998) pp. 425–450.
304. L. Ntaimo and S. Sen, "A Branch-and-Cut algorithm for two-stage stochastic mixed-binary programs with continuous first-stage variables," *International Journal of Computational Science and Engineering* 3 (2008a) pp. 231–241.
305. L. Ntaimo and S. Sen, "A comparative study of decomposition algorithms for stochastic combinatorial optimization," *Computational Optimization and Applications* 40 (2008b) pp. 299–319.
306. S. Parikh, *Lecture Notes on Stochastic Programming* (University of California, Berkeley, CA, 1968).
307. M.V.F. Pereira and L.M.V.G. Pinto, "Stochastic optimization of a multireservoir hydroelectric system—A decomposition approach," *Water Resources Research* 21 (1985) pp. 779–792.
308. M.V.F. Pereira and L.M.V.G. Pinto, "Multistage Stochastic Optimization Applied to Energy Planning," *Mathematical Programming* 52 (1991) pp. 359–375.
309. G.Ch. Pflug, "Stepsize rules, stopping times and their implementation in stochastic quasigradient algorithms" in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 353–372.
310. G.Ch. Pflug, "Scenario tree generation for multiperiod financial optimization by optimal discretization," *Mathematical Programming, Ser. B* 89 (2001) pp. 251–271.
311. G.Ch. Pflug and L. Halada, "A note on the recursive and parallel structure of the Birge and Qi factorization," *Computational Optimization and Applications* 24 (2003) pp. 251–265.

312. J. Pintér, "Deterministic approximations of probability inequalities," *ZOR—Methods and Models of Operations Research, Series Theory* 33 (1989) pp. 219–239.
313. A.B. Philpott and Z. Guan, "On the convergence of stochastic dual dynamic programming and related methods," *Operations Research Letters* 36 (2008) pp. 450–455.
314. E.L. Plambeck, B-R. Fu, S.M. Robinson, and R. Suri, "Sample-path optimization of convex stochastic performance functions," *Mathematical Programming* 75 (1996) pp. 137–176.
315. W.B. Powell, "A comparative review of alternative algorithms for the dynamic vehicle allocation program" in: B. Golden and A. Assad, Eds., *Vehicle Routing: Methods and Studies* (North-Holland, Amsterdam, 1988).
316. W.B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality* (Wiley, New York, 2007).
317. A. Prékopa, "Logarithmic concave measures with application to stochastic programming," *Acta Sci. Math. (Szeged)* 32 (1971) pp. 301–316.
318. A. Prékopa, "Contributions to the theory of stochastic programs," *Mathematical Programming* 4 (1973) pp. 202–221.
319. A. Prékopa, "Programming under probabilistic constraints with a random technology matrix," *Mathematische Operationsforschung und Statistik* 5 (1974) pp. 109–116.
320. A. Prékopa, "Logarithmically concave measures and related topics" in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980).
321. A. Prékopa, "Boole-Bonferroni inequalities and linear programming," *Operations Research* 36 (1988) pp. 145–162.
322. A. Prékopa, *Stochastic Programming* (Kluwer Academic Publishers, Dordrecht, Netherlands, 1995).
323. A. Prékopa and T. Szántai, "On optimal regulation of a storage level with application to the water level regulation of a lake," *Survey of Mathematical Programming (Proc. Ninth Internat. Math. Programming Sympos., Budapest, 1976)*, Vol. 2 (North-Holland, Amsterdam, 1976).
324. H.N. Psaraftis, "On the practical importance of asymptotic optimality in certain heuristic algorithms," *Networks* (1984) pp. 587–596.
325. L. Qi, "Forest iteration method for stochastic transportation problem," *Mathematical Programming Study* (1985) pp. 142–163.
326. L. Qi, "An alternating method for stochastic linear programming with simple recourse," *Stochastic Processes and Their Applications* 841 (1986) pp. 183–190.
327. H. Raiffa, *Decision Analysis* (Addison-Wesley, Reading, MA, 1968).
328. H. Raiffa and R. Schlaifer, *Applied Statistical Decision Theory* (Harvard University, Boston, MA, 1961).
329. R. Ravi and A. Sinha, "Hedging uncertainty: Approximation algorithms for stochastic optimization problems," *Mathematical Programming Ser. A* 108 (2006) pp. 97–114.
330. W. Rei, J.-F. Cordeau, M. Gendreau and P. Soriano, "Accelerating Benders' decomposition by local branching," *INFORMS Journal on Computing* 21 (2009) pp. 333–345.
331. H. Robbins and S. Monro, "A stochastic approximation method," *Annals of Mathematical Statistics* 22 (1951) pp. 400–407.
332. S.M. Robinson and R.J-B Wets, "Stability in two-stage stochastic programming," *SIAM Journal on Control and Optimization* 25 (1987) pp. 1409–1416.
333. R.T. Rockafellar, *Convex Analysis* (Princeton University Press, Princeton, NJ, 1969).
334. R.T. Rockafellar, *Conjugate Duality and Optimization* (Society for Industrial and Applied Mathematics, Philadelphia, PA, 1974).
335. R.T. Rockafellar, "Monotone operators and the proximal point algorithm," *SIAM Journal on Control and Optimization* 14 (1976a) pp. 877–898.
336. R.T. Rockafellar, *Integral Functionals, Normal Integrands and Measurable Selections* (Lecture Notes in Mathematics 543, 1976b).
337. R.T. Rockafellar and S. Uryasev, "Optimization of Conditional Value-At-Risk," *The Journal of Risk* 2:3 (2000) pp. 21–41.
338. R.T. Rockafellar and S. Uryasev, "Conditional Value-at-Risk for general loss distributions," *Journal of Banking and Finance* 26 (2002) pp. 1443–1471.

