

# Programmazione operativa

## GESTIONE DELLA PRODUZIONE

Uglietti Luigi

14 maggio 2009

[l.uglietti@wisefrogs.com](mailto:l.uglietti@wisefrogs.com)

# La programmazione operativa

La programmazione operativa (o di breve periodo, o scheduling), ha lo scopo di tradurre gli ordini di produzione (intesi come richieste) in ordini di produzione operativi (intesi come decisioni)

# Le fasi dello scheduling

- Allocazione delle operazioni:
  - Operazioni → Macchine
- Allocazione della produzione nel tempo
  - Operazioni → Istante temporale
- Sequenziamento lavori sulle macchine
  - Tempo → Macchine

# Le tecniche di scheduling (1/3)

Sono stati sviluppate numerose tecniche applicabili a diverse situazioni produttive, ma che hanno solo in minima parte trovato applicazione nella realtà. Questo a causa di fattori quali:

- Dati estremamente numerosi
- Dati soggetti a continui variazione
- Notevole aleatorietà dei tempi
- Multiobiettivo

# Le tecniche di scheduling (2/3)

Esistono fattori che hanno invece aumentato l'interesse verso queste problematiche

- Crescente grado di automazione e flessibilità dei sistemi produttivi
- Sviluppo di sistemi informativi di produzione sempre più integrati verso il basso
- Evoluzione degli Strumenti hardware e software

# Le tecniche di scheduling (3/3)

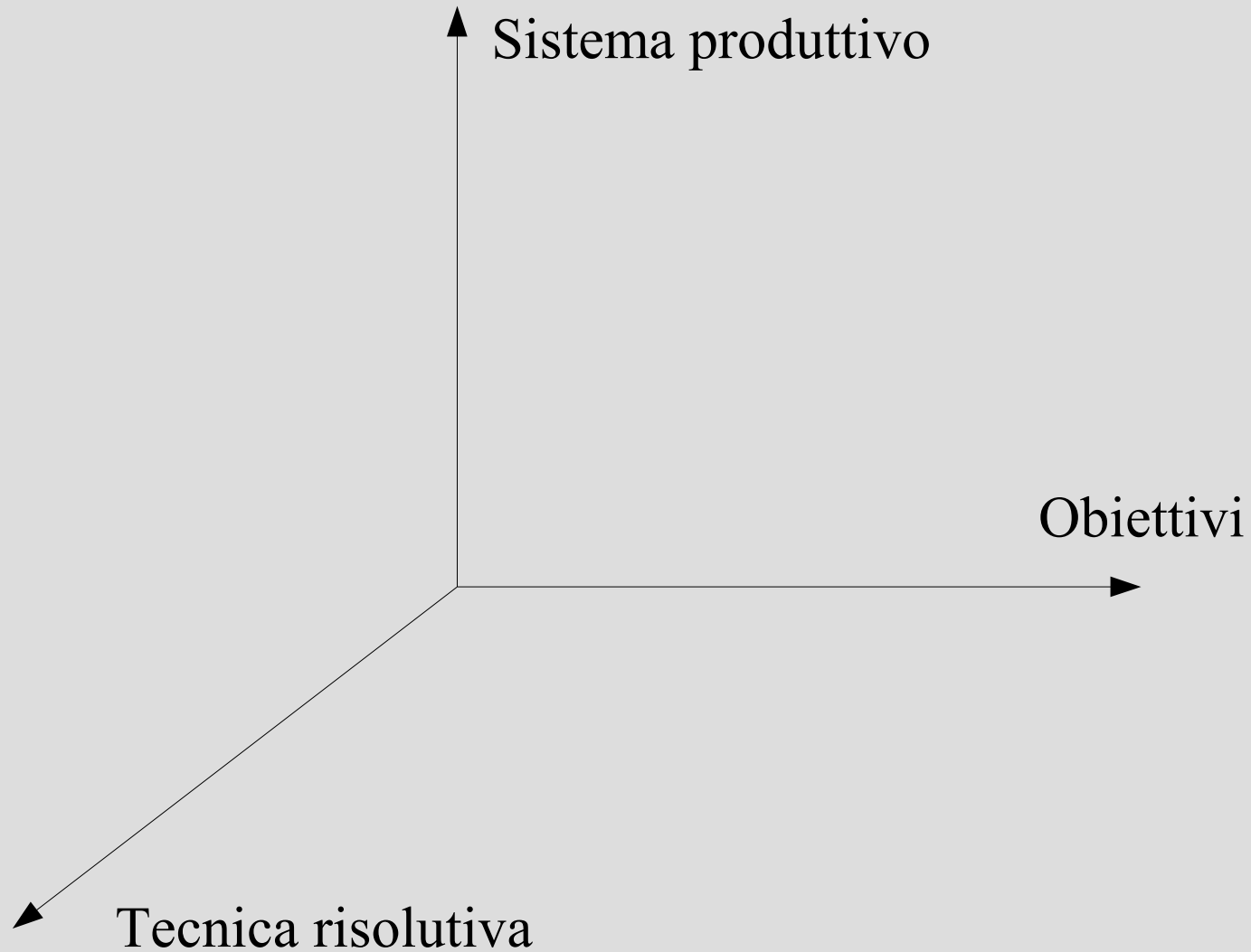
Alcune definizioni:

