

Politecnico di Torino

Corso di laurea triennale in ingegneria gestionale L-8

**Applicazione per la gestione e l'ottimizzazione
delle prestazioni di una linea di produzione
industriale tramite simulazione**



Politecnico di Torino

Relatore: Prof. Corno Fulvio

Studente: Manzetti Federico

Proposta di progetto

Studente proponente

S213240 Manzetti Federico

Titolo della proposta

Applicazione per la gestione e l'ottimizzazione delle prestazioni di una linea di produzione industriale tramite simulazione

Descrizione del problema proposto

L'applicazione sviluppata si prefigge di ricavare, tramite simulazione di funzionamento e permettendo l'inserimento dei singoli parametri delle workstation da parte dell'utente, i parametri di lavoro ottimali di una linea di produzione industriale, cercando di ottimizzarne il comportamento adattandosi ai dati di domanda quotidiani multiperiodali ricavati da un database esterno, il tutto con riferimento puntuale alle euristiche e alle formule di ottimizzazione derivate dalla Gestione della produzione e dai dati di benchmarking interno della linea.

Descrizione della rilevanza gestionale del problema

La rilevanza gestionale del problema risiede nella capacità dell'applicazione di ottimizzare i parametri di lavorazione in base ad una data domanda ricavata dall'esterno, tale da permettere una gestione della produzione misurata sulle esigenze e sulle capacità delle linee a disposizione, oltre che nella possibilità di generare in modo modulare un'intera linea con parametri scelti

dall'utente, permettendo quindi una maggiore personalizzazione della situazione analizzata ed un maggior numero di possibili casistiche affrontabili.

Descrizione dei data-set per la valutazione

L'applicativo utilizzerà un dataset fornito dal professor Chiabert riguardante l'andamento della domanda quotidiana di 3 prodotti distinti. I dati riguarderanno gli anni 2019-2021, e, nello specifico, consisteranno nella domanda definita giorno per giorno dei 3 prodotti finiti, ognuno dei quali di competenza di un anno differente nell'intervallo considerato.

Descrizione preliminare degli algoritmi coinvolti

- Creazione e gestione di varie linee di produzione, con specifiche relative alla natura della linea e degli imprevisti coinvolti incluse in appositi parametri (guasti, setup, rilavorazioni)
- Gestione delle query al database per ricavarne le vendite previste
- Simulazione multiperiodale, giorno per giorno, del funzionamento delle varie linee di produzione in base alle vendite previste del prodotto finito uscente dalla suddetta linea e proposta di miglioramenti sulla suddetta nei casi di problemi, imprevisti o di inefficienze. Si effettuerà un'operazione di ricorsione tra tutte le possibili combinazioni di parametri delle workstation per poter ottenere la linea più efficiente

Possibile, dati i valori dei vincoli iniziali sui parametri considerati.

Descrizione preliminare delle funzionalità previste per l'applicazione software

L'applicazione sarà fisicamente accessibile tramite un'interfaccia grafica che permetterà di dare in input i parametri di ogni workstation a disposizione per poter creare la linea di produzione ipotetica personalizzata su cui effettuare la simulazione. Tramite un apposito tasto di avvio, si eseguirà la simulazione di produzione sulla linea selezionata per un dato prodotto e, una volta fatto, si otterranno in output l'elenco delle stazioni della linea in questione, i parametri di funzionamento fondamentali della linea (Tempo Ciclo, Throughput e Work in Process) ed una diagnosi con focus

sull'utilizzazione (parametro che fa da ago della bilancia tra corretto funzionamento e blocco/sottoutilizzo della stazione).

La simulazione consisterà nella determinazione quotidiana dell'input di materia prima da immettere a monte della linea di produzione, la quale verrà volta per volta processata da ogni workstation in modo da poter garantire un throughput tale da soddisfare la domanda del giorno in questione (si suppone di dover produrre per poter vendere o stoccare entro la sera del giorno considerato). Per ogni workstation si valuterà il tempo effettivo di lavorazione, parametro fondamentale per inquadrarne l'efficienza, dati i parametri di normale funzionamento, di varianza, o di funzionamenti particolari precedentemente preimpostati tramite apposita classe (utile ad inquadrare la variabilità di questi valori in una casistica reale).

Nel caso non ci fossero ostacoli alla normale e prevista produzione, ogni giorno la quantità in uscita dalla linea sarebbe uguale alla domanda, mentre un'utilizzazione troppo alta o troppo bassa porterebbero ad un blocco del flusso di lavoro, rispettivamente per saturazione del collo di bottiglia e per sottoutilizzo.

Durante la progettazione della sopracitata linea, si può fare anche un'analisi di benchmarking interno, valutando la vicinanza del comportamento della stessa rispetto ad un funzionamento supposto ideale (bassissima variabilità nei tempi effettivi di lavorazione)

L'obiettivo finale diventa quindi la determinazione dei parametri ottimali di ogni workstation tramite simulazione per poter garantire il soddisfacimento della domanda nel modo più efficiente possibile.