339. R.T. Rockafellar and R.J-B Wets, "Stochastic convex programming: basic duality," *Pacific Journal of Mathematics* 6 (1976a) pp. 173–195.
340. R.T. Rockafellar and R.J-B Wets, "Stochastic convex programming, relatively complete recourse and induced feasibility," *SIAM Journal on Control and Optimization* 14 (1976b) pp. 574–589.
341. R.T. Rockafellar and R.J-B Wets, "A Lagrangian finite generation technique for solving linear-quadratic problems in stochastic programming," *Mathematical Programming Study* 28 (1986) pp. 63–93.
342. R.T. Rockafellar and R.J-B Wets, "Scenarios and policy aggregation in optimization under uncertainty," *Mathematics of Operations Research* 16 (1991) pp. 119–147.
343. W. Römisch, "Stability of stochastic programming problems," in A. Ruszczyński and A. Shapiro (eds.), *Handbooks in Operations Research and Management Science, Volume 10: Stochastic Programming* (Elsevier, Amsterdam, 2003) pp. 483–554.
344. W. Römisch and R. Schultz, "Distribution sensitivity in stochastic programming," *Mathematical Programming* 50 (1991a) pp. 197–226.
345. W. Römisch and R. Schultz, "Stability analysis for stochastic programs," *Annals of Operations Research* 31 (1991b) pp. 241–266.
346. C. Roos, T. Terlaky, and J-P. Vial, *Interior Point Methods for Linear Optimization*, Second Edition (Springer, New York, 2005).
347. S.M. Ross, *Introduction to Stochastic Dynamic Programming* (Academic Press, New York, London, 1983).
348. H.L. Royden, *Real Analysis* (Macmillan, London, NY, 1968).
349. R.Y. Rubinstein, *Simulation and the Monte Carlo Method* (John Wiley Inc., New York, NY, 1981).
350. A. Ruszczyński, "A regularized decomposition for minimizing a sum of polyhedral functions," *Mathematical Programming* 35 (1986) pp. 309–333.
351. A. Ruszczyński, "Parallel decomposition of multistage stochastic programming problems," *Mathematical Programming* 58 (1993a) pp. 201–228.
352. A. Ruszczyński, "Regularized decomposition of stochastic programs: algorithmic techniques and numerical results," Working Paper WP-93-21, International Institute for Applied Systems Analysis, Laxenburg, Austria (1993b).
353. A. Ruszczyński, "Probabilistic programming with discrete distributions and precedence constrained knapsack polyhedra," *Mathematical Programming* 93 (2002) pp. 195–215.
354. G. Salinetti, "Approximations for chance constrained programming problems," *Stochastics* 10 (1983) pp. 157–169.
355. B. Sandıkçı, N. Kong, and A.J. Schaefer, "A hierarchy of bounds for stochastic mixed-integer programs," Chicago Booth Research Paper No. 09-21 (Chicago, IL, June 3, 2009); available at SSRN: <http://ssrn.com/abstract=1413774>.
356. Y.S. Sathe, M. Pradhan, and S.P. Shah, "Inequalities for the probability of the occurrence of at least m out of n events," *Journal of Applied Probability* 17 (1980) pp. 1127–1132.
357. H. Scarf, "A minimax solution of an inventory problem" in: K.J. Arrow, S. Karlin, and H. Scarf, Eds., *Studies in the Mathematical Theory of Inventory and Production* (Stanford University Press, Stanford, CA, 1958).
358. R. Schultz, "Continuity properties of expectation functionals in stochastic integer programming," *Mathematics of Operations Research* 18 (1993) pp. 578–589.
359. R. Schultz, L. Stougie and M.H. van der Vlerk, "Solving stochastic programs with integer recourse by enumeration: A framework using Gröbner basis reductions," *Mathematical Programming* 83 (1998) pp. 229–252.
360. N. Secomandi and F. Margot, "Reoptimization approaches for the vehicle-routing problem with stochastic demands," *Operations Research* 57 (2009) pp. 214–230.
361. S. Sen, "Algorithms for stochastic mixed-integer programming", in: K. Aardal, G.L. Nemhauser, R. Weismantel, Eds., *Handbooks in Operations Research and Management Science* (Elsevier, Amsterdam, 2005) pp. 515–558.
362. S. Sen and J.L. Higle, "The C^3 theorem and a D^2 algorithm for large scale stochastic mixed-integer programming," *Mathematical Programming* 104 (2005) pp. 1–20.

363. S. Sen and H.D. Sherali, "Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming," *Mathematical Programming* 106 (2006) pp. 203–223.
364. D.B. Shmoys and C. Swamy, "An approximation scheme for stochastic linear programming and its application to stochastic integer programs," *Journal of the ACM* 53 (2006) pp. 978–1012.
365. A. Shapiro, "Asymptotic analysis of stochastic programs," *Annals of Operations Research* 30 (1991) pp. 169–186.
366. A. Shapiro, "Inference of statistical bounds for multistage stochastic programming problems," *Mathematical Methods of Operations Research* 58 (2003) pp. 57–68.
367. A. Shapiro and T. Homem-de-Mello, "On the rate of convergence of optimal solutions of Monte Carlo approximations of stochastic programs," *SIAM Journal on Optimization* 11 (2000) pp. 70–86.
368. W.F. Sharpe, "Capital asset prices: a theory of market equilibrium under conditions of risk," *Journal of Finance* 19 (1964) pp. 425–442.
369. D.B. Shmoys and C. Swamy, "An approximation scheme for stochastic linear programming and its application to stochastic integer programs," *Journal of the ACM* 53 (2006) pp. 978–1012.
370. S.A. Smolyak, "Interpolation and quadrature formula for the class W_s^a and E_s^a ," *Dokl. Akad. Nauk SSSR* 131 (1960) pp. 1028–1031.
371. L. Somlyódi and R.J-B Wets, "Stochastic optimization models for lake eutrophication management," *Operations Research* 36 (1988) pp. 660–681.
372. G.J. Stigler, "The cost of subsistence," *Journal of Farm Economics* 27 (1945), 303–314.
373. L. Stougie, *Design and Analysis of Algorithms for Stochastic Integer Programming* (Centrum voor Wiskunde en Informatica, Amsterdam, 1987).
374. B. Strazicky, "Some results concerning an algorithm for the discrete recourse problem," in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980).
375. A.H. Stroud, *Approximate Calculation of Multiple Integrals* (Prentice-Hall, Inc., Englewood Cliffs, NJ, 1971).
376. J. Sun, L. Qi, and K-H. Tsai, "A simplex method for network programs with convex separable piecewise linear costs and its application to stochastic transshipment problems," in: D.Z. Du and P.M. Pardalos, Eds., *Network Optimization Problems: Algorithms, Applications and Complexity* (World Scientific Publishing Co., London, 1993) pp. 281–300.
377. C. Swamy and D.B. Shmoys, "Sampling-based approximation algorithms for multistage stochastic optimization," in: *Proceedings of FOCS 2005* (IEEE Computer Society, Los Alamitos, CA, 2005) pp. 357–366.
378. C. Swamy and D.B. Shmoys, "Approximation Algorithms for 2-Stage Stochastic Optimization Problems," *ACM SIGACT News* 37:March (2006) pp. 33–46.
379. G.H. Symonds, "Chance-constrained equivalents of stochastic programming problems," *Operations Research* 16 (1968) pp. 1152–1159.
380. T. Szántai, "Evaluation of a special multivariate gamma distribution function," *Mathematical Programming Study* 27 (1986) pp. 1–16.
381. G. Taguchi, *Introduction to Quality Engineering* (Asian Productivity Center, Tokyo, Japan, 1986).
382. G. Taguchi, E.A. Alsayed, and T. Hsiang, *Quality Engineering in Production Systems* (McGraw-Hill Inc., New York, NY, 1989).
383. H.A. Taha, *Operations Research: An Introduction*, Fifth edition (Macmillan, New York, NY, 1992).
384. S. Takriti, "On-line Solution of Linear Programs with Varying Right-Hand Sides," Ph.D. Dissertation, Department of Industrial and Operations Engineering, University of Michigan (Ann Arbor, MI, 1994).
385. S. Takriti and J.R. Birge, "Using integer programming to refine Lagrangian-based unit commitment solutions," *IEEE Transactions on Power Systems* 15 (2000a), pp. 151–156.
386. S. Takriti and J.R. Birge, "Lagrangian solution techniques and bounds for loosely-coupled mixed-integer stochastic programs," *Operations Research* 48 (2000b) pp. 91–98.

