

POLITECNICO DI TORINO

Corso di Laurea

Magistrale

in

Ingegneria Gestionale

Tesi di Laurea Magistrale

Utilizzo di un sistema M.E.S. per la
pianificazione in tempo reale della
produzione



Relatore

Prof. Giulia Bruno

Candidato

Matteo Moroni

Anno accademico 2019/2020

Lavoro soggetto a licenza

Tutti i diritti riservati

ABSTRACT

La pianificazione della produzione è uno degli elementi fondamentali per il raggiungimento dell'efficienza aziendale, tradizionalmente una persona viene incaricata di redigere la schedulazione delle attività a priori. Questo metodo ha dei limiti in termini di tempo impiegato e di reattività nel caso di imprevisti. Lo sviluppo tecnologico ha permesso la nascita di nuovi strumenti, utili alla gestione del processo produttivo, che, integrati all'interno della linea produttiva, permettono di aumentare efficacia ed efficienza delle funzioni gestionali dell'azienda. Questo studio intende porsi nel mezzo dei due sistemi per verificare i limiti del primo e le potenzialità del secondo. Per perseguire questi fini è stato realizzato, su software Matlab, un algoritmo per la risoluzione di un problema di schedulazione in contesto flexible job shop, di classe NP-hard. La soluzione ottenuta viene quindi confrontata con il risultato di una schedulazione tradizionale e con quella prodotta dal sistema MES implementato in una azienda italiana. Il confronto è mirato a valutare l'algoritmo sviluppato e, soprattutto, la robustezza del risultato prodotto dal sistema MES il cui risultato è prodotto tenendo in considerazione un maggior numero di vincoli. Il risultato ottenuto ha permesso di verificare che il sistema MES in oggetto permette di ottenere tempistiche di lavorazioni simili a quelle ottenute dall'algoritmo studiato ad hoc.

ELENCO DEGLI ACRONIMI

- **MES:** Manufacturing Execution System
- **MESA:** Manufacturing Execution System Association
- **OKP:** One Of A Kind Production
- **FJSP:** Flexible Job Shop Problem
- **JSP:** Job Shop Problem
- **MESA:** Manufacturing Enterprise Solutions Associations
- **ISA:** International Society of Automation
- **ANSI:** American National Standard Intitute
- **PDP:** piani di produzione aziendale
- **FIFO:** first in first out
- **FASPS:** first arrival at the shop first to be served
- **GA:** Genetic Algorhms
- **ACO:** Ant Colony Optimization

LISTA DELLE FIGURE

Figura 1: attività svolte dai sistemi di gestione e flussi informativi.....	17
Figura 2: schema dei livelli di una gerarchia funzionale generica con individuazione della posizione in cui il MES deve intervenire.....	19
Figura 3: illustrazione della struttura della gerarchia di equipaggiamento con riferimento alla gerarchia funzionale.	20
Figura 4: Esempio di layout open shop (Yalaoui, et al. 2020).....	32
Figura 5: Esempio di layout job shop (Yalaoui, et al. 2020).....	32
Figura 6: Esempio di layout flow shop (Yalaoui, et al. 2020).....	33
Figura 7: metodi matematici per la soluzione di problemi job shop flessibile (Jain e Meeran 1999).....	35
Figura 8: esempio di ottimo locale e ottimo globale	36
Figura 9: rappresentazione grafica del comportamento degli individui nella ricerca dell'ottimo. a) rappresenta il percorso iniziale, in b) viene introdotto un vincolo, in c) gli individui esplorano le diverse soluzioni e in d) viene rappresentata la soluzione migliore. (Slowik 2020).....	47
Figura 10: ant colony optimization flow chart	48
Figura 11: ciclo del prodotto di Eurodies	50
Figura 12: logo Octavic PTS	51
Figura 13: Confronto sulle tempistiche di avvio delle prime operazioni	65
Figura 14: andamento delle durate complessive di lavorazione.....	65
Figura 15: confronto sulle deadline rispetto alla pianificazione di Eurodies.	66
Figura 16: confronto tra le durate complessive ottenute su Matlab con quelle elaborate dal sistema MES.....	66

Figura 17: confronto tra i ritardi subiti nell'avvio di produzione tra l' algoritmo sviluppato e il sistema MES..... 67

Figura 18: confronto tra i margini sulle deadline tra script Matlab e sistema MES..... 67

LISTA DELLE TABELLE

Tabella 1: Parte dei dati inseriti come input per la schedulazione.	56
Tabella 2: confronto tra i risultati dell' algoritmo sviluppato in contrapposizione con quelli prodotti dalla pianificazione di Eurodies e quelli ottenuti dal sistema MES di Octavic	64

LISTA DEGLI ALLEGATI

Allegato 1: Input per la schedulazione	71
Allegato 2: output di schedulazione	72
Allegato 3: confronto tra pianificazioni.....	73
Allegato 4: pianificazione Octavic	74
Allegato 5: pianificazione Eurodies	75
Allegato 6: pianificazione Matlab	76

SOMMARIO

SISTEMI DI PRODUZIONE ONE-OF-A-KIND.....	13
IL SISTEMA MES	15
2.1 MRP LIMITI E FUNZIONALITÀ	16
2.2 STANDARD ANSI/ISA-95	17
2.2.1 Modelli e Terminologia.....	18
2.2.2 Parte 3: Modelli di Attività di Gestione delle Operazioni Produttive.....	23
SCHEDULAZIONE DELLA PRODUZIONE	26
3.1 LA SCHEDULAZIONE.....	26
3.1.1 Attività.....	28
3.1.2 Risorse	28
3.1.3 Vincoli	29
3.1.4 Obiettivi.....	30
3.2 I DETERMINANTI DELLA SCHEDULAZIONE	31
3.2.1 Sistema di produzione	31
3.2.2 Direzione della schedulazione	33
3.3 MODELLI MATEMATICI.....	34
3.3.1 Exact Methods	35
3.3.2 Approximate Methods	35
CASO STUDIO.....	49
4.1 I SOGGETTI COINVOLTI	49

4.1.1 Eurodies	49
4.1.2 Octavic PTS.....	50
4.1.3 Università di Lund.....	52
4.2 PROCESSO PRODUTTIVO E CICLI DI LAVORAZIONE.....	53
4.2.1 Il ciclo produttivo	53
4.2.2 Reparto aziendale di interesse	54
4.3 SVILUPPO DELL'ALGORITMO	55
4.3.1 Assunzioni	55
4.3.2 Dati di Input.....	56
4.3.3 Prima operazione: sgrossatura	57
4.3.4 Seconda operazione: finitura	60
4.3.5 Terza operazione: taglio	61
4.3.6 Ripetizione del programma	62
4.3.7 Output.....	63
4.4 RISULTATI	63
4.5 OSSERVAZIONI.....	67
CONCLUSIONI.....	69
ALLEGATI	71
BIBLIOGRAFIA.....	77

INTRODUZIONE

Questa tesi è il frutto degli studi sull'applicazione di algoritmi di schedulazione in un contesto particolare come quello della produzione One-Of-A-Kind.

In continuità con gli studi del Dott. Mattia Orlando (Orlando, Bruno e Lombardi 2020), di cui si prosegue il lavoro, questa tesi partecipa ad un progetto del Politecnico di Torino, finanziato da Horizon 2020, un programma di ricerca e innovazione dell'Unione Europea. L'obiettivo finale del progetto è quello di integrare il software Production Efficiency Tracker sviluppato da Octavic PTS Company, all'interno del reparto produttivo di Eurodies, pmi italiana che si occupa sia della produzione e progettazione delle componenti necessarie alla realizzazione di componenti in lamiera per veicoli prototipali, sia della produzione dei componenti stessi.

Il tipo di produzione in questione rende difficoltosa la programmazione della produzione, questo studio quindi mira a valutare i risultati ottenuti dall'applicazione del sistema.

Nella tesi verranno confrontati i risultati prodotti da uno script realizzato con il software Matlab, e quindi, creato ad hoc per il caso studio, con quelli forniti dal software implementato in azienda.

L'obiettivo della tesi non è quello di migliorare un sistema già collaudato, bensì, di valutarne i punti di forza e le debolezze in modo da poterne valutare le potenzialità e adattarlo maggiormente alle esigenze.

La tesi è articolata in quattro capitoli:

- Nel primo andremo a caratterizzare il tipo di produzione One-Of-A-Kind (in seguito OKP), valutandone i punti di forza e le criticità che questo contesto produttivo può riservare al suo interno.

- Il secondo capitolo verterà sul Manufacturing Execution System (in seguito MES), analizzandone le caratteristiche, gli standard, le potenzialità applicative.
- Il terzo capitolo sarà incentrato sulla revisione letteraria relativa ai metodi di schedulazione generali ed applicati a sistemi di produzione OKP e FJSP
- Il quarto capitolo conterrà il lavoro relativo al caso studio, la descrizione dell'algoritmo sviluppato e il confronto con i risultati portati dal sistema MES implementato presso l'azienda Eurodies.

CAPITOLO 1

SISTEMI DI PRODUZIONE ONE-OF-A-KIND

La customizzazione è una delle strategie applicabili per aumentare la competitività aziendale. All'aumentare del grado di personalizzazione si riducono le quantità di prodotti uguali tra loro, con conseguente riduzione della ripetitività del processo produttivo.

L'estremizzazione della situazione sopra riportata si ha nel caso di produzione prototipale, detta anche OKP nella quale ogni prodotto differisce dagli altri e i volumi sono ridotti a poche se non addirittura una singola unità.

Possiamo distinguere diversi scenari per lo sviluppo del prodotto distinti in base all'informazione necessaria all'avvio della produzione:

- *Make-to-stock* in questo scenario l'informazione necessaria allo start della produzione è riferita a dati storici e previsioni di mercato, l'azienda produce determinati quantitativi di prodotto prevedendone la vendita. L'azienda si assume tutto il rischio di non vendita dei prodotti.
- *Assemble-to-order* prevede che il prodotto necessiti esclusivamente della fase di assemblaggio per essere venduto, in questo caso quindi si accorciano notevolmente i tempi di consegna e i rischi per l'azienda in quanto non subisce anticipatamente i costi di assemblaggio.
- *Make to order* prevede che il cliente fornisca il design del prodotto, lasciando solo la fase di produzione e assemblaggio. Le unità per essere prodotte richiedono sia componenti standard sia alcuni realizzati su misura.
- *Engineer-to-order* permette al cliente di suggerire un'idea nuova o derivante da qualcosa di esistente, con precise specifiche e funzionalità, vincolando tempi e

budget. La lista dei lavori è dunque dettata esclusivamente dal cliente e non è possibile all'azienda produrre in anticipo scorte per il magazzino.

Il sistema di produzione OKP si basa in particolare su quest'ultimo, in quanto l'unicità del prodotto deve essere garantita per intero ed il grado di customizzazione è tale che può essere definito in ogni dettaglio.

La produzione OKP è una delle più complesse esistenti (Jia, Jiang e Li 2015), guasti macchine, modifiche progettuali, tempistiche uniche per ogni prodotto, ciclo di produzione non standard, sono solo alcune delle problematiche che possono sorgere. Ne consegue che l'organizzazione aziendale possa beneficiare di un sistema che possa gestire in tempo reale le variazioni al pianificato.

CAPITOLO 2

IL SISTEMA MES

L'alto sviluppo tecnologico apportato nell'ultimo ventennio ha contribuito significativamente allo sviluppo di sistemi di raccolta e analisi dei dati dei processi di produzione industriale.