Tutte le informazioni riguardanti i parametri verranno inserite in input dall'utente, in quanto si ipotizza di eseguire il set-up ex novo della linea, senza riutilizzare workstation già in azienda e per permettere una maggiore elasticità di applicazione del programma e rendere possibile la costruzione modulare di una linea ipotetica.

La produzione si suppone essere, per amor di semplicità, lineare e "continua", ovvero supponendo la continuità del servizio di forniture a monte della produzione e tenendo fuori dal computo delle possibilità eventuali scarsità di materiali, prefabbricati o consumabili, da imputare in questo caso a livelli di gestione più alti e non di competenza dell'applicazione o dell'utente.

Descrizione del problema affrontato

Ogni volta che una qualunque azienda industriale decide di mettere in produzione un nuovo prodotto, dopo tutti i vari accertamenti precedenti quali la convenienza effettiva nell' immettersi nel business e l'aver identificato da una parte i fornitori e dall' altra il mercato a cui puntare (ovvero i clienti) dovrà valutare l'effettiva possibilità di produzione con le risorse a disposizione.

Un qualunque prodotto destinato ad un vasto mercato viene creato su di una linea di produzione modulare, ovvero un insieme di stazioni di lavoro poste in sequenza le quali, prendendo in input la materia prima necessaria alla produzione, restituiranno in output il prodotto finito.

I maggiori problemi decisionali di questo tipo di approccio alla produzione riguardano la corretta calibrazione dei parametri di efficienza delle varie workstation (le linee di lavoro), da una parte per soddisfare la domanda entro un determinato arco temporale preso in considerazione, e dall'altra per fare tutto questo con la massima efficienza possibile, ovvero il minor spreco di risorse possibile nel caso di specie.

In fase di progettazione ad ogni workstation vengono assegnati due valori fondamentali:

- Il tempo di processo medio al pezzo t_0 , detto tempo medio naturale (rappresenta il tempo medio di produzione di un pezzo sulla workstation ipotizzando che la workstation lavori sempre in condizioni ottimali, ovvero senza guasti, rilavorazioni o necessità di eseguire dei setup di tanto in tanto).
- Una varianza del tempo di processo v_0 , detta varianza naturale (rappresenta la naturale varianza del processo di produzione, sempre presente in qualunque processo umano).

Dato che si sta parlando di processi legati al lavoro umano, bisogna valutare il grado di incertezza dei tempi utilizzando delle variabili stocastiche.

Gli studiosi di gestione della produzione preferiscono tuttavia usare il coefficiente di variabilità (v_0 / t_0) alla varianza per questioni di maggiore efficacia rappresentativa della variabilità nel momento in cui si confrontano processi con unità di misura diverse.

I parametri naturali non tengono però conto delle eventuali problematiche che può avere una workstation in fase di lavoro. Di fatto esistono tre macro-tipologie che raggruppano tutti i problemi che possono sorgere durante il funzionamento delle macchine/lavoro dell'operaio:

- Guasti. I guasti rappresentano tutte le eventualità che possono avvenire in qualunque momento della produzione e che possono bloccarla momentaneamente. I guasti aggiungono 4 parametri allo studio delle stazioni di lavoro:

- Mf** (mean time to failure): l'intervallo di tempo medio tra un guasto e l'altro.
- Cf**: coefficiente di variabilità di Mf.
- Mr** (mean time to repair): tempo medio necessario a far tornare la workstation guasta al completo funzionamento.
- Cr**: coefficiente di variabilità di Mr.

Le formule che rappresentano come i guasti influenzano il tempo medio di produzione e la rispettiva varianza sono le seguenti:

$$A = \frac{m_f}{m_f + m_r}$$

$$t_e = \frac{t_0}{A}$$

$$\sigma_e^2 = \left(\frac{\sigma_0}{A} \right)^2 + \frac{(m_r^2 + \sigma_r^2)(1-A)t_0}{Am_r}$$

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = c_0^2 + A(1-A)\frac{m_r}{t_0} + c_r^2 A(1-A)\frac{m_r}{t_0}$$

La "e" di fianco al tempo di processo, alla varianza e al coefficiente di variabilità sta per "effective" e sta ad indicare come una volta introdotti i parametri di guasto, il tempo di processo effettivo da considerare da lì in poi sarà T_e e così poi per gli altri parametri, che di fatto diventeranno così "corretti" per rispettare i fattori di variabilità in gioco.

