



**Politecnico
di Torino**

POLITECNICO DI TORINO

Corso di laurea triennale in ingegneria gestionale L-8

A.A. 2022/2023

Sessione di laurea Luglio 2023

Software per la schedulazione e la simulazione produttiva

Relatore:

Corno Fulvio

Candidato:

Proglio Davide s272199

SOMMARIO

1. PROPOSTA DI PROGETTO.....	4
1.1 STUDENTE PROPONENTE	
1.2 TITOLO DELLA PROPOSTA	
1.3 DESCRIZIONE DEL PROBLEMA PROPOSTO	
1.4 DESCRIZIONE DELLA RILEVANZA GESTIONALE DEL PROGETTO	
1.5 DESCRIZIONE DEI DATA-SET PER LA VALUTAZIONE	
1.6 DESCRIZIONE PRELIMINARE DEGLI ALGORITMI COINVOLTI	
1.7 DESCRIZIONE PRELIMINARE DELLE FUNZIONALITA' PREVISTE PER L'APPLICAZIONE SOFTWARE	
2. DESCRIZIONE DEL PROBLEMA AFFRONTATO.....	6
3. DESCRIZIONE DEL DATA-SET.....	7
4. DESCRIZIONE DI ALTO LIVELLO DELLE STRUTTURE DATI E DEGLI ALGORITMI UTILIZZATI.....	8
4.1 LA STRUTTURA DATI	
4.2 L'ALGORITMO RICORSIVO	
4.3 L'ALGORITMO DI SIMULAZIONE	
5. DIAGRAMMA DELLE CLASSI.....	11
6. L'APPLICAZIONE E I RISULTATI SPERIMENTALI.....	12
6.1 L'INTERFACCIA	
6.2 SCHEDULAZIONE PRODUTTIVA	
6.3 SIMULAZIONE PRODUTTIVA	
7. VALUTAZIONI E CONCLUSIONI.....	17

1. PROPOSTA DI PROGETTO

1.1 STUDENTE PROPONENTE

S272199 Proglia Davide

1.2 TITOLO DELLA PROPOSTA

Software per la schedulazione e la simulazione produttiva

1.3 DESCRIZIONE DEL PROBLEMA PROPOSTO

Il software si propone di realizzare un piano di produzione per stabilire le quantità mensili di un determinato prodotto finito da produrre su un orizzonte temporale di un anno, in modo da soddisfare la domanda della clientela. A tale scopo viene impiegato un algoritmo ricorsivo nella fase di Lot Sizing in modo da ottenere la schedulazione di produzione ottimale nel rispetto dei parametri limite introdotti dall'utente.

La seconda parte del software procederà con la simulazione produttiva di una certa quantità di prodotto finito data in input, su una linea produttiva scelta dall'utente, in modo da poter osservare tempistiche e performance.

1.4 DESCRIZIONE DELLA RILEVANZA GESTIONALE DEL PROBLEMA

In ambito produttivo è fondamentale stabilire le quantità da produrre periodicamente per ogni singolo prodotto in modo da riuscire a soddisfare la domanda dei clienti in tempo utile, rispettando la capacità produttiva a disposizione e minimizzando i costi (produzione e stoccaggio).

Le linee di produzione che si occupano della realizzazione di questi ordini produttivi devono avere performance adeguate, è quindi un problema centrale capire come quest'ultime operano in modo da comprendere se sono possibili miglioramenti che giovino alle performance.

1.5 DESCRIZIONE DEI DATA-SET PER LA VALUTAZIONE

Il dataset su cui il software opera è stato reperito sul sito Kaggle.com (comunità di data scientist in cui è possibile trovare e pubblicare set di dati). I dati rappresentano l'ordinazione di prodotti da parte di diversi clienti dal 2018 al 2022. La tabella è costituita da 4 colonne e circa un milione di righe:

- Identificativo dell'ordine
- Identificativo del cliente
- Data dell'ordine
- Identificativo del prodotto

1.6 DESCRIZIONE PRELIMINARE DEGLI ALGORITMI COINVOLTI

- Schedulazione piano produttivo:

- Selezione ordini di un determinato prodotto dato in input e calcolo dei fabbisogni mensili.
- Lot Sizing: la dimensione dei lotti di produzione mensile è calcolata attraverso un algoritmo ricorsivo che stabilisce quanto produrre per ogni slot temporale in modo da minimizzare i costi (dati dai costi di produzione e di stoccaggio inseriti dall'utente), rispettando il vincolo di capacità produttiva mensile per il prodotto dato in input dall'utente.

- Simulazione produttiva:

- Selezione di una linea produttiva tra quelle disponibili (differenti per numero di workstations, tempistiche e variabilità) e della quantità di prodotto di cui si desidera simulare il processo produttivo.
- Viene simulato il processo produttivo ricavando il tempo impiegato (in ore e in giorni) e le prestazioni della linea (Throughput, Cycle Time, WIP).

1.7 DESCRIZIONE PRELIMINARE DELLE FUNZIONALITA' PREVISTE PER L'APPLICAZIONE SOFTWARE

Il software sarà utilizzabile dall'utente attraverso un'interfaccia grafica divisa in due sezioni.

