

Minimizzazione del makespan mediante un software di scheduling in un Job Shop ed in un Flexible Flow Shop

A partire da una classificazione dell'architettura delle linee produttive, viene proposto un caso applicativo per mostrare le potenzialità degli algoritmi di simulazione delle sequenze di operazioni su un determinato parco macchine.

■ Alessio Puppato, Barbara Fuoco, Andrea Rossi, Michele Lanzetta
Dipartimento di Ingegneria Meccanica, Nucleare e della Produzione Università di Pisa

Le attuali tendenze del mercato e la varietà della domanda dei consumatori hanno spinto le aziende ad affrontare il problema della riduzione dei costi di produzione attraverso una migliore gestione delle risorse disponibili. La soluzione a questo problema è proprio lo scheduling che con i suoi algoritmi consente di raggiungere tali obiettivi.

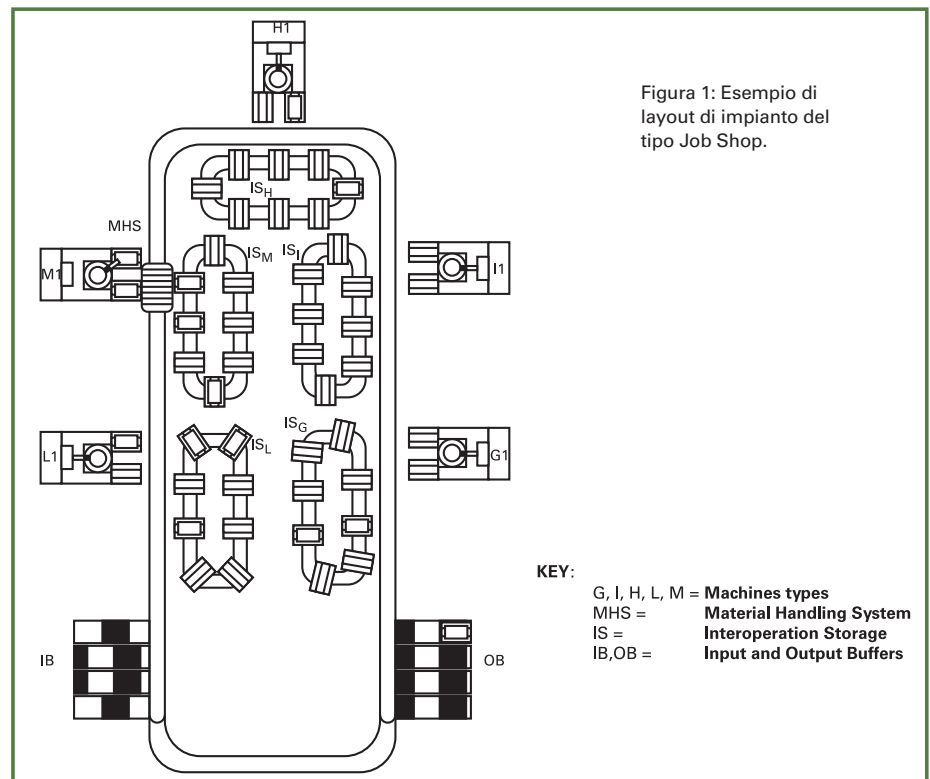
Lo scheduling

I principali modelli di schedulazione sono:

- Modello Macchina Singola
- Macchine Parallele
- Job Shop
- Flow Shop (modello di Johnson)
- Flexible Job Shop
- Flexible Flow Shop
- Open Shop

I problemi di scheduling sono problemi decisionali in cui riveste importanza fondamentale il fattore tempo, inteso come risorsa (scarsa) da allocare in modo ottimale.

Con il termine macchina si indica una qualsiasi risorsa come ad esempio: una macchina utensile, un centro di lavoro, un reparto, un operatore, etc. Le attività, che possono consistere in una sola o più operazioni elementari (ciclo di lavorazione) e sono indicate con il termine job. A ciascun job possono essere associate molteplici informazioni:



- Tempo di processamento (processing time) t_{ij} : rappresenta il tempo necessario al completamento del job j -esimo sulla i -esima macchina. Se il tempo di processamento è indipendente dalla macchina si indica semplicemente con τ_j .

- Tempo di rilascio (release date) r_j : indica l'istante di tempo (rispetto a un tempo iniziale) prima del quale non è possibile iniziare l'esecuzione del job

j -esimo. Può rappresentare l'istante in cui diventano disponibili le materie prime o i semilavorati per la lavorazione di un certo prodotto.