- **Setup.** I setup rappresentano eventualità che avvengono solamente in intervalli di tempo più rigidi, cioè appunto quando bisogna eseguire il setup della workstation (per esempio cambiare un utensile di lavoro). Vengono aggiunti 3 nuovi parametri di studio:
 - Ns**: numero medio di pezzi processati tra un setup e l'altro.
 - Ts**: tempo medio di setup.
 - Cs**: coefficiente di variabilità del tempo di setup.

$$t_e = t_0 + \frac{t_s}{N_s}$$

$$\sigma_e^2 = \sigma_0^2 + \frac{\sigma_s^2}{N_s} + \frac{(N_s - 1)}{N_s^2} t_s^2$$

$$c_e^2 = \frac{\sigma_e^2}{t_e^2}$$

Rilavorazioni: si introducono nel caso sia presente un sistema in grado di effettuare il controllo qualità dei prodotti. Nel caso dovesse essere rilevato un difetto in output, sarà necessaria una rilavorazione in modo tale da soddisfare gli standard qualitativi richiesti. Si introduce un nuovo parametro:

-P (la probabilità che il semilavorato o il prodotto finito non rispettino gli standard qualitativi minimi e debbano essere rilavorati).

$$t_e = \frac{t_0}{1-p}$$

$$\sigma_e^2 = \frac{\sigma_0^2}{1-p} + \frac{p t_0^2}{(1-p)^2}$$

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = c_0^2 + p(1-c_0^2)$$



Una volta determinati i valori di tempo medio e varianza effettivi si può iniziare a valutare quantitativamente prima le performances delle varie workstations e poi infine della linea nel suo complesso.

Dal punto di vista delle workstations, le performances vengono analizzate considerando un solo parametro fondamentale, ovvero l'utilizzazione u .

L'utilizzazione rappresenta in poche parole quanto una data stazione di lavoro sta lavorando. Per avere un sistema senza blocchi u deve sempre essere minore di 1, in quanto se fosse maggiore vorrebbe dire che ho un input medio di materia prima troppo grande rispetto al tempo che impiego a generare output. L'utilizzazione si ricava dalla seguente formula:

$$u = \frac{r_a t_e}{m}$$

r_a = numero di pezzi in input per unità di tempo

M = numero di macchine parallele che lavorano simultaneamente sulla data workstation (più macchine ho più suddivido il lavoro, quindi sovraccarico di meno la stazione di lavoro)

Quando ci si presta a progettare una linea in vista di un determinato piano di produzione, bisogna sempre stare attenti a garantire un funzionamento senza blocchi per ogni workstation e per ogni giorno di funzionamento, è inoltre buona regola cercare di garantire un'utilizzazione il più alta possibile (ma minore di 1) per sfruttare al meglio la capacità produttiva della stazione di lavoro, in quanto un valore molto basso di utilizzazione vorrebbe dire che ho una workstation "troppo veloce" per le reali necessità e che quindi ho sprecato risorse inutilmente in fase di progettazione.

Il software si cura quindi di garantire primariamente questo equilibrio riguardo l'utilizzazione per ogni workstation.

Diretta conseguenza della performance delle singole stazioni di lavoro è la performance della linea nel suo complesso. La performance della linea è caratterizzata quantitativamente da tre indici:

- **TH** (throughput): indica in media quanti pezzi per unità di tempo escono a valle della linea di produzione. Se tutte le workstations hanno $u < 1$ questo valore coinciderà con il tasso di arrivo della linea (e cioè con l'input R_a della workstation a monte della linea).
- **CT** (cycle time): indica il tempo medio impiegato dalla linea per generare un output di 1 pezzo.
- **WIP** (work in process): indica il numero medio di semilavorati presenti all'interno della linea durante il suo funzionamento.

Questi tre indici sono legati dalla legge di Little:

$$WIP = TH \cdot CT$$

Per calcolare gli indici globali di linea è necessario prima calcolare il cycle time per ogni workstation (come se fosse una micro-linea di un solo modulo) e tramite la legge di Little trovare poi il work in process di ogni workstation. Una volta fatto questo, i valori totali di WIP e CT saranno semplicemente la somma dei valori delle singole workstations e il TH, se non ci sono blocchi, sarà uguale al tasso di arrivo della prima workstation della filiera.

Il modo per calcolare il CT per ogni stazione di lavoro utilizzato in questo progetto è il modello $G/G/m/\infty$, che è anche il modo più generale e corretto di valutare le prestazioni della linea, secondo il quale:

$$CT = \left(\frac{c_a^2 + c_e^2}{2} \right) \left(\frac{u^{\sqrt{2(m+1)}-1}}{m(1-u)} \right) t_e + t_e$$

dove C_a sta ad indicare il coefficiente di variabilità del tempo di arrivo della materia prima (o dei semilavorati nella workstation successive alla prima).

Questo valore viene posto generalmente uguale ad 1 (variabilità media) nella prima workstation a monte della linea, in quanto lì il tasso di arrivo è determinato da fattori esterni alla linea che non possono essere presi in considerazione in queste formule, mentre cambia per ogni altra workstation in quanto influenzato dalle variabilità delle workstation precedenti. La formula utilizzata in questo caso è la seguente:

$$c_d^2 = 1 + (1 - u^2)(c_a^2 - 1) + \frac{u^2}{\sqrt{m}}(c_e^2 - 1)$$

dove C_d sta per "departure coefficient", che sarebbe C_a visto però dal punto di vista della workstation precedente.

