La pianificazione della produzione

La pianificazione della produzione è un processo complesso che coinvolge larga parte dell'azienda (funzioni e risorse) e deve ragionare su previsioni a lungo periodo.

Gli elementi du un piano di produzione sono:

- MPS (Master Production Schedule)
- MRP (Material Requirement Planning)
- Inventory Management
- Capacity Planning
- Scheduling
- Production Control

Data la complessità del problema, i diversi decisori e il lungo orizzonte temporale il processo di pianificazione si realizza normalmente per successive approssimazioni:

a . piano a lungo termine molto aggregato

b . piani successivi con orizzonti minori e maggior dettaglio

Production Planning : tentativo di pianificazione delle quantità su un *orizzonte temporale* lungo (6 mesi, un anno).
Include dimensionamenti della forza lavoro, delle giacenze di magazzino, della materia prima e altre risorse.
Opera per quantità aggregate (per tipologia o periodo)

Production Scheduling (dispatching): piano di produzione dettagliato che considera individualmente ogni prodotto/ risorsa e definisce esattamente cosa "fanno nel tempo.
L'orizzonte di pianificazione è limitato (1, 2 settimane, un mese)
Nonostante il dettaglio il piano viene modificato nella pratica a causa di: necessità o urgenze impreviste, rottura di macchine, utensili, pezzi, aleatorietà dei dati.

La pianificazione si basa su risorse <u>fisse</u> e <u>variabili</u> definite dal management (es: impianti, equipaggiamenti, magazzini, forza lavoro, etc.)

Si basa inoltre su stime della <u>domanda futura</u> che si assume nota e valida. Questa stima entra quindi nella fase di pianificazione e nei modelli come un *dato di input*. La sua determinazione, problema fondamentale nell'azienda (*forcasting*), può essere effettuata con il supporto di vari metodi e tecniche della Ricerca Operativa (descritti negli opportuni corsi).

La pianificazione deve determinare la scelta *ottima* considerando i :
1. costi di immagazzinamento e obsolescenza (*oversupply*)
2. costi dovuti al non soddisfare la domanda (interamente o con ritardo)
3. costi dovuti ad alternative di produzione (lavoro straordinario, nuovo personale, produzione esterna)
4. costi di modifica del livello di produzione

I parametri del MPS

1. Unità temporale (mese, settimana, giorno, ora..)
2. Orizzonte temporale
3. Livello di aggregazione dei prodotti (per prodotto, per categoria,..)
4. Livello di aggregazione delle risorse
5. Frequenza di ripianificazione
6. Numerosità e struttura dei piani

1. Unità temporale

▷ Di norma si usano le settimane o i mesi.
▷ Piccola unità ⇔ maggior dettaglio, pianificazione più difficile

2. Orizzonte temporale

▷ Di norma discende naturalmente dalla natura della produzione (prodotti stagionali, prodotti di largo consumo, etc.)
▷ Il piano parte un breve periodo dal momento attuale; da ora all'inizio del piano si considerano decisioni non modificabili (*congelate*)

3. Livello di aggregazione dei prodotti

▷ Scopo dell'aggregazione è: (a) concentrare l'attenzione sulle risorse/costi più rilevanti; (b) sviluppare un piano che sia relativamente semplice da sviluppare/analizzare
▷ Il livello ottimale di aggregazione dipende dalla natura dei costi, dalla struttura produttiva, dalla stabilità della situazione
▷ Esiste poi il problema di disaggregare il piano per la fase di scheduling (spesso si usano piani intermedi)

4. Livello di aggregazione delle risorse

▷ Valgono le considerazioni del punto 3.

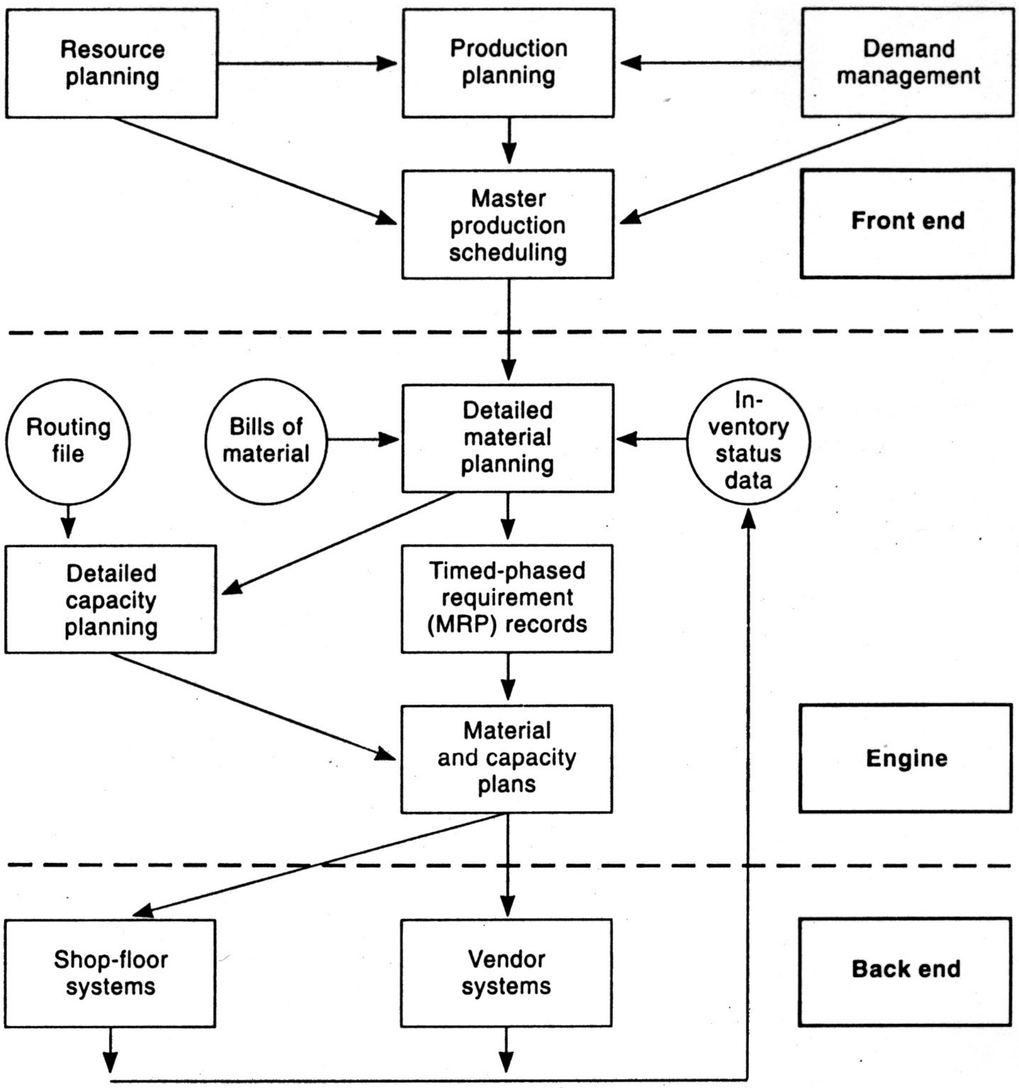
5. Frequenza di ripianificazione

▷ Per loro natura i MPS non vengono implementati esattamente
▷ L'MPS è su un lungo orizzonte, ma viene ricalcolato più volte in questo periodo
▷ Spesso i piani sono usati in *rolling mode*: si sviluppa un piano per T periodi e si inizia a realizzarlo, ma dopo $\tau \ll T$ periodi si calcola un nuovo piano (ovvero solo i primi τ periodi di ogni piano sono utilizzati)
▷ In pratica i piani sono rivisti di frequenza data l'instabilità dei sistemi e l'aleatorietà dei dati