Nella prima sezione all'utente sarà possibile selezionare un prodotto d'interesse tra quelli presenti nel database per calcolarne il fabbisogno mensile su un orizzonte temporale di un anno, inoltre potrà inserire la capacità produttiva per il prodotto scelto (ordinaria e straordinaria), il costo unitario di produzione (ordinaria, straordinaria ed esterna), il costo di stoccaggio per singolo prodotto e il prezzo di vendita del prodotto. Questi valori saranno impiegati nella fase di Lot Sizing per determinare le quantità di prodotto ottimali da produrre mensilmente e da delegare a produttori esterni per soddisfare gli ordini e minimizzare i costi.

Come risultato della prima sezione di software all'utente saranno presentati il piano di produzione per il prodotto scelto, i costi di gestione (minimi) e il profitto atteso.

Nella seconda sezione all'utente sarà richiesto di inserire in input la quantità di prodotto di cui intende simulare la produzione.

Inoltre l'utente dovrà selezionare una linea di produzione tra quelle disponibili, ognuna con un determinato tempo di interarrivo e composta da un certo numero di workstations, caratterizzate da:

- Fermi macchina preemptive (non schedulati): tempo medio tra due guasti successivi (m_f), tempo medio di riparazione (m_r).
- Fermi macchina non-preemptive (schedulati): numero medio di jobs tra setups consecutivi (N_s), tempo medio di setup (t_s).

A partire da questi dati verrà simulato il processo di produzione del prodotto e come risultato saranno restituiti all'utente il tempo impiegato per produrre la quantità di prodotto fornita in input e le performance della linea produttiva: throughput (TH), tempo ciclo (CT) e work in progress (WIP).

2. DESCRIZIONE DEL PROBLEMA AFFRONTATO

Al giorno d'oggi la produzione industriale è diventata estremamente tecnica, ciò è imputabile a livelli concorrenziali elevati che fanno innalzare gli standard di mercato e rendono dettagli apparentemente piccoli estremamente strategici. Nell'attuale contesto manifatturiero per soddisfare la clientela sono necessari sistemi per il controllo della produzione e delle risorse, in modo da assicurarsi di non effettuare scelte antieconomiche e condurre un'attività di successo.

Il software proposto si suddivide in due parti.

La prima sezione consente di creare il piano produttivo ottimale relativo alla domanda di un determinato prodotto di interesse. Nel settore industriale è fondamentale stabilire quanto e quando produrre per soddisfare la richiesta della clientela, limitando il più possibile i costi legati al processo produttivo. Il costo di produzione per la realizzazione del prodotto finale non è l'unico parametro ad impattare sul prodotto finito, a questo si aggiungono i costi di immagazzinamento costituiti dai mancati interessi per effetto dell'immobilizzazione (il materiale a magazzino non produce profitto finché non viene venduto), ma anche da costi diretti come ad esempio l'eventuale affitto dei locali di stoccaggio e la loro manutenzione oppure l'obsolescenza del materiale. Il costo unitario di produzione può assumere valori differenti a seconda che il prodotto venga realizzato con capacità produttiva ordinaria o straordinaria. Talvolta può anche essere necessario ricorrere all'outsourcing, quindi delegare parte della produzione ad un'altra azienda in modo da riuscire a soddisfare la domanda, pur accontentandosi di profitti inferiori. Di conseguenza per la realizzazione di un piano produttivo è necessario tenere anche conto della capacità produttiva a disposizione, in modo che la programmazione produttiva sia sostenibile.

Tenendo conto di tutte le variabili citate in precedenza il software propone all'utente la schedulazione produttiva mese per mese, per l'anno e il prodotto di

interesse, che minimizza i costi e quindi garantisce il massimo profitto nel rispetto dei vincoli di capacità.

Nella realtà una volta determinato il piano produttivo, si passa al processo di produzione attraverso il quale vengono creati i prodotti finiti da vendere ai clienti.

La produzione avviene lavorando le materie prime su una successione di stazioni di lavoro dette workstation, al termine della quale si ottiene il prodotto finale. Generalmente le stazioni effettuano lavorazioni differenti con tempi di processo diversi.

Il processo di lavorazione non è deterministico, bensì intervengono alcune fonti di variabilità che è bene controllare per assicurare efficienza e buoni risultati. Ogni workstation è soggetta a una variabilità naturale che non è in alcun modo esplicitabile, a questa fonte se ne sovrappongono altre come guasti e setups che determinano il comportamento effettivo della macchina.

I guasti sono fermi macchina non programmati che si verificano casualmente e richiedono un tempo di ripristino a sua volta variabile, in entrambe i casi la variabilità è strettamente legata al tipo di macchinario impiegato nel processo.

I setups sono invece fermi macchina programmati, quindi a differenza della tipologia precedente non interrompono la lavorazione di un job, questi si verificano in maniera deterministica dopo un certo numero di jobs processati.