Buona prassi è, dopo aver calcolato le prestazioni di linea, eseguire un benchmarking interno della filiera ovvero un'analisi di quanto la linea in questione abbia lavorato al pieno delle sue potenzialità. Per far ciò si esegue solitamente la cosiddetta analisi di **Practical Worst Case (PWC)**: un'analisi che permette di dare una valutazione quantitativa di quanto la linea si sia scostata dalla sua performance peggiore (nella pratica) possibile. Non avrebbe senso, in questa sede, dilungarmi sui principi teorici alla base di quest'analisi, basti sapere che una linea "passa" positivamente l'analisi di PWC quando:

- $TH > \frac{w}{W_0 + w - 1} r_b$
- $CT < T_0 + \frac{w - 1}{r_b}$

dove:

- T_0 è la somma dei tempi medi di processo di ogni workstation (sarebbe il CT ideale, ovvero l'eventuale tempo di processo di linea nel caso non ci fossero variabilità e guasti/setup/rilavorazioni)
- r_b è il tasso di produzione della workstation che fa da collo di bottiglia, e quindi il TH massimo dell'intera linea (sarà il tasso di produzione della workstation con l'utilizzazione più alta, e quindi in questo caso più lenta)

- **W₀** è, dalla legge di Little, uguale a $T_0 \cdot r_b$ e rappresenta il massimo valore di WIP in linea che non genera congestione e che allo stesso tempo sfrutta al meglio le potenzialità della linea.
- **w** = WIP

Il software si occupa quindi di:

1. Effettuare una simulazione utilizzando come parametri delle workstations gli estremi peggiori inseriti dall'utente e restituire in output le prestazioni medie di utilizzazione e di linea, oppure rilevare un eventuale blocco in uno dei giorni di simulazione.
2. Ottimizzare il processo sopra descritto, valutando quale sia la combinazione di parametri ottimale per garantire valori di utilizzazione medi il più alti possibile ma senza generare blocchi. L'ottimizzazione si ripercuote poi di conseguenza anche sui parametri di linea in quanto direttamente collegati all'utilizzazione delle singole stazioni di lavoro.

Ovviamente i risultati così ottenuti dovranno anche essere valutati in funzione della spesa necessaria a progettare, acquistare e mettere in funzione macchine e componenti necessarie a garantire la performance attesa.

Descrizione del data-set utilizzato

Il data-set utilizzato è un derivato di un file Excel fornitomi dal Prof. Chiabert riguardante la domanda giornaliera di una certa quantità di tre prodotti A, B, C. Inizialmente il file Excel era in uno stato "grezzo", ovvero con molteplici errori di battitura e informazioni superflue che ho dovuto interpretare e quindi modificare. Ciò mi ha portato alla creazione di una tabella venditeprodotti, alla quale ho affiancato una tabella workstation per memorizzarci le varie stazioni di lavoro create da parte dell'utente. Qui di seguito la struttura delle due tabelle:

venditeprodotti	
Prodotto	VARCHAR (3)
Quantità	INT (11)
Data	DATE

workstation	
Nome	VARCHAR (50)
t0	DOUBLE
m	TINYINT (3)
c0	DOUBLE
isGuasti	TINYINT (1)
MfMax	DOUBLE
MfMin	DOUBLE
MrMax	DOUBLE
MrMin	DOUBLE
CfMax	DOUBLE
CfMin	DOUBLE
CrMax	DOUBLE
CrMin	DOUBLE
isSetup	TINYINT (1)
NsMax	DOUBLE
NsMin	DOUBLE
TsMax	DOUBLE
TsMin	DOUBLE
CsMax	DOUBLE
CsMin	DOUBLE
isRilavorazioni	TINYINT (1)
PMax	DOUBLE
PMin	DOUBLE

Descrizione ad alto livello delle strutture dati e degli algoritmi utilizzati

Il software è suddiviso in tre packages:

- **CONTROLLER**, che contiene le classi Main e Controller per l'interazione con l'utente.
- **DB**, all'interno del quale si trovano le classi necessarie per interfacciarsi con il database, ovvero DBConnect e la classe Dao con i metodi necessari per interrogare e/o modificare il database.
- **MODEL**, il cuore dell'applicativo, contenente le classi Java Bean, la classe Model che gestisce la parte algoritmica principale e la classe Simulazione tramite la quale si esegue l'algoritmo di simulazione della linea.