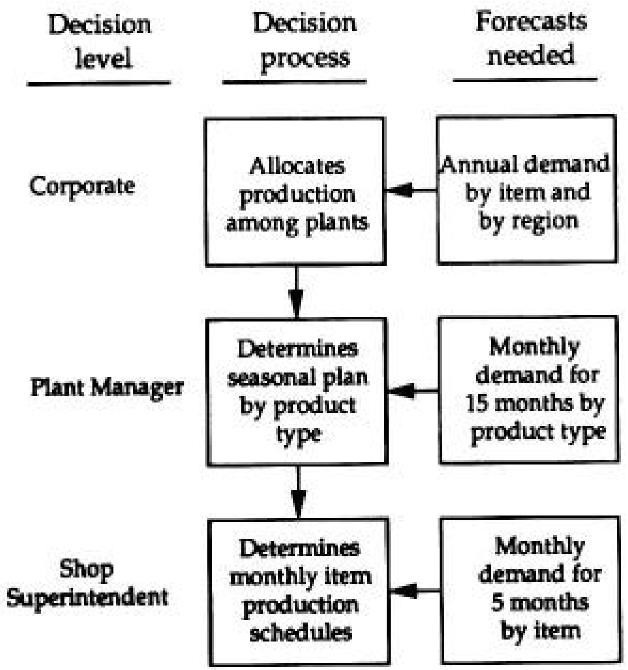
6. Numerosità e struttura dei piani

▷ Spesso le aziende non hanno un solo piano, ma molti piani a diversi livelli di aggregazione
▷ I diversi piani vengono utilizzati a diversi livelli decisionali e hanno diversi orizzonti temporali e unità di misura

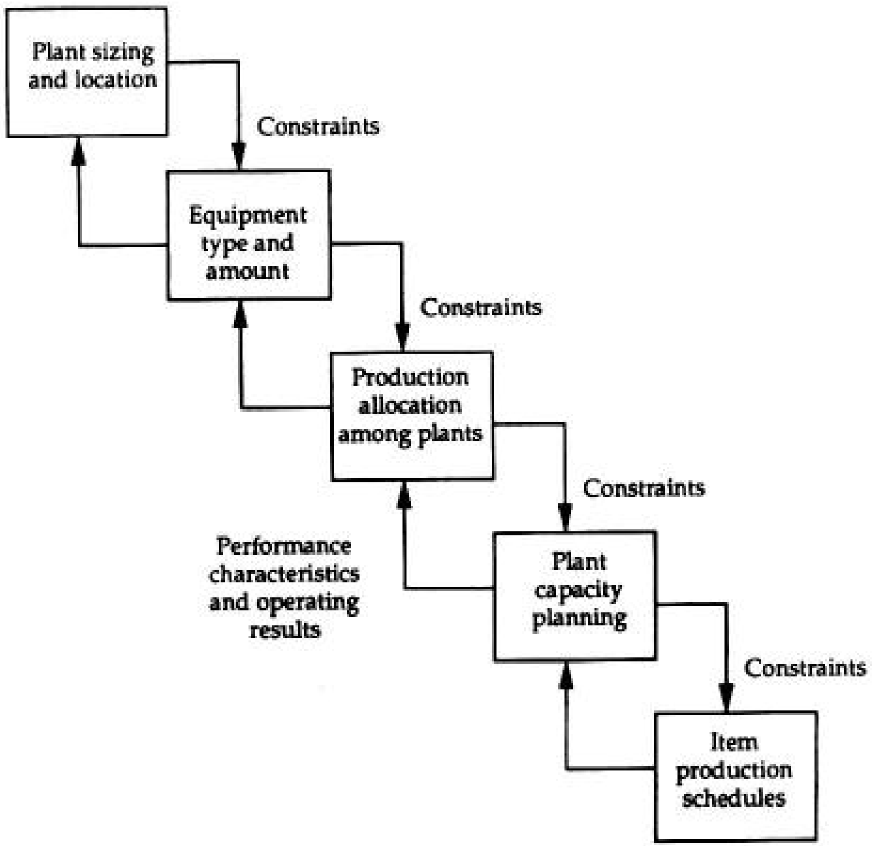
Le fasi della Pianificazione



Manufacturing Planning and Control System. Vollmann, Berry and Whybark (1992), 3rd edition, Irwin, Homewood

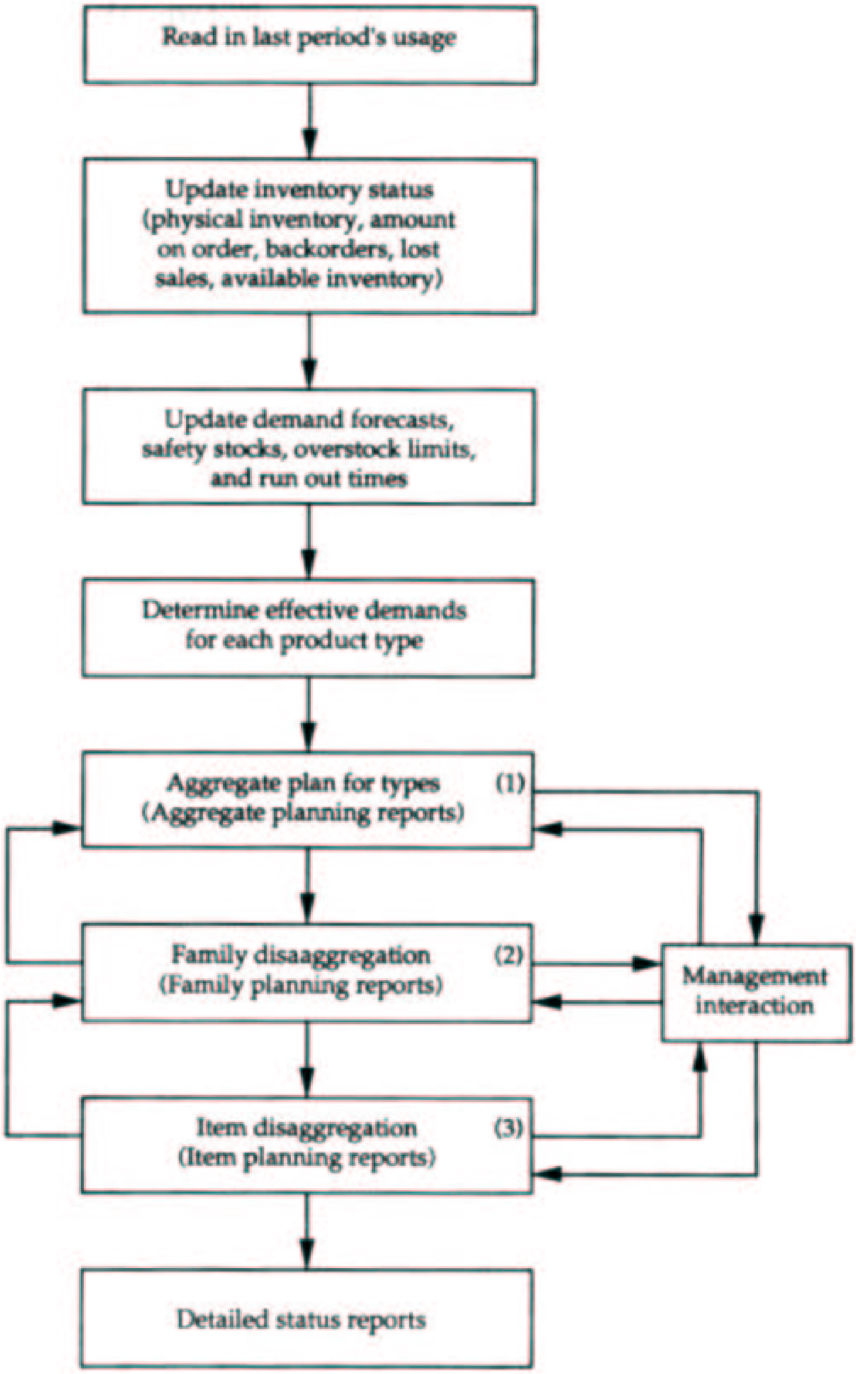


Hierarchical Planning Process



Production Planning Decision Hierarchy

Hierarchical Production Planning, Bitran and Tirupati (1993) , in Logistics Of Production and Inventory, Graves, Rinnoy Kan and Zipkin (eds), Handbooks of Operations Research and Management Science, North-Holland



Hierarchical Production Planning, Bitran and Tirupati (1993) , in Logistics Of Production and Inventory, Graves, Rinnoy Kan and Zipkin (eds), Handbooks of Operations Research and Management Science, North-Holland

Modelli aggregati

➢ Si tenta di rappresentare nei modelli i principali elementi di costo e le principali risorse

➢ Vi sono vari modelli a seconda delle diverse 'filosofie' di rappresentazione dei costi e delle diverse aggregazioni