387. K.T. Talluri and G.J. van Ryzin, *Theory and Practice of Revenue Management* (Springer, New York, 2005).
388. M.J. Todd and B.P. Burrell, "An extension of Karmarkar's algorithm for linear programming using dual variables," *Algorithmica* 1 (1986) pp. 409–424.
389. D.M. Topkis and A.F. Veinott, Jr., "On the convergence of some feasible Eddirection algorithms for nonlinear programming," *SIAM Journal on Control* 5 (1967) pp. 268–279.
390. C. Toregas, R. Swain, C. Revelle, and L. Bergmann, "The location of emergency service facilities," *Operations Research* 19 (1971) pp. 1363–1373.
391. S. Uryasiev, "Adaptive stochastic quasigradient methods" in: Y. Ermolieva and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988) pp. 373–384.
392. F.A. Valentine, *Convex Sets* (McGraw-Hill Inc., New York, NY, 1964).
393. M. H. van der Vlerk, "Convex approximations for complete integer recourse models," *Mathematical Programming* 99 (2004) pp. 297–310.
394. M. H. van der Vlerk, *Stochastic Programming Bibliography on the World Wide Web*, <http://mally.eco.rug.nl/splib.html>, 1996–2007.
395. R. Van Slyke and R.J-B Wets, "L-shaped linear programs with application to optimal control and stochastic programming," *SIAM Journal on Applied Mathematics* 17 (1969) pp. 638–663.
396. L. Vandenberghe and S. Boyd, " Semidefinite Programming," *SIAM Review* 38 (1996) pp. 49–95.
397. P. Varaiya and R.J-B Wets, "Stochastic dynamic optimization approaches and computation" in: M. Iri and K. Tanabe, Eds., *Mathematical Programming: Recent Developments and Applications* (Kluwer, Dordrecht, Netherlands, 1989) pp. 309–332.
398. O. Vasiček, "Probability of loss on loan portfolio," KMV Corporation, Technical Report (San Francisco, CA, 1987); available at:
www.moodyskmv.com/research/files/wp/Probability_of_Loss_on_Loan_Portfolio.pdf.
399. O. Vasiček, "Limiting loan loss portfolio distribution," KMV Corporation, Technical Report (San Francisco, CA, 1991); available at:
www.moodyskmv.com/research/files/wp/Probability_of_Loss_on_Loan_Portfolio.pdf.
400. O. Vasiček, "Loan portfolio value," *Risk* 15:12 (2002) pp. 160–162.
401. J.A. Ventura and D.W. Hearn, "Restricted simplicial decomposition for convex constrained problems," *Mathematical Programming* 59 (1993) pp. 71–85.
402. B. Verweij, S. Ahmed, A.J. Kleywegt, G. Nemhauser, and A. Shapiro, "The sample average approximation method applied to stochastic routing problems: a computational study," *Computational Optimization and Applications* 24 (2003) pp. 289–333.
403. J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior* (Princeton University Press, Princeton, NJ, 1944).
404. A. Wald, *Statistical Decision Functions* (John Wiley, Inc. New York, NY, 1950).
405. D. Walkup and R.J-B Wets, "Stochastic programs with recourse," *SIAM Journal on Applied Mathematics* 15 (1967) pp. 1299–1314.
406. D. Walkup and R.J-B Wets, "Stochastic programs with recourse II: on the continuity of the objective," *SIAM Journal on Applied Mathematics* 17 (1969) pp. 98–103.
407. S.W. Wallace, "Decomposing the requirement space of a transportation problem into polyhedral cones," *Mathematical Programming Study* 28 (1986a) pp. 29–47.
408. S.W. Wallace, "Solving stochastic programs with network recourse," *Networks* 16 (1986b) pp. 295–317.
409. S.W. Wallace, "A piecewise linear upper bound on the network recourse function," *Networks* 17 (1987) pp. 87–103.
410. S.W. Wallace, "Decision making under uncertainty: is sensitivity analysis of any use?" *Operations Research* 48 (2000) pp. 20–25.
411. S.W. Wallace and R.J-B Wets, "Preprocessing in stochastic programming: the case of linear programs," *ORSA Journal on Computing* 4 (1992) pp. 45–59.
412. S.W. Wallace and T.C. Yan, "Bounding multi-stage stochastic programs from above," *Mathematical Programming* 61 (1993) pp. 111–129.

413. S.W. Wallace and W.T. Ziemba, Eds., *Applications of Stochastic Programming: MPS-SIAM Book Series on Optimization 5* (SIAM/MPS, Philadelphia, PA, 2005).
414. R.J-B Wets, “Programming under uncertainty: the equivalent convex program,” *SIAM Journal on Applied Mathematics* 14 (1966) pp. 89–105.
415. R.J-B Wets, “Characterization theorems for stochastic programs,” *Mathematical Programming* 2 (1972) pp. 166–175.
416. R.J-B Wets, “Stochastic programs with fixed recourse: the equivalent deterministic problem,” *SIAM Review* 16 (1974) pp. 309–339.
417. R.J-B Wets, “Convergence of convex functions, variational inequalities and convex optimization problems” in: R.W. Cottle, F. Giannessi and J.-L. Lions, Eds., *Variational Inequalities and Complementarity Problems* (John Wiley, Inc., New York, NY, 1980a) pp. 375–404.
418. R.J-B Wets, “Stochastic multipliers, induced feasibility and nonanticipativity in stochastic programming” in: M.A.H. Dempster, Ed., *Stochastic Programming* (Academic Press, New York, NY, 1980b).
419. R.J-B Wets, “Solving stochastic programs with simple recourse,” *Stochastics* 10 (1983a) pp. 219–242.
420. R.J-B Wets, “Stochastic programming: solution techniques and approximation schemes” in: A. Bachem, M. Grötschel, and B. Korte, Eds., *Mathematical Programming: State-of-the-Art 1982* (Springer-Verlag, Berlin, 1983b) pp. 560–603.
421. R.J-B Wets, “Large-scale linear programming techniques in stochastic programming” in: Y. Ermoliev and R. Wets, Eds., *Numerical Techniques for Stochastic Optimization* (Springer-Verlag, Berlin, 1988).
422. R.J-B Wets, “Stochastic programming” in: G.L. Nemhauser, A.H.G. Rinnooy Kan, and M.J. Todd, Eds., *Optimization* (Handbooks in Operations Research and Management Science; Vol. 1, North-Holland, Amsterdam, Netherlands, 1990).
423. R.J-B Wets and C. Witzgall, “Algorithms for frames and lineality spaces of cones,” *Journal of Research of the National Bureau of Standards Section B* 71B (1967) pp. 1–7.
424. P. Whittle, *Risk-sensitive Optimal Control* (John Wiley & Sons, Chichester, UK, 1990).
425. A.C. Williams, “A stochastic transportation problem,” *Operations Research* 11 (1963) pp. 759–770.
426. A.C. Williams, “Approximation formulas for stochastic linear programming,” *SIAM Journal on Applied Mathematics* 14 (1966) pp. 668–677.
427. E.L. Williamson, “Airline Network Seat Control,” Ph.D. Dissertation, MIT (Cambridge, MA, 1992).
428. R.J. Wittrock, “Advances in a nested decomposition algorithm for solving staircase linear programs,” Technical Report SOL 83-2, Systems Optimization Laboratory, Stanford University (Stanford, CA, 1983).
429. R. Wollmer, “Two stage linear programming under uncertainty with 0-1 integer first stage variables,” *Mathematical Programming* 19 (1980) pp. 279–288.
430. L. Wolsey, *Integer Programming* (John Wiley and Sons, New York, 1998).
431. H. Woźniakowski, “Average-case complexity of multivariate integration,” *Bulletin of the American Mathematical Society (new series)* 24 (1991) pp. 185–194.
432. S.E. Wright, “Primal-dual aggregation and disaggregation for stochastic linear programs,” *Mathematics of Operations Research* 19 (1994) pp. 893–908.
433. D. Yang and S.A. Zenios, “A scalable parallel interior point algorithm for stochastic linear programming and robust optimization,” in: A. Murli and G. Toraldo, Eds., *Computational Issues in High Performance Software for Nonlinear Optimization* (Springer, New York, 1997) pp. 143–158.
434. Y. Ye, *Interior Point Algorithms: Theory and Analysis* (John Wiley and Sons, New York, 1997).
435. J.W. Yen and J.R. Birge, “A stochastic programming approach to the airline crew scheduling Problem,” *Transportation Science* 40 (2006) pp. 3–14.
436. J. Žáčková, “On minimax solutions of stochastic linear programming problems,” *Časopis pro Pěstování Matematiky* 91 (1966) pp. 423–430.