La raccolta e la gestione delle informazioni permettono di monitorare l'efficienza di un sistema produttivo, analizzarne le criticità e di conseguenza gli sprechi generati.

Per soddisfare questo bisogno di raccolta e analisi delle informazioni sono stati quindi sviluppati dei sistemi informatici specializzati per la gestione della produzione.

I software industriali specializzati per la gestione, il monitoraggio e il controllo dei processi produttivi delle aziende prendono il nome di Manufacturing Execution System.

I tratti fondamentali del MES sono definiti a livello internazionale dal Manufacturing Enterprise Solutions Associations (MESA) e possono essere riassunti nelle seguenti attività che rappresentano i “pilastri” di un sistema MES:

- Gestione dei processi
- Gestione della manodopera
- Gestione degli ordini
- Raccolta e acquisizione dei dati
- Controllo dei dati e confronto con parametri forniti dai macchinari
- Allocazione delle risorse
- Controllo della qualità

- Tracciamento del prodotto
- Analisi delle attività non a valore aggiunto
- Analisi delle performance produttive

I paragrafi successivi prenderanno come riferimento (Laganà 2019) in cui è descritto in modo più che esaustivo il sistema da cui è partito lo sviluppo dei sistemi MES e la sintesi delle normative a cui fa riferimento.

2.1 MRP LIMITI E FUNZIONALITÀ

Introduciamo le due componenti che principalmente componevano il sistema gestionale di un'azienda:

- MRP
- ERS

Con MRP si individuano i sistemi di Management Requirements Planning, questi permettevano di pianificare gli approvvigionamenti, di prendere in carico gli ordini e di associare gli strumenti da utilizzare.

L'Enterprise Resource Planning, è il sistema utilizzato per gestire le attività commerciali quotidiane, come ad esempio contabilità, procurement, project management, gestione del rischio e operazioni per la supply chain.

Il punto di contatto dei due sistemi è definito dalle informazioni relative all'avanzamento lavori che dovevano essere inserite manualmente all'interno del sistema.

In questa interconnessione tra le componenti risiede il limite principale del sistema, ovvero, quello di non essere in grado di creare piani di produzione efficienti e integrati al contesto produttivo in quanto la mancanza di trasmissione dell'informazione in tempo reale crea per la sua natura ritardi comunicativi.

Per risolvere questo handicap è fondamentale il ruolo di collegamento decisionale apportato dal MES.

Tra i principali standard che il sistema MES deve rispettare possiamo sottolineare:

- Drastica riduzione delle tempistiche e aumento della precisione del flusso di informazione. L'aumento della qualità informativa si rispecchia in un conseguente aumento dell'efficienza delle performances aziendale.
- Aumento della reattività e della flessibilità del sistema produttivo che si traduce con la produzione di PDP maggiormente attendibili, in quanto avvalorati dal controllo delle attività produttive passate, e adattati alle capacità aziendali e alle risorse disponibili.
- Elaborazione di KPI per la valutazione delle performance a breve medio termine, questi possono variare seguendo le necessità aziendali.

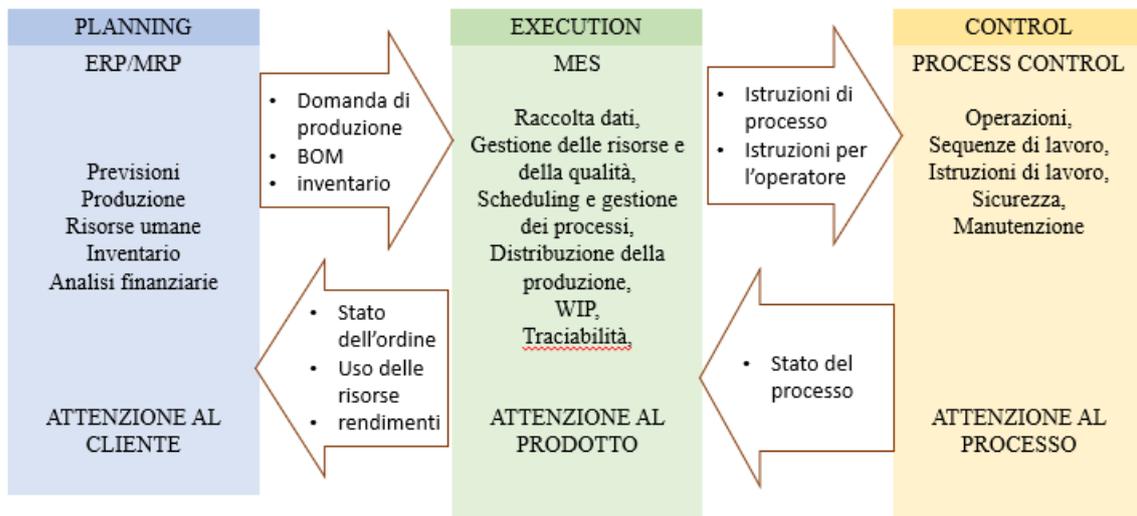


Figura 1: attività svolte dai sistemi di gestione e flussi informativi

2.2 STANDARD ANSI/ISA-95

Un sistema MES per essere definito tale deve rispettare le normative definite dallo Standard Internazionale ANSI/ISA-95, nato nel 1995 grazie alla collaborazione tra l'International

Society of Automation (ISA), società non a scopo di lucro per coloro che sono interessati ad attività inerenti l'automazione aziendale, e l'American National Standard Institute (ANSI), organizzazione privata non a scopo di lucro il cui obiettivo è quello di promuovere la conformità e l'adesione a standard comuni, tramite la definizione delle norme da rispettare, per accrescere competitività e qualità delle imprese americane.

In particolare, l'ANSI/ISA-95 è composto da sei parti:

2.2.1 Modelli e Terminologia

La prima sezione è dedicata alla definizione terminologica e identifica i modelli di oggetti utili alla gestione del flusso di informazione.

In particolare, sono trattati:

- I modelli gerarchici
- I modelli a flusso di dati
- Funzioni aziendali

2.2.1.1 I Modelli Gerarchici

Una struttura organizzativa di questo tipo permette di delineare correttamente e inequivocabilmente le funzioni aziendali, assegnandone precisamente responsabilità e attività.

Vengono identificati quindi i due modelli gerarchici:

Gerarchia Funzionale

Con questo modello si vuole dare importanza alle macro-funzioni aziendali che vengono quindi disposte mediante struttura gerarchica.

Viene posto al vertice dell'impresa uno staff di specialisti, capaci di gestire quei processi che sono comuni per più funzioni aziendali. Il gruppo è inoltre competente a delineare la strategia globale che deve essere seguita.

Ogni funzione aziendale è a sé stante e si occupa quindi della propria area di affari.

Sempre lo standard descrive i cinque livelli a cui la gerarchia funzionale fa riferimento:

- 4° LIVELLO: Logistica e Business Planning, detta le linee guida per l'azienda e la sua strategia;
- 3° LIVELLO: Manufacturing Operations Management, definisce il flusso operativo e controlla le attività produttive mediante ottimizzazione del processo produttivo, gestione delle operazioni e tracciamento del prodotto;
- 2° LIVELLO: destinato al monitoraggio e controllo delle attività alla base del processo produttivo;
- 1° LIVELLO: definisce fisicamente quali attività debbano essere incluse nel processo produttivo;
- 0 LIVELLO: include gli operatori che agiscono fisicamente sul processo

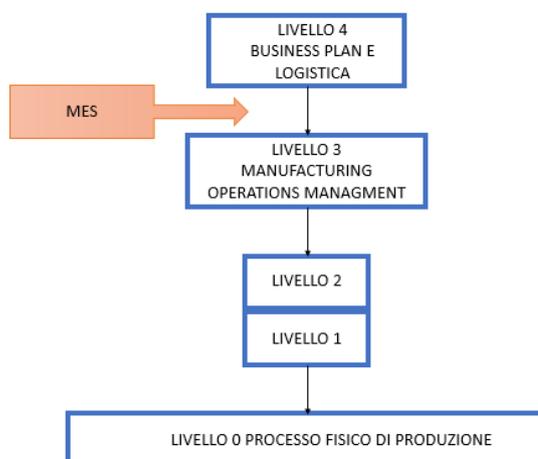


Figura 2: schema dei livelli di una gerarchia funzionale generica con individuazione della posizione in cui il MES deve intervenire.

La posizione d'interesse in cui il MES deve essere collocato è quindi identificata in modo implicito all'interno delle definizioni delle informazioni scambiate tra i vari livelli.

Gerarchia di Equipaggiamento

Secondo questa struttura viene esplicitata l'area funzionale di riferimento, mostrando per ogni livello l'asset fisico sfruttato dall'impresa.

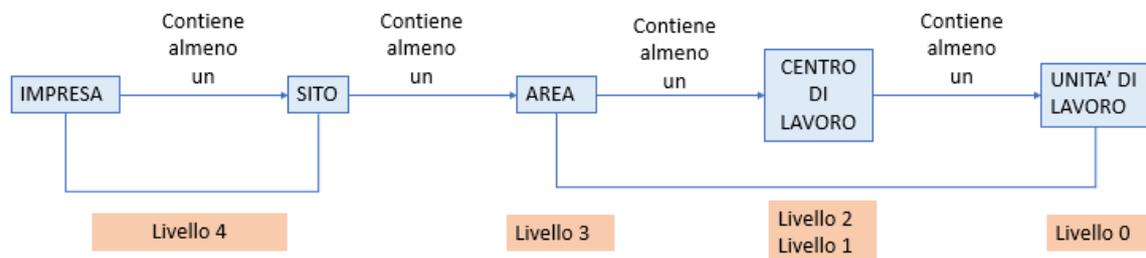


Figura 3: illustrazione della struttura della gerarchia di equipaggiamento con riferimento alla gerarchia funzionale.

La struttura della gerarchia è così distinta:

- L'impresa paragonabile al quarto livello funzionale, si occupa della strategia aziendale e programma la produzione definendo i prodotti e le modalità di lavorazione e associandoli ai siti che si occuperanno della produzione.
- Il sito definisce quella parte di azienda caratterizzata da una determinata locazione geografica. Nel sito è possibile eseguire le lavorazioni dei prodotti assegnati dall'impresa. Ogni sito contiene almeno un'area.
- Le aree sono vincolate ai siti e contengono uno o più centri di lavoro.
- I centri di lavoro contengono le unità produttive, i sistemi di movimentazione, le linee di produzione e i magazzini dedicati a ricezione delle materie prime e alla conservazione e spedizione dei prodotti finiti o semilavorati.

- Infine le unità di lavoro corrispondenti al livello zero della gerarchia funzionale, quelle zone in cui viene effettuata direttamente la lavorazione, che sia una macchina o una risorsa umana a compiere l'azione pratica.

2.2.1.2 Modelli a Flusso di Dati

Definiti anche con la terminologia inglese di Data Flow Diagram, sono tra i modelli maggiormente sfruttati per i sistemi informativi e le analisi dei flussi in quanto permettono di evidenziare schematicamente lo scambio informativo interno al sistema.

Uno dei fattori critici per una azienda è comprendere la locazione dei dati, per capirne la provenienza e l'affidabilità, individuare da quali componenti del sistema vengono elaborati e quale sia la loro destinazione.

I Data Flow Diagram sono definiti dai seguenti componenti:

- Le funzioni, responsabili dell'elaborazione delle informazioni, che permettono di trasformare i dati di input in output;
- Il flusso di dati, che indica la direzione delle informazioni;
- Gli archivi, responsabili della conservazione dei dati e delle informazioni che devono essere elaborate;
- Gli agenti esterni al sistema, fonti o destinatari del flusso di dati.