➢ La stima dei costi non è facile. In particolare è complicato dare costi che *omogenizzano* quantità non monetarie (es. costo di non soddisfacimento di un ordine)

➢ Si devono prevedere modalità per passare dai piani aggregati a quelli disaggregati

➢ I modelli sono normalmente molto grandi e li si risolve tramite *rilassamenti lagrangiani* o *decomposizione*

Nel seguito supponiamo di avere definito:

n tipi di prodotti
T unità temporali
d_{it} domanda per ogni prodotto i nel periodo t

Modello 1

$$\min z = \sum_{i=1}^{n} \sum_{t=1}^{T} (cv_{it} x_{it} + ci_{it} I_{it}) + \sum_{t=1}^{T} (cw_t W_t + co_t O_t)$$

$$x_{it} + I_{i,t-1} = d_{it} + I_{it} \qquad \forall i, \forall T \quad (1)$$

$$\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t \qquad \forall t \quad (2)$$

$$W_t \leq WCap_t \qquad \forall t \quad (3)$$

$$O_t \leq OCap_t \qquad \forall t \quad (4)$$

$$x_{it}, I_{it} \geq 0 \qquad \forall i, \forall t$$

$$W_t, O_t \geq 0 \qquad \forall t$$

Variabili

x_{it} = numero prodotti famiglia i realizzati nel periodo t

I_{it} = prodotti famiglia i in magazzino alla fine del periodo t

W_t = ore di lavoro regolare nel periodo t

O_t = ore di lavoro straordinario nel periodo t

Costanti

k_i = ore necessarie per realizzare un prodotto famiglia i

$WCap_t$ = ore di lavoro regolari disponibili nel periodo t

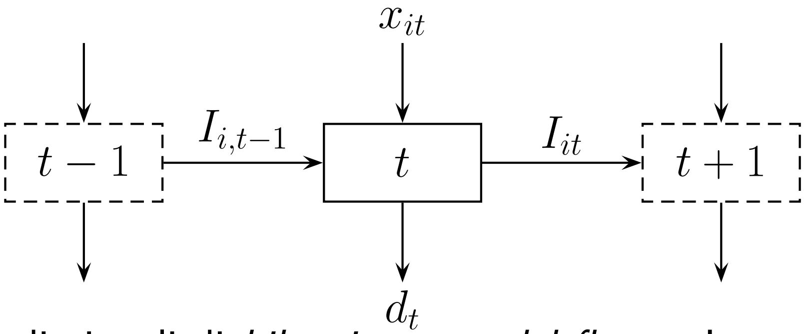
$OCap_t$ = ore di lavoro regolari disponibili nel periodo t

$c\star_{it}$, $c\star_t$= costi

$$x_{it} + I_{i,t-1} = d_{it} + I_{it} \quad \forall i, \forall T \qquad (1)$$

Si deve 'soddisfare' la domanda del prodotto i, periodo t, con:
- produzione del periodo t
- scorte di magazzino alla fine del periodo $t-1$

Le quantità di cui sopra devono essere sufficienti a coprire anche la giacenza di magazzino alla fine del periodo t



➤ Si tratta di vincoli di *bilanciamento del flusso*. Le variabili I partono da I_0, ovvero la Giacenza a magazzino all'inizio della programmazione.

$$\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t \;\forall t \qquad (2)$$

Le ore necessarie per realizzare tutti i prodotti devono essere coperte con lavoro regolare o straordinario.
➤ Sono vincoli di tipo *knapsack*
N.B. Si suppone avere manodopera omogenea: chiunque può lavorare qualunque prodotto. Nella disaggregazione si potranno avere problemi !

$$W_t \leq WCap_t, \;\forall t \qquad (3)$$
$$O_t \leq OCap_t, \;\forall t \qquad (4)$$

Le ore regolari e di straordinario non devono eccedere le massime disponibilità dichiarate.

Rilassamenti

Eliminazione dei vincoli (2) $\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t$

La coppia (x, I) e la coppia (W, O) non sono più legate: il problema si separa in due problemi indipendenti

$$
\begin{aligned}
(P1)\ \ \min z = &\sum_{i=1}^{n}\sum_{t=1}^{T}(cv_{it}x_{it} + ci_{it}I_{it}) & \\
&x_{it} + I_{i,t-1} = d_{it} + I_{it} & \forall i, \forall T \\
&x_{it}, I_{it} \geq 0 & \forall i, \forall t \\
(P2)\ \ \min z = &\sum_{t=1}^{T}(cw_t W_t + co_t O_t) & \\
&W_t \leq WCap_t & \forall t \\
&O_t \leq OCap_t & \forall t \\
&W_t, O_t \geq 0 & \forall t
\end{aligned}
$$

P1 è un problema di *flusso di costo minimo*, mentre P2 ha soluzione banale $W_t = O_t = 0\ \forall t$.
Rilassando in maniera lagrangiana i vincoli (2) la f.o di P1 diventa

$$\min z = \sum_{i=1}^{n}\sum_{t=1}^{T}((cv_{it} + \lambda_t k_i)x_{it} + ci_{it}I_{it}) \qquad (\lambda_t \geq 0)$$

e quella di P2

$$\min z = \sum_{t=1}^{T}((cw_t - \lambda_t)W_t + (co_t - \lambda_t)O_t) \qquad (\lambda_t \geq 0)$$

P1 rimane un problema di flusso; P2 si risolve per ispezione

Modello 2

$$\min z = \sum_{i=1}^{n}\sum_{t=1}^{T}(cv_{it}x_{it} + ci_{it}I_{it} + cb_{it}B_{it}) + \sum_{t=1}^{T}(cw_t W_t + co_t O_t + ca_t A_t + cl_t L_t)$$

$$x_{it} + I_{i,t-1} + B_{it} = d_{it} + B_{i,t-1} + I_{it} \qquad \forall i, \forall T \qquad (5)$$

$$\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t \qquad \forall t \qquad (6)$$

$$W_t = W_{t-1} + A_{t-1} - L_t \qquad \forall t \qquad (7)$$

$$O_t \leq \alpha W_t \qquad \forall t \qquad (8)$$

$$x_{it}, I_{it}, B_{it} \geq 0 \qquad \forall i, \forall t$$

$$W_t, O_t, H_t, F_t \geq 0 \qquad \forall t$$

Variabili

x_{it} = numero prodotti famiglia i realizzati nel periodo t

I_{it} = prodotti famiglia i in magazzino alla fine del periodo t

B_{it} = domanda prodotti famiglia i non soddisfatta nel periodo t

W_t = ore di lavoro regolare nel periodo t

O_t = ore di lavoro straordinario nel periodo t

A_t = personale assunto nel periodo t

L_t = personale licenziato nel periodo t

Costanti

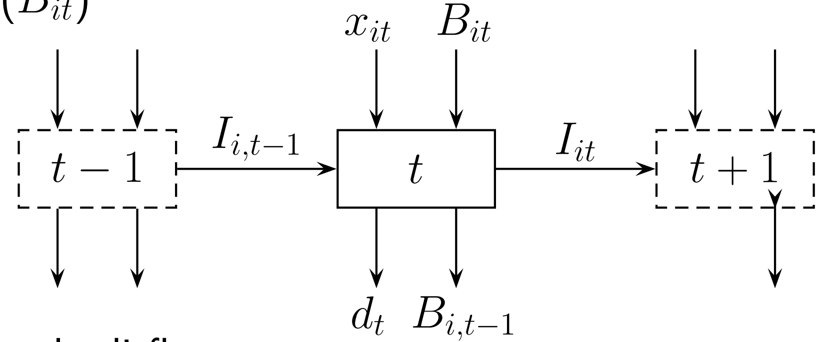
k_i = ore necessarie per realizzare un prodotto famiglia i

α $(0 < \alpha < 1)$ = coefficiente di massimo utilizzo di ore straordinarie (vs ore regolari)

$c\star_{it}$, $c\star_t$= costi

$$x_{it} + I_{i,t-1} + B_{it} = d_{it} + B_{i,t-1} + I_{it} \quad \forall i, \forall T \quad (5)$$

Nel periodo t si deve soddisfare la domanda del prodotto i e il non prodotto del periodo $t-1$ ($B_{i,t-1}$)
La richiesta di produzione può essere decurtata del non prodotto del periodo t (B_{it})