Anche il tempo che occorre per effettuare il setup è soggetto a una certa variabilità, generalmente di entità inferiore a quella che interviene per i guasti. La variabilità di una stazione si propaga alle successive creando code e tempi di attesa tra una workstation e l'altra, influenzando così l'intero processo produttivo. Da questo aspetto si può comprendere quanto sia importante controllare le proprie linee produttive.

La seconda sezione del software consente di testare una linea produttiva composta da più workstations, simulando su di essa la produzione di una determinata quantità di prodotto per poterne saggiare tempistiche e performance. L'applicazione restituisce all'utente il tempo impiegato per la produzione della quantità desiderata, inoltre calcola il tempo ciclo CT (tempo medio di attraversamento del sistema da parte di un job), il throughput TH (quanto esce dal sistema nell'unità di tempo) e il work in progress WIP (numero medio di jobs nel sistema). Attraverso questi parametri è possibile farsi un'idea di quanto il sistema impiega a produrre una data quantità di prodotto e come si comporta.

3. DESCRIZIONE DEL DATA-SET

Il database sul quale si basa il software è stato reperito sul sito Kaggle.com (comunità di data scientist in cui è possibile trovare e pubblicare set di dati), ed è disponibile nel progetto come file "ordiniProd.sql" nella cartella "database". Il data-set è costituito da un'unica tabella da circa un milione di righe, essa contiene gli ordini acquisiti da un'impresa dall'anno 2018 al 2022:

Ordini	
Nome	Tipo
Id_order	int
Id_customer	int
date	date
Id_prod	varchar

Il campo id_order contiene l'identificativo dell'ordine relativo a un certo acquirente contraddistinto dall' id_customer per il prodotto individuato dall' id_prod. Il campo data si riferisce a quando l'ordine viene acquisito (si assume che vada completato nel mese in cui viene effettuato).

4. DESCRIZIONE DI ALTO LIVELLO DELLE STRUTTURE DATI E DEGLI ALGORITMI UTILIZZATI

4.1 LA STRUTTURA DATI

Il software è stato realizzato separando la parte logica/applicativa da quella deputata all'interazione con la struttura dati e da quella riservata all'interfaccia grafica. Il progetto è stato sviluppato in linguaggio Java, seguendo i pattern MVC (Model View Controller) e DAO (Data Access Object). L'interfaccia grafica è

```

v [Icona] > tesi [tesi master]
  v [Icona] src/main/java
    v [Icona] it.polito.tdp.tesi
      > [Icona] EntryPoint.java
      > [Icona] FXMLController.java
      > [Icona] Main.java
    v [Icona] it.polito.tdp.tesi.db
      > [Icona] DBconnect.java
      > [Icona] ordiniDAO.java
      > [Icona] TestDAO.java
    v [Icona] it.polito.tdp.tesi.model
      > [Icona] Disposizione.java
      > [Icona] Domanda.java
      > [Icona] Event.java
      > [Icona] LotSizing3.java
      > [Icona] Model.java
      > [Icona] Produzione.java
      > [Icona] Simulazione3.java
      > [Icona] TestLotSizing.java
      > [Icona] TestSimulazione.java
      > [Icona] WorkStation.java
  
```

stata realizzata in JavaFX con l'ausilio di SceneBuilder.

Il programma è costituito da tre package principali:

- It.polito.tdp.tesi che contiene la parte attraverso la quale all'utente è possibile interagire con il tool.

L'interfaccia utente è realizzata nella classe "FXMLController", mentre la classe "Main" si occupa di lanciare il programma e "EntryPoint" configura interfaccia e logica applicativa.

- It.polito.tdp.tesi.db attraverso il quale il software accede alla struttura e svolge le sue interrogazioni. L'accesso ai dati viene effettuato attraverso la classe "DBconnect", mentre le query per selezionare i dati da elaborare e su cui si basa il programma, si trovano nella classe "ordiniDAO". La classe "TestDAO" non ha

rilevanza nell'economia del progetto e ha avuto funzione di supporto nella fase realizzativa.

- It.polito.tdp.tesi.model nel quale è realizzata la logica applicativa del software. Le classi di maggiore interesse sono "LotSizing3", che contiene l'algoritmo ricorsivo attraverso il quale viene realizzato il piano produttivo ottimale, e la classe "Simulazione3" in cui è implementato il simulatore per il testing delle linee produttive. La classe "Model" è quella che funge da "passacarte" al controller fornendogli i metodi applicativi per svolgere le funzioni del software. Le restanti classi definiscono oggetti e corrispondenti metodi che vengono impiegati nel processo ricorsivo ("Disposizione", "Domanda" e "Produzione") e di simulazione ("Event" e "WorkStation").