- **JOB:** lotto di pezzi (al limite unitario) lavorato nel sistema. L'unità elementare di programmazione è infatti il singolo item.
- **ROUTING:** è il ciclo di lavorazione (o tecnologico) del job; insieme ordinato di operazioni da eseguire per la lavorazione del job.

# Ipotesi nella programmazione operativa

- Risorse note e fisse
- Una sola risorsa critica
- Job completamente definiti
- Tempi di trasporto trascurabili
- Non si considera l'influenza di buffer
- Tutti i job assegnati devono essere compiuti
- Una macchina può lavorare un solo job alla volta
- Un job non può essere lavorato contemporaneamente su più macchine → un'operazione inizia solo al termine della precedente (*no lap-phasing*)
- Trascurati i costi di mantenimento a scorta

# Classificazione dei modelli





# Tecniche risolutive (1/6)

- Metodi di ottimizzazione analitici
  - Esiste una 'formula risolutiva'
  - E' possibile una ulteriore distinzione tra metodi continui e discreti (in relazione al dominio in cui si cercano le soluzioni)
  - Trovano impiego in situazioni in cui si voglia tenere in considerazione fattori non deterministici (Teoria del controllo)

# Tecniche risolutive (2/6)

- Metodi di ottimizzazione algoritmici
  - Sequenza di passi per costruire della soluzione
  - E' possibile distinguere tra algoritmi general purpose e specifici
  - Tra i metodi general purpose è possibile distinguere tra metodi enumerativi e metodi di calcolo

# Tecniche risolutive (3/6)

- Metodi euristici per sostituzione di obiettivo  
Rimpiazzano l'obiettivo del problema con un altro scelto in modo che:
  - Abbia una dipendenza meno complessa con le variabili decisionali
  - La soluzione così ottenuta sia 'buona' (anche se non ottima)

# Tecniche risolutive (4/6)

- Metodi euristici miopi
  - Trascurano alcune variabili decisionali del problema (senza implicare un peggioramento significativo della soluzione)
  - Si distinguono tra metodi miopi rispetto allo spazio (neighborhood search) e al tempo (hill-climbing)
  - Sono in genere altamente specifici rispetto al problema che risolvono

# Tecniche risolutive (5/6)

- Sistemi esperti
  - Basati sulle tecniche di intelligenza artificiale
  - Cercano di formalizzare il patrimonio di conoscenza tipico dei decisori dei sistemi produttivi
  - Fuzzy theory, Neural Networks, Algoritmi genetici
- Metodi interattivi
  - Soluzione ottenuta attraverso una serie di intuizioni, tentativi e correzioni da parte di un decisore umano

# Tecniche risolutive (6/6)

- Simulazione
  - A rigore non dovrebbero essere considerati metodi di risoluzione in quanto sono strumenti di verifica
  - Non generano soluzioni, ma permettono di valutare la bontà delle soluzioni proposte da altri metodi risolutivi
  - E' fondamentale il Design Of Experiment (DOE)