> È un vincolo di flusso

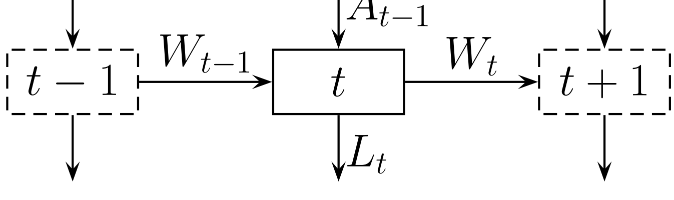
$$\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t \ \forall t \quad (6)$$

Le ore necessarie per realizzare tutti i prodotti devono essere coperte con lavoro regolare o straordinario

> È un vincolo di tipo knapsack con risorse omogenee

$$W_t = W_{t-1} + A_{t-1} - L_t \ \forall t \quad (7)$$

Il personale del periodo t è uguale a quello del periodo $t-1$
più le persone assunte in $t-1$, meno quelle licenziate in t



> È un vincolo di flusso

$$O_t \leq \alpha W_t \ \forall t \quad (8)$$

Le ore di straordinario devono essere una percentuale delle ore regolari

Rilassamenti

Eliminazione dei vincoli (6) $\sum_{i=1}^{n} k_i x_{it} \leq W_t + O_t \forall t$

Le variabili (x, I, B) e le variabili (W, O, A, L) non sono più legate: il problema si separa in due problemi indipendenti

$$(P2-1) \min z = \sum_{i=1}^{n} \sum_{t=1}^{T} (cv_{it}x_{it} + ci_{it}I_{it} + cb_{it}B_{it})$$

$$x_{it} + I_{i,t-1} + B_{it} = d_{it} + B_{i,t-1} + I_{it} \qquad \forall i, \forall t$$

$$x_{it}, I_{it}, B_{it} \geq 0 \qquad \forall i, \forall t$$

$$(P2-2) \min z = \sum_{t=1}^{T} (cw_tW_t + co_tO_t + ca_tA_t + cl_tL_t)$$

$$W_t = W_{t-1} + A_{t-1} - L_t \qquad \forall t$$

$$O_t \leq \alpha W_t \qquad \forall t$$

$$W_t, O_t, A_t, L_t \geq 0 \qquad \forall t$$

P2-1 e P2-2 sono problemi di *flusso di costo minimo* (in P2-2 $O_t = 0 \; \forall t$)

Miglior LB si ottiene con un rilassamento lagrangiano

Modello 3: Lot sizing (gestione dei costi fissi di setup)

- Un solo prodotto
- m risorse produttive, ciascuna con capacità CAP_i;
- cs_i = costo di preparazione setup della risorsa i
- cv_{it} = costo unitario di produzione su risorsa i nel periodo t

$$y_i = \begin{cases} 1 & \text{se la risorsa } i \text{ è utilizzata} \\ 0 & \text{altrimenti} \end{cases}$$

x_{it} = prodotti realizzati sulla risorsa i nel periodo t

$$\min \sum_{i=1}^{m} \sum_{t=1}^{T} cv_{it} x_{it} + \sum_{i=1}^{m} cs_i y_i \quad (1)$$

$$s.t. \sum_{i=1}^{m} x_{it} \geq d_t \qquad t = 1, \ldots, T \quad (2)$$

$$\sum_{t=1}^{T} x_{it} \leq CAP_i y_i \qquad i = 1, \ldots, m \quad (3)$$

$$x_{it} \geq 0, \text{intere }, y_i \in \{0, 1\} \quad (4)$$

$$\min \sum_{i=1}^{m} \sum_{t=1}^{T} cv_{it} x_{it} + \sum_{i=1}^{m} cs_i y_i \tag{1}$$

$$s.t. \sum_{i=1}^{m} x_{it} \geq d_t \qquad t = 1, \ldots, T \tag{2}$$

$$\sum_{t=1}^{T} x_{it} - CAP_i y_i \leq 0 \qquad i = 1, \ldots, m \tag{3}$$

$$x_{it} \geq 0, \text{intere }, y_i \in \{0, 1\} \tag{4}$$

- La funzione obiettivo (1) minimizza la somma dei costi di produzione e trasporto (prime due sommatorie) e dei costi di setup;
- i vincoli (2) impongono di soddisfare tutte le domande
- i vincoli (3) impongono i limiti di capacità delle risorse e il legame tra x_{it} e y_i
- i vincoli (4) impongono la non negatività di x e booleanità di y

Rilassamenti lagrangiani

Si possono ottenere rilassamenti lagrangiani:

- rilassando i vincoli (2)
- rilassando i vincoli (3)

➤ rilassando i vincoli (2) in si ottiene:

$$L1(\lambda) = \min \sum_{i=1}^{m} \sum_{t=1}^{T} cv_{it} x_{it} + \sum_{i=1}^{m} cs_i y_i + \sum_{t=1}^{T} \lambda_t (d_t - \sum_{i=1}^{m} x_{it}) \quad (5)$$

$$s.t. \sum_{t=1}^{T} x_{it} - CAP_i y_i \leq 0 \quad i = 1, \ldots, m \quad (6)$$

$$x_{it} \geq 0, \text{intere } , y_i \in \{0, 1\}, \lambda_t \geq 0 \quad (7)$$

$$L1(\lambda) = \sum_{t=1}^{n} \lambda_t d_t + \min \left(\sum_{i=1}^{m} \sum_{t=1}^{T} (cv_{it} - \lambda_t) x_{it} + \sum_{i=1}^{m} cs_i y_i \right)$$
$$s.t.(6) \text{ and } (7) \quad (8)$$

➤ rilassando i vincoli (3) si ottiene un problema con sole x nei vincoli

$$L2(\pi) = \min \sum_{i=1}^{m} \sum_{t=1}^{T} cv_{it} x_{it} + \sum_{i=1}^{m} cs_i y_i + \sum_{i=1}^{m} \pi_i (\sum_{t=1}^{T} x_{it} - CAP_i y_i) \quad (9)$$

$$s.t. \sum_{i=1}^{m} x_{it} \geq d_t \quad t = 1, \ldots, T \quad (10)$$

$$x_{it} \geq 0, \text{intere } , y_i \in \{0, 1\}, \pi_i \geq 0 \quad (11)$$

$$L2(\pi) = \min \left(\sum_{i=1}^{m} \sum_{t=1}^{T} (cv_{it} + \pi_i) x_{it} + \sum_{i=1}^{m} (cs_i - \pi_i CAP_i) y_i \right)$$
$$s.t.(10) \text{ and } (11) \quad (12)$$

Soluzione di L1

$$L1(\lambda) = \sum_{t=1}^{T} \lambda_t d_t + \min \left(\sum_{i=1}^{m} \sum_{t=1}^{T} (cv_{it} - \lambda_t) x_{it} + \sum_{i=1}^{m} cs_i y_i \right)$$

$$s.t. \sum_{t=1}^{n} x_{it} - CAP_i y_i \leq 0 \quad i = 1, \ldots, m$$

$$x_{it} \geq 0, \text{intere } , y_i \in \{0,1\}, \lambda_t \geq 0$$

- Non ci sono domande da soddisfare
  $\Rightarrow$ la soluzione banale $x = 0; y = 0$ è ammissibile (non lo è per il problema originale)
  $\Rightarrow$ la soluzione ottima di $L1$ ha valore $\sum_{t=1}^{T} \lambda_t d_t + \Theta$ con $\Theta \leq 0$
- Se decido di usare una risorsa i pago un costo cs_i, allora mi conviene usarla solo se i termini in x portano ad un calo del valore della f.o.