437. S.A. Zenios, *Financial Optimization* (Cambridge University Press, Cambridge, UK, 1993).
438. W.T. Ziemba, “Computational algorithms for convex stochastic programs with simple recourse,” *Operations Research* 18 (1970) pp. 414–431.
439. W.T. Ziemba and R.G. Vickson, *Stochastic Optimization Models in Finance* (Academic Press, New York, NY, 1975).
440. M. Zinkevich, “Online convex programming and generalized infinitesimal gradient ascent,” in: T. Fawcett and N. Mishra, Eds., *Proceedings of the Twentieth International Conference on Machine Learning* (The AAAI Press, Menlo Park, CA, 2003) pp. 928–936.
441. P. Zipkin, “Bounds for row-aggregation in linear programming,” *Operations Research* 28 (1980a) pp. 903–916.
442. P. Zipkin, “Bounds on the effect of aggregating variables in linear programs,” *Operations Research* 28 (1980b) pp. 403–418.

Author Index

- Abrahamson, 275
Adelman, 439
Agarwal, 91
Ahmed, 263, 311, 405
Anstreicher, 226
Ariyawansa, 222
Artzner, 85
Ashford, 445
Asmussen, 389
Attouch, 382
Avriel, 171

Bahn, 236
Bayraksan, 412–414
Bazaraa, 116, 121, 246, 254
Beale, 59, 247, 251, 445, 446
Bellman, 89
Ben-Tal, 67, 86, 346
Benders, 182
Bereanu, 108
Berger, 87
Berman, 69
Bertsekas, 436
Bertsimas, 86, 362
Bienstock, 332
Billingsley, 381
Birge, 120, 160, 168, 170, 171, 199, 200,
 226, 229, 235, 242, 251, 252,
 266, 268, 275, 286, 301, 347,
 349, 352, 357, 362, 366, 367, 370, 371, 374, 376–379, 381,
 382, 384, 412, 414, 423, 425,
 431, 433, 435, 436, 441, 443,
 444, 446
Bitran, 426, 440, 445, 446
Blackwell, 90
Blair, 136
Borell, 126
Boyd, 86, 360
Brumelle, 50
Burrell, 226

Califiore, 404
Campi, 404
Cariño, 429
Carpenter, 234
Carøe, 301, 333
Ceder, 67
Chao, 170
Charnes, 25, 49, 124, 128
Chen, 412, 414
Chiu, 69
Chung, 56
Chvátal, 57, 73, 97
Cipra, 374, 407
Clarke, 382
Conn, 209
Cooper, 25, 49, 124, 128
Dai, 412

- Dantzig, 49, 57, 59, 73, 182, 237, 238, 245, 372, 373, 390, 392, 446
- Dawson, 361
- de Farias, 404
- Deák, 362, 389, 405
- DeGroot, 87
- Delbaen, 85
- Dempster, 91, 108, 115, 160, 256, 429
- Demyanov, 263
- Dentcheva, 379
- Donohue, 433–435
- du Merle, 236
- Dulá, 368, 374, 376, 377
- Dupač, 399
- Dupačová, 373, 411, 429
- Dye, 263
- Dyer, 265, 403
- Eber, 85
- Edirisinghe, 364, 374
- Edmundson, 346, 350
- Eisner, 122
- Eppen, 67
- Epstein, 92
- Ermoliev, 49, 263, 301, 374, 399, 402
- Escudero, 49
- Feller, 358, 360
- Ferguson, 49, 245
- Flåm, 160
- Flaxman, 91, 263
- Fleming, 91
- Forrest, 445
- Fourer, 26, 245
- Fox, 414
- Frantzeskakis, 440, 441
- Frauendorfer, 346, 347, 350, 363–365, 373
- Freund, 235
- Frieze, 263
- Gaivoronski, 374, 400
- Gartska, 92, 129, 222
- Gassmann, 49, 200, 218, 221, 268, 275, 349, 362, 405
- Gay, 26, 228
- Gendreau, 148, 301
- Geoffrion, 237, 356
- Gilboa, 92
- Glassey, 266
- Glynn, 389, 390, 392
- Goffin, 236
- Gondzio, 236
- Gould, 209
- Gröwe, 429
- Grinold, 155, 423
- Grothey, 236
- Guan, 434
- Gupta, 263
- Halada, 236
- Hansen, 92
- Harrison, 92, 431
- Haugland, 222
- Hazan, 91
- Hearn, 255
- Heath, 85
- Hemmecke, 311
- Heyman, 89
- Higle, 318, 389, 390, 395–397
- Hiriart-Urruty, 120
- Hjörring, 301
- Ho, 266
- Hoeffding, 358, 405
- Hogan, 128
- Holmes, 229
- Holt, 301
- Homem-de-Mello, 411
- Howard, 90
- Huang, 346
- Huber, 86, 409
- Hudson, 222
- Hunsaker, 311
- Høyland, 429
- Iancu, 86
- Infanger, 275, 392
- Jagganathan, 378, 407
- Jaillet, 70
- Jarvis, 246
- Jasin, 439, 447

- Jensen, 166, 346
Jeroslow, 136

Kalai, 91
Kale, 91
Kall, 89, 112, 115, 208, 222, 346, 347, 374, 382
Kallberg, 21, 126, 244, 284
Kallio, 152
Kalman, 92
Kan, 125
Kannan, 403
Kao, 49
Karmarkar, 226
Karr, 372
Kemperman, 373
Kernighan, 26
Kibzun, 125
King, 49, 255, 381, 385, 386, 409
Kiwiel, 263
Klaassen, 431
Klein Haneveld, 122, 446
Kong, 177, 311
Kouwenberg, 429
Krein, 372
Kreps, 92, 431
Krivelevich, 263
Kuhn, 423
Kumar, 439, 447
Kurbakovskiy, 125
Kushner, 91, 399
Kusy, 284