Possiamo considerare un sistema MES come il mezzo che elabora i dati provenienti dalla produzione, nonché il tramite per la trasmissione dei dati elaborati al sistema di gestione aziendale. In questo caso il flusso dei dati è bidirezionale in quanto l'informazione deve ritornare alla produzione per garantire la corretta comunicazione tra le funzioni aziendali. Il sistema MES è quindi l'intermediario del dialogo informativo tra area produttiva e area gestionale dell'impresa.

2.2.1.3 Coinvolgimento delle funzioni aziendali

La prima parte della normativa è dedicata alla definizione delle attività di supporto alle funzioni aziendali:

1. **Elaborazione dell'ordine:** gli ordini di produzioni ricoprono un ruolo fondamentale per la gestione dei flussi di materiale. Le informazioni fornite dagli ordini ricoprono un ruolo importante anche per le previsioni dei piani di produzione futuri.
2. **Pianificazione della produzione:** la comunicazione della pianificazione della produzione è fondamentale per definire la capacità produttiva, il piano di produzione e valutare lo stato dell'ordine.
3. **Controllo della produzione:** fondamentale per creare gli standard per efficientare il processo produttivo.
4. **Gestione della manutenzione:** uno dei fattori critici per una azienda è gestire la manutenzione senza compromettere la capacità produttiva.
5. **Controllo qualità:** la gestione della qualità permette all'azienda di ridurre gli sprechi e quindi i costi, massimizzando l'efficienza.
6. **Controllo sui materiali e sul consumo di energia:** anche in questo caso valutare il consumo di materiale ed energia rientra in quelle funzioni che permettono all'azienda di abbattere i costi.
7. **Approvvigionamento:** le attività di approvvigionamento efficienti permettono di ridurre le scorte a magazzino e i costi.
8. **Inventario di produzione:** questa funzione permette di valutare l'inventario prodotti finiti o semilavorati per evitare sovrapproduzioni e quindi costi extra.
9. **Contabilità e costi prodotto:** permette di definire dove vi siano effettivamente gli sprechi e quindi quali siano le criticità aziendali.

10. Spedizione del prodotto e sua amministrazione: funzione dedicata alla gestione della fase di trasporto del prodotto finito.

11. Marketing e vendite: le analisi di mercato risultano fondamentali per il lancio di un nuovo prodotto. Per quanto riguarda i prodotti esistenti il marketing si occupa di monitorare e pianificare le vendite per rispondere alle variazioni del mercato

12. Ricerca e sviluppo prodotto: l'innovazione tecnologica è alla base del mercato attuale, il ciclo vita del prodotto limitato deve necessariamente essere accompagnato dallo sviluppo tecnologico, questa funzione è quindi fondamentale per ogni azienda che vuole ricoprire una posizione dominante all'interno del mercato.

2.2.2 Parte 3: Modelli di Attività di Gestione delle Operazioni Produttive

In questo capitolo della normativa vengono descritte quelle funzioni che saranno gestite da un sistema MES: produzione, manutenzione, qualità, inventario.

2.2.2.1 Le attività di produzione

Con attività di produzione vengono definite tutte quelle che permettono di trasformare le materie prime in prodotti finiti a valore aggiunto idonei a soddisfare le esigenze di mercato e clienti. Al fine di ottenere il risultato vengono sfruttate risorse materiali, quali macchinari, componenti, attrezzature, etc. e risorse immateriali quali, energia, tempo etc.

Il modello definito nello standard prevede alcune attività specifiche tra le quali vengono scambiate informazioni:

- Definizione e gestione del prodotto;
- Gestione delle risorse;
- Pianificazione della produzione;
- Tracciabilità;

- Allocazione prodotti-unità produttive
- Gestione ed esecuzione della produzione
- Raccolta dati
- Analisi delle performances.

2.2.2.2 Le attività per la gestione della qualità

Vengono considerate tutte quelle attività che mirano a controllare e monitorare la qualità dell'output di lavorazione, in modo che questo possa rispettare gli standard qualitativi imposti da azienda e cliente.

Il flusso informativo generato permette di risalire a quelle criticità produttive che minano la conformità dei pezzi. L'intervento sulle aree critiche è fondamentale per ridurre al minimo le difettosità e i pezzi non conformi, con conseguente riduzione degli sprechi.

Tra le attività che vengono svolte possiamo identificare:

- Monitoraggio delle materie prime per valutarne l'adeguatezza agli standard aziendali
- Monitoraggio del prodotto finito per valutarne l'adeguatezza agli standard imposti dal cliente
- Test di qualità richiesti dal livello 4 della struttura gerarchica
- Comunicazione dei dati per l'aggiornamento dei livelli interessati, in caso di non conformità dovranno essere informate tutte quelle entità che interagiscono con il componente difettoso.

2.2.2.3 Le attività di gestione della manutenzione

Uno dei fattori critici all'interno della linea produttiva è quello della manutenzione, in quanto limita le risorse disponibili alla produzione.

Vengono identificate tre tipologie di manutenzione:

- Manutenzione correttiva: è quella non pianificata che intercorre in caso di guasto macchina;
- Manutenzione programmata: sono le attività pianificate di manutenzione che vogliono prevenire i guasti;
- Manutenzione per ottimizzazione: include tutti quegli interventi che mirano a migliorare le prestazioni aziendali.

2.2.2.4 Le attività di gestione dell'inventario

Includiamo in questo campo tutte quelle attività che mirano a tenere traccia del flusso di prodotto, a partire dall' immagazzinamento della materia prima fino allo stoccaggio dei prodotti finiti.

Queste attività sono cruciali per tracciare i prodotti e risalire alla causa origine dei problemi.

I sistemi MES Contribuiscono a tracciare i prodotti creando uno storico delle operazioni.

CAPITOLO 3

SCHEDULAZIONE DELLA PRODUZIONE

La programmazione del lavoro è, ad oggi, sempre più importante nella realtà della produzione industriale in quanto necessaria per ottimizzare la sequenza delle lavorazioni che devono essere svolte nell'officina.

I sistemi MES permettono, grazie alla analisi dei dati e delle informazioni, di ottimizzare la produzione.

Le operazioni che vengono svolte per gestire la produzione sono:

- *Schedulazione*: è l'operazione che assegna le priorità alle attività e le alloca agli opportuni centri di lavorazione che dovranno processarle. In seguito alla schedulazione, che avviene in forma teorica, si procede all'esecuzione pratica delle attività.
- *Monitoraggio*: è la sequenza di operazioni che va a confrontare lo stato di avanzamento lavori effettivo con quello previsto dallo scheduling. È necessario perché un eventuale ritardo può andare a creare una catena di eventi che porterebbe infine un danno.
- *Controllo*: è la fase che va di pari passo al monitoraggio; nel momento in cui dovesse venir riscontrata una discrepanza è necessario intervenire in maniera repentina. L'intervento può causare anche una nuova schedulazione della produzione.

3.1 LA SCHEDULAZIONE

Lo scheduling consiste nella realizzazione di un programma, che prende il nome di schedule, visualizzabile in forma grafica tramite diagramma di GANTT che riporta sui suoi assi le attività pianificate in funzione del tempo. Ogni attività viene identificata tramite una barra

la cui lunghezza rappresenta la durata. Per indicare la successione delle attività in alcune rappresentazioni vengono poste delle frecce che collegano le varie attività.

Le connessioni tra le attività possono essere di tre tipi:

- *Start to start*: il vincolo temporale per l'inizio di una attività è legato all'inizio di un'altra;
- *End to start*: l'inizio di una attività è vincolata alla fine di un'altra;
- *Start to end*: l'attività può finire solo nel caso in cui un'altra sia iniziata;
- *End to end*: la fine di un'attività è vincolata alla fine di un'altra.

In contesto produttivo la connessione più logica e frequente è *end to start* per due ragioni principali:

- L'attività che deve essere processata da un determinato centro di lavoro deve attendere che questo sia libero e che abbia, quindi, concluso l'attività precedente.
- Una sequenza di operazioni comporta che per effettuare la successiva sia terminata quella precedente; salvo alcune eccezioni è difficile effettuare due lavorazioni successive sullo stesso pezzo contemporaneamente.

“Optimization of Logistics” (Yalaoui, et al. 2020) identifica quattro componenti che definiscono un problema di schedulazione:

- Attività
- Risorse
- Vincoli
- Obiettivi

3.1.1 Attività

Un'attività è una sequenza di operazioni che permette di trasformare un input in un output con valore economico maggiore.

“L'analisi delle a., l'analisi della trasformazione di beni o fattori in altri beni o prodotti, accompagnata dalla definizione dei rapporti fissi in cui si verifica il processo di trasformazione” (Enciclopedia Treccani s.d.)

Le attività possono essere caratterizzate in base a:

- Data di inizio al più presto: prima di questa l'attività non può assolutamente iniziare.
- Data di fine al più tardi: è la data ultima possibile per terminare l'attività senza sfociare in ritardi.
- Data di inizio e di fine: le date in cui vengono assegnate rispettivamente l'inizio e la fine dell'attività, la distanza tra loro dipende dalla durata.
- Durata: il periodo di tempo necessario per processare l'attività
- Predecessori: sono quelle attività che devono essere svolte prima senza le quali l'attività in esame non può iniziare.
- Successori: le attività che devono essere sequenziate dopo quella in esame.

3.1.2 Risorse

In un problema di scheduling ogni attività deve essere svolta da una o più risorse.

Per schedulare le attività all'interno di un'officina di produzione vengono considerati i singoli centri di lavorazione, la cui definizione è ben definita nel testo Operation Management (Jacobs, et al. 2020)

“Un centro di lavorazione è un’area dell’azienda in cui sono organizzate le risorse produttive e in cui viene eseguita un’attività specifica. Il centro di lavorazione può essere un’unica macchina, una serie di macchine o un reparto dedicato per una specifica attività.”

Il centro di lavorazione è quindi composto da un insieme di macchinari più o meno automatizzati e da un numero variabile di risorse umane.

Una risorsa può essere:

- Non disponibile: nel caso in cui sia assente, guasta o in manutenzione;
- Disponibile: se può processare le attività, in questo caso vengono individuati i seguenti stati:
 - Assegnata: la risorsa sta processando l’operazione;
 - In attesa: la risorsa sta aspettando la sua prossima attività da processare.

Lo scheduling si fa carico di allocare le risorse necessarie alle attività.

3.1.3 Vincoli

In ogni problema di scheduling vi sono dei vincoli derivanti da contesto produttivo, natura delle lavorazioni, caratteristiche delle risorse.

Introduction to Production Planning (Villa 2006) identifica alcune assunzioni che sono necessarie in un problema di schedulazione:

- ❖ “Ogni job è trattato come una realtà inseparabile. Ne consegue che viene considerato l’insieme dei componenti da processare nel caso in cui siano più di uno. Questa limitazione obbliga a considerare i lotti nel loro complesso prima di essere trasportati nel centro di lavoro successivo.
- ❖ Ogni risorsa processa un pezzo per volta, specialmente nel caso di lavorazioni con macchine utensili. Questa affermazione, valida per tutte le lavorazioni su macchina

utensile, cade in alcuni contesti produttivi particolari quali tessile alimentare o in alcuni trattamenti particolari quali termici e chimici.