Se $(cv_{it} - \lambda_t) \geq 0$ non mi conviene usare la risorsa i per il periodo t
Se $(cv_{it} - \lambda_t) < 0$ può convenire attivare la risorsa i

$$\Delta_i = \min_{t=1,\ldots,T} (\ cv_{it} - \lambda_t\ )$$

sia $t(i) : cv_{i,t(i)} - \lambda_{t(i)} = \Delta_i$ (il periodo più profittevole per la risorsa i)

se $cs_i < -\Delta_i CAP_i$ conviene attivare la risorsa i dedicando tutta la sua capacità al periodo $t(i)$

$\Rightarrow y_i = 1$, $x_{i,t(i)} = CAP_i$, $x_{it} = 0\ \forall t : t \neq t(i)$

se $cs_i \geq -\Delta_i CAP_i \Rightarrow y_i = 0$, $x_{it} = 0\ \forall t$

Soluzione di L2

$$L2(\pi) = \sum_{i=1}^{m} \pi_i + \min \left(\sum_{i=1}^{m} \sum_{t=1}^{T} (cv_{it} + \pi_i) x_{it} + \sum_{i=1}^{m} (cs_i - \pi_i CAP_i) y_i \right)$$

$$s.t. \sum_{i=1}^{m} x_{it} \geq d_t \quad t = 1, \ldots, T$$

$$x_{it} \geq 0, \text{intere } , y_i \in \{0, 1\}, \pi_i \geq 0$$

- Le variabili y non sono vincolate in alcun modo
  $\Rightarrow y_i = 0$ se $(cs_i - \pi_i CAP_i) \geq 0$; $y_i = 1$ se $(cs_i - \pi_i CAP_i) < 0$
- se $\exists\ \ell : (cv_\ell + \pi_\ell) < 0$ il problema è illimitato
  $\Rightarrow$ è ottima la soluzione con $x_\ell = +\infty$
- se $(cv_{it} + \pi_i) \geq 0\ \forall i$

$$\overline{\Delta}_t = \min \sum_{i=1}^{m} (cv_{it} + \pi_i) x_{it}$$

sia $i(t) : cv_{i(t),t} + \pi_{i(t)} = \overline{\Delta}_t$ (la risorsa più favorevole per t)

La soluzione ottima, per ogni periodo t, è:
$x_{i(t),t} = d_t$, $x_{it} = 0\ \forall i \neq i(t)$

Esempio

$n = 2$, $m = 8$

i	cv_{it}		CAP_i	cs_i
	1	2		
1	21	73	14	40
2	45	41	17	45
3	86	50	8	27
4	105	35	6	20
5	110	54	12	39
6	42	22	9	30
7	38	26	13	42
8	63	65	15	46
d_t	31	28		

Lo stesso modello vale per:

- periodo temporale fisso
- n clienti dove ogni cliente j ha domanda d_j
- m stabilimenti di produzione
- cpt_{ij} = costo unitario produzione e trasporto da stabilimento i a cliente j

$$\begin{aligned}
&\min \sum_{i=1}^{m} \sum_{j=1}^{n} cpt_{ij} x_{ij} + \sum_{i=1}^{m} cs_i y_i \\
&s.t. \sum_{i=1}^{m} x_{ij} \geq d_j && j = 1, \ldots, n \\
&\sum_{j=1}^{n} x_{ij} - CAP_i y_i \leq 0 && i = 1, \ldots, m \\
&x_{ij} \geq 0, \text{intere}\ , y_i \in \{0, 1\}
\end{aligned}$$

Modello 4: costi di setup, sotto-utilizzo/carenza risorse

$$\min z = \sum_{i=1}^{n} \sum_{t=1}^{T} (ci_{it} I_{it} + cs_{it} y_{it}) + \sum_{k=1}^{K} \sum_{t=1}^{T} (cv_{kt} OT_{kt} + cu_{kt} UT_{kt})$$

$$\beta_i x_{i,t-L(i)} + I_{i,t-1} = d_{it} + I_{it} + \sum_{j=1}^{n} comp_{ij} x_{jt} \quad \forall i, t \tag{1}$$

$$\sum_{i=1}^{n} (k_{ki} x_{it} + s_{ki} y_{it}) + UT_{kt} - OT_{kt} = CAP_k \quad \forall k, t \tag{2}$$

$$x_{it} \leq M y_{it} \quad \forall i, t \tag{3}$$

$$x_{it}, I_{it}, UT_{kt}, OT_{kt} \geq 0$$

$$y_{it} \in \{0, 1\}$$

Variabili

x_{it} = numero prodotti famiglia i realizzati nel periodo t

y_{it} = 1 se si effettua un setup per prodotti famiglia i nel periodo t

OT_{kt} = eccesso di capacità sulla risorsa k nel periodo t

UT_{kt} = carenza di capacità sulla risorsa k nel periodo t

Costanti

β_i = rendimento della produzione del prodotto i

$L(i)$ = Lead Time per la produzione del prodotto i

k_{ki} = ore necessarie per realizzare un prodotto di famiglia i sulla risorsa k

$comp_{ij}$ = numero di prodotti famiglia i usati per realizzare un prodotto famiglia j

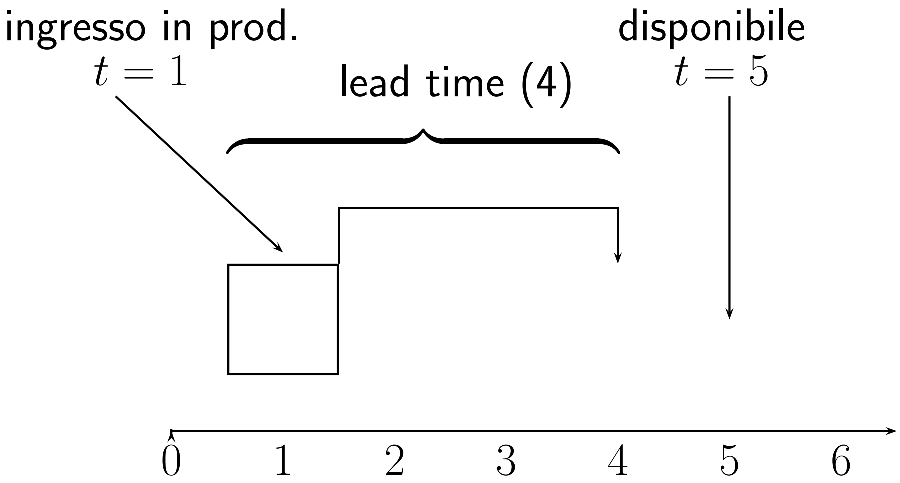
s_{ki} = tempo di setup per la risorsa k prodotto famiglia i

CAP_k = capacità risorsa k

M = numero $\gg 0$

$$\beta_i x_{i,t-L(i)} + I_{i,t-1} = d_{it} + I_{it} + \sum_{j=1}^{n} comp_{ij} x_{jt} \ (1)$$

Per ogni famiglia i di prodotti si deve soddisfare la domanda, la scorta di magazzino alla fine del periodo t e la richiesta di questo prodotto per realizzare gli altri prodotti di famigliaP j tramite i prodotti disponibili nel periodo t e la giacenza di magazzino al periodo $t-1$.
N.B. Il modello rende disponibili all'istante τ i prodotti che entrano in produzione all'istante $\tau - L(i)$. Il carico delle risorse viene considerato solo nella "bocca d'ingresso del manufacturing.



$$\sum_{i=1}^{n}(k_{ki}x_{it} + s_{k1}y_{it}) + UT_{kt} - OT_{kt} = CAP_k \ (2)$$

Le ore richieste per la produzione ed il setup devono essere commisurate alla capacità di ogni risorsa (si generano eccessi o carenze di capacità che vengono pesati nella funzione obiettivo)

$$x_{it} \leq My_{it} \ (3)$$

Si può produrre il bene i nel periodo t solo se si effettua il corrispondente setup

Modello disaggregato

> Le quantità aggregate per famiglia (x_{it}) devono essere separate nei singoli prodotti per definire la produzione effettiva del periodo
> Si deve generare un piano di dettaglio coerente con il piano globale
> Non sempre è possibile avere piani coerenti in quanto il piano aggregato trascura vincoli che devono essere inclusi nel piano particolare

$J(i)$ = insieme dei prodotti di dettaglio che appartengono al tipo i

y_{jt} = quantità del j-esimo prodotto da realizzare nel periodo t

cs_j = costo di setup per il prodotto j

$$\min \sum_{j \in J(i)} cs_j \frac{d_j}{y_{jt}}$$

$$\sum_{j \in J(i)} y_{jt} = x_{it} \tag{4}$$

$$lb_{jt} \leq y_{jt} \leq ub_{jt} \qquad j \in J(i) \tag{5}$$

NOTE

$$lb_{jt} = \max(0, d_{jt} + ss_{jt} - ai_{jt}) \quad ub_{jt} = \max(0, os_{jt} + d_{jt} - ai_{jt})$$

per il prodotto j, periodo t:
d_{jt} = domanda; ss_{jt}= scorta di sicurezza;
ai_{jt}= disponibilità a magazzino; os_{jt}= massima scorta