Laporte, 148, 293, 301
Larson, 69
Lasdon, 199
Lemaréchal, 263
Linderoth, 209, 222
Lo, 380
Louveaux, 33, 65, 136, 141, 146, 148, 153, 170, 199, 200, 212, 214, 266, 277, 278, 282, 284, 293, 301, 321, 325, 332
Luedtke, 405
Lustig, 234

Madansky, 164, 166, 237, 346, 349, 350
Maddox, 362, 446
Mak, 412
Mangasarian, 166
Manne, 49, 170, 266
Margot, 301
Markowitz, 67
Marti, 379
Martin, 67, 107
Mayer, 208
McGill, 50
McMahon, 91
Mehrotra, 414
Mercure, 148
Merton, 404
Michel, 382
Miller, 127
Minty, 257
Mirzoachmedov, 403
Monro, 401
Mordukhovich, 382
Morgenstern, 67
Morris, 128
Morton, 268, 412–414
Mulvey, 20, 234, 256, 286
Murty, 57, 253

Natarajan, 362
Nazareth, 242, 247, 251
Nedeva, 374
Nemhauser, 136
Nemirovski, 86, 360
Nesterov, 403
Niederreiter, 414
Nielsen, 256
Noël, 262, 266, 275
Norkin, 301
Ntiamo, 318
Nudel'man, 372

Olsen, 122
Papagaki-Papoulias, 108
Parikh, 127, 128
Parrilo, 86
Peeters, 65

- Penot, 382
 Pereira, 266, 433
 Pflug, 236, 301, 403, 429
 Philpott, 434
 Pintér, 358, 360
 Pinto, 266, 433
 Plambeck, 263
 Popescu, 362
 Porteus, 92, 152
 Powell, 436, 440, 441
 Prékopa, 25, 49, 126, 127, 358, 360, 361, 386
 Pradhan, 361
 Psaraftis, 67
 Qi, 120, 229, 246, 251, 252, 362, 381, 382, 384
 Queyranne, 49
 Römisch, 118, 411, 414, 429
 Raiffa, 88, 163
 Ravi, 263
 Rei, 301
 Richels, 49
 Rishel, 91
 Robbins, 401
 Robinson, 118
 Rockafellar, 85, 108, 120, 122, 157–160, 255–257, 356, 357, 376, 383, 409, 444
 Roos, 227
 Rosa, 275
 Rosen, 166
 Ross, 89
 Royden, 118, 372
 Rubinstein, 362
 Ruszczyński, 202, 205, 208, 268, 286, 301, 379
 Rutenberg, 222
 Séguin, 148, 301
 Sahinidis, 311
 Salinetti, 362
 Sandıkçı, 177
 Sankoff, 361
 Sargent, 92
 Sarkar, 445
 Sathe, 361
 Scarf, 378
 Schaefer, 177, 311
 Schlaifer, 163
 Schmeidler, 92
 Schrage, 67
 Schultz, 118, 136, 146, 311, 333, 411
 Secomandi, 301
 Sen, 318, 389, 390, 395–397
 Shah, 361
 Shapiro, 332, 360, 411, 431
 Sherali, 246, 318
 Shetty, 116, 121, 254
 Shmoys, 263, 265, 403, 431
 Sim, 86
 Sinha, 263
 Smeers, 33, 170, 262, 266, 275, 282
 Smolyak, 414
 Sobel, 89
 Somlyódy, 49, 255
 Stigler, 73
 Stougie, 137, 146, 263, 265, 311, 403
 Stoyan, 346
 Strazicky, 222
 Stroud, 342
 Sun, 246, 252
 Swamy, 263, 265, 403, 431
 Symonds, 49, 129
 Szántai, 49, 360, 362, 405
 Taguchi, 36
 Taha, 446
 Takriti, 286
 Talluri, 67, 438
 Tawarmalani, 311
 Taylor, 445
 Teboulle, 67, 379
 Teo, 362
 Terlaky, 227
 Tharakan, 67
 Thompson, 128
 Tind, 301
 Todd, 226
 Toint, 209

- Tomasgard, 263
Topkis, 255
Toregas, 72
Tsai, 246, 252
Tsitsiklis, 436

Uriasiev, 403
Uryasev, 85

Valentine, 377
van der Vlerk, 141, 146, 311, 321, 325
Van Roy, 404
van Ryzin, 67, 438
Van Slyke, 182, 267
Vandenbergh, 360
Vanderbei, 236
Vanhamme, 301
Varaiya, 27
Vasiček, 405
Vasiliev, 263
Veinott, 255
Ventura, 255
Verweij, 414
Vial, 227, 236, 403
Vickson, 20
Vladimirou, 20, 256, 286
von Neumann, 67

Wagner, 127
Wald, 87, 372
Walkup, 111, 212
Wallace, 49, 89, 219, 222, 246, 347, 366,
 367, 371, 429, 444
Watson, 446
Wein, 92

Wets, 27, 49, 92, 108, 111–113, 115,
 117, 118, 120, 122, 124, 126,
 129, 160, 182, 212, 218, 219,
 221, 222, 242, 243, 247, 251,
 255, 256, 267, 347, 349, 352,
 357, 367, 370, 371, 374, 378,
 379, 381, 382, 384–386, 411,
 443, 444
White, 244, 284
Whittle, 92
Williams, 63, 171, 247
Williamson, 438
Wittrock, 268
Witzgall, 113
Woźniakowski, 414
Wolfe, 182
Wolsey, 136, 300
Wood, 412
Wright, 209, 222, 423

Yan, 444
Yanassee, 426, 440, 445
Yang, 236
Ye, 227
Yen, 301

Žáčková, 373
Zenios, 20, 236, 256
Zhao, 436
Ziemba, 20, 21, 49, 126, 244, 247, 251,
 284, 346, 349
Zinkerich, 91
Zinn, 92
Zipkin, 423

Subject Index

- L*-shaped, 182, 196, 198–202, 204, 208–210, 213, 217, 218, 222, 226, 237, 238, 241, 245–247, 253, 263, 294, 352
integer, 293, 301
 ∞ -norm, 209
 ρ -approximation, 320
- a priori optimization, 70
a.s., *see* almost surely
abridged nested decomposition, 433
absolutely continuous, 112, 116, 137, 141, 247
abstract linear program, 372
active set, 208, 247, 251, 276
adjusted random sample, 429
ADP, *see* approximate dynamic programming
affine, 98
 hull, 98, 350
 space, 98
affine scaling, *see* scaling
aggregation, 31, 266, 422
airline crew, *see* crew scheduling
almost surely, 60, 124
ancestor, 152, 267, 277
annuity, 31
approximate dynamic programming, 367, 436
approximation, 39, 144, 341
midpoint, 342
polynomial, 342
quadratic, 251
trapezoidal, 342, 350
- arbitrage, 429
arborescent, 152
artificial variable, 94, 95
assembly, 74
athletics, 53
atom, 346
augmented Lagrangian, *see* Lagrangian
ball, 98
barycentric, 368
 coordinates, 350
basis, 94, 107
 factorization, 222
 forest structure, 252
 function, 436
 working, 224
Bayesian, 93, 407, 427
Bellman-Hamilton-Jacobi equation, 92
Benders decomposition, *see* decomposition
bias, 393
bid-ask spread, 430
bid-price, 438
block angular, 182
block separable, *see* separable
block separable recourse, *see* recourse