- Tempo di consegna (due date) d_j : indica l'istante di tempo (rispetto a un tempo iniziale) entro il quale l'esecuzione del j -esimo job dovrebbe essere terminata. In genere, la violazione di un tempo di consegna comporta dei costi, delle penalità

che si applicano sul ricavo pattuito con l'ordinativo (o commessa d'ordine).

- Peso w_j : tale quantità rappresenta l'importanza relativa del job j -esimo rispetto agli altri; può rappresentare, ad esempio, il costo di mantenimento del job nel sistema (ad es. costo di immagazzinamento). E' dunque un indice di priorità del job j -esimo rispetto agli altri.

Consideriamo ora le caratteristiche relative all'ambiente produttivo.

I sistemi più semplici sono i seguenti:

- Macchina Singola: in questo sistema tutti i job richiedono la medesima risorsa produttiva per poter essere completati.
- Macchine Parallele: si possono avere tre tipologie di architettura.
- Macchine parallele identiche: in questo caso i prodotti possono essere indifferentemente lavorati su m macchine identiche.
- Macchine parallele generiche: in questo caso ci sono ancora m macchine come nel caso precedente ma i tempi di lavorazione τ_{ij} dipendono oltre che dal job j anche dalla macchina i su cui il lavoro viene eseguito. Ogni macchina è caratterizzata da una sua velocità v_i .
- Macchine parallele scorrelate: il sistema produttivo è una generalizzazione del caso precedente. La macchina i può eseguire il job j con una velocità v_{ij} . Se le velocità delle macchine fossero indipendenti dai job (cioè $v_{ij} = v_i$ per ogni j) si ricadrebbe nel caso precedente.
- Flow Shop: in questo caso il sistema consiste di m macchine in serie e ciascun job deve essere eseguito da ciascuna delle

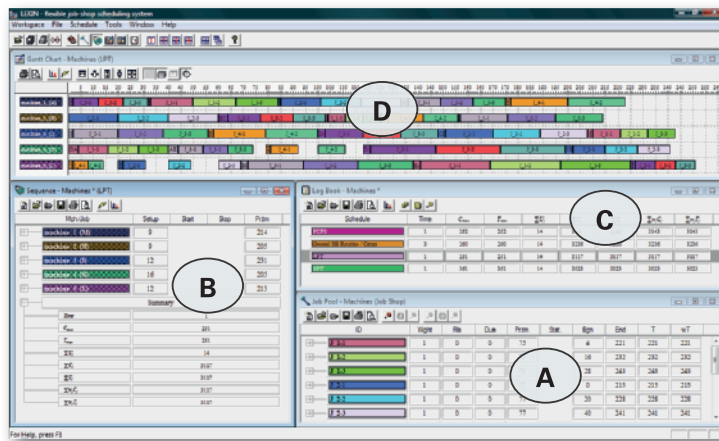


Figura 2: Screenshot del software Legin Scheduler [6] riguardante il caso esaminato, con (A) dati relativi al prodotto, (B) al sistema produttivo inseriti dall'utente, (C) i risultati ottenuti con i vari algoritmi e (D) il piano calcolato.

macchine successivamente, ossia l'ordine in cui i job visitano le macchine è lo stesso per tutti i job (routing fisso). L'ordine delle operazioni è fisso per ogni job ma i tempi di lavorazione su ogni macchina τ_{ij} possono essere differenti da job a job.

- Flexible Flow Shop: è una generalizzazione dei sistemi produttivi di Flow Shop e di Macchine Parallele. Invece di avere m macchine in serie, vi sono c stadi di produzione in serie, ognuno con un certo numero di identiche macchine in parallelo. Ogni job deve essere lavorato prima allo stadio 1, poi allo stadio 2 e così via, ed in ogni stadio il job può essere lavorato su una qualunque delle macchine di quello stadio.

- Job Shop: anche in questa architettura vi sono m macchine, ma ciascun job ha un proprio ordine con cui visitarle. L'ordine delle operazioni (routing) per ciascun job è fisso ed è diverso da job a job e i tempi di lavorazione τ_{ij} su ogni macchina possono essere differenti da job a job.