4.2 L'ALGORITMO RICORSIVO

Il metodo "schedula" che svolge la funzione ricorsiva, riceve per ogni mese una lista ("parziale") contenente lo stato capacitativo del sistema per il mese d'interesse e per quelli precedenti ad esso in cui potrebbe essere conveniente produrre per soddisfare la domanda del mese corrente. L'algoritmo ricorrendo distribuisce la domanda sui vari slot produttivi, dopodiché tra le disposizioni che soddisfano la richiesta, salva quella a costo di gestione minimo e aggiorna il piano produttivo annuale.

```
private void schedula(ArrayList<Disposizione> parziale, Domanda d, int slotProduttivo) {
    if(d.getQta() <= 0) {
        double ct = calcolaCostoGestione(parziale, d.getMese());
        if(ct < this.ctMin) {
            for(Disposizione disp : parziale) {
                if(disp.getTipo().equals("ord")) {
                    this.schedOttima.get(disp.getMese()-1).setQtaProdOrd(disp.getQta());
                }
                if(disp.getTipo().equals("straord")) {
                    this.schedOttima.get(disp.getMese()-1).setQtaProdStraord(disp.getQta());
                }
                if(disp.getTipo().equals("est")) {
                    this.schedOttima.get(disp.getMese()-1).setQtaProdEst(disp.getQta());
                }
            }
            this.ctMin = ct;
        }
    }
    else {
        for(int i=slotProduttivo; i<parziale.size(); i++) {
            if(parziale.get(i).getQta() < getCapacita(parziale.get(i).getTipo())) {
                int qtaI = parziale.get(i).getQta();
                int qtaAgg = 0;
                if(qtaI+d.getQta() <= getCapacita(parziale.get(i).getTipo())) {
                    parziale.get(i).setQta(parziale.get(i).getQta()+d.getQta());
                    qtaAgg = d.getQta();
                    d.setQta(0);
                }
                else {
                    if(parziale.get(i).getQta() < getCapacita(parziale.get(i).getTipo())) {
                        qtaAgg = getCapacita(parziale.get(i).getTipo())-parziale.get(i).getQta();
                        parziale.get(i).setQta(getCapacita(parziale.get(i).getTipo()));
                        d.setQta(d.getQta()-qtaAgg);
                    }
                    else {
                        slotProduttivo = i+1;
                    }
                }
            }
            schedula(parziale, d, slotProduttivo);
            parziale.get(i).setQta(qtaI);
            d.setQta(d.getQta()+qtaAgg);
        }
        slotProduttivo++;
    }
}
```

4.3 L'ALGORITMO DI SIMULAZIONE

Il metodo "processEvent" si occupa di simulare il processo produttivo sulla linea selezionata dall'utente una volta che la coda degli eventi è stata caricata con i jobs da svolgere e i guasti sulla workstation di riferimento.

Nel caso l'evento che si verifichi sia un nuovo job si controlla se questo si può eseguire o va messo in coda in quanto la workstation è occupata. Inoltre man mano che i jobs sono processati vengono eseguiti setups con una certa frequenza in base alla stazione della linea.

```
private void processEvent(Event e, int nw) {
    WorkStation wk = this.linea.get(nw);
    switch (e.getType()) {
        case NUOVO_JOB:
            if(!wk.getIngressi().containsKey(e.getnProd())) {
                wk.getIngressi().put(e.getnProd(), e.getTempo());
            }
            if(e.getTempo()>=wk.getTime()) { //wk libera
                BetaDistribution bTe = new BetaDistribution(3,3);
                double te = (int)((wk.getT0()-wk.getStdv0())
                    +((wk.getT0()+wk.getStdv0())-(wk.getT0()-wk.getStdv0()))*bTe.sample());
                wk.setTime(e.getTempo()+te);
                wk.getUscite().put(e.getnProd(), wk.getTime());
                if(wk.getUscite().size()==wk.getNs()) { //necessario setup
                    BetaDistribution bSu = new BetaDistribution(3,3);
                    double tsu = (int)((wk.getTs()-wk.getStdvS())
                        +((wk.getTs()+wk.getStdvS())-(wk.getTs()-wk.getStdvS()))*bSu.sample());
                    wk.setTime(wk.getTime()+tsu);
                }
                if(this.linea.size()==nw) {
                    this.pezziCompletati++;
                }
            }
            else { //wk occupata, job da accodare
                this.queue.add(new Event(wk.getTime(), EventType.NUOVO_JOB, e.getnProd(), nw));
            }
        break;
    }
}
```