- ❖ I centri di lavorazione sono sempre disponibili. Vengono trascurati eventi imprevisti come malfunzionamenti o rotture dei macchinari. È dato per scontato che la manutenzione preventiva effettuata alle macchine sia sufficiente a prevenire le interruzioni della produzione e che venga effettuata al di fuori delle ore di produzione. Resta limitante per alcuni tipi di produzione, quali quelle automobilistiche per le quali non possono essere escluse a priori delle interruzioni della produzione.
- ❖ Un problema classico di schedulazione è deterministico, ne consegue che tutti i dati necessari siano noti a priori. A conseguenza di ciò eventuali update della schedulazione possono interferire solo con il periodo di tempo successivo la schedulazione già effettuata. Non sono neanche considerate altre fonti di incertezze quali variabilità delle date di arrivo dei componenti, rotture o blocchi macchina. Per integrare elementi di incertezza o variabilità delle date è necessario ricorrere ad una programmazione probabilistica.”

3.1.4 Obiettivi

L'obiettivo della schedulazione è quello massimizzare l'efficienza del sistema produttivo, questo può essere fatto rispettando alcuni criteri. È importante sottolineare come la scelta di quali criteri debbano essere considerati dipende dal contesto produttivo, dal sistema scelto, e dalle esigenze dell'azienda.

3.2 I DETERMINANTI DELLA SCHEDULAZIONE

Dopo aver introdotto quali elementi, vincoli e obiettivi siano necessari per effettuare una programmazione delle attività, vengono analizzati quei fattori che determinano le priorità di assegnazione.

3.2.1 Sistema di produzione

La base di partenza per gestire attività e algoritmi è data dalla conoscenza del sistema produttivo aziendale.

Questo è determinato dalla quantità, dalla tipologia e dalla varietà dell'output di produzione.

Possiamo identificare tre diversi sistemi di produzione distinti in base alla sequenzialità delle operazioni (Jia, Jiang e Li 2015) (Kuznetsov e Noskova 2020):

- *Flow shop*: tutti i job devono subire il medesimo processo di lavorazione, determinato a priori, vengono sequenziate le operazioni su tutte le risorse presenti all'interno della linea produttiva.
- *Job shop*: ogni job ha una sua precisa sequenza di lavorazione che è determinata a priori ma varia di caso in caso. Alcune macchine possono essere omesse per alcuni job.
- *Open shop*: ogni job deve essere processato su tutte le macchine, senza alcuna restrizione riguardo la sequenza. Lo schedulatore ha la libertà di determinare la sequenza per ogni job, questa può variare di caso in caso in quanto non determinata a priori.

Questa categorizzazione è abbastanza restrittiva in quanto non prevede centri di lavorazione in cui più risorse lavorano in parallelo

Includendo questa ipotesi vengono introdotte flow shop e job shop flessibili (Pinedo 2005).

- *Flow shop flessibile*: è una generalizzazione del caso riportato in precedenza che include centri di lavorazione con macchine in parallelo. Vi sono un numero finito di stage che lavorano in serie, come in un *flow shop* tradizionale in ognuno dei quali

possono essere presenti una o più macchine che lavorano in parallelo. Ogni lavorazione necessita di essere processata da una singola macchina per stadio e necessita di essere processata da ogni stadio.

- *Job shop flessibile*: in questo caso il sistema di produzione è del tipo job shop con più macchine che possono lavorare in parallelo per ogni stadio. Similmente a quanto detto in precedenza ogni prodotto necessita di essere processato da una singola macchina per stage, ma essendo le sequenze di lavorazioni diverse per ogni job, l'ordine degli stage cambia di volta in volta.

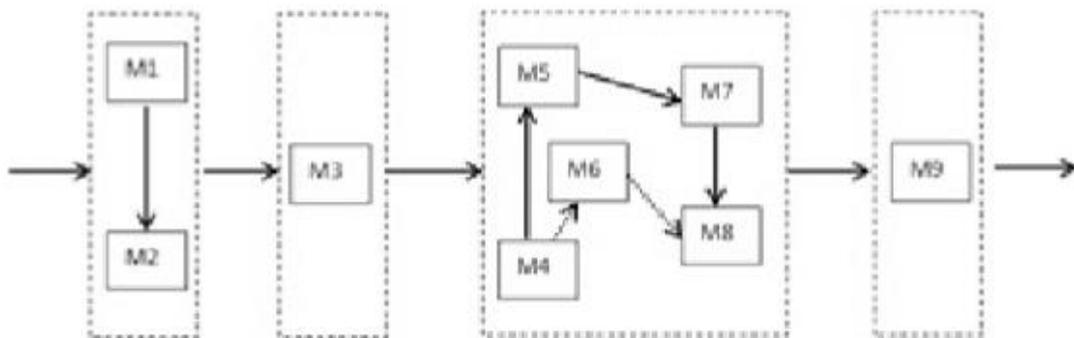


Figura 4: Esempio di layout open shop (Yalaoui, et al. 2020)

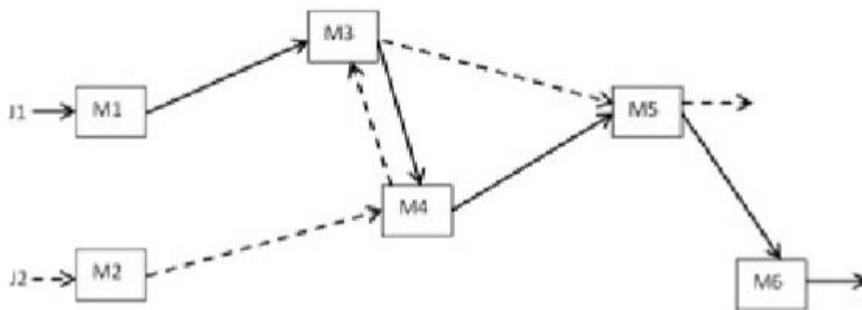


Figura 5: Esempio di layout job shop (Yalaoui, et al. 2020)

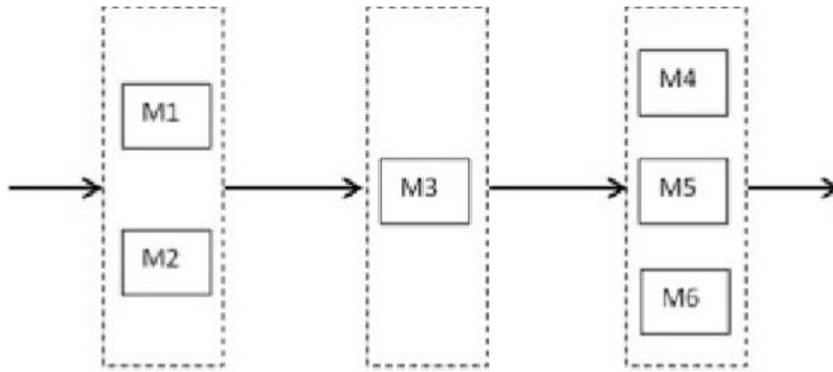


Figura 6: Esempio di layout flow shop (Yalaoui, et al. 2020)

3.2.2 Direzione della schedulazione

Un ulteriore elemento che caratterizza la programmazione è la data di partenza scelta per allocare le attività: uno schedule può essere generato backward o forward. La scelta di quale sia il più adatto dipende dagli obiettivi.

3.2.2.1 Forward scheduling

La programmazione parte dalla prima attività, la cui data di inizio viene fissata as soon as possible.

Considerando x_i come la data di inizio al più presto di una qualsiasi attività e con d_i la durata di tale attività possiamo calcolare la data di inizio dell'attività successiva come:

$$x_{i+1} = x_i + d_i$$

Eccezion fatta per la prima attività la cui data di inizio è forzata da vincoli esterni la programmazione.

Seguendo questa direzione otterremo anche le date di fine al più presto per la sequenza. Queste sono molto importanti in quanto permettono di visualizzare lo scarto tra l'ipotetica fine e la data di consegna.

3.2.2.2 Backward scheduling

Iniziamo la programmazione partendo dalla data ultima in cui l'attività deve terminare, solitamente in campo industriale coincide con la consegna del componente.

Fissato quindi il termine della sequenza di N attività y_N è possibile ricavare le date di fine al più tardi per ognuna

$$y_{i-1} = y_i - d_i$$

Questa procedura permette di valutare il quantitativo di ritardo iniziale massimo oltre al quale la consegna è in ritardo.

Entrambe i metodi sono validi, tuttavia, il primo dei due ha il vantaggio di evidenziare il tempo rimanente tra la fine delle operazioni e la data di consegna.

Considerando che in qualsiasi contesto produttivo non esiste il rischio zero, è sempre bene tutelarsi terminando i lavori con uno scarto rispetto la data di consegna.

3.3 MODELLI MATEMATICI

Il problema della schedulazione job shop consiste nell'assegnare un set di attività ad un set di macchinari. Il problema di programmazione dei job shop (JSP) è noto per essere uno tra i più difficili ed è stato oggetto di molti sforzi di ricerca per diversi decenni (Wisittipanich e Kachitvichyanukul 2012). In contesto flessibile la difficoltà aumenta in quanto la possibilità di lavorare su macchine in parallelo accresce le possibilità del sistema. Possiamo suddividere i modelli in due macrocategorie (Jain e Meeran 1999) ovvero i metodi esatti e quelli approssimativi.

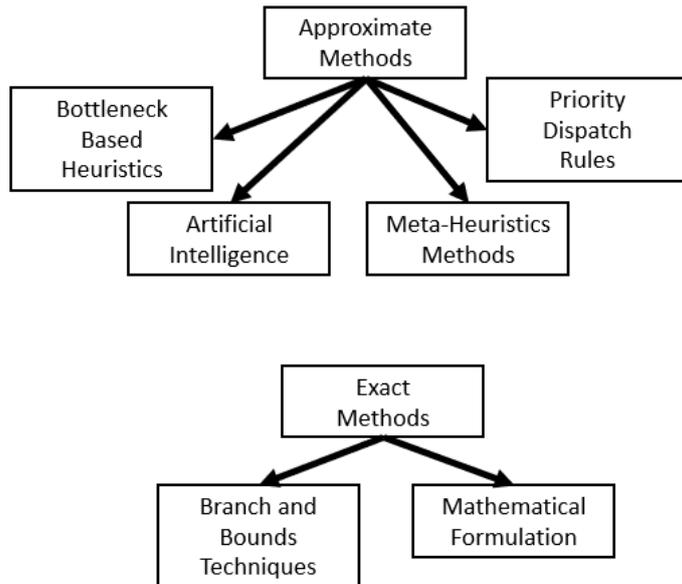


Figura 7: metodi matematici per la soluzione di problemi job shop flessibile (Jain e Meeran 1999)

3.3.1 Exact Methods

I metodi esatti sono solitamente rappresentati da algoritmi *branch and bound* (Rego and Duarte 2009) o da formulazioni matematiche, tuttavia, a causa della complessità dei problemi di schedulazione *flexible job shop*, che come già detto rientrano nei problemi di classe NP-ardui, questi metodi sono applicabili esclusivamente a problemi di dimensioni modeste. (M.R. Gary 1979)

È necessario quindi applicare metodi di approssimazione (euristici) per trovare soluzioni a problemi più ampi. (Rego and Duarte 2009)

3.3.2 Approximate Methods

I metodi approssimati prendono il nome di metodi euristici e si basano su assunzioni plausibili, ma non matematiche, delle proprietà della soluzione ottima del problema.