- Flexible Job Shop: è una

generalizzazione dei sistemi produttivi a Job Shop ed a Macchine Parallele. Invece di m macchine ci sono c workcenter (isole, aree produttive, celle) ognuno con un certo numero di identiche macchine in parallelo. Ogni job deve essere processato, seguendo un suo ordine ben preciso, nei vari workcenter e in ognuno di questi può essere lavorato su qualsiasi macchina.

- Open Shop: in questo caso ciascun job può essere processato sulle m macchine senza alcun ordine specifico (routing non fisso). La scelta del routing di ogni job può essere determinata sulla base di considerazioni logistico-gestionali non essendo imposta da ragioni tecnologiche.

Oltre alle caratteristiche dei job e del sistema produttivo, per definire esaurientemente un problema di scheduling occorrono ulteriori informazioni, tra le quali:

- Tempi di setup: indica il tempo necessario a riconfigurare la macchina nel passaggio da un job j -esimo ad un altro k -esimo e si indica con s_{jk} . Se il setup dipende dalla precedente operazione eseguita sulla macchina si parla di sequence dependent setup times, mentre nel caso in cui il setup non dipenda dalla precedente operazione nel routing del job si parla di sequence independent setup time.

- Preemption (priorità o importanza): caratteristica che, se presente, indica che è consentito interrompere la realizzazione di un job a favore di altri con priorità più alta. I job in sospenso saranno ripresi in un secondo momento, dopo che i job con priorità maggiore sono stati ultimati.

- Precedence constraints (vincoli di

	Jobs Type				
	J1	J2	J3	J4	J5
Job #	J1-1 J1-2 J1-3	J2-1 J2-2 J2-3	J3-1 J3-2 J3-3	J4-1 J4-2 J4-3	J5-1 J5-2 J5-3
	Machine – Setup - Time				
	4 – A - 12	2 – B - 20	1 – C - 10	5 – D - 6	3 – E - 18
	1 – A - 17	5 – B - 9	2 – C - 13	3 – D - 21	4 – E - 10
	2 – A - 7	1 – B - 16	3 – C - 15	4 – D - 11	5 – E - 22
	5 – A - 28	3 – B - 19	4 – C - 26	2 – D - 14	1 – E - 13
	3 – A - 11	4 – B - 13	5 – C - 8	1 – D - 23	2 – E - 19
	4 – A - 12	2 – B - 20	1 – C - 10	5 – D - 6	3 – E - 18

Tabella 1: Per ciascuno dei job sono riportati per colonne: i routing tecnologici, i set-up richiesti (A, B, C, D, E) ed i tempi di processamento.

precedenza): significa che esistono delle relazioni di precedenza tra i vari job; può accadere infatti che un job debba aspettare il completamento degli altri prima di essere eseguito.

- Machine breakdown: avaria di un apparato o di un dispositivo della macchina che non ne consente temporaneamente l'utilizzo. Il lasso di tempo durante il quale il macchinario non è disponibile si assume noto (ad esempio a seguito di una manutenzione programmata).

I parametri

Prima di presentare le funzioni obiettivo più usate nei problemi di scheduling, occorre introdurre i seguenti parametri da associare ai job.

La lettera i viene utilizzata come pedice per riferirsi alla macchina, mentre con la lettera j ci si riferisce al job.

- Tempo di completamento c_{ij} : indica il tempo (rispetto a un tempo iniziale) necessario affinché il j-esimo job sia completato sulla macchina i-esima. Se il pedice i viene omesso c_j si riferisce all'intervallo di tempo necessario perché il job j-esimo concluda la sua lavorazione sull'ultima macchina in cui doveva essere processato.

- Lateness L_j : rappresenta la differenza tra il tempo di completamento e il tempo di consegna del j-esimo job, ovvero $L_j = c_j - d_j$. Se tale differenza è positiva, la lateness

indica un ritardo, se negativa un anticipo rispetto al tempo di consegna.

- Tardiness T_j : il tardiness di un job j-esimo è definito come $T_j = \max(0; c_j - d_j) = \max(0; L_j)$. Essa è quindi una quantità sempre positiva o nulla, per cui a differenza del lateness, quest'ultimo non considera un vantaggio il fatto che un job sia in anticipo rispetto al relativo tempo di consegna.

- Earliness E_j : è definito come $E_j = \min(0; L_j)$. Esso è uguale al lateness se il job è in anticipo (ovvero $L_j < 0$) ed è nullo se in ritardo. Tale valore risulta di interesse in un ottica di just in time.

- Unit penalty U_j : è definita come $U_j = 1$ se $C_j > d_j$ mentre $U_j = 0$ se $C_j \leq d_j$ [1].