N.B.
- É prefissato il periodo che si considera (t)
- i valori delle x sono dati del problema

$$\min \sum_{j \in J(i)} cs_j \frac{d_j}{y_{jt}}$$

Definita la domanda annuale d_j che sarà soddisfatta dai vincoli (4) si vuole minimizzare il costo di setup
È la stessa formula che si usa per il lotto economico (EOQ):
(d_j/y_{jt} è il numero di riprese della produzione di j)

$$\sum_{j \in J(i)} y_{jt} = x_{it} \qquad (4)$$

Il totale della produzione dei prodotti in $J(i)$ deve essere uguale a quanto previsto per la famiglia i

$$\max(0, d_{jt} + ss_{jt} - ai_{jt}) = lb_{jt} \leq y_{jt}$$

La produzione del periodo, insieme alla scorta di magazzino deve soddisfare la domanda, lasciando disponibile una scorta di sicurezza a magazzino

$$y_{jt} \leq ub_{jt} = \max(0, os_{jt} + d_{jt} - ai_{jt})$$

La scorta di magazzino a fine periodo non deve superare un limite massimo di stock (os_{jt}).

Il problema delle incongruenze

➤ La disaggregazione di un MPS non sempre é possibile

Dati di input per una famiglia di 2 prodotti

	domanda			magazzino
Prodotto	1	2	3	iniziale
1	5	17	30	9
2	3	12	30	20
domanda per MPS	8	29	60	29

Soluzione ammissibile per MPS

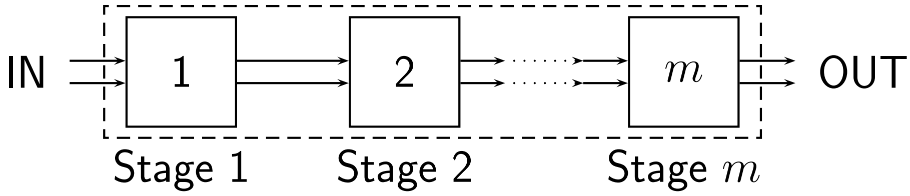
	periodi		
Variabile	1	2	3
x (produzione)	0	8	60
I (mag. fine periodo)	21	0	0
domanda	8	29	60

La domanda del primo periodo viene soddisfatta con la giacenza iniziale di magazzino
Con questa scelta a fine primo periodo rimangono 4 unitá del prodotto 1 e 18 del prodotto 2

$\Rightarrow$ non é possibile soddisfare la domanda del prodotto 1 periodo 2 !

Linea di produzione

➤ Sistema produttivo caratterizzato dalla *sequenzialitá* delle operazioni



- Ogni prodotto richiede n lavorazioni
- Ogni stazione (stage) puó eseguire qualunque operazione
- Processing time indipendente dalla stazione che la esegue (p_j)
- Carico lavorativo di ogni stazione C

Il *carico di lavoro* C di una stazione é il tempo che la stazione lavora in un ciclo di produzione, ovvero nel tempo che occorre per realizzare un prodotto

C é determinato dalla stazione piú carica

In pratica a regime nella linea vi sono prodotti in ogni stazione. Ad ogni ciclo esce un prodotto finito dall'ultima stazione.

> Problema 1 (dimensionamento):
assegnare le lavorazioni alle stazioni minimizzando il numero di stazioni

$$x_{ij} = \begin{cases} 1 & \text{se l'operazione } j \text{ é eseguita dalla stazione } i \\ 0 & \text{altrimenti} \end{cases}$$

$$y_i = \begin{cases} 1 & \text{se la stazione } i \text{ é usata} \\ 0 & \text{altrimenti} \end{cases}$$

$$\min \sum_{i=1}^{U} y_i$$

$$\sum_{i=1}^{U} x_{ij} = 1 \qquad j = 1 \ldots, n \qquad (6)$$

$$\sum_{j=1}^{n} p_j x_{ij} \leq C y_i \qquad i = 1 \ldots, U \qquad (7)$$

$$x_{ij} \in \{0, 1\}$$

U = upper bound sul numero di stazioni necessario (si puó porre $U = n$)

(6) ogni operazione deve essere assegnata a una stazione

(7) il tempo di lavoro di una stazione non puó eccedere il carico di lavoro

Precedenze tra operazioni

$$A = \{(k, k') : k \prec k',\ k, k' = 1, \ldots, n\}$$

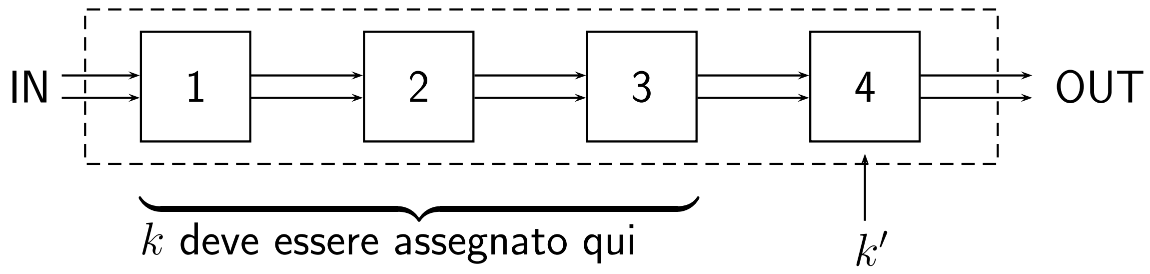
Se $(k, k') \in A$ l'operazione k deve essere eseguita prima di k'

$$x_{ik'} \leq \sum_{\ell=1}^{i-1} x_{\ell k} \quad i = 1, \ldots, U$$

k' é assegnata alla stazione i

$\Downarrow$

k deve essere assegnata ad una stazione in $1 \ldots i-1$



> **Problema 2 (bilanciamento)**:
dato il numero m di stazioni assegnare le lavorazioni alle stazioni bilanciando i carichi

Il carico ideale è $\mu = \dfrac{\sum_j p_j}{m}$: si vuole minimizzare lo scostamento da μ.

Sia $C_i = \sum_{j=1}^{n} p_j x_{ij}$ il carico del processore i

$$\min \sum_{i=1}^{m} |C_i - \mu|$$

La funzione *valore assoluto* non è lineare : occorre linearizzare con

$$\begin{aligned}
&\min \sum_{i=1}^{m} z_i \\
&\qquad z_i \geq C_i - \mu \qquad && i = 1, \ldots, m \\
&\qquad z_i \geq \mu - C_i \qquad && i = 1, \ldots, m \\
&\qquad \ldots
\end{aligned}$$

In alternativa si minimizza il *makespan* C_{max}

$$\begin{aligned}
&\min z \\
&\quad \sum_{i=1}^{m} x_{ij} = 1 \qquad && j = 1 \ldots, n \\
&\quad \sum_{j=1}^{n} p_j x_{ij} \leq z \qquad && i = 1 \ldots, m \\
&\quad x_{ij} \in \{0, 1\}
\end{aligned}$$

Ulteriori vincoli

- Coppie di operazioni da eseguire su stessa stazione (S)

$$x_{ik} = x_{ik'} \quad (k, k') \in S$$

- Coppie di operazioni da eseguire su stessa stazioni diverse (D)

$$x_{ik} + x_{ik'} \leq 1 \quad (k, k') \in D$$

Notazione

I sistemi di schedulazione sono caratterizzati da numerosi elementi:

- *macchine* (processori) o *risorse*
- *configurazioni* e *caratteristiche* delle macchine
- livello di automatizzazione
- sistemi di movimentazione
- ...

Chiameremo genericamente *macchina* ogni risorsa (che può essere un tornio, un computer, un operatore, un reparto, una stanza, etc.)