ALGORITMO DI SIMULAZIONE:

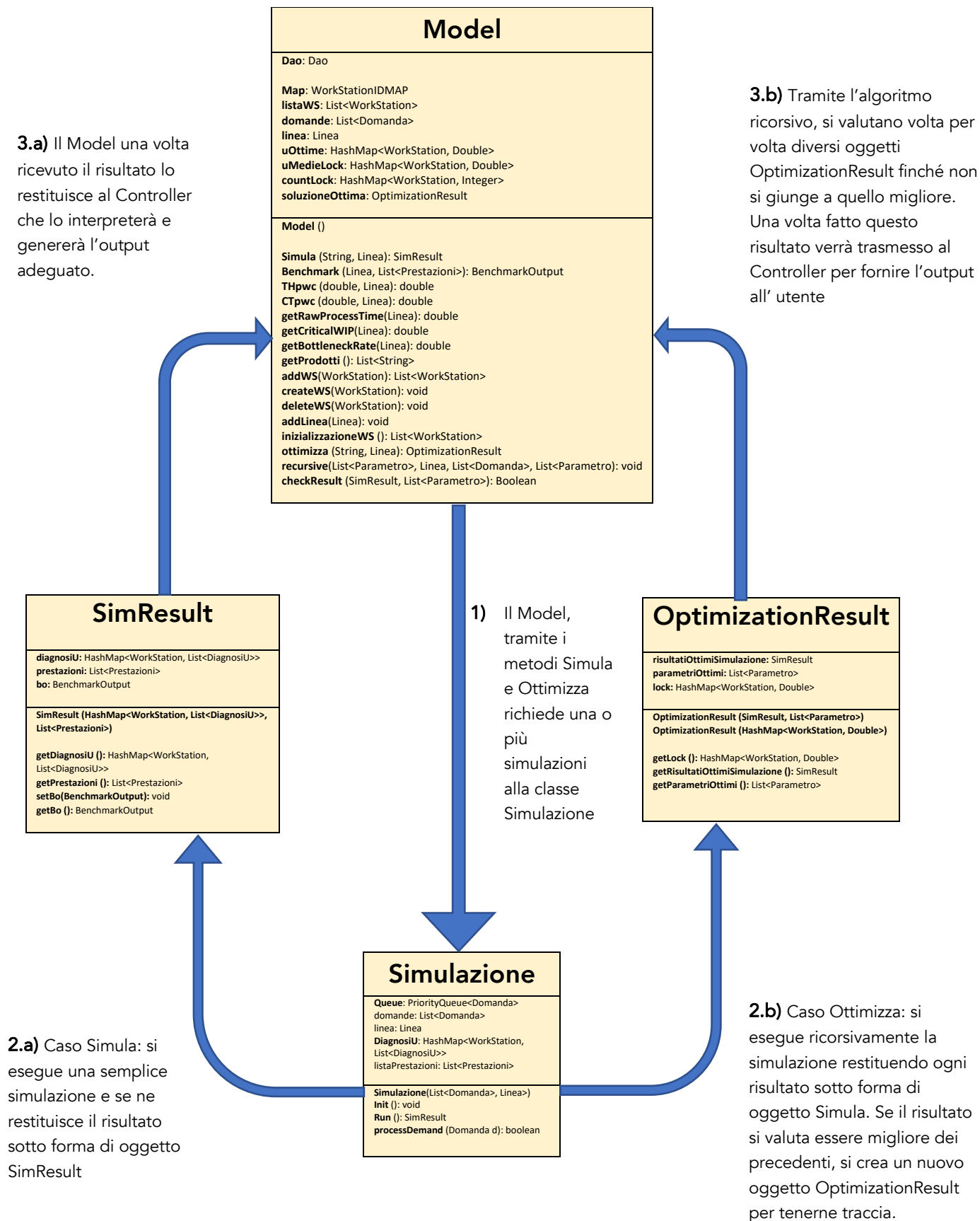
- a. Vengono caricati i valori relativi alle quantità delle domande quotidiane del prodotto selezionato.
- b. Una volta creata la linea si esegue la simulazione considerando (per ipotesi) gli estremi peggiori per ogni parametro, ovvero i valori massimi di ogni parametro, eccetto che per M_f e N_s dei quali verranno presi i valori minimi.
- c. Durante la simulazione vengono calcolate le utilizzazioni di ogni stazione di lavoro per ogni giorno di funzionamento. Si suppone, in questo progetto, che le macchine lavorino 365 giorni l'anno 24h/24 ed inoltre il tasso di arrivo viene sempre calcolato in modo da avere un flusso di lavoro costante durante il giorno.
- d. Si calcolano le prestazioni di linea (WIP, CT e TH).
- e. Si ricalcola il valore del coefficiente di variabilità in arrivo per (l'eventuale) prossima workstation della filiera.
- f. Si controlla se in quel giorno c'è stato un blocco su di una determinata workstation ($u > 1$). In caso affermativo la simulazione si interrompe e viene comunicato il blocco all'utente, altrimenti la simulazione continua fino all'esaurirsi dei giorni di simulazione.
- g. Viene comunicato in output all'utente se ci sono stati problemi durante la simulazione (blocchi o utilizzazioni troppo basse, cioè se minori di 0.4, come da prassi) più il numero di giorni di simulazione nei quali le prestazioni di linea sono state peggiori del Practical Worst Case.