Obiettivi dello scheduling

Una soluzione di un problema di scheduling prende il nome di schedule o piano di produzione (production plan).

In termini generali, uno schedule è una descrizione completa dell'utilizzo temporale delle macchine da parte dei job che devono essere processati.

Dopo l'introduzione di queste grandezze, si possono esaminare alcuni esempi di funzioni obiettivo.

- Minimizzazione del makespan;
- Minimizzazione del numero di job in ritardo;
- Minimizzazione del lateness;
- Minimizzazione del tardiness;

Setup	A	B	C	D	E
A		1	2	3	4
B	2		4	2	3
C	1	3		4	2
D	4	2	3		1
E	3	4	1	2	

Tabella 2: Tempi di set-up necessari in funzione del set-up già presente sulla macchina (penalità).

- Minimizzazione del flow time o tempo di attraversamento;

- Minimizzazione del work in process WIP, cioè del numero di job in lavorazione;

- Massimizzazione della saturazione delle macchine, il quale coincide con la minimizzazione del makespan;

- Minimizzazione del tempo di setup complessivo.

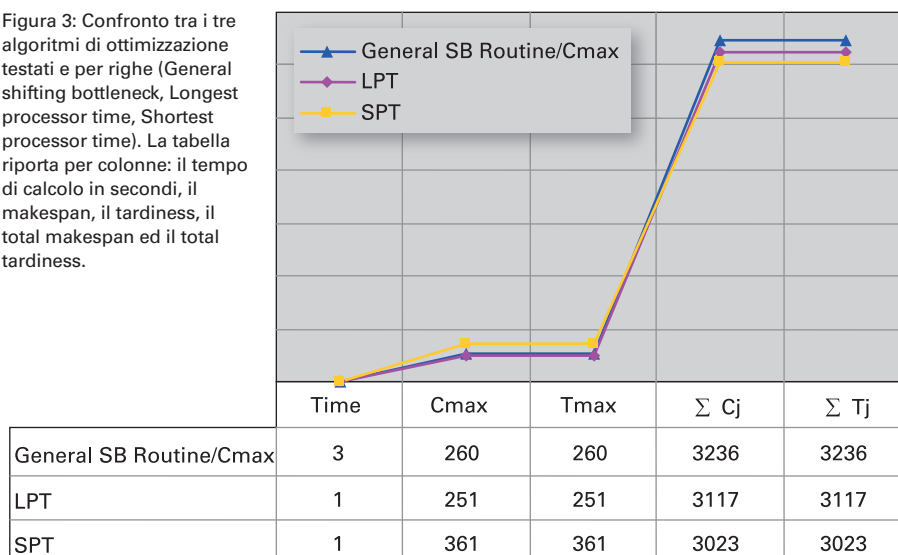
- Makespan o tempo di completamento C_{max} : è equivalente al tempo di completamento dell'ultimo job in lavorazione ovvero $C_{max} = \max(c_j)$. Minimizzare questo valore permette di ottenere un alto coefficiente di utilizzazione delle macchine.

- Maximum lateness L_{max} : è definita come $L_{max} = \max(L_j)$, individua la massima violazione della data di consegna, cioè lo scarto maggiore tra il tempo in cui si finisce di lavorare un job e quello in cui il job dovrebbe essere già consegnato. Potrebbe anche essere negativo nel caso in cui tutti i job terminino prima del tempo di consegna previsto.

- Total weighted completion time $\sum w_j C_j$: tale quantità, nota altresì con il nome flow time, è definita come la somma pesata dei tempi di completamento dei job (weighted flow time). Essa è un indicatore del work in process aziendale.

- Total weighted tardiness $\sum w_j T_j$: funzione di costo più generica rispetto al total weighted completion time, rappresenta la somma pesata dei ritardi nell'esecuzione dei job rispetto alle loro date di consegna.

Figura 3: Confronto tra i tre algoritmi di ottimizzazione testati e per righe (General shifting bottleneck, Longest processor time, Shortest processor time). La tabella riporta per colonne: il tempo di calcolo in secondi, il makespan, il tardiness, il total makespan ed il total tardiness.