- booking limit, 439
- Boole-Bonferroni inequalities, *see* inequality
- Borel field, 348, 420
- bounded, 98
- bounding, *see* bounds
- bounds, 171, 381, 441, 444
- branch-and-bound, 242, 299, 318, 335
- branch-and-cut, 312
- branching
 - on tenders, 304, 307, 312
 - solutions, 434
- bunching, 140, 218, 219, 275
- bundle method, 255, 263
- buy-and-hold, 27
- call option, 380, 429
- capacity expansion, 151–153, 222
- Carathéodory's theorem, 349, 377
- cell, 211, 277, 347
- central limit theorem, 391, 411
- chance constraint, *see* probabilistic constraint
- Chebyshev inequality, *see* inequality
- Cholesky factor, 230, 233, 428
- closed, 98
- coherent risk measure, 85, 86
- column splitting, 233
- common cut coefficient, 314
- compact, 98
- complement, 361
- complementarity, 129
- complementary, 240, 267
 - slackness, 96
 - system, 124
- complete recourse, *see* recourse
- complexity, 228, 230, 263, 414, 438
- concave, 20, 22, 84, 98, 107
- conditional expectation, *see* expectation
- conditional value-at-risk, 85
- cone, 98, 106, 113, 117, 205, 207
 - positive, 113, 218
- confidence
 - interval, 125, 392, 415, 433, 435
 - region, 403
- conjugate, 100, 356
- connected, 125, 377
- constraint
 - relaxation, 265
 - subtour elimination, 148, 300
- contingent payoff, 429
- continuous, 13
 - relaxation, 286, 290, 326
 - time, 92
- control, 20, 27
 - limit, 92
- convergence, 100, 181, 196, 197, 204, 238, 241, 247, 251, 256–259, 261, 268, 275, 278, 286, 287, 342, 347, 356, 381–383, 390, 392, 395–397, 400–403, 409, 411–413, 415, 431, 432, 434
- bounded, 120
- geometric, 259, 261
- in distribution, 381, 383
- pointwise, 100
- superlinear, 256
- uniform, 99
- convex, 13, 32, 157
 - combination, 97
 - complex, 211
 - function, 98
 - proper, 98
 - hull, 97, 238, 254, 356, 370, 377
 - set, 97
 - simplex method, 251
- cover, 133, 134
- covering, 146
- crew scheduling, 301
- cumulative probability distribution, 16
- cut, 202
 - disjunctive, 289, 317–319, 331, 336–338
 - feasibility, 184, 191, 192, 196, 203, 276, 289, 293, 306, 326–329, 353, 391
 - optimality, 184, 185, 188–190, 196, 197, 203, 215, 276, 290, 291, 293, 294, 296, 299, 301, 322–326, 329, 434

- Dantzig-Wolfe, *see* decomposition decision, 57
analysis, 87, 88, 163
rule, 92
theory, 88
tree, 25, 88, 427
- decomposition, 151, 181, 212, 213, 218, 219, 222, 224, 226, 245, 277, 289, 311, 389, 401, 417, 432, 444
- Benders, 182, 196, 266, 301
- Dantzig-Wolfe, 182, 196, 198, 199, 237, 275
- Datnzig-Wolfe, 238
- dual, 322
- nested, 266, 273, 275, 277, 433, 434, 438
- nested quadratic, 276
- regularized, 202, 204, 208, 209, 279
- scenario, 333
- simplicial, 255
- stochastic, *see* stochastic deflection, 37
- degeneracy, 275
- density, 12, 20, 56, 122, 126, 142, 145, 146, 392, 393, 395, 410
- DEP, *see* deterministic-equivalent derivative, 16, 206, 263, 377
directional, 98, 99
financial, 380
Hadamard, 99
security, 429
- descendant, 152, 267
- design, 84
- deterministic, 28
equivalent, 34, 60, 72, 104, 125, 127, 135, 146, 150, 151, 182, 263, 265, 289
model, 20, 25, 26, 31
- diagonal quadratic approximation, 286
- dictionary, 94–96, 195, 221, 328, 330
- differentiable, 16, 98, 112, 146
continuously, 409
G- or Gâteaux, 99
- dimension, 98
- directional derivative, *see* derivative discount, 52, 90, 407, 423, 436
- discounting, 18, 89
- discrete variables, *see* integer variables
- disjunction, 337
- disjunctive cut, *see* cut distribution
- Bernoulli, 363, 404
- binomial, 449
- Dirichlet, 408
- empirical, 132
- exponential, 130, 143, 450
- function, 16
- gamma, 408, 450
- lognormal, 427, 432
- multivariate gamma, 360
- normal, 73, 83, 127, 145, 149, 299, 362, 363, 391, 410, 442, 446, 450
- Poisson, 83, 122, 147, 149, 297–299, 322, 449
- problem, 108, 164
- triangular, 36, 110
- uniform, 122, 142, 143, 149, 168, 449, 450
- dom, *see* effective domain
- dominance, 134, 135
set, 133, 134
- downside risk, *see* risk-downside
- dual, 96, 118, 122, 233, 370
ascent, 254
block angular, 182
Lagrangian, 99
program, 356
simplex, 97
- duality, 57, 158
gap, 122
strong, 100
weak, 100
- dualization, 265, 371
- dynamic, 28
program, 87, 89, 92, 150
- dynamic programming operator, 436
- E-model, 124

- Edmundson-Madansky bound, *see* inequality
 EF, *see* stochastic-program-extensive form
 effective domain, 98, 158
 electric power, *see* power
 emergency, 52, 69, 72, 155
 empirical, 132, 389, 407, 408
 measure, 385
 end effects, 31, 270, 423
 energy, 30, 49, 51, 170, 275
 entering variable, 94
 environment, 275
 EPEV, *see* expectation-of pairs expected value
 epi-convergence, 382
 epigraph, 98, 240, 322, 382
 equivalent martingale measure, 380, 430
 essentially bounded, 119
 event, 10, 33, 56, 58–60, 64, 66, 69, 70, 104, 105, 300, 361, 418, 426, 432
 EVPI, *see* expected-value of perfect information
 exhaustible resources, 170
 expectation, 10, 57
 conditional, 343, 367, 419, 424
 of pairs expected value, 174
 expected
 shortage, 141, 146
 surplus, 141, 146
 value of perfect information, 9, 163, 429
 value of sample information, 407
 value problem, 165
 value solution, 9, 24, 165
 extensive form, *see* stochastic program
 extremal measure, 373
 extreme
 direction, 378, 379
 point, 94, 182, 222, 226, 237, 238, 240, 241, 337, 347–351, 353, 354, 364–366, 372, 373, 377, 379, 418, 420–422
 ray, 237, 238, 241, 242
 solution, 240
 face value, 380
 factorization, 208, 229
 basis, *see* basis
 QR, 208
 failure rate, 127
 Farkas lemma, 97
 feasibility
 set, 105, 109, 111, 138, 139, 152, 158, 196, 291, 308, 326, 331, 390, 430
 second-stage, 138, 210
 feasibility cut, *see* cut
 feasible region, 91, 97–99, 103, 115, 156, 157, 241, 269, 414, 415
 Fenchel duality, 158
 filtration, 160
 finance, 20, 84, 91, 244, 284, 358, 380, 429
 financial crisis of 2007–2010, 358
 financial planning, 20, 21, 150, 151, 155, 429, 430, 432, 435
 finite generation, 255
 first-order stochastic dominance, 85
 first-stage, 8, 10, 104
 binary, 18
 decision, 58
 fleet assignment, 49, 245
 forestry, 49, 51
 Frank-Wolfe method, 247, 253, 263
 free variable, 96
 full decomposability, 218
 G-differentiable, *see* differentiable
 GATT, 219
 generalized
 network, 27, 245
 programming, 238, 245, 247, 248, 356, 373
 upper bound, 335
 generalized moment, *see* moment
 Gomory function, 136, 149, 327
 Gröbner basis, 311
 gradient, 98
 GUB, *see* generalized-upper bound
 Hamiltonian tour, 299