ALGORITMO DI OTTIMIZZAZIONE:

- a. Tramite un algoritmo ricorsivo eseguo la simulazione della linea con tutte le varie combinazioni di parametri delle stazioni di lavoro (entro gli estremi indicati dall'utente in fase di creazione).
- b. Valuto la combinazione di parametri che mi possa garantire l'utilizzazione ottima per ogni workstation (cioè l'utilizzazione più grande possibile ma che non genere blocchi) analizzando i risultati di ogni simulazione.
- c. Una volta concluso il processo di ottimizzazione, se non si sono verificati problemi, restituisco in output i valori medi di utilizzazione per ogni stazione di lavoro, il numero di giorni che non hanno superato l'analisi di Practical Worst Case nella simulazione con i parametri ottimi più questi ultimi. Potrebbe accadere che con nessuna combinazione di parametri si riesca ad evitare il blocco della linea in almeno un giorno dell'anno di produzione, in tal caso verranno restituite in output le indicazioni per risolvere il problema.

Si noti che l'ottimizzazione si focalizza esclusivamente sui valori di utilizzazione, in quanto si può dimostrare che, a parità di variabilità, una buona utilizzazione delle varie workstations garantisce lo sfruttamento migliore possibile delle potenzialità della linea.

Diagramma delle classi principali dell'applicazione



Ovviamente ci sono numerose altre classi necessarie per il funzionamento dell'applicazione, nello schema sopra mostrato vengono evidenziate solamente le classi concettualmente "centrali" per il funzionamento del programma.

Si noti inoltre che buona parte del lavoro interpretativo dell'output, che trasforma il risultato "grezzo" in un risultato comprensibile dall'utente, viene fatto dal Controller, che tuttavia ho preferito non includere all'interno dello schema in quanto non fondamentale dal punto di vista algoritmico.

Schermate dell'applicazione

The screenshot shows the 'Simulazione linea di produzione industriale' application interface. The title bar reads 'Simulazione linea di produzione industriale'. The interface is divided into a top control area and a large empty workspace below.

Top Control Area:

- Buttons:** 'Crea WS', 'SET linea', 'Workstations' (dropdown), 'Aggiungi WS', 'Crea linea', 'Prodotto' (dropdown), 'Simulazione linea', 'Ottimizza'.
- Inputs:** 'Ws3' (text field), 'Parametri naturali: t0 17 c0 0.', 'Num macchine 2'.
- Checkboxes:** 'Guasti' (unchecked), 'Setup' (checked), 'Rilavorazioni' (unchecked).
- Parameter Groups:**
 - min MAX:** Mf, Mr, Cf, Cr (all empty).
 - Ns:** 15, 17 (input fields).
 - P:** (empty input field).
 - Ts:** 2, 2. (input fields).
 - Cs:** 0., 1 (input fields, with '1' highlighted).
- Buttons:** 'Modifica ...' (next to Cs inputs).

The main workspace below is a large, empty rectangular area.

Creazione WorkStation

Simulazione linea di produzione industriale

Crea WS: Parametri naturali: Num macchine:

☒ Guasti ☐ Setup ☐ Rilavorazioni

min MAX min MAX min MAX

Mf: Ns: p:

Mr: Ts:

Cf: Cs:

Cr:

SET linea:

Creazione Linea

Simulazione linea di produzione industriale

Crea WS: Parametri naturali: Num macchine:

☐ Guasti ☐ Setup ☐ Rilavorazioni

min MAX min MAX min MAX

Mf: Ns: p:

Mr: Ts:

Cf: Cs:

Cr:

SET linea:

Benchmarking interno:
la linea ha avuto performances peggiori di THpwc e CTpwc 154.0 giorni,
solo di THpwc 0.0 giorni,
e solo di CTpwc 0.0 giorni

Esempio di output di Simulazione in un caso senza problemi

Simulazione linea di produzione industriale

Parametri naturali:

 Num macchine

☐ Guasti
 ☐ Setup
 ☐ Rilavorazioni

	min	MAX		min	MAX		min	MAX
Mf	<input type="text"/>	<input type="text"/>	Ns	<input type="text"/>	<input type="text"/>	P	<input type="text"/>	<input type="text"/>
Mr	<input type="text"/>	<input type="text"/>	Ts	<input type="text"/>	<input type="text"/>			
Cf	<input type="text"/>	<input type="text"/>	Cs	<input type="text"/>	<input type="text"/>			
Cr	<input type="text"/>	<input type="text"/>						

ws2

P_A

La workstation ws1 ha un' utilizzazione media troppo bassa (0.004519596651445983)

Benchmarking interno:
 la linea ha avuto performances peggiori di THpwc e CTpwc 363.0 giorni,
 solo di THpwc 0.0 giorni,
 e solo di CTpwc 2.0 giorni

Esempio di output di Simulazione in un caso di utilizzazione troppo bassa

Simulazione linea di produzione industriale

Parametri naturali:

 Num macchine

☐ Guasti
 ☐ Setup
 ☐ Rilavorazioni

	min	MAX		min	MAX		min	MAX
Mf	<input type="text"/>	<input type="text"/>	Ns	<input type="text"/>	<input type="text"/>	P	<input type="text"/>	<input type="text"/>
Mr	<input type="text"/>	<input type="text"/>	Ts	<input type="text"/>	<input type="text"/>			
Cf	<input type="text"/>	<input type="text"/>	Cs	<input type="text"/>	<input type="text"/>			
Cr	<input type="text"/>	<input type="text"/>						

ws3

P_B

La workstation ws1 ha un' utilizzazione media di 0.41912852914389825

La workstation ws3 ha un' utilizzazione media di 0.47948303734061914

Benchmarking interno:
 la linea ha avuto performances peggiori di THpwc e CTpwc 154.0 giorni,
 solo di THpwc 0.0 giorni,
 e solo di CTpwc 0.0 giorni

ws1 P 0.2
 ws3 Ns 15.0
 ws3 Ts 2.4
 ws3 Cs 1.0

Esempio di output di ottimizzazione in un caso senza problemi

Simulazione linea di produzione industriale

Crea WS Parametri naturali: Num macchine

☐ Guasti ☐ Setup ☐ Rilavorazioni

min MAX min MAX min MAX

Mf Ns p

Mr Ts

Cf Cs

Cr

Utilizzazione eccessiva su una o più ws della linea!!! Valori medi di utilizzazione per ws:
ws1: 1.9798900462966733
ws3: 0.39918537168750057
Si consiglia di abbassare i valori dei parametri con le utilizzazioni medie maggiori

Esempio di output di ottimizzazione in un caso di eccessiva utilizzazione

LINK AL VIDEO DIMOSTRATIVO:

https://youtu.be/Yo0r2SNnY_w

Tabelle con risultati sperimentali ottenuti

Ws1 con rilavorazioni:

- $t_0 = 18s$
- $c_0 = 0.7$
- $m=3$
- $p_{Min} = 0.1$
- $p_{Max}=0.2$

Ws2 con setup:

- $t_0 = 17s$
- $c_0 = 0.5$
- $m=2$
- $Ns_{Min} = 14$
- $Ns_{Max} = 14$
- $Ts_{Min}=1.2$
- $Ts_{Max}=1.4$
- $Cs_{Min}=0.8$
- $Cs_{Max}=0.9$

Ws3 con guasti:

- $t_0 = 10s$
- $c_0 = 1.2$
- $m = 1$
- $Mf_{Min} = 70$
- $Mf_{Max} = 72$
- $Mr_{Min} = 1$
- $Mr_{Max} = 1.3$
- $Cf_{Min} = 0.8$
- $Cf_{Max} = 1$
- $Cr_{Min} = 1$
- $Cr_{Max} = 1$
- $Cr_{Min} = 1$
- $Cr_{Max} = 1$

Ws4 come ws3 ma

$t_0 = 85.188 s$

Ws5 come ws3 ma

$t_0=8.5$

Ws6 con setup:

- $t_0 = 8.553s$
- $c_0 = 0.8$
- $m=1$
- $Ns_{Min} = 9$
- $Ns_{Max} = 10$
- $Ts_{Min}=1$
- $Ts_{Max}=2$
- $Cs_{Min}=0$

Ws7 con rilavorazioni:

- $t_0 = 15.6$
- $c_0 = 1.1$
- $m=2$
- $p_{Min} = 0.4$
- $p_{Max}=0$

Linea: ws1-ws2, prodotto C, simulazione normale:

- Utilizzazione media troppo bassa su ws1 (0.01807 circa)
- 361 giorni di performances peggiori sia di THpwc che di CTpwc
- 4 giorni di performances peggiori solo di CTpwc

Linea: ws1-ws5, prodotto B, simulazione normale:

- Nessun problema di utilizzazione
- 280 giorni di performances peggiori sia di THpwc che di CTpwc

Linea: ws1-ws2, prodotto B, ottimizzazione:

- U media di ws1: 0.41912852914389825
- U media di ws2: 0.47780652322404393
- 151 giorni di performances peggiori sia di THpwc che di CTpwc
- Parametri ottimi:
 - Ws1 P=0.2
 - Ws2 Ns=14
 - Ws2 Ts=1.4
 - Ws2 Cs=0.9

Linea: ws1-ws3, prodotto B, ottimizzazione:

- Problema di utilizzazione troppo grandi, indicazioni:
 - U di ws1: 0.3631847993826513
 - U di ws3: 0.521036891311944
- Si consiglia di abbassare i valori dei parametri con le utilizzazioni medie maggiori

Linea: ws1-ws3, prodotto A, ottimizzazione:

- U media ws1: 0.004519596651445983

- U media ws3: 0.006138042690439946
- 363 giorni di performances peggiori sia di THpwc che di CTpwc
- 2 giorni di performances peggiori di CTpwc
- Parametri ottimi:
 - Ws1 P=0.2
 - Ws3 Mf=70
 - Ws3 Mr=1.3
 - Ws3 Cf=1
 - Ws3 Cr=1

Linea: ws1-ws4, prodotto A, ottimizzazione:

- U media ws1: 0.004519596651445983
- U media ws4: 0.052065619701172225
- 363 giorni di performances peggiori sia di THpwc che di CTpwc
- 2 giorni di performances peggiori di CTpwc
- Parametri ottimi:
 - Ws1 P=0.2
 - Ws3 Mf=70.3
 - Ws3 Mr=1
 - Ws3 Cf=1
 - Ws3 Cr=1

Linea: ws6-ws7, prodotto B, ottimizzazione:

- U media ws6: 0.4847479690461075
- U media ws7: 0.4843263003440602
- 366 giorni di performances peggiori sia di THpwc che di CTpwc
- Parametri ottimi:
 - Ws6 Ns = 9.9
 - Ws6 Ts = 1.2
 - Ws6 Cs = 0.3
 - Ws7 P = 0.1

Nota 1: Data la pesantezza dell'algoritmo ricorsivo misto alla simulazione, per le prove ho sempre usato linee di massimo 2 workstations, altrimenti il tempo di calcolo sarebbe aumentato eccessivamente.

Nota 2: eccetto che negli ultimi due risultati, i valori dei parametri ottimi sono, per ovvi motivi, sempre posti agli estremi. Per ottenere dei valori intermedi dei parametri ottimi bisogna avvicinarsi ad un valore di utilizzazione molto vicino a 1 "giocando" sul valore di tempo naturale di processo (o anche sul numero di macchine).

Nota 3: i valori di u media della simulazione normale vanno interpretati in modo diverso da quelli relativi all'ottimizzazione. Nel primo caso la u media è semplicemente la media aritmetica delle utilizzazioni nell'anno del prodotto considerato; nel secondo caso è la media delle medie di tutte le simulazioni. Dare un valore relativo alle u medie nel caso dell'ottimizzazione serve solamente ad avere un'idea della "zona" in cui si colloca l'utilizzazione.

Valutazioni sui risultati ottenuti e conclusioni

L'applicativo implementa due funzionalità: la simulazione e l'ottimizzazione.

Nel primo caso, il software agisce sostanzialmente da verifica nel caso si abbia già un'idea della linea di produzione che si vuole utilizzare.

Nel secondo caso si vuole cercare di far fronte ad una situazione di maggiore incertezza dal punto di vista progettuale e dare quindi un'indicazione sulla combinazione migliore di parametri dal punto di vista esclusivo dell'efficienza della linea.

Nel complesso l'obiettivo si può considerare raggiunto, bisogna tuttavia tener conto che, in un caso aziendale reale, andrebbero considerati numerosi altri aspetti quantitativi (per esempio il consumo energetico delle macchine, il tasso di inquinamento, controlli secondo normative ecc ...) ed inoltre sarebbe probabilmente più efficace eseguire un'analisi il meno discretizzata possibile per poter affinare l'accuratezza con la quale si valutano i valori dei parametri ottimi.

Va anche considerato che nello studio dei flussi di linea si applicano molte leggi empiriche che si rifanno in molti casi alle formule dello studio dei fluidi (vedi legge di Little) e di conseguenza riesce poco naturale applicare uno studio discretizzato ad un fenomeno che si studia con formule a variabili continue. Ho voluto tuttavia tentare di creare un software a riguardo in quanto questo è stato un argomento dal mio punto di vista molto interessante e poco affrontato algoritmicamente.

I risultati ottenuti in fase di ottimizzazione mostrano come nella maggior parte dei casi i valori dei parametri ottimi rimangono agli estremi. In effetti per dare un'effettiva utilità a questo software bisognerebbe utilizzarlo solamente quando ci si trova intorno a valori di utilizzazione molto prossimi a 1, quando cioè una valutazione umana rischierebbe di essere poco efficace. In tutti gli altri casi è ovvio che se ho un'utilizzazione molto bassa mi conviene aumentare al massimo i

parametri in modo da sfruttare al meglio la linea e allo stesso tempo risparmiare nelle performances richieste da parte delle workstations (e quindi indirettamente risparmiando anche potenza elettrica per esempio) mentre si può notare già ad occhio quando valori troppo alti dei parametri naturali conducono ad utilizzazione ben oltre il valore di 1.

Quest'opera è stata rilasciata con licenza Creative Commons
Attribuzione - Non commerciale - Condividi allo stesso modo 2.5 Italia.
Per leggere una copia della licenza visita il sito web
<http://creativecommons.org/licenses/by-nc-sa/2.5/it/> o spedisci una
lettera a Creative Commons, PO Box 1866, Mountain View, CA 94042,
USA.