Un procedimento euristico è un approccio ad un problema che non segue un preciso percorso ma che analizza diverse opzioni cercando di arrivare ad una soluzione di ottimo.

Un metodo può essere applicato sia a problemi lineari sia a problemi non lineari. La differenza principale è nell'individuazione dell'ottimo, in quanto in un problema lineare, una soluzione di ottimo locale coincide con la soluzione di ottimo globale, situazione plausibile ma non assoluta per i problemi non lineari che necessitano di ulteriore ricerca per l'ottimizzazione globale. (Kumar, Jain e Sharma 2019)

Uno dei principali limiti da superare con un approccio euristico è, quindi, quello di oltrepassare le soluzioni di ottimo relative in caso di problemi non lineari. L'esempio in figura rappresenta due soluzioni di ottimo locale X_1 e X_2 .

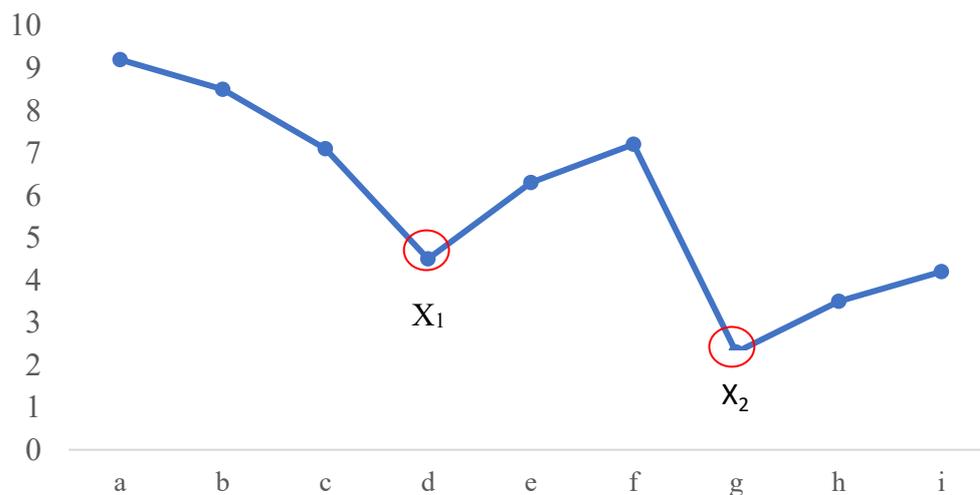


Figura 8: esempio di ottimo locale e ottimo globale

Sulle ascisse presentiamo le ipotetiche soluzioni derivate dall'applicazione di un metodo euristico

Sulle ordinate gli ipotetici giorni necessari a portare a termine una schedulazione

I punti contrassegnati in rosso identificano un punto di ottimo locale.

Considerando il punto X_1 possiamo notare come a partire dal primo risultato le soluzioni siano progressivamente migliorative, tuttavia il punto in questione presenta un flesso e le due soluzioni successive si evidenziano come peggiorative. Questo può portare ad una trappola in quanto, se la schedulazione viene effettuata da un operatore, per ridurre i tempi necessari a trovare l'ottimo assoluto, in questo caso X_2 , potrebbe accontentarsi della prima soluzione, mentre se l'algoritmo viene applicato su software ma non è stato ben sviluppato potrebbe anche questa volta accontentarsi della prima soluzione, soprattutto nel caso di algoritmi che sfruttano la convergenza, come vedremo in seguito.

3.3.2.1 Priority Dispatching Rules

La prima tecnica euristica presentata prevede di assegnare un valore di priorità ai vari job, solitamente in ordine crescente, quindi minore è il valore assegnato maggiore sarà l'urgenza di terminare quel dato job. (Sculli e Tsang 1990)

Vengono analizzate le principali regole sfruttate per assegnare le priorità alle diverse attività. L'efficacia della regola applicata dipende fortemente dal contesto produttivo e dagli obiettivi che devono essere perseguiti dallo schedulatore. (Jacobs, et al. 2020)

First In First Out

La logica FIFO prevede di dare precedenza cronologica alle lavorazioni in base alla data di arrivo.

Questo permette di evitare lunghe rimanenze a magazzino materie prime o semilavorati.

Ovviamente è più facile visualizzare questa tecnica più su un magazzino rispetto che applicata ad una schedulazione, all'atto pratico questa dare precedenza agli ordini di cui il materiale è giunto prima.

In alternativa questa regola può prendere il nome di FASPS (*first arrival at the shop first to be served*).

Last In First Out

Questa regola è facilmente visualizzabile pensando ad una cesta in cui l'ultimo oggetto inserito è il primo ad uscire.

Per seguire questa regola è necessario dare priorità agli ordini appena arriva il materiale, anche posticipando ordini precedenti.

Shortest Processing Time o Shortest Operating Time

Seguendo questa la priorità viene assegnata in base alla durata delle attività, a partire dalla più breve.

Earliest Due Date

Uno dei requisiti più importanti per un centro di lavorazione è quello di rispettare le date di consegna, seguendo questa regola si ordinano i job in base alle date di consegna.

Slack Time Remaining

Per introdurre questa regola è necessario specificare cosa si intenda con slittamento, ovvero, l'intervallo di tempo entro cui un'attività può iniziare o terminare senza causare ritardi alla consegna. Se lo slittamento libero è nullo l'attività è definita critica e ogni suo ritardo compromette la data di consegna.

La regola prevede di calcolare lo slittamento come periodo di tempo che manca alla data di consegna escluso della durata dell'attività. La priorità deve essere quindi assegnata a quelle attività che hanno minore slittamento, a partire da quelle critiche.

È anche possibile rapportare lo slittamento alle operazioni da fare, andando a distribuirlo uniformemente su tutte le operazioni.

Critical Ratio

Similmente al caso precedente si possono confrontare i giorni rimanenti alla consegna, o per meglio dire i giorni necessari a completare l'attività, con i giorni lavorativi ancora disponibili, in questo caso rapportandoli.

Essendo auspicabile che il tempo mancante sia superiore rispetto alla durata dell'attività questo rapporto si pone $CR \leq 1$ per cui la priorità deve essere assegnata quelle attività con CR più prossimo all'unità.

Longest Remaining Operation

Assegna la priorità a quei job che hanno un tempo maggiore per il completamento esclusa quella che è in processo.

3.3.2.2 Evolutionary Algorithms

Negli ultimi anni sono stati sviluppati approcci meta-euristici, chiamati *Evolutionary Algorithms* (EA), sviluppati a partire dalla conoscenza di alcuni rami scientifici, tra i quali matematica, chimica, fisica, biologia, ingegneria e altre.

Caratteristiche principali (Kumar, Jain e Sharma 2019) degli algoritmi evolutivi sono:

- Flessibilità rispetto alla diversità dei problemi
- Robustezza, possono essere applicati anche in caso di incertezza
- Adattabilità a eventi dinamici
- Autonomia dall'intervento umano

Le principali aree di applicazione degli EA sono (Kumar, Jain e Sharma 2019):

- Pianificazione delle routines e schedulazione
- Design di circuiti elettronici e network neurali
- Identificazione delle funzioni predittive per dati medici
- Identificazione di spam

Genetic Algorithms

Gli algoritmi genetici derivano dalla rielaborazione deterministica della teoria dell'evoluzione di Darwin, e quindi delle continue mutazioni genetiche che avvengono nelle specie.

La selezione naturale e la sopravvivenza del più adatto rappresentano il metodo di scelta delle soluzioni.

Le mutazioni del genoma rappresentano il metodo in cui vengono generate nuove soluzioni.

Vengono identificate principalmente due cause per le mutazioni del genoma:

- Mutazioni randomiche sviluppate durante la riproduzione cellulare a causa di esposizioni chimiche o radiative.
- Ricombinazioni genetiche causate dalla riproduzione

Genericamente l'algoritmo procede seguendo alcuni step (Holland 1975):

- 1) Viene generata una popolazione iniziale, in cui ogni soluzione prende il nome di *cromosoma*.
- 2) Viene controllata la validità di ogni individuo della popolazione.
- 3) Gli individui che non superano la selezione vengono scartati.
- 4) Se la popolazione non è adatta si procede applicando tre operatori:
 - a. Selezione;
 - b. Crossover;
 - c. Mutazione.
- 5) Viene quindi generata la prole di cui vengono memorizzati i migliori cromosomi identificati.
- 6) Ripetere da 2) per il numero di volte scelto.

L'interruzione dell'algoritmo può essere generata da alcuni criteri impostati in partenza, questi possono essere un numero finito di iterazioni che si vogliono far fare al programma, un tempo computazionale prefissato o altri criteri in base alle necessità. (Farashahi et al. 2011).

Uno dei punti di forza dell'algoritmo genetico è il suo funzionamento non seriale, dovuto alla varietà e alle proprietà degli operatori che vengono applicati e che aumentano il grado di casualità del processo permettendo di muoversi nello spazio delle soluzioni in tutte le direzioni.

Un altro benefit apportato da questo algoritmo è, come già detto, la capacità di superare le soluzioni di ottimo locali, in quanto non sfrutta la convergenza per arrivare all'ottimo.

Di contro è necessario inizializzare una popolazione per rendere il problema ammissibile e, inoltre, la convergenza non garantita può allungare i tempi necessari alla computazione del programma.

Analizziamo dunque alcuni casi in cui può essere adattato l'algoritmo (Kumar, Jain e Sharma 2019)

Combinatorial Optimization

È la procedura di ricerca dell'ottimo in cui il dominio della funzione obiettivo è limitato ma lo spazio di ricerca è molto ampio.

Uno dei casi che si presta bene come esempio è quello della schedulazione di problemi job-shop.

Per risolvere problemi di questo tipo sono necessarie alcune modifiche all'algoritmo e viene aumentato il tasso di mutazione.

Multi-Objective Optimization

Gli algoritmi genetici possono essere applicati anche nel caso in cui vi siano un numero variabile di funzioni da ottimizzare, in questo caso l'ottimizzazione della singola funzione può andare in contrasto con i risultati delle altre.

È necessario quindi un'ottimizzazione globale del problema, che all'atto pratico fornisce un set di soluzioni che vadano a soddisfare gli obiettivi ad un livello accettabile senza compromettere le altre soluzioni.

3.3.2.3 Simulated Annealing

Gli algoritmi di tempra simulata prendono ispirazione dalla fisica e in particolare dal processo di tempra dei metalli, nel quale gli atomi ad alte temperature, quindi con elevata energia e disordine, vengono raffreddati in maniera controllata per bloccare la loro posizione ed evitare che si formino difettosità a livello molecolare.

Lo sviluppo di questo algoritmo parte dagli studi sulla simulazione del comportamento di un insieme di atomi in equilibrio termico ad una certa temperatura (Metropolis, et al. 1953).

L'obiettivo di questo algoritmo non è solo quello di accettare cambiamenti che siano migliorativi per la funzione obiettivo, ma anche di accettare con una certa probabilità modifiche che a primo impatto risultino peggiorative.

Probabilità determinata da:

$$p(E) = \exp\left(-\frac{\delta E}{k_B T}\right)$$

In cui δE indica il cambiamento energetico, T è la temperatura di controllo, k_B è la costante di Boltzmann.

Per applicare l'algoritmo si assume la variazione energetica pari alla variazione della funzione obiettivo δf opportunamente corretta con una costante, che per motivi pratici può essere considerata unitaria:

$$\delta E = \gamma \delta f$$

Da cui:

$$p(\delta f; T) = e^{-\frac{\delta f}{k_b T}}$$

Un'ultima approssimazione percorribile è quella di assegnare valore unitario anche alla costante di Boltzmann.