# Sistemi di produzione (1/6)

- Macchina singola
  - Unica risorsa produttiva (impianto schematizzabile come una singola macchina) es. produzioni di processo, produzioni in cui uno stadio è più critico
  - Job indipendenti o dipendenti (in relazione all'esistenza di relazioni di precedenza)
  - Preemption ammessa o non ammessa (interruzione e ripresa di un job dopo la lavorazione di altri job)

# Sistemi di produzione (2/6)

- Macchine parallele identiche
  - Unico stadio (come macchina singola), ma esistono un numero di macchine su cui i job possono essere lavorati indifferentemente)
  - Rispetto al caso di macchina singola, oltre al problema del sequenziamento, si aggiunge anche quello di allocazione dei job alle macchine



# Sistemi di produzione (3/6)

- Macchine parallele generiche
  - Come nel caso di macchine parallele identiche, esiste un certo numero di macchine in grado di lavorare i job
  - La caratterizzazione di ogni operazione dipende però dalla coppia job-macchina

# Sistemi di produzione (4/6)

- Open shop
  - Ciclo tecnologico che richiede l'intervento di più macchine successive, ma in cui l'ordine di esecuzione (routing) può essere qualsiasi
  - Ogni operazione è perciò identificata da una coppia di indici che identificano il job e la macchina

# Sistemi di produzione (5/6)

- Flow shop
  - Ciclo tecnologico che richiede l'intervento di più macchine diverse, ma in cui l'ordine di esecuzione delle operazioni è lo stesso per tutti i job
  - E' possibile distinguere tra flow shop puro o generico
  - Passing consentito o no passing

# Sistemi di produzione (6/6)

- Job shop
  - Ciclo tecnologico che richiede più macchine diverse e in cui l'ordine di esecuzione varia da job a job
  - La caratterizzazione dell'operazione è data da una terna di indici  $(j,i,k)$  in cui  $j$  è il job,  $i$  è la macchina e  $k$  l'operazione

# Obiettivi

- Qualunque sia la tecnica utilizzata, si tratta di massimizzare o minimizzare uno o più parametri di prestazione, funzione di variabili decisionali e esogene (vincoli)
- Le variabili decisionali sono costituite dall'assegnazione delle operazioni alle macchine e al timing
- I vincoli sono invece dettati dalle caratteristiche fisiche e tecnologiche dell'impianto

# Obiettivi

- Parametri di prestazione (1/3)
  - Tempi di lavorazione dei job  $(t_j, t_{ji}, t_{jik})$
  - Data di possibile inizio della produzione  $r_j$
  - Data di consegna  $d_j$
  - Data di ingresso del job nel sistema  $l_j$
  - Data di completamento  $C_j$
  - Il lateness  $L_j = C_j - d_j$
  - Il tardiness  $T_j = \max\{0, L_j\}$  (no anticipo)

# Obiettivi

- Parametri di prestazione (2/3)

- Il tempo di attraversamento (flowtime)

$$F_j = C_j - I_j$$

- Medio lateness

$$LM = \frac{\sum L_j}{N}$$

- Medio tardiness

$$TM = \frac{\sum T_j}{N}$$

- Medio flowtime

$$FM = \frac{\sum F_j}{N}$$

- Job in ritardo

$$NT = \sum \delta(T_j) \quad \text{dove}$$

$$\delta(T_j) = 1 \text{ se } T_j > 0$$

$$\delta(T_j) = 0 \text{ se } T_j = 0$$

- Makespan  $MAK = \max_j \{C_j\} - \min_j \{I_j\}$

# Obiettivi

- Parametri di prestazione (3/3)

- Coefficiente di saturazione della macchina  $i$

$$TS_i = \frac{\sum t_{ji}}{MAK} \quad \text{dove} \quad t_{ji} = \text{tempo di lavorazione}$$

- Coefficiente di saturazione medio del sistema

$$TSM = \frac{\sum \sum t_{ji}}{M \text{ MAK}} \quad \text{dove} \quad M = \text{numero di macchine}$$