Le macchine devono eseguire dei *lavori* (job) che possono consistere in una sola operazione o in una serie di operazioni che compongono un processo produttivo (ciclo)

Le specifiche caratteristiche delle macchine, dei lavori e della funzione obiettivo determinano problemi di schedulazione di natura molto differente

Dati di input

Processing time (p_{ij}) tempo di esecuzione del job j sull macchina i. Se l'esecuzione è indipendente dalla macchina p_j

Release date (r_j) data minima di inizio della lavorazione del job j

Due date (d_j) data di consegna attesa. A volte si hanno *deadline* ovvero date di consegna che non possono essere superate

Setup time (s_{jk}) tempo di allestimento della macchina per passare dal job j al job k

Precedenze $(i \prec j)$ indica che il job j può iniziare solo se è finito il job i

Output della schedulazione

Completion time (C_{ij}) istante di completamento dell'operazione i del job j

Starting time (S_{ij}) istante di inizio dell'operazione i del job j $(S_{ij} = C_{ij} - d_{ij})$

Se il problema lo richiede si fornisce in output anche la scelta della macchina che processa ciascun operazione

Problemi a macchina singola

➤ Molti sistemi di produzioni vengono assimilati a singoli processori

- quando una macchina è il "collo di bottiglia che determina la velocità di un intero sistema
- quando un reparto può essere aggregato e studiato come un unica unità

Problemi a macchina singola si usano come sottoproblemi da ottimizzare nella risoluzione di problemi più complessi.

Vi sono numerosi problemi ben studiati in letteratura.

Alcuni problemi si risolvono con algoritmi semplici di ordinamento dei job
(es. SPT *Shortest Processing Time*, EDD *Earliest Due Date*)

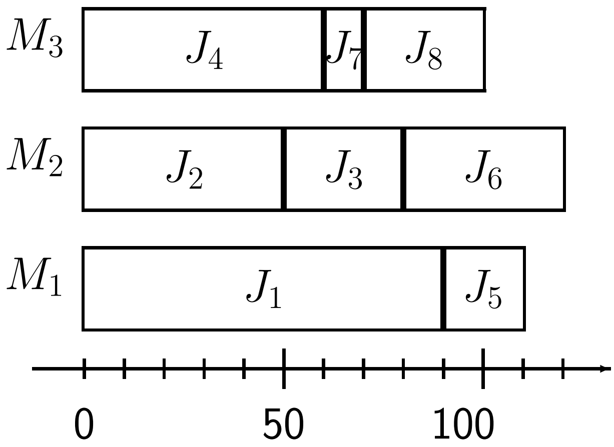
Altri problemi richiedono tecniche di enumerazione implicita.

I job sono di norma singole operazioni.

Problemi a macchine parallele

Vi sono m macchine che possono eseguire i job.
I job sono si norma operazioni singole.

Occorre: (a) assegnare i job alle macchine; (b) determinare gli istanti di inizio e fine dei job per ogni macchina



Se le macchine sono *identiche* il tempo di processamento di un job è indipendente dalla macchina (p_j)

Se le macchine sono *uniformi* il tempo di processamento di un job è il rapporto tra il tempo di processamento "standard" del job e la velocità della macchina (rappresenta problemi con macchine dello stesso tipo, ma di diversa "classe", es. processore Pentium III, 500Mhz, Pentium 4, 2Ghz)

Se le macchine sono comletamente diverse (*non correlate*) il tempo di processamento dipende dalla coppia macchina-job (p_{ij}) (ad esempio se le macchine sono dei programmatori uno sarà molto veloce a programmare in C++ e lento in Java ed un altro viceversa)

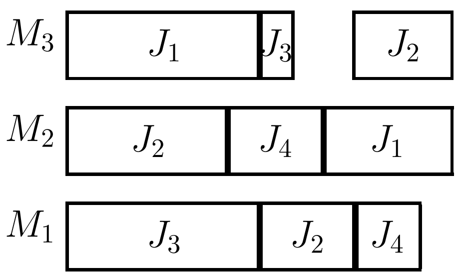
Shop problem

I job sono costituiti da più operazioni da eseguire su più macchine.

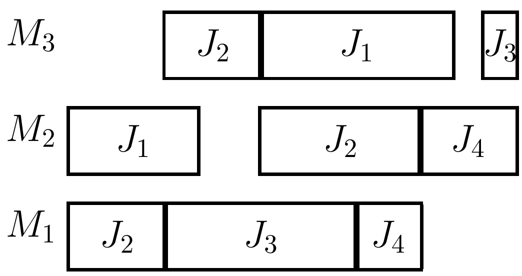
Di norma ogni job operazione di un job è assegnata a una macchina data

Open shop

- Non vi è alcuna precedenza tra le operazioni



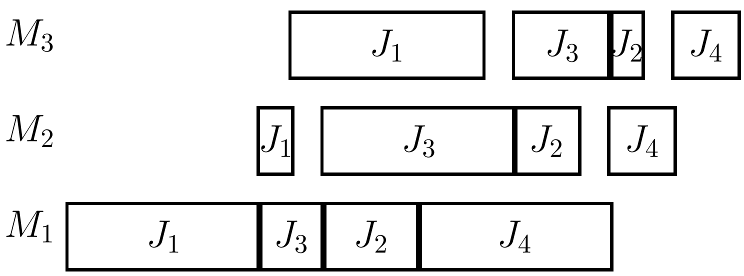
Soluzione 1



Soluzione 2

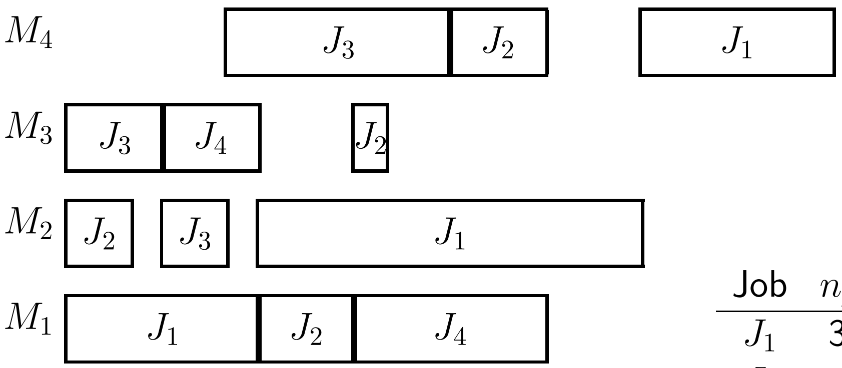
Flow shop

- Ogni job ha m operazioni da eseguire in sequenza sulle macchine $1, 2, \ldots, m$, rispettivamente



Job shop

➤ Ogni job j ha n_j operazioni da eseguire in sequenza sulle macchine $\mu_1, \mu_2, \ldots, \mu_{n_j}$, rispettivamente

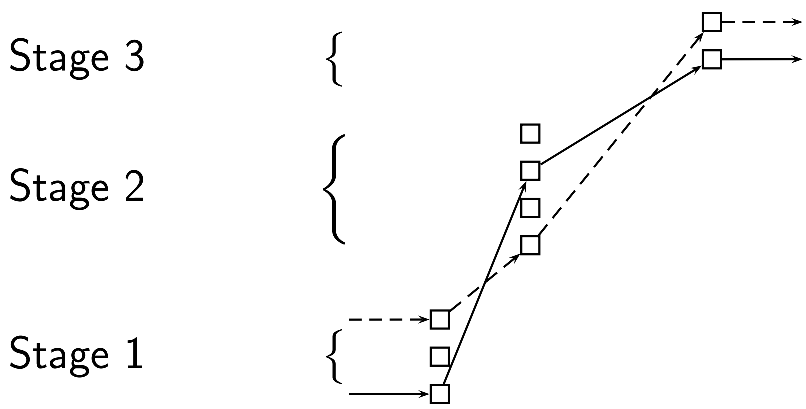


Job	n_j	μ_1	μ_2	μ_3	μ_4
J_1	3	1	2	4	
J_2	4	2	1	3	4
J_3	3	3	2	4	
J_4	2	3	1		

Flexible shop problem

➤ Shop problem con più macchine parallele in ogni stadio di produzione.

Le precedenze non sono più tra macchine, ma tra stadi di produzione



Funzioni obiettivo

> **Makespan** $C_{max} = \max_j(C_j)$ istante di completamento dell'ultima operazione

Determina il tempo necessario per terminare tutta la produzione.
E' una delle misure più studiate.