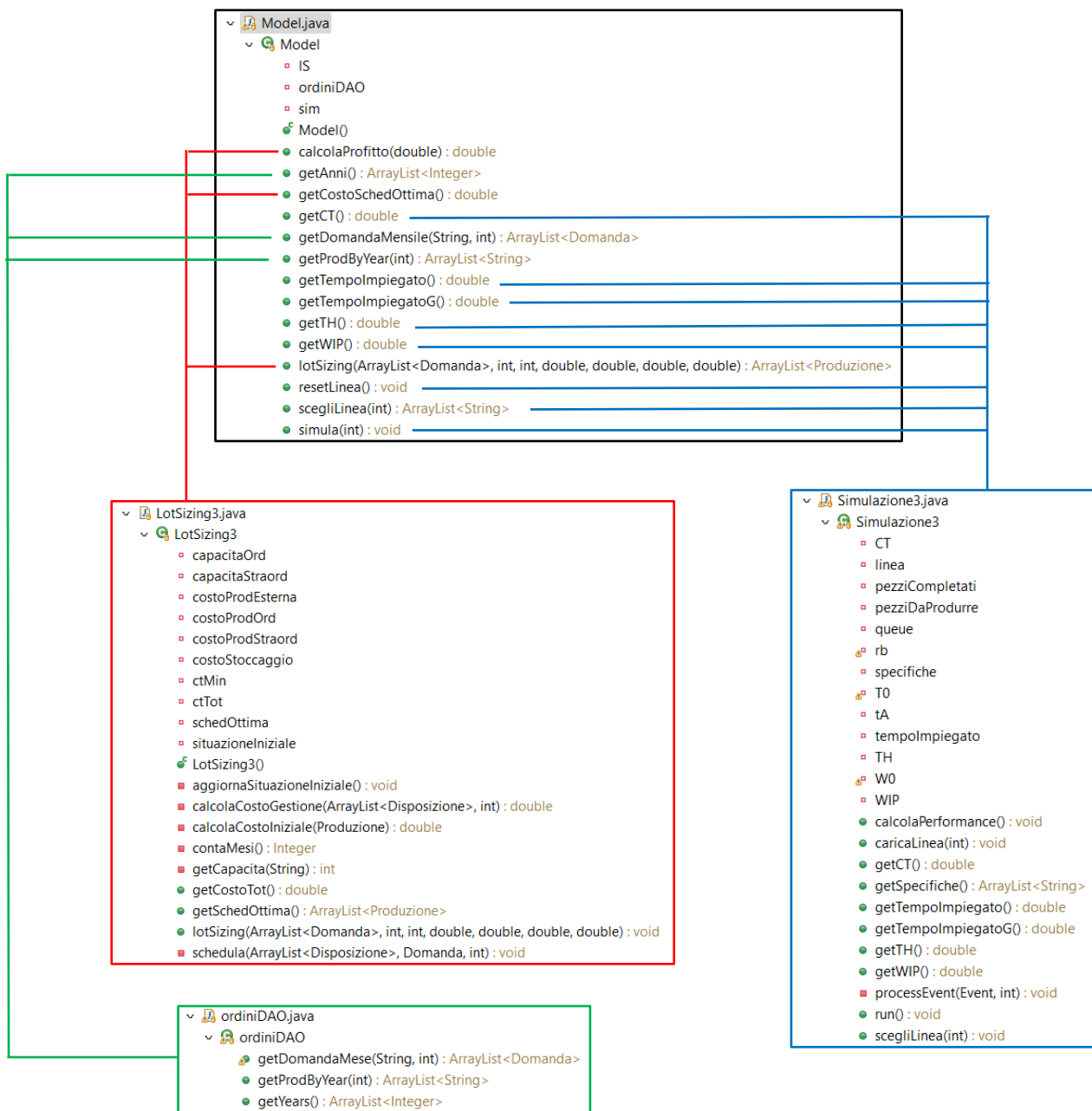
Se si verifica un guasto anche in questo caso si osserva se la stazione al momento del guasto è libera o no e quindi si propaga il ritardo dovuto al ripristino dell'operatività (si assume che il job venga interrotto nel momento del guasto e ripreso dallo stesso punto dopo il ripristino).

```
        case GUASTO:
            if(e.getTempo()>=wk.getTime()) { //il guasto si verifica quando la macchina è libera
                BetaDistribution bMr = new BetaDistribution(3,3);
                double tr = (int)((wk.getMr()-wk.getStdvR())
                    +((wk.getMr()+wk.getStdvR())-(wk.getMr()-wk.getStdvR()))*bMr.sample());
                wk.setTime(wk.getTime()+tr);
            }
            else { //il guasto si verifica quando la macchina è occupata
                boolean trovato = false;
                HashMap<Integer, Double> uscite = new HashMap<Integer, Double>(wk.getUscite());
                int ultimaUscita = uscite.size();
                while(trovato==false) {
                    if(uscite.get(ultimaUscita)<=e.getTempo()) { //job uscito prima del guasto?
                        trovato=true;
                    }
                    else {
                        ultimaUscita--;
                    }
                }
                BetaDistribution bMr = new BetaDistribution(3,3);
                double tr = (int)((wk.getMr()-wk.getStdvR())
                    +((wk.getMr()+wk.getStdvR())-(wk.getMr()-wk.getStdvR()))*bMr.sample());
                for(int i=ultimaUscita+1; i<=wk.getUscite().size(); i++) {
                    wk.getUscite().replace(i, wk.getUscite().get(i)+tr);
                }
                if(wk.getUscite().get(wk.getUscite().size())>wk.getTime()) {
                    wk.setTime(wk.getUscite().get(wk.getUscite().size()));
                }
            }
        break;
    }
}
```

I tempi di interarrivo e interguasto vengono estratti da una distribuzione di Poisson, mentre tempi di setup, di processo e di riparazione vengono estratti da una distribuzione Beta.

L'impiego di queste distribuzioni e dei relativi metodi è stato possibile importando nel progetto la libreria "org.apache.commons.math3.distribution".