- Work-in-process

$$WIP = \frac{1}{b-a} \int_a^b WIP(t) dt \quad \text{dove} \quad \begin{aligned} a &= \min_j \{I_j\} \\ b &= \max_j \{C_j\} \end{aligned}$$

- Tempo di setup complessivo

$$SUC = \sum_1^M SU_i \quad \text{dove} \quad SU_i = \text{setup complessivo sulla macchina } i$$



# Obiettivi

- Obiettivi della programmazione operativa
  - Minimizzazione del medio lateness
  - Minimizzazione del medio tardiness
  - Minimizzazione del medio flowtime
  - Minimizzazione del numero job in ritardo
  - Minimizzazione del makespan
  - Massimizzazione del coefficiente di saturazione medio del sistema
  - Minimizzazione del WIP
  - Minimizzazione del tempo di setup totale

# Modello di Karg-Thompson (1/3)

- Ipotesi
  - N job indipendenti tra loro, disponibili al tempo 0
  - Date di consegna non rilevanti
  - Non è ammessa la preemption dei job
  - Tempi di setup dipendenti dalla sequenza
- Funzione obiettivo
  - Minimizzazione del tempo complessivo di setup

# Modello di Karg-Thompson (2/3)

- $\min \text{SUC} \leftrightarrow \min \text{MAK}$  (per le ipotesi)
- Metodo Euristico
- Monomacchina
- Passi
  - Selezionare casualmente due job
  - Selezionare un nuovo job e provare a disporlo in una posizione della sequenza corrente calcolando il SU
  - Allocare il job al passo con in minimo SU e tornare al passo 2

# Modello di Karg-Thompson (3/3)

- Il risultato dipende da
  - La scelta della coppia iniziale
  - Dall'ordine con cui gli altri job vengono considerati per l'inserimento nella sequenza
- Per migliorare il risultato si può ripetere l'algoritmo più volte, iniziando sempre con coppie diverse

# Modello di Hodgson (1/3)

- Ipotesi
  - N job indipendenti tra loro, disponibili al tempo 0
  - Date di consegna note
  - Preemption non ammessa
  - Tempi di setup nulli o indipendenti dalla sequenza
- Funzione obiettivo
  - Minimizzazione del numero di job in ritardo

# Modello di Hodgson (2/3)

- Metodo di ottimizzazione algoritmico
- Monomacchina
- Passi
  - Creare l'insieme  $E^*$  con i job in ordine crescente di date di consegna e  $L^* = \{ \}$
  - Se in  $E^*$  non ci sono job in ritardo  $E = E^*$  e  $L = L^*$ , altrimenti identificare il primo job  $k$  in ritardo
  - Spostare il job con tempo di lavorazione più lungo tra i primi  $k$  in  $E^*$  in  $L^*$  e tornare al passo 2

# Modello di Hodgson (3/3)

$Job_j$	$t_j$	$d_j$
1	1	2
2	5	7
3	3	8
4	9	13
5	7	11

# Modello di Johnson (1/3)

- Ipotesi
  - Flowshop con  $M=2$  sempre disponibili
  - $N$  job indipendenti tra loro, disponibili al tempo 0
  - Date di consegna non rilevanti
  - Preemption non ammessa
  - Tempi di setup nulli o indipendenti dalla sequenza
- Funzione obiettivo
  - Minimizzazione del makespan



# Modello di Johnson (2/3)

- Metodo di ottimizzazione algoritmico
- Flowshop
- Passi
  - Per tutti i job calcolare  $\min_j \{t_{j1}, t_{j2}\}$
  - Se il minimo è sulla macchina 1, mettere il job nella prima posizione disponibile della sequenza e andare allo step 3
  - Se il minimo è sulla macchina 2, mettere il job nell'ultima posizione disponibile della sequenza
  - Rimuovere il job assegnato e tornare al passo 1

# Modello di Johnson (3/3)

<b>Job<sub>j</sub></b>	<b>t<sub>j1</sub></b>	<b>t<sub>j2</sub></b>
1	3	6
2	5	2
3	1	2
4	6	6
5	7	5