Job Pool - Machines (Job Shop)										
ID	Wght	Rls	Due	Pr.trm	Stat.	Bgn	End	T	wT	
J_1-1	1	0	0	75	A	4	221	221	221	
										machine_1(G)
										machine_1(M)
										machine_2(H)
										machine_3(I)
J_1-2	1	0	0	75	A	16	232	232	232	
										machine_1(G)
										machine_1(M)
										machine_2(H)
										machine_3(I)
J_1-3	1	0	0	75	A	28	243	243	243	
										machine_1(G)
										machine_1(M)
										machine_2(H)
										machine_3(I)
J_2-1	1	0	0	77	A	0	215	215	215	
										machine_2(H)
										machine_1(M)
										machine_1(G)
										machine_3(I)
J_2-2	1	0	0	77	B	20	228	228	228	
										machine_2(H)
										machine_1(M)
										machine_1(G)
										machine_3(I)
J_2-3	1	0	0	77	B	40	241	241	241	
										machine_2(H)
										machine_1(M)
										machine_1(G)
										machine_3(I)
J_3-1	1	0	0	72	C	2	235	235	235	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_3-2	1	0	0	72	C	12	243	243	243	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_3-3	1	0	0	72	C	22	251	251	251	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_4-1	1	0	0	75	D	2	200	200	200	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)

Job Pool - Machines (Job Shop)										
ID	Wght	Rls	Due	Pr.trm	Stat.	Bgn	End	T	wT	
J_4-2	1	0	0	75	D	8	223	223	223	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_5-1	1	0	0	82	D	2	176	176	176	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_5-2	1	0	0	82	D	18	195	195	195	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)
J_5-3	1	0	0	82	D	38	214	214	214	
										machine_1(M)
										machine_2(H)
										machine_3(I)
										machine_1(G)

Figura 4: Il "job pool" del programma mostra il routing tecnologico di ogni job. Sono riportati in dettaglio tutte le caratteristiche dei job, quali, tempi di processamento, tempi di rilascio e consegna, pesi, istanti di inizio e fine processo, tardiness e weighted tardiness (A).

- Weighted number of tardy jobs $\sum w_j U_j$: rappresenta il numero di job in ritardo rispetto alla propria due date ed è spesso utilizzata nella pratica.

- Total setup time $\sum SU_i$: rappresenta il tempo di setup complessivo ed è definito come:

$$\sum_{i=1}^m SU_i$$

dove SU_i è il tempo di setup sulla macchina i -esima per lavorare l'insieme dei job assegnati.

In passato la ricerca si è concentrata soprattutto su modelli che cercavano di minimizzare funzioni costituite da un solo obiettivo, mentre attualmente si è cominciato ad analizzare e studiare anche modelli caratterizzati da funzioni multiobiettivo (multiple objective functions).

Algoritmi di simulazione

I principi generali esposti in precedenza (problema di scheduling e ottimizzazione) possono essere estesi ad un particolare caso aziendale.

Nel caso esaminato, l'azienda deve produrre una commessa di 5 prodotti diversi, adottando un sistema produttivo a Job Shop, dove ogni manufatto ha un determinato routing tecnologico fissato da esigenze meccaniche e di corretto funzionamento dei prodotti stessi (Tabelle 1 e 2).

Il programma

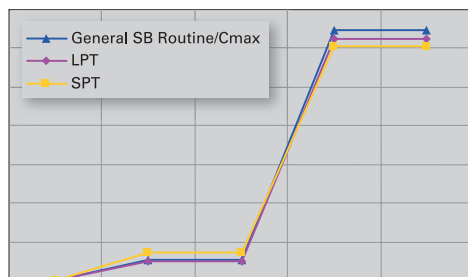
Ogni macchina può eseguire solo un job e ciascun job può essere lavorato solo da una macchina alla volta (capacity constraints). Per la soluzione del problema viene utilizzato il software per la pianificazione automatica dei cicli Lekin Sheduler. In Figura 2 è illustrata l'interfaccia grafica del programma.

Il programma permette di testare diversi algoritmi di ottimizzazione al fine di determinare il minimo della funzione obiettivo, nel caso specifico il makespan.

Si riportano i seguenti algoritmi più noti: - Shortest processing time (SPT): ordinamento secondo il quale il job che richiede il più basso tempo di processamento è il prossimo job ad essere processato. Si ordina per tempi di processamento crescenti.