5. DIAGRAMMA DELLE CLASSI



6. L'APPLICAZIONE E I RISULTATI SPERIMENTALI

6.1 L'INTERFACCIA

L'interfaccia grafica è stata sviluppata con JavaFX attraverso SceneBuilder, si suddivide in due sezioni contigue la prima intitolata "schedulazione produttiva" e la seconda "simulazione produttiva" che svolgono le due omonime funzioni. Di seguito viene riportata la videata che appare all'utente una volta avviato il programma.

Software per la schedulazione e la simulazione produttiva

SCHEDULAZIONE PRODUTTIVA

Selezionare anno e prodotto di interesse

Anno

Prodotto

Cerca

Mese	Domanda
Nessun contenuto nella tabella	

Inserire i valori richiesti per elaborare il piano produttivo ottimale relativo al prodotto selezionato

capacità ordinaria

costo prod ordinaria

capacità straordinaria

costo prod straordinaria

costo prod esterna

costo di stoccaggio

prezzo di vendita

Elabora piano

Mese	Qta prod ordinaria	Qta prod straordinaria	Qta prod esterna
Nessun contenuto nella tabella			

Costo gestione minimo:

Profitto:

SIMULAZIONE PRODUTTIVA

Selezionare la quantità di prodotto da produrre (max 10000 unità)

Quantità da produrre

Seleziona la linea da testare

Linea 1

Linea 2

Linea 3

Simula

Dati linea

TEMPO IMPIEGATO:

TEMPO CICLO (CT):

THROUGHPUT (TH):

WORK IN PROGRESS (WIP):

Un video dimostrativo sul funzionamento dell'applicazione è disponibile al link:
https://youtu.be/MzEbOgbXo_Q

6.2 SCHEDULAZIONE PRODUTTIVA

Nella prima sezione l'utente può selezionare un anno e un prodotto di interesse dai rispettivi combo box per visualizzarne la domanda mensile cliccando il bottone "Cerca". Dopo la scelta del prodotto di interesse è possibile creare il piano produttivo ottimale inserendo i parametri richiesti e confermando su "Elabora piano". A questo punto vengono presentati all'utente la migliore schedulazione possibile, il relativo costo e il profitto atteso.

Di seguito viene riportata a titolo di esempio la schermata visualizzata dall'utente, per due prodotti diversi.

Software per la schedulazione e la simulazione produttiva

SCHEDULAZIONE PRODUTTIVA

Selezionare anno e prodotto di interesse

2019
12162CR
Cerca

Mese	Domanda
1	175
2	175
3	373
4	183
5	176

Inserire i valori richiesti per elaborare il piano produttivo ottimale relativo al prodotto selezionato

Schedulazione ottimale

Mese	Qta prod ordinaria	Qta prod straordinaria	Qta prod esterna
1	150	25	0
2	150	50	0
3	150	50	148
4	150	33	0
5	150	50	0

Costo gestione minimo: 16490.0 € Profitto: 18765.92 €

Elabora piano

Software per la schedulazione e la simulazione produttiva

SCHEDULAZIONE PRODUTTIVA

Selezionare anno e prodotto di interesse

2019
16451S
Cerca

Mese	Domanda
1	264
2	244
3	284
4	540
5	225

Inserire i valori richiesti per elaborare il piano produttivo ottimale relativo al prodotto selezionato

Schedulazione ottimale

Mese	Qta prod ordinaria	Qta prod straordinaria	Qta prod esterna
1	282	0	0
2	300	0	0
3	300	75	0
4	300	75	0
5	300	75	0

Costo gestione minimo: 33234.0 € Profitto: 58979.1 €

Elabora piano

Consideriamo ad esempio il prodotto "12162CR" relativo all'anno 2019 (prima immagine d'esempio) e fissiamo la capacità produttiva ordinaria a 150 unità/mese e quella straordinaria a 50 unità/mese. Analizziamo come varia il piano produttivo al variare delle voci di costo:

Mese	Domanda
1	175
2	175
3	373
4	183
5	176
6	624
7	281
8	494
9	358
10	153
11	123
12	93

costo prod. ordinaria	3.75 €/u
costo prod. straordinaria	4.50 €/u
costo prod. esterna	7.50 €/u
costo stoccaggio	2.00 €/u

Mese	Qta prod ord	Qta prod straord	Qta prod est
1	150	25	0
2	150	50	0
3	150	50	148
4	150	33	0
5	150	50	0
6	150	50	400
7	150	50	81
8	150	50	294
9	150	50	158
10	150	3	0
11	123	0	0
12	93	0	0

costo prod. ordinaria	3.75 €/u
costo prod. straordinaria	5.50 €/u
costo prod. esterna	7.50 €/u
costo stoccaggio	2.00 €/u



Mese	Qta prod ord	Qta prod straord	Qta prod est
1	150	25	0
2	150	25	0
3	150	50	173
4	150	33	0
5	150	26	0
6	150	50	424
7	150	50	81
8	150	50	294
9	150	50	158
10	150	3	0
11	123	0	0
12	93	0	0

costo prod. ordinaria	3.75 €/u
costo prod. straordinaria	4.50 €/u
costo prod. esterna	8.50 €/u
costo stoccaggio	1.50 €/u



Mese	Qta prod ord	Qta prod straord	Qta prod est
1	150	50	0
2	150	50	0
3	150	50	123
4	150	50	0
5	150	50	0
6	150	50	383
7	150	50	81
8	150	50	294
9	150	50	158
10	150	3	0
11	123	0	0
12	93	0	0

Nel primo caso si può notare che non conviene produrre in anticipo, ciò che eccede la capacità straordinaria viene prodotto con forza lavoro esterna.