Per accettare una soluzione è quindi necessario imporre

$$p(\delta f; T) = e^{-\frac{\delta f}{T}} > r$$

Con r numero soglia casuale preso da una distribuzione uniforme $[0,1]$.

È importante notare come T sia un fattore fondamentale dell'equazione, deve essere quindi scelto con cautela in quanto se T è un valore alto le probabilità di accettazione della soluzione sono elevate mentre a valori di T prossimi a 0 la probabilità si riduce a tal punto da incombere nella trappola dei minimi locali (Yang 2019).

Per la scelta del valore T si possono applicare un metodo lineare o geometrico:

- $T = T_0 - \beta t$
- $T(t) = T_0 \alpha^t$

In cui α e β sono le costanti di raffreddamento, dalla pratica $\alpha = 0.7 - 0.99$ mentre β dipende dal massimo numero di iterazioni t_f che si vogliono fare per cui $\beta = \frac{T_0}{t_f}$.

3.3.2.4 Differential Evolution

Differential Evolution algorithm è stato sviluppato verso la fine degli anni '90 da R. Storn e K. Price, e risulta quindi uno dei più recenti (Storn e Price 1997).

È un algoritmo con tendenze ad auto-organizzarsi e che necessita di una ridotta memoria computazionale, in termine di tempi di lavoro della CPU (Qian e al. 2008), a causa della sua struttura molto semplice. Questo lo rende molto efficiente per la soluzione di problemi

continui (Kumar, Jain e Sharma 2019). Inoltre, essendo le stringhe di soluzione composte da numeri reali, non è necessaria alcuna codifica e decodifica (Yang 2019).

Come altri algoritmi basati sulla popolazione, ad esempio i genetic algorithm, gli algoritmi di evoluzione differenziale vengono inizializzati con una popolazione generata in maniera randomica.

Le soluzioni sono rappresentate da vettori D-dimensionali, composti, come detto in precedenza, da numeri reali.

Al momento dell'inizializzazione del programma vengono generati n vettori.

Il tratto caratteristico di questo algoritmo sono le operazioni che vengono effettuate, ovvero, mutazioni, crossover e selezione. Queste vengono iterate fino a che non si realizza la condizione che termina la simulazione (Wisittipanich and Kachitvichyanukul 2012).

3.3.2.5 Swarm Intelligence Algorithms

La categoria *Swarm Intelligence* comprende quegli algoritmi che sfruttano la conoscenza portata dall'interazione di singole unità che interagiscono singolarmente nel contesto per trovare una soluzione globale al problema. Tutto questo prende spunto da elementi esistenti in natura, quali stormi di uccelli, banchi di pesci, sciami di api o colonie di formiche.

Non è previsto un controllo globale delle singole unità che agiscono in maniera randomica condividendo informazioni tra loro per ottenere una soluzione globale.

Particle Swarm Optimization

In particolare, la *Particle Swarm Optimization*, ha origine nel 1995 (Kennedy e Eberhart 1995) quando per la prima volta è stata sfruttata per un problema di ottimo, prendendo spunto da una simulazione del 1990 in cui veniva simulato uno stormo di uccelli. L'obiettivo del

primo test effettuato è stato quello di realizzare un programma che fosse in grado di cercare diverse soluzioni di ottimo locale e non solo quello globale.

Le “particelle” dell’algoritmo sono in grado di muoversi nello spazio delle soluzioni ad una certa velocità. Ogni posizione dello spazio occupata da una particella rappresenta una possibile soluzione.

Generalmente le “particelle” sono governate dalle seguenti equazioni:

- Equazione di velocità:

$$v_i(t + 1) = w v_i(t) + c_1 r_1 (p_{best,i} - x_i(t)) + c_2 r_2 (g_{best} - x_i(t))$$

- Equazione di posizione:

$$x_i(t + 1) = x_i(t) + v_i(t + 1)$$

Di cui

- $i = 1, 2, \dots, p$; p : numero di particelle
- $p_{best,i}$ indica la posizione migliore ottenuta dalla particella i -esima
- g_{best} indica la posizione migliore in assoluto trovata
- v : velocità della particella nella singola dimensione
- t : numero dell’iterazione
- w : costante di inerzia
- $c_1; c_2$ costanti di accelerazione
- $r_1; r_2 \forall [0; 1]$: numero casuale che assegna un peso alla componente sociale o del singolo.

Nell’equazione della velocità si possono distinguere tre componenti:

- $w v_i(t)$ indica l’inerzia che la particella si porta dietro dalla simulazione precedente, è necessario che all’aumentare delle simulazioni questa sia decrescente per evitare che la particella schizzi via, è importante scegliere un valore di w che permetta di

dare una certa convergenza alle soluzioni. È possibile anche assegnare un valore randomico.

- $c_1 r_1 (p_{best,i} - x_i(t))$ è la componente della singola particella mentre $c_2 r_2 (g_{best} - x_i(t))$ indica la componente sociale. c_1 e c_2 indicano le costanti di accelerazione che permettono alla particella di allontanarsi dalle soluzioni trovate in precedenza, queste costanti possono variare di simulazione in simulazione. r_1 e r_2 sono valori randomici che permettono una esplorazione valida dello spazio.

Ant Colony Optimization

Lo sviluppo degli algoritmi ACO parte nel 1991 dagli studi di Dorigo (Dorigo, Maniezzo e Colomi, Positive Feedback as a Research Strategy 1991), ed è stato implementato con trattazioni successive degli stessi autori e di altri. (Dorigo, Maniezzo e Colomi 1996) (Dorigo e Stutzle 2004).

Come altri algoritmi appartenenti alla categoria *swarm intelligence*, anche questo sfrutta un framework composto da un certo numero di individui, coordinati in maniera decentrata e tramite *self-association*.

L'idea alla base del programma è sita nel metodo usato dalle formiche per ricercare il cibo, ovvero uno sviluppo probabilistico di un sentiero al quale vengono aggiunti iterativamente segmenti. L'associazione effettuata per applicare l'algoritmo in un problema di flexible job shop è quella di far coincidere l'inizio e la fine delle attività da schedulare con la tana e il cibo, ovvero i punti di inizio e fine del percorso delle formiche. (Deepalakshmi e Shankar 2020).

Per ottenere una simulazione corretta e più rapida, le formiche devono muoversi in maniera simultanea e non in sequenza.

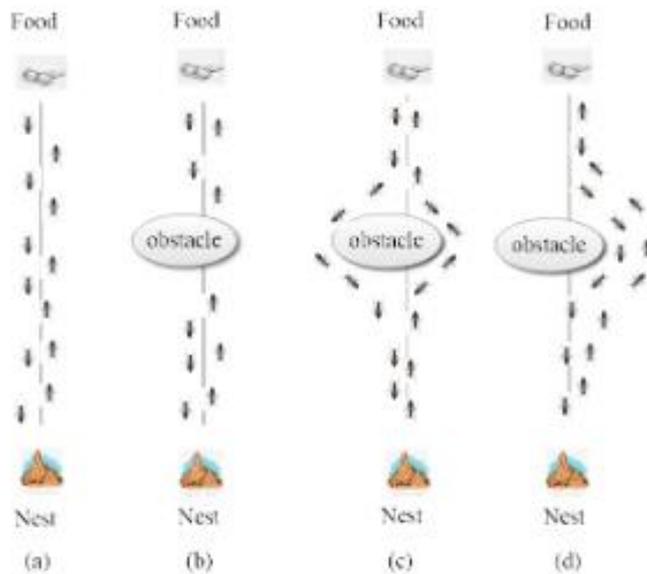


Figura 9: rappresentazione grafica del comportamento degli individui nella ricerca dell'ottimo. a) rappresenta il percorso iniziale, in b) viene introdotto un vincolo, in c) gli individui esplorano le diverse soluzioni e in d) viene rappresentata la soluzione migliore. (Slowik 2020)

L'algoritmo è definito in partenza dai punti seguenti (Slowik 2020):

- Tutte le formiche sono inizializzate al punto di partenza;
- Una certa quantità di feromone è assegnata ad ogni individuo.

Ogni individuo segue determinate regole:

- Preferisce seguire il percorso con la quantità maggiore di feromone
- I punti che sono già stati percorsi sono vietati fino alla fine del circuito
- Quando il percorso termina viene aggiornata la quantità di feromone lasciata su ogni sentiero.

La quantità di feromone rilasciata su ogni punto del sentiero, essendo in partenza una quota fissa, dipende dalla lunghezza del sentiero percorso, a maggiori lunghezze corrispondono minori quantità rilasciate.

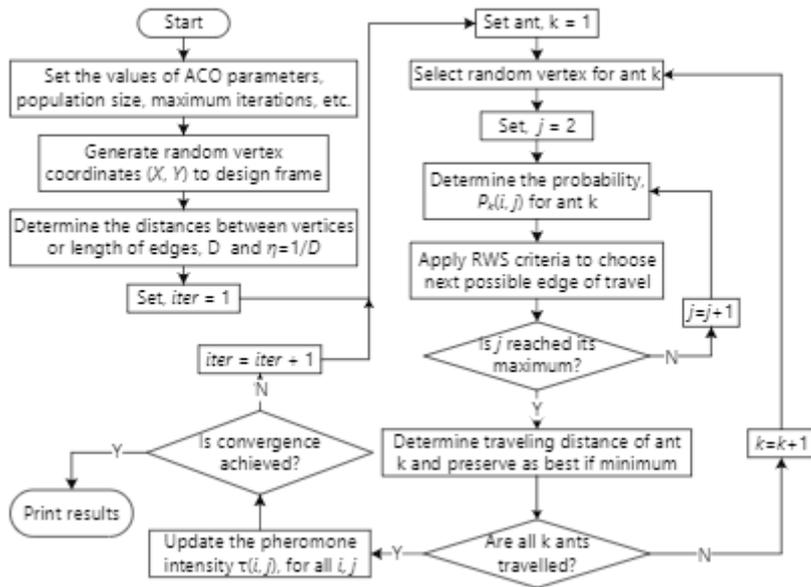


Figura 10: ant colony optimization flow chart

CAPITOLO 4

CASO STUDIO

Alla base degli studi effettuati questa tesi vi è un progetto supportato da IoT4Industry e finanziato da Horizon 2020, il programma più importante e corposo dell'Unione Europea per la ricerca e l'innovazione. Questo progetto, che prende il nome di RTPP, Real Time Production Planning, prevede l'implementazione del sistema Octavic PET all'interno della catena produttiva dell'azienda Eurodies, con la partecipazione dell'Università di Lund e del Politecnico di Torino.

4.1 I SOGGETTI COINVOLTI

4.1.1 Eurodies

Eurodies è una piccola media impresa italiana, con sede ad Avigliana (TO), che si occupa della realizzazione di componenti in lamiera e alluminio per il settore automotive. In particolare, fornisce componenti per case automobilistiche quali: BMW, Audi, Skoda, Volkswagen, Opel, Ford, Porsche, FCA, Seat, Mercedes, Alfa Romeo, Lamborghini.

Le componenti realizzate dall'azienda sono destinate ai prototipi di vetture che devono essere impiegati sia per test di assemblaggio sia per test invasivi quali i crash test.

Trattandosi di prototipi le case madri non hanno la convenienza a realizzare da se questi componenti per tanto delegano la realizzazione ad aziende esterne.