- hedging, 9
- here-and-now, 164
- Hessian, 99, 256
- heuristic, 6, 163, 335
- history process, 160, 427
- Hoeffding inequality, *see* inequality
- homogeneous self-dual, 227
- horizon, 21, 25, 31, 150, 270
- hospital, 52
- hull
 - convex, 321
- hypercube, 304, 306–309, 311
- hyperplane, 98, 402
 - separating, 106, 111, 198
 - supporting, 99, 189, 190, 196, 352, 356
- implicit representation, *see* stochastic program
- importance sampling, 390
- improving direction, 97
- independence
 - linear, 373
- indicator function, 97, 166
- induced constraint, 68, 193, 326, 328
- inequality
 - Bonferroni, 405
 - Boole-Bonferroni, 360
 - Chebyshev, 358
 - cover, 335, 338
 - Edmundson-Madansky, 346, 350, 418
 - Hoeffding, 405
 - Jensen, 166, 346, 360, 418
 - triangle, 42
 - valid, 133, 312, 335, 336
- infeasible, 95
- infinite dimensional, 372
- infinite horizon, 89, 417, 422, 423, 435, 436
- inner linearization, 181, 182, 199, 237, 255, 265, 266
- int, *see* interior
- integer variables, 35
- integrable, 118, 158
- integration, 158, 345, 346, 414
- multiple, 342
- numerical, 113, 341–343, 350
- interior, 98
- interior point method, 222, 276
- Jensen's inequality, *see* inequality
- just-in-time, 282
- K-K-T, *see* Karush-Kuhn-Tucker
- Kalman filtering, 92
- Karush-Kuhn-Tucker, 14, 82, 99, 116, 211, 214, 276, 283, 375
- knapsack, 133
- kurtosis, 429
- Lagrangian, 99, 253, 265, 286, 333
 - augmented, 256
- large-deviation bounds, 389, 412
- large-scale optimization, 152, 182
- large-scale programming, *see* large-scale optimization
- leaving variable, 94
- Lebesgue measure, 384
- level set, 374
- linear
 - program, 5, 57
 - solver, 185
 - quadratic, 255
 - quadratic Gaussian, 91
 - linearization, 246, 275
 - inner, *see* inner
 - outer, *see* outer
 - Lipschitz, 99, 112, 409
 - locally, 99, 382
 - local, 99
 - location, 61, 69, 72, 332
 - uncapacitated facility, 61
 - logarithmic barrier, 227
 - logarithmically concave, 126, 127
 - lower semicontinuous, 136, 157, 383
 - LP, *see* linear-program
 - LP-relaxation, *see* continuous-relaxation
 - LQG, *see* linear-quadratic Gaussian
 - machine learning, 90
 - major iteration, 200

- manufacturing, 92
- mapping
 - multifunction, 385
- marginal, 367
 - value, 96
- Markov decision process, 87, 89, 155
- mathematical expectation, *see* expectation
- max-min utility, 92
- maximal monotone operator, 257
- mean value problem, *see* expected-value problem
- mean-variance model, 67
- measurable, 118, 156, 385
- measure, 55, 118
- min-max, 93
- mixed integer, 131, 330, 331
- modeling language, 26
- moment, 57
 - generalized, 362, 372
 - generating function, 360
 - second, 110–112, 114, 115, 124, 152, 342, 345, 358, 372–374, 376, 377, 381, 429, 441
- Monte Carlo
 - method, 266, 389
- MQSP, *see* decomposition-nested quadratic
- multicut, 199, 202, 275, 322, 329
- multifunction, 385
- multiple integration, *see* integration
- multiplier, 96, 191, 267
 - dual, 374
- multistage, 18, 25, 28, 65, 149, 265, 332
- natural probability, 431
- nested decomposition, *see* decomposition
- network, 242, 245, 286, 362
 - generalized, *see* generalized network
- network revenue management, 438
- neuro-dynamic programming, 436
- news vendor, 3, 14, 15, 251
- newsboy, *see* news vendor
- Newton step, 256
- Neyman-Pearson lemma, 372
- no-arbitrage condition, 430
- node
 - terminal, 327
- nonanticipative, 21, 25, 26, 91, 118, 150, 159, 234, 256, 257, 333, 418, 420, 421
- nonanticipativity, *see* nonanticipative
- nonconvex, 382
- nondifferentiable, 116, 255, 342
- nonlinear, 21, 27, 40, 156, 441
 - programming, 97, 343
- normal cone, 117, 159, 207
- normal distribution, *see* distribution
- NP-hard, 263
- numerical integration, *see* integration
- numerical stability, 208
- oil spills, 67
- online optimization, 90
- optimality condition, 115, 116
- optimality cut, *see* cut
- order of merit rule, 35
- outer linearization, 182, 266
- P-model, 124
- pairs problem, 172
- parallel processing, 222, 226, 236, 256, 268, 276
- parallel subspace, 98
- parametric optimization, 376
- path-dependent, 427
- path-following, 227
- Peano's rule, 342
- penalty, 91
- period, 65
- PERT network, 362, 446
- PHA, *see* progressive hedging
- phase one, 94, 326, 373
- phase two, 95
- piecewise
 - constant, 143, 149
 - convex, 212
 - linear, 22, 99, 143, 149, 342
 - quadratic, *see* quadratic
- pivot, 94