Nel terzo caso invece conviene produrre fino a due mesi d'anticipo, mentre nel secondo si produce al massimo un mese prima, altrimenti si ricorre a forza lavoro esterna.

Il tool oltre a calcolare la schedulazione produttiva ottimale è utile per capire come muta il profitto al variare dei singoli costi unitari e può essere impiegato anche per valutare l'impatto di un investimento per l'aumento della capacità produttiva sui risultati.

6.3 SIMULAZIONE PRODUTTIVA

Nella seconda parte l'utente stabilisce quante unità di prodotto produrre completando il campo dedicato e su quale linea simularne la produzione. Spuntando la linea di interesse vengono presentate all'utente le sue specifiche, riportate di seguito. La linea 1 è composta da due stazioni identiche, la linea numero 2 da una workstation più veloce e da una più lenta, mentre l'ultima linea è costituita da tre macchinari con velocità via via decrescente.

LINEA 1	
Tempo interarrivo	1 h
Workstation 1	
Tempo di processo	0.5 h
Tempo interguasto	8 h
Tempo riparazione guasto	0.83 h
Numero di job tra setups	20 jobs
Tempo di step	0.33 h
Workstation 2	
Tempo di processo	0.5 h
Tempo interguasto	8 h
Tempo riparazione guasto	0.83 h
Numero di job tra setups	20 jobs
Tempo di step	0.33 h

LINEA 2	
Tempo interarrivo	3.33 h
Workstation 1	
Tempo di processo	1.67 h
Tempo interguasto	27.22 h
Tempo riparazione guasto	3.33 h
Numero di job tra setups	15 jobs
Tempo di step	2.78 h
Workstation 2	
Tempo di processo	2.64 h
Tempo interguasto	35.78 h
Tempo riparazione guasto	4.17 h
Numero di job tra setups	20 jobs
Tempo di step	2.22 h

LINEA 3	
Tempo interarrivo	1.94 h
Workstation 1	
Tempo di processo	0.83 h
Tempo interguasto	41.67 h
Tempo riparazione guasto	1.94 h
Numero di job tra setups	25 jobs

Tempo di step	0.56 h
Workstation 2	
Tempo di processo	1.39 h
Tempo interguasto	36.11 h
Tempo riparazione guasto	1.11 h
Numero di job tra setups	10 jobs
Tempo di step	1.39 h
Workstation 3	
Tempo di processo	1.67 h
Tempo interguasto	27.78 h
Tempo riparazione guasto	1.39 h
Numero di job tra setups	17 jobs
Tempo di step	1.39 h

Premendo sul bottone “Simula” viene simulato il processo produttivo e vengono presentati all’utente il tempo impiegato e le performance della linea.
Di seguito viene riportata una schermata d’esempio.

SIMULAZIONE PRODUTTIVA

Selezionare la quantità di prodotto da produrre (max 10000 unità)

Seleziona la linea da testare

☒ Linea 1
☐ Linea 2
☐ Linea 3

Simula

Dati linea

tempo interarrivo medio della linea: 1.0 h
numero workstation: 2
WORKSTATION 1
tempo medio di processo: 30.0 min
tempo medio tra guasti successivi: 8.0 h
tempo medio di riparazione guasto: 0.83 h
numero di jobs tra setups consecutivi: 20 jobs
tempo medio di setup: 0.33 h

TEMPO IMPIEGATO: 999.25 h, equivalenti a 41.64 giorni

TEMPO CICLO (CT): 1.101 h THROUGHPUT (TH): 1.001 unità/h WORK IN PROGRESS (WIP): 1.102 unità

Altre combinazioni sono disponibili sotto forma di tabelle di seguito.

Quantità di prodotto: 1000 unità					
Linea	Tempo impiegato [h]	Tempo impiegato [giorni]	CT [h]	TH [u/h]	WIP [u]
1	999.25	41.64	1.101	1.0	1.102
2	3338.14	139.09	7.438	0.3	2.228
3	1949.57	81.23	5.185	0.513	2.66

Quantità di prodotto: 10000 unità					
Linea	Tempo impiegato [h]	Tempo impiegato [giorni]	CT [h]	TH [u/h]	WIP [u]
1	10003.29	416.18	1.175	1.0	1.175
2	33335.3	1388.97	7.247	0.3	2.174
3	19442.68	810.11	5.584	0.514	2.872

Quantità di prodotto: 6754 unità					
Linea	Tempo impiegato [h]	Tempo impiegato [giorni]	CT [h]	TH [u/h]	WIP [u]
1	6753.74	281.41	1.161	1.0	1.161
2	22521.79	938.41	7.218	0.6	2.165
3	13134.61	547.28	5.556	0.514	2.857

Le linee implementate sono senza perdite (ciò che entra nel sistema esce), di conseguenza il throughput, in questo caso calcolato sperimentalmente, deve coincidere con il tasso d'arrivo della linea.