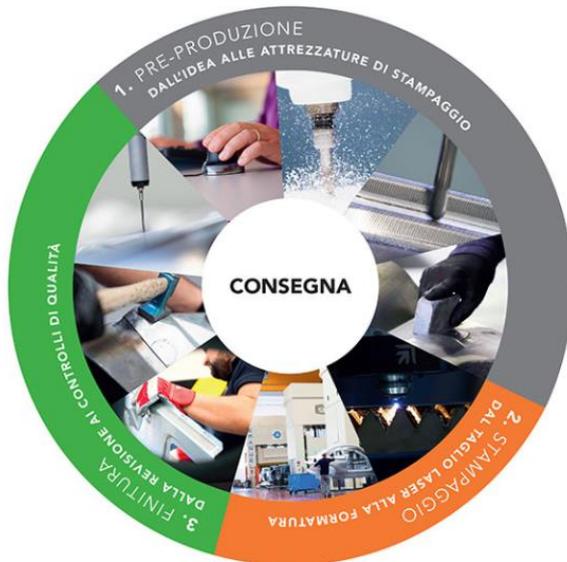


Figura 11: ciclo del prodotto di Eurodies

L'azienda si occupa del prodotto dalle fasi di progettazione, realizza gli stampi necessario alle lavorazioni, effettua le lavorazioni necessarie e invia il prodotto finito.

Le dimensioni aziendali sono in crescita e il numero di dipendenti è pressoché triplicato negli ultimi 20 anni passando dai 40 di inizio millennio agli oltre 160 di oggi.

La crescita dell'azienda è associata, come spesso accade, ad un bisogno di ristrutturazione interna da qui l'integrazione della soluzione proposta da Octavic PTS.

L'implementazione presso Eurodies, vista la sua alta complessità produttiva, di un sistema MES permette di valutare sicuramente dell'efficienza del sistema e l'adattabilità. Adattabilità basata sui feedback ricevuti dalla produzione di Eurodies.

4.1.2 Octavic PTS

Octavic PTS è una startup recente, fondata in Danimarca e Romania, la cui missione è quella di offrire la migliore piattaforma decisionale sul mercato.

La sua storia inizia con l'implementazione di una soluzione Andon, presso l'azienda Plexus, per l'assemblaggio di alto livello per equipaggiamento medico. Di qui ha continuato a

lavorare con il campo farmaceutico con la realizzazione in uno degli stabilimenti Sanofi, di un sistema per digitalizzare tutte le carte necessarie alla produzione di farmaci, in conformità agli standard GMP, i più alti standard del settore farmaceutico.

L'azienda conta nove impiegati e conta tra i suoi clienti azienda come: Zentiva, Gedeon-Richter, Alvogen, Sanofi, Draexlmaier.



Figura 12: logo Octavic PTS

4.1.2.1 La soluzione MES di Octavic

Octavic PET è un sistema di Overall Equipment Effectiveness, che vuole valutare l'efficacia globale di un impianto.

Il framework sviluppato da Octavic è composto da una piattaforma web utilizzata per l'archiviazione e l'analisi dei dati, con l'intento di sfruttarli per fornire previsioni future, dare riscontri in tempo reale sugli eventi grazie a KPI e scale di valori flessibili.

Oltre alla piattaforma web Octavic ha costruito due versioni di dispositivi hardware con caratteristiche industriali, dotati di touch screen, che sono posizionati accanto alle apparecchiature di produzione e che sono in grado di raccogliere dati in tempo reale e di collegarli con la conoscenza dell'operatore attraverso flussi visivi guidati. Quindi l'uomo è in grado di contestualizzare, con i suoi input al sistema, tutti quei dati che vengono forniti dalle macchine, quali arresti, scarti, prodotti, parametri di processo etc.

Un aspetto chiave del framework è la sua gestione della linea produttiva dalla quale raccoglie informazioni rilevanti per ogni livello organizzativo.

Il sistema pianifica le turnazioni, gestisce le sostituzioni e contestualizza in modo autonomo un gran numero di dati che arrivano dalle macchine.

Oltre a tutto ciò è stato ideato e prototipato il sistema per la pianificazione della produzione, oggetto di esame in questa trattazione, che ad oggi è in fase di verifica per essere applicato a scenari reali ed essere implementato nella produzione.

Il framework creato da Octavic per la raccolta e contestualizzazione dei dati fornisce una migliore comprensione della radice delle problematiche riscontrate, analizzando le azioni che sono state intraprese e producendo un feedback in tempo reale.

I dati raccolti vengono processati e quelli rilevanti vengono presentati ai vari livelli decisionali aziendali.

Lo sviluppo del prototipo per la pianificazione in tempo reale è stato sviluppato da Octavic in collaborazione con Nordic IoT, di cui fa parte l'università di Lund. Una delle funzioni di questo prototipo è quella di fornire la stima dei tempi di esecuzione degli ordini futuri sulla base di dati attuali e di quelli storici.

Uno degli step per l'implementazione del software è quello di valutarne l'efficienza per poter essere adattato al meglio alle esigenze aziendali.

4.1.3 Università di Lund

Uno degli aspetti più rilevanti del progetto deriva dalla complessità del contesto produttivo in cui il sistema deve essere implementato. Il ruolo riservato all'Università di Lund e, più precisamente, al dipartimento di Informatica della Facoltà di Ingegneria, è quello sviluppare le tecniche per la gestione dei vincoli e la loro applicazione.

4.1.3.1 JaCoP

Creato nel 2001 il risolutore di vincoli “Java Constraint Programming” è un programma scritto interamente in Java e permette di lavorare una vastissima quantità di vincoli dei quali alcuni non disponibili in nessun altro risolutore. L’implementazione di JaCoP si basa sullo stato dell’arte degli algoritmi di programmazione vincolata supportato da diversi anni di sperimentazione con svariati progetti. (Kuchcinski e Radoslaw 2013)

JaCoP è implementato con oltre novanta mila righe di codice esclusi esempi e commenti. Tra le sue funzionalità bisogna annoverare i metodi di ricerca che permettono di sfruttare metodi euristici di ricerca oltre che metodi che sfruttano sequenzialità e metodi gerarchici.

4.2 PROCESSO PRODUTTIVO E CICLI DI LAVORAZIONE

4.2.1 Il ciclo produttivo

La produzione prototipale di Eurodies gestisce l’ordine nel suo intero a partire dalle matematiche che vengono fornite dal cliente fino alla realizzazione del prodotto finito.

L’estrema personalizzazione che caratterizza i prodotti ha conseguenze sul ciclo produttivo che può essere comunque riassunto in una sequenza di fasi comune su tutte le commesse (Orlando, Bruno e Lombardi 2020):

- Ricezione di disegni CAD: inviati dal cliente riportano le matematiche necessarie per la commessa.
- Definizione del ciclo produttivo e disegno degli stampi: le operazioni che Eurodies esegue sotto pressa sono quelle di imbutitura, flangiatura e tranciatura.
- Simulazione del processo di stampaggio: tramite software Autoform viene simulato il processo in caso di esito negativo deve essere rivisto il modello CAD.

- Progettazione percorsi utensili: il reparto CAM è responsabile della progettazione dei percorsi utensili necessari alle lavorazioni di fresatura
- Produzione degli stampi: è la fase di maggior complessità e durata. Inizia con l'arrivo degli stampi grezzi dalla fonderia. Le lavorazioni di fresatura a cui sono sottoposti gli stampi possono essere riassunte in tre macro-fasi:
 - Spianatura;
 - Sgrossatura;
 - Finitura.
- Costruzione della struttura per il taglio laser 3D: i componenti in lamiera dopo essere stati formati sotto pressa vengono staffati e preparati per procedere al taglio.
- Messa a punto: i primi componenti, visti i tempi stringenti della produzione, possono essere soggetti a problematiche. È fondamentale che le difettosità vengano identificate nel minor tempo possibile per poter ridurre gli sprechi.
- Taglio laser: verificata l'idoneità dei semilavorati la lamiera in eccesso viene asportata mediante taglio laser. Dopo il taglio laser il componente viene lavorato manualmente dai battilastra che effettuano tutte le lavorazioni non convenienti da eseguire sotto pressa.
- Collaudo: prima della spedizione è necessario effettuare un controllo a campione per verificarne il rispetto delle tolleranze di forma e dimensione.
- Spedizione

4.2.2 Reparto aziendale di interesse

La fase di processo che interessa lo studio è quella iniziale che parte dall'arrivo delle matematiche del cliente e si conclude con la produzione dello stampo per le operazioni di stampaggio.

In particolare, verrà analizzata la schedulazione riguardante le lavorazioni operate nel reparto di fresatura.

Il codice alfanumerico che identifica i macchinari riporta in prima posizione la lettera F che indica una fresatrice e due numeri il primo dei quali indica la fila in cui è posizionata la macchina e il secondo la sua posizione all'interno della fila, quindi la macchina F 2_3 indicherà la fresa che occupa la terza posizione della seconda fila.

All'atto pratico le sette macchine considerate non hanno le stesse caratteristiche ma per semplificare lo sviluppo dell'algoritmo di schedulazione in seguito presentato, sono state divise in sole due macrocategorie: il primo gruppo, che contiene quattro macchine, è destinato alle operazioni di sgrossatura e taglio, il secondo composto da tre macchinari è destinato alle operazioni di finitura.

4.3 SVILUPPO DELL'ALGORITMO

Per analizzare la robustezza della soluzione sviluppata dall'Università di Lund è stato creato uno script utilizzando il software Matlab.

Per lo sviluppo dell'algoritmo sono state valutate diverse soluzioni metaeuristiche, tuttavia dalla letteratura è stato riscontrato che queste soluzioni non sono state raramente applicate a problemi di elevata complessità come questo, e lo sviluppo di un programma di tale complessità richiede conoscenze approfondite per lo sviluppo e tempistiche molto elevate.

Per questi motivi è stato sviluppato un programma con ottimizzazione minore e tempi di ricerca della soluzione di ottimo decisamente elevati.

4.3.1 Assunzioni

Le seguenti assunzioni sono il punto di partenza per lo sviluppo dell'algoritmo:

- Ciclo di lavorazione unico per ogni prodotto;

- Sono trascurati i tempi di trasferimento dei componenti tra le diverse operazione;
- Sono trascurati i tempi di set-up delle macchine;
- Le macchine appartenenti allo stesso sottogruppo sono state considerate equivalenti tra loro;
- Sono stati esclusi fermi macchina e i guasti, le uniche attività di manutenzione possibili sono quelle programmate.

4.3.2 Dati di Input

Lo sviluppo di un programma di schedulazione richiede, solitamente, di dover adattare i dati di input alle caratteristiche del programma. Nel nostro caso, volendo confrontare i risultati con quelli prodotti da un altro software, è stato necessario modellare il programma sulla base dei dati di input che sono:

- 73 macro-operazioni suddivise su 39 attività
- Codice univoco identificante il prodotto
- Data di arrivo del grezzo da fonderia
- Deadline di consegna
- Operazioni a cui deve essere sottoposto il prodotto
- Durata delle operazioni espressa in giorni

codice	arrivo	consegna	op1	durata1	op2	durata2	op3	durata3
4817 C 3015	10/11/2020	03/12/2020	FRE_45	5,176			FRE_60	0,067
4817 C 3020	10/11/2020	04/12/2020	FRE_40	2,904	FRE_50	2,3076		
4822 C 3015	10/11/2020	04/12/2020	FRE_45	5,176			FRE_60	0,067
4828 D 1520	10/11/2020	18/12/2020	FRE_40	5,363	FRE_50	4,2521		
4828 S 1520	10/11/2020	16/12/2020	FRE_40	5,363	FRE_50	4,2521		

Tabella 1: Parte dei dati inseriti come input per la schedulazione.