- polar matrix, 112
- polynomial approximation, *see* approximation
- Pontryagin's maximum principle, 92
- pos, *see* cone-positive, *see* cone-positive
- positive
 - cone, *see* cone
 - definite, 93, 210
 - hull, 198
 - semi-definite, 210, 277
- positive linear basis, 368
- positively homogeneous, 108, 367
- posterior distribution, 93
- power generation, 28, 31, 193, 286
- PQP, *see* quadratic-piecewise
- premium, 380
- preprocessing, 222, 335
- price effect, 17
- primal-dual, 121
- probabilistic constraint, 34, 47, 124, 128, 146, 345, 357, 404
- probabilistic programming, 3, 25, 71
- probability, 56
 - space, 55, 56
- production, 49, 74, 418, 425
- progressive hedging, 161, 256–258, 284, 285, 444
- projection, 98, 160, 232, 400
- proper convex function, 115
- proximal point method, 257
- pseudo-random, 414
- PSPACE-hard, 265
- quadratic, 27, 40, 93, 99, 202, 276
 - piecewise, 210, 212, 214, 277
- quadrature, 341, 342, 345
 - Gaussian, 345
- quality, 37
- quantile, 17, 57, 73, 125, 404
- quasi-concave, 125, 127
- quasi-random, 414
- racing, 52
- random
 - continuous, 16
 - variable, 55, 58, 66
- continuous, 11, 32, 56, 104
- discrete, 10, 32, 56, 104, 144
- normal, 391
- vector, 10, 11, 110
- rc, *see* recession cone
- recession
 - cone, 115, 117
 - direction, 115, 237, 239
- recourse, 164
 - block separable, 32, 154
 - complete, 113, 118, 193
 - fixed, 10, 103, 150, 156, 168
 - function, 11, 104
 - integer
 - simple, 319
 - matrix, 104
 - network, 246
 - nonlinear, *see* nonlinear
 - problem, 24
 - program, 57
 - relatively complete, 113, 117, 119, 120, 122, 124, 155, 159, 160, 193, 277, 278, 293, 306, 317, 411, 414, 433, 438
 - simple, 40, 49, 64, 113, 116, 128, 239, 242, 246–248, 284, 343, 367, 440
 - simple integer, 140, 146, 289, 322
 - rectangular region, 350
 - recursion, 150
 - reduced gradient, 251
 - refinement, 347, 357
 - reformulation, 312
 - regret, 90
 - regularity, 99, 157
 - condition, 99, 100, 120, 160
 - regularized decomposition, *see* decomposition
 - relative interior, 98, 158
 - reliability, 3, 34, 35, 40, 124, 127, 359, 360, 408, 426
 - revenue management, 50, 67, 418
 - ri, *see* relative interior
 - risk
 - attitude, 128

- aversion, 18, 66, 67
- downside, 67
- preference, 379
- risk-neutral measure, 430
- risk-sensitive, 93
- riskfree rate, 380
- robust, 84, 86, 92, 358
 - optimization, 86
 - risk-measure, 86
- route, 148
- s-neighbors, 294
- SAA, *see* sample average approximation
- salvage value, 31, 440
- sample average approximation, 390, 392, 409, 414, 431
- sample information, 407
- sampling measure, 385
- scaling
 - affine, 227
 - projective, 227, 230, 233, 235, 236
- scenario, 21, 22, 56, 67, 130, 152, 163, 172
 - generation, 266, 426
 - reduction, 266, 427, 437, 438
 - reference, 172, 177
- Schur complement, 233, 246
- second moment, *see* moment
- second-stage, 8, 10, 58, 104
 - integer, 18
 - value function, 60
- self-dual, 235
- semi-definite program, 360, 362
- separability, *see* separable
- separable, 99, 140, 239, 242, 247, 248, 251, 297, 343, 350, 356, 366, 367, 441
 - block, 20, 153, 154, 156, 332
 - function, 114
 - time, 92, 275
- sequential sampling, 393, 411, 413, 414
- serial independence, 427
- shadow price, 96
- sharp minimum, 411
- Sherman-Morrison-Woodbury formula, 235
- short-selling, 429, 430
- shortage, 22, 141, 319
- sifting, 222
- simple integer recourse, *see* recourse
- simple recourse, *see* recourse
- simplex, 350, 368
- simplex algorithm, 94
- simplicial decomposition, *see* decomposition
- simplicial region, 349
- SIP, *see* stochastic-program-integer
- skewness, 429
- slack variable, 94, 95
- Slater condition, 99, 157
- solution, 94
 - basic, 94
 - feasible, 94
 - optimal, 94
- SOS, *see* special-ordered set
- sparse grid, 414
- special-ordered set, 335
- SPEV, *see* sum of pairs expected values
- sports, 49, 53
- SQG, *see* stochastic-quasi-gradient
- SQM, *see* stochastic-queue median
- SSM, *see* sequential sampling
- stability, 118
- staffing, 49, 52
- stage, 57, 65, 90, 150
- state, 90, 91, 151
 - of the world, 56
 - prices, 430
 - variables, 27
- static, 28
- statistical decision theory, 87
- Steiner tree, 263
- stochastic
 - control, 87, 91
 - decomposition, 389, 395, 397, 398
 - dominance, 379
 - independence, 350
 - program
 - extensive form, 8, 11, 68, 139, 182, 265
 - implicit representation, 11, 68

- integer, 135, 286, 289, 414
 - with recourse, 149, 156
- quasi-gradient, 399, 401
- queue median, 69
- subgradient, *see* subgradient, 403
- stochastic dual dynamic programming, 433
 - stopping criteria, 352
 - strategic, 56
 - stress, 37, 38, 40
 - subadditive, 85, 136
 - subdifferential, 99, 114, 117
 - generalized, 382
 - subgradient, 99, 116, 159, 167, 213, 362, 399
 - method, 254
 - stochastic, 400
 - sublinear, 374
 - suboptimization, 384
 - subtour elimination, *see* constraint
 - sum of pairs expected values, 172
 - superadditive, 311
 - support, 60, 104, 150, 182, 219
 - supporting hyperplane
 - seehyperplane, 196
 - surplus, 22, 141, 319
 - tail risk, 429
 - technology matrix, 104
 - tender, 105, 140, 242, 251
 - terminal conditions, 150
 - test sets, 311
 - time horizon, *see* horizon
 - time-additive, *see* separable-time
 - time-separable, *see* separable
 - total second moment, 374
 - totally unimodular, 139
 - transaction cost, 20, 27, 91, 430
 - translation, 98
 - transportation, 252
 - transportation model, 63
 - trapezoidal approximation, *see* approximation
 - traveling salesperson problem, 42–45, 47, 48, 58, 70, 299, 302
 - tree, 22
 - decision, 22
 - triangle inequality, *see* inequality
 - triangular distribution, *see* distribution
 - trust region, 222
 - trust-region method, 209
 - TSP, *see* traveling salesperson problem
 - two-point support, 377
 - two-stage, 65, 103
 - stochastic program with recourse, 10, 59, 156
 - UFLP, *see* location-uncapacitated facility
 - unbiased estimates, 406
 - unbounded, 94
 - uncertainty set, 86
 - unit commitment, 286
 - utility, 21, 22, 25, 67, 89, 90
 - von Neumann-Morgenstern, 67, 84
 - V-model, 124
 - valid inequality, *see* inequality
 - value function, 11, 136
 - value of information, 160
 - value of the stochastic solution, 9, 17, 24, 165
 - value-at-risk, 84
 - variance, 57
 - reduction, 390, 405
 - vehicle, 42, 148, 299, 440
 - allocation, 418
 - location, 155
 - routing, 40, 299, 301
 - VRP, *see* vehicle-routing
 - VSS, *see* value of the stochastic solution
 - wait-and-see, 164, 302
 - water resource, 49, 50, 255
 - working basis, *see* basis
 - worst case, 18, 228
 - yield management, 50
 - zero-coupon bond, 380