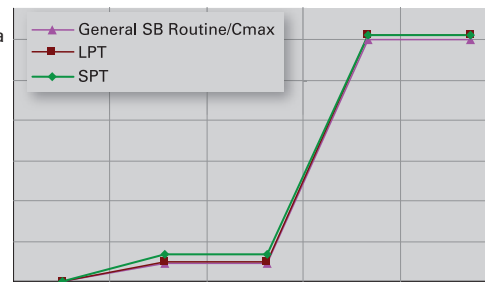
- Longest processing time (LPT): ordinamento secondo il quale il job che richiede il più alto tempo di processamento è il prossimo job ad essere processato. Si ordina per tempi di processamento decrescenti.

General Shifting Bottleneck (General SB): identifica la macchina che influenza in modo più consistente il valore del makespan rispetto alle altre e tale macchina viene detta "risorsa bottleneck" ("collo di bottiglia"). L'algoritmo ad ogni passo individua la macchina che costituisce il bottleneck corrente e fissa il sequenziamento su quest'ultima. Al passo successivo, ricerca la macchina che, tra quelle ancora non sequenziate, risulta maggiormente critica, sequenzia quest'ultima e così via. Siccome il ruolo di risorsa bottleneck viene rivestito da una macchina diversa ad ogni passo, l'algoritmo viene detto shifting bottleneck. Gli algoritmi citati sono stati testati sul caso in esame tramite il programma descritto e i risultati sono sintetizzati nella Figura 3. Dall'analisi dei tempi di processamento (zona B in Figura 2) si nota come le isole di produzione con più alti tempi operazione siano la "machine 3 (I)" con processor time pari a 231 unità di tempo e la "machine 1 (M)" con processor time pari a 214 unità di tempo. Si propone quindi una modifica del layout del sistema produttivo aggiungendo una nuova macchina identica a quella già



	Time	Cmax	Tmax	Σ Cj	Σ Tj
General SB Routine/Cmax	3	260	260	3236	3236
LPT	1	251	251	3117	3117
SPT	1	361	361	3023	3023

Figura 5: Confronto tra i medesimi algoritmi di Figura 3 e con la stessa commessa ma con la modifica del sistema produttivo a Flexible Flow Shop.



	Time	Cmax	Tmax	Σ Cj	Σ Tj
General SB Routine/Cmax	6	226	226	2986	2986
LPT	1	243	243	3033	3033
SPT	1	329	329	3042	3042

presente per ciascuna delle suddette isole. In questa nuova simulazione, ogni job potrà quindi passare indifferentemente su entrambe le macchine presenti nelle isole "machine 3 (I)" ed "machine 1 (M)". Con questa modifica il sistema passa da Job Shop a Flexible Job Shop. Questo consente di ridurre ulteriormente il makespan di processo (Figura 5) ed aumentare di conseguenza la saturazione del parco macchine ma evidentemente richiede un investimento che deve essere valutato.

In particolare, confrontando le Figure 4 e 5 si nota che il beneficio risultante è di ridurre da 251 unità di tempo ottenibile con un algoritmo LPT a 226 unità di tempo necessarie per il piano produttivo, ottenuto con l'algoritmo euristico General SB Routine/Cmax.

Conclusioni

Il semplice caso applicativo proposto consente di evidenziare i vantaggi derivanti dall'uso di un software di scheduling.

Infatti con essi è possibile ricavare oltre al makespan qui trattato, numerose altre informazioni quali il numero di job in ritardo, il work in process, la saturazione delle macchine e le altre descritte precedentemente.

Tutto ciò può essere fatto in poco tempo (Figura 3 e Figura 5) e simulando diversi layout produttivi, quindi questi software sono anche un ottimo strumento per valutare la necessità e la convenienza di dotarsi di nuove macchine ed apparecchiature. ■



Guardia del corpo

Certe volte siamo veramente fieri di noi stessi e con estrema chiarezza dichiariamo che "safety at work" per noi non è solo una parte del nostro lavoro. Il nostro impegno, la nostra motivazione ed esperienza sono al Vostro servizio con l'obiettivo di rendere ancora più sicure le vostre applicazioni.

Benvenuti dagli specialisti dei sensori di sicurezza industriali - Benvenuti da the sensor people

Il nuovo standard per le multiraggio di sicurezza; MLD 500 (tipo 4 e PL_e) e MLD 300 (tipo 2 PL_d) con muting integrato e nuovo transceiver a tre raggi.

Ulteriori informazioni e dettagli sul prodotto www.leuze.com



Leuze electronic S.r.l. - Via Soperga 54 - 20127 Milano
Tel. +39 02 26110643 - www.leuze.com

 **Leuze electronic**

the sensor people