I dati di input in questo forma non sono compatibili tra loro per poter eseguire l’algoritmo, è stato necessario quindi inizializzare i dati per renderli più comodi per il programma.

In primis sono stati azzerati i dati mancanti, assegnando un codice “X” alle operazioni e il valore 0 alle durate.

```
TF1 = isnan(jobs.durata1);
TF2 = isnan(jobs.durata2);
TF3 = isnan(jobs.durata3);

jobs.durata1(TF1) = 0;
jobs.durata2(TF2) = 0;
jobs.durata3(TF3) = 0;

TF1 = isundefined(jobs.op1);
TF2 = isundefined(jobs.op2);
TF3 = isundefined(jobs.op3);

jobs.op1(TF1) = "X";
jobs.op2(TF2) = "X";
jobs.op3(TF3) = "X";
```

Viene identificato come tempo zero in minimo delle date di arrivo dei grezzi da materiale, le altre date, sia di arrivo sia di consegna vengono riferite a quella. Le durate vengono trasformate in ore.

```
rif = min(jobs.arrivo);
jobs.arrivo = hours(jobs.arrivo - rif);
jobs.consegna = hours(jobs.consegna - rif);
jobs.durata1 = hours(days(jobs.durata1));
jobs.durata2 = hours(days(jobs.durata2));
jobs.durata3 = hours(days(jobs.durata3));
```

4.3.3 Prima operazione: sgrossatura

La prima fase del programma assegna in modo randomico le attività alla macchina che deve effettuare la prima operazione.

```
jobs.macchina = randi(4,Njob,1);
```

Viene quindi creato un elenco di lavorazioni per ognuna delle quattro macchine che effettuano le operazioni di sgrossatura.

```
idxM1 = find(jobs.macchina == 1);
idxM2 = find(jobs.macchina == 2);
idxM3 = find(jobs.macchina == 3);
idxM4 = find(jobs.macchina == 4);
```

```
daProcessareM1 = jobs (idxM1, :);  
daProcessareM2 = jobs (idxM2, :);  
daProcessareM3 = jobs (idxM3, :);  
daProcessareM4 = jobs (idxM4, :);
```

Per ordinare le operazioni su ogni macchine vengono applicate le *Priority Dispatching Rules* in particolare, le operazioni vengono sequenziate in primis seguendo la logica FIFO, in secondo luogo la logica EDD ed infine in caso di pareggio tra più operazioni queste vengono sequenziate seguendo la regola STP.

A questo punto l'algoritmo prevede processare le operazioni su ogni macchina nel seguente modo:

- La data di inizio della prima lavorazione è pari alla data di arrivo del pezzo da fonderia.
- Viene registrata la data di fine dell'operazione data dalla somma tra la data di arrivo e la durata della prima operazione.
- Dalla seconda operazione in poi viene fatto un controllo sulla data di arrivo del pezzo dalla fonderia e il termine dell'operazione precedente:
 - o Se la data di arrivo del grezzo è precedente rispetto la fine dell'operazione precedente allora la data di inizio dell'operazione è pari a quella di fine del pezzo precedente;
 - o Se la data di arrivo del grezzo è successiva alla fine dell'operazione precedente allora la data di inizio è pari a quella di arrivo da fonderia.

Per effettuare ciò è stata creata una funzione che sfrutta un contatore che simula lo scorrere del tempo denominato tempo passato.

```
function jobs = mymachine(jobs, daProcessareM, idxM, stepTime)  
  
m = daProcessareM(1, :);  
t = hours(daProcessareM.arrivo(1));
```

```

tempoPassato = hours(0);
idxJob = 1;

daProcessareM(1,:) = [];

while ~isempty(m)

    if tempoPassato >= t + hours(m.duratal) && tempoPassato >=
hours(m.arrivo)

        if idxJob ~= 1
            if m.arrivo <= jobs.fineOp1(idxM(idxJob -1 ))
                fineop = m.duratal + jobs.fineOp1(idxM(idxJob -1));
            else

                fineop = m.arrivo + m.duratal;
            end
        else
            fineop = m.arrivo + m.duratal;
        end

        t = hours(fineop);
        jobs.fineOp1(idxM(idxJob)) = fineop;

        if ~isempty(daProcessareM)
            m = daProcessareM(1,:);
            daProcessareM(1,:) = [];
        else
            m(1,:) = [];
        end

        idxJob = idxJob + 1;

    end

    tempoPassato = tempoPassato + hours(stepTime);
end

end

```

l'output di questa fase è la definizione del tempo di fine della prima operazione, necessario per la seconda parte dell'algoritmo.

Viene quindi creata all'interno della tabella che conterrà tutti i dati rilevanti per ogni attività la colonna che contiene il tempo di fine della prima operazione sempre riferito al tempo zero calcolato in precedenza.

4.3.4 Seconda operazione: finitura

Per processare la seconda lavorazione vengono riordinati i dati di input in base alle date di fine della prima operazione, seguendo sempre la logica FIFO.

In questo caso l'assegnazione delle macchine avviene in base alla data di fine della prima operazione. La tabella viene quindi riorganizzata con il comando:

```
jobs = sortrows (jobs,11);
```

Per processare questa operazione è stato applicato un ragionamento a stati che controlla se una macchina sia disponibile o meno.

```
while ~stop

    if idxJob <= Njob
        if freeM5 && ~isempty(incorso)
            m5 = incorso(idxJob,:);
            freeM5 = false;
            tstartM5 = max([hours(m5.fineOp1) tfineM5]);
            tdeltaM5 = hours(m5.durata2);
            indM5 = idxJob;
            jobs.MacchOp2(idxJob) = 5;

            idxJob = idxJob+1;
        end

        if freeM6 && ~isempty(incorso)
            m6 = incorso(idxJob,:);
            freeM6 = false;
            tstartM6 = max([hours(m6.fineOp1) tfineM6]);
            tdeltaM6 = hours(m6.durata2);
            indM6 = idxJob;
            jobs.MacchOp2(idxJob) = 6;

            idxJob = idxJob+1;
        end

        if freeM7 && ~isempty(incorso)
            m7 = incorso(idxJob,:);
            freeM7 = false;
            tstartM7 = max([hours(m7.fineOp1) tfineM7]);
            tdeltaM7 = hours(m7.durata2);
            indM7 = idxJob;
            jobs.MacchOp2(idxJob) = 7;

            idxJob = idxJob+1;
        end
    else
```

```

        if freeM5 && freeM6 && freeM7
            stop = true;
        end
    end

    % Macchina 5
    if tempoPassato - tstartM5 >= tdeltaM5
        tfineM5 = tstartM5 + tdeltaM5;
        jobs.fineOp2(indM5) = tfineM5;
        freeM5 = true;
    end

    % Macchina 6
    if tempoPassato - tstartM6 >= tdeltaM6
        tfineM6 = tstartM6 + tdeltaM6;
        jobs.fineOp2(indM6) = tfineM6;
        freeM6 = true;
    end

    % Macchina 7
    if tempoPassato - tstartM7 >= tdeltaM7
        tfineM7 = tstartM7 + tdeltaM7;
        jobs.fineOp2(indM7) = tfineM7;
        freeM7 = true;
    end

    tempoPassato = tempoPassato + hours(stepTime);
end

```

L'output di questa parte del programma è la data di fine della seconda operazione per ogni attività. Nel caso di attività che non prevedono questa operazione il tempo di fine coinciderà con quello della prima operazione.

4.3.5 Terza operazione: taglio

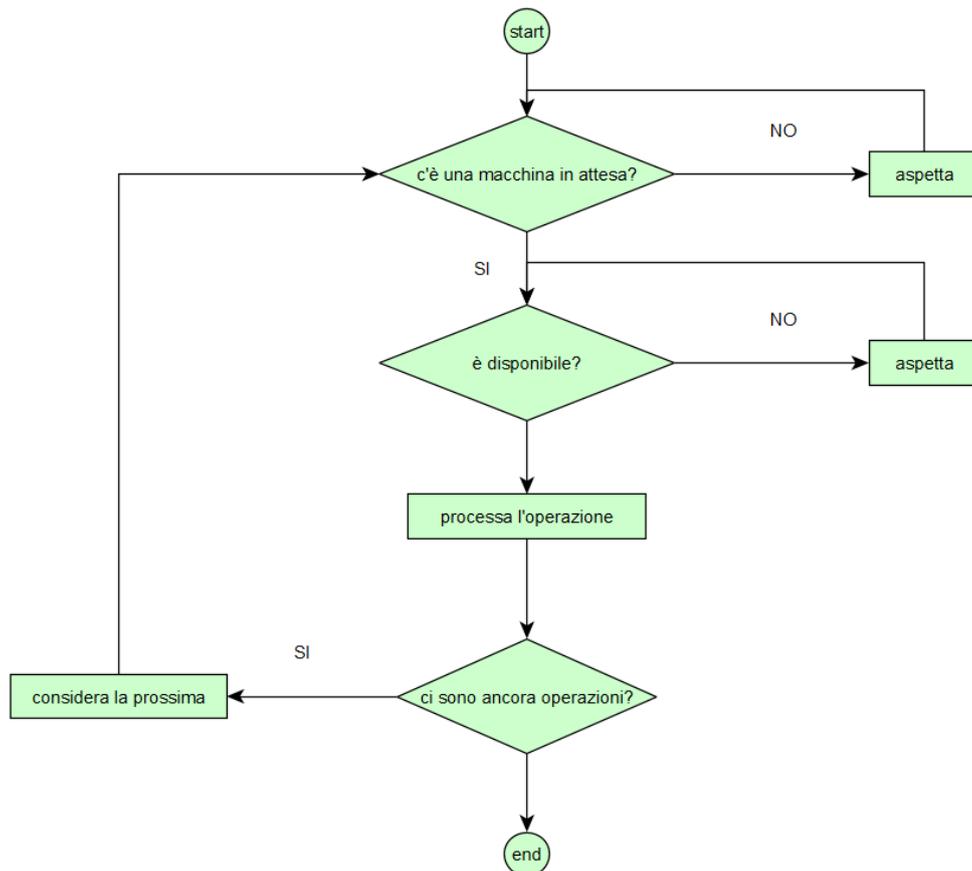
La terza operazione deve essere svolta sulle stesse macchine che hanno svolto la prima, è stato quindi necessario identificare i momenti in cui le macchine fossero in attesa tra un'operazione e l'altra, che verranno denominati idle, per poter collocare queste operazioni.

È stato nuovamente riorganizzata la tabella dei dati in base alle date di fine delle seconde operazioni.

È stato necessario quindi di nuovo un ragionamento per stato ma in questo caso gli stati da considerare sono stati tre:

- In attesa tra due operazioni precedentemente assegnate
- Disponibile a iniziare un'operazione tra quelle ancora da processare
- Indisponibile perché occupata da una operazione

Per ogni prodotto che necessita di essere sottoposto a taglio è stato quindi considerato il tempo di fine dell'operazione di finitura effettuata la seguente valutazione:



L'output fornito è quindi la data di fine della terza e ultima operazione.

4.3.6 Ripetizione del programma

Il programma, visto la randomicità delle assegnazioni delle macchine della prima operazione, necessita la ricerca della soluzione di ottimo.

È stato quindi necessario identificare una funzione o un valore da minimizzare