> **Tempo medio di attraversamento** $\sum_{j=1}^{n} C_j, \sum_{j=1}^{n} w_j C_j$

Viene usato come indicatore dei costi di Work-in-Process (quanità di materiale acquistato, ma non venduto in quanto in attesa di lavorazioni)

> **Valutazione dei ritardi di consegna**

$L_j = C_j - d_j$	*Lateness* (può essere positiva o negativa)
$T_j = max(0, C_j - d_j$	*Tardiness* (ritardo di consegna)
$E_j = max(0, d_j - d_j)$	*Earliness* (anticipo di consegna)

Si usa minimizzare sia il massimo che la somma delle misure

$$L_{max} = \max_j(L_j), \quad \sum_{j=1}^{n} L_j, \quad \sum_{j=1}^{n} w_j L_j$$

$$T_{max} = \max_j(T_j), \quad \sum_{j=1}^{n} T_j, \quad \sum_{j=1}^{n} w_j T_j$$

La earliness misura l'anticipo rispetto alla data di consegna. Diventa un costo in un ottica di Just-In-Time

Classificazione

Si utilizza una notazione a tre campi http:
(//www.mathematik.uni-osnabrueck.de/research/OR/class/)

$$\alpha|\beta|\gamma$$

$\alpha = (\alpha_1\alpha_2)$ caratteristiche delle macchine
$\alpha_1 \in \{\emptyset, P, Q, R, PMPM, QMPM, G, X, O, F, J\}$
$\alpha_2 \in \{\emptyset,$numero$\}$

$\beta =$ caratteristiche dei job
$(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6)$
β_1 : preemption
β_2 : precedenze
β_3 : release time
β_4 : caratteristiche dei processing time
β_5 : deadline
β_6 : batch

$\gamma =$ funzione obiettivo
data una funzione f_i è del tipo f_{max} oppure $\sum f_i$

Esempi

$P||C_{max} =$ macchine parallele identiche, minimizzare il makespan

$P|prec|C_{max} =$ macchine parallele identiche, precedenze generiche tra job, minimizzare il makespan

$1|r_j|L_{max} =$ macchina singola, release time, minimizzare la massima lateness

$J||C_{max} =$ Job shop, minimizzare il makespan

Tecniche di soluzione generali

- Dispatching rules (euristica)
- Metaeuristiche (euristica)
- Branch-and-Bound (esatta/euristica)
- Programmazione Dinamica (esatta/euristica)
- Constraint programming (esatta/euristica)
- Programmazione matematica (esatta/euristica)

I package commerciali sviluppati sino a metà degli anni '90 usavano essenzialmente dispatching rules.

Verso la fine degli anni '90 sono stati sviluppati package basati su metaeuristiche ed euristiche basate su applicazioni limitate nel tempo e nelle scelte di programmazione lineare, branch-and-bound o constraint programming.
I problemi reali sono di norma molto complessi e complicati da numerosissimi "strani" vincoli, quindi il ricorso alle euristiche è d'obbligo.

Anche gli algoritmi euristici possono però utilizzare tecniche molto sofisticate !

Dispatching rules

L'idea è quella di *ordinare secondo priorità* le operazioni da assegnare alle macchine.

Sono state studiate e applicate per vari decenni.
Regole statiche la priorità di un job viene definita una volta per tutte
Regole dinamiche la priorità di un job cambia durante la creazione della soluzione

Algoritmo sequenziale

Input: $n, m,$ caratteristiche dei job e delle macchine
Ordina i job secondo una priorità definita
for $j := 1$ to n **do**
  assegna il j-esimo job della lista al prima possibile, rispettando i vincoli (non sovrapposizione, precedenze, etc)
end for

Algoritmo parallelo

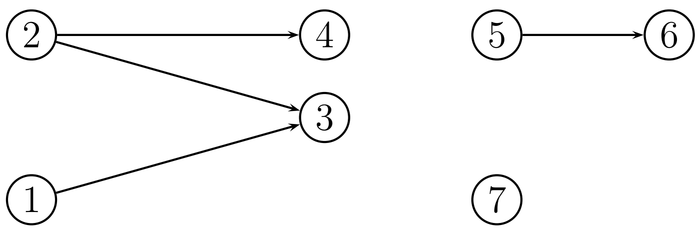
Input: $n, m,$ caratteristiche dei job e delle macchine
Ordina i job secondo una priorità definita
$t := 0$; $J = \{1, \ldots, n\}$
while $(J \neq \emptyset)$ **do**
  Determina l'insieme $S \subseteq J$ dei job che possono iniziare in t
  eventualmente riordina i job di S secondo priorità
  while $(S \neq \emptyset)$ **do**
    estrai il primo job j e schedulalo dall'istante t, se possibile
  end while
  $J := J \backslash S$; $t :=$minimo istante $> t$ cui può essere assegnato un job
end while

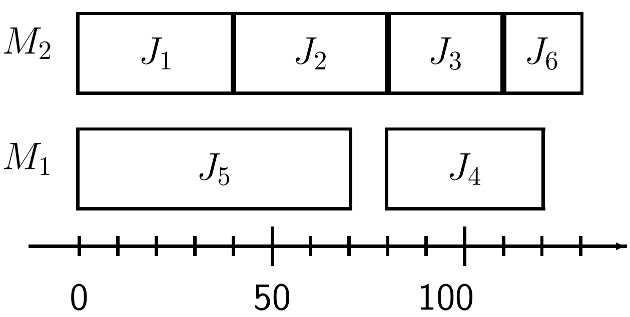
Esempio: $P|prec|C_{max}$

Regola LPT (Longest Processing Time)

6 job, 2 macchine parallele identiche
processing time $p_j = (40, 40, 30, 30, 70, 20)$

Precedenze

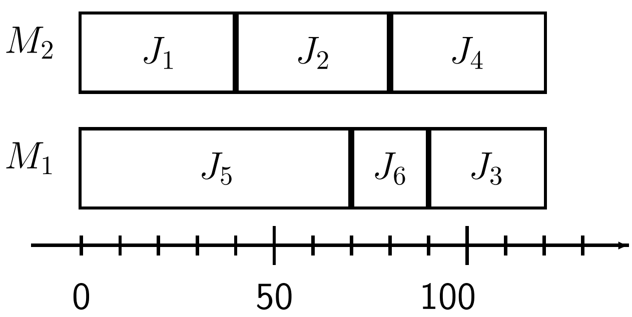




Algoritmo sequenziale

Ordine

$J_5 \quad J_1 \quad J_2 \quad J_3 \quad J_4 \quad J_6$



Algoritmo parallelo

t	S
0	$\{J_5, J_1, J_2\}$
40	$\{J_2\}$
70	$\{J_6\}$
80	$\{J_4, J_3\}$
90	$\{J_3\}$

Principali regole di ordinamento

ERD (Earliest Release Date): equivale a F.I.F.O.

LPT (Longest Processing Time): in ordine decrescente di durata

SPT (Shortest Processing Time): in ordine crescente di durata

WSPT (Weighted Shortest Processing Time): in ordine crescente del rapporto durata/peso

EDD (Earliest Due Date): in ordine crescente di due-date (o deadline)

MS (Minimum Slack): in ordine crescente di differenza tra l'istante attuale e la due date (se usato con algoritmo statico è come EDD)

SST (Shortest Setup Time) : in ordine crescente di setup

CP (Critical Path) : seleziona il job che inizia il cammino più lungo

	Regola	**Dati**	**Obiettivo**
dipende da	ERD	r_j	somma delle varianze
release date	EDD	d_j	L_{max}
o due date	MS	d_j	L_{max}
dipende da	LPT	p_j	bilanciamento di macchine identiche, makespan
processing	SPT	p_j	tempo medio di completamento
time	WSPT	p_j, w_j	tempo medio di completamento
	CP	p_j, prec	makespan
varie	SST	-	tempo medio di completamento, makespan

- Di norma le dispatching rules forniscono soluzioni euristiche. Per casi speciali forniscono la soluzione ottima.

$1 \vert\vert \sum C_j$	SPT (Smith's rule (1956))
$1 \vert\vert \sum w_j C_j$	WSPT (Smith's ratio rule (1956))
$1 \vert\vert L_{max}$	EDD (Jackson's rule (1955))

Efficacia delle tecniche di soluzione