La linea 1 ha un tempo ciclo medio di poco più di 1h, un valore vicino al tempo di processo che i job dovrebbero attendere per essere convertiti in prodotto finale (30min sulla prima stazione e altrettanti sulla seconda) se il comportamento della linea fosse ideale (senza fonti di variabilità di qualsiasi tipo). Sulla linea per la maggior parte del tempo è presente un solo job, probabilmente si potrebbe velocizzare il tasso d'arrivo in modo da aumentare il tasso di occupazione per avere maggiore utilizzazione delle risorse.

La seconda linea ha un tempo ciclo medio di più di 7h, ben più elevato rispetto al tempo ideale di 4.3h (100min sulla prima stazione e 158.33min sulla seconda). Questo aspetto indica che ogni job spende circa 2.7h in attesa, nonostante ciò il livello di wip resta controllato (leggermente superiore a due unità), grazie ad un tempo di interrivo della linea non troppo ridotto. Limitando i tempi morti che il job deve subire, le performance della linea possono sicuramente essere migliorate.

Infine la terza linea presenta un tempo ciclo di più di 5h, sapendo che il tempo ideale sarebbe di circa 3.88h si può comprendere che anche in questo caso ogni job è interessato da tempi d'attesa. Analogamente al caso precedente la frequenza con cui i jobs arrivano alla linea assicura un CT e un WIP sotto controllo, altrimenti si creerebbero lunghi accodamenti.

7. VALUTAZIONI E CONCLUSIONI

Per quanto riguarda la parte di schedulazione, il programma proposto permette di creare il piano produttivo ottimale nel rispetto della capacità produttiva, in un tempo generalmente breve. Il risultato consente di farsi un'idea di quanto saranno i costi da sostenere in fase produttiva e i profitti, nonché a quanta forza lavoro esterna e straordinaria si dovrà ricorrere. Inoltre l'utente ha la possibilità di osservare come mutano profitti e costi al variare della capacità produttiva

(valutazione investimento in capacità produttiva) oppure dei costi produttivi unitari.

Il piano produttivo viene creato a partire da una domanda deterministica (nota a priori), realisticamente nella maggior parte dei casi questa non è nota anche se può essere stimata attraverso metodi di forecasting.

La seconda sezione relativa alla simulazione produttiva permette di testare una linea produttiva in modo da stimare il tempo che impiegherà a produrre una certa quantità di prodotto e le sue performance.

L'analisi delle performance permette di capire come si comporta la linea e se sono possibili miglioramenti. Il software progettato permette di testare tre linee prestabilite, di conseguenza per poter osservare il comportamento di sistemi differenti sarebbe necessaria una configurazione ad hoc. D'altra parte il testing di una generica linea avrebbe richiesto all'utente l'inserimento di un gran numero di parametri, molti dei quali poco intuitivi e strettamente legati al particolare sistema d'interesse (distribuzioni di probabilità e misure di variabilità). Il progetto permette quindi di testare alcune linee d'esempio, ma può essere riconfigurato per analizzare qualsiasi altra linea.

Va inoltre sottolineato che i sistemi produttivi reali sono per lo più a rete e presentano funzionamenti più complessi (lavorazioni parallele, buffer, lavorazioni multiprodotto, scarti, rilavorazioni), di cui la linea rappresenta l'unità base. Il simulatore proposto è un buon punto di partenza su cui strutturare il proprio sistema di interesse specifico.

La pianificazione e il controllo sono sicuramente elementi indispensabili a livello industriale per ottenere risultati soddisfacenti, in conclusione il software implementato può sicuramente avere un'utilità, soprattutto se adattato al particolare caso d'interesse.

Questa relazione tecnica è rilasciata con licenza Creative Commons BY-NC-SA.

Tu sei libero di:

- Condividere – riprodurre, distribuire, comunicare al pubblico, esporre in pubblico, rappresentare, eseguire e recitare questo materiale con qualsiasi mezzo e formato.
- Modificare – remixare, trasformare il materiale e basarti su di esso per le tue opere.

Alle seguenti condizioni:

- Attribuzione – Devi riconoscere una menzione di paternità adeguata, fornire un link alla licenza e indicare se sono state effettuate delle modifiche. Puoi fare ciò in qualsiasi maniera ragionevole possibile, ma

non con modalità tali da suggerire che il licenziante avalli te o il tuo utilizzo del materiale.

- Non Commerciale – Non puoi utilizzare il materiale per scopi commerciali.
- Stessa Licenza – Se remixi, trasformi il materiale o ti basi su di esso, devi distribuire i tuoi contributi con la stessa licenza del materiale originario.

Per visualizzare una copia di questa licenza, visitare:

<http://creativecommons.org/licenses/by-nc-sa/2.5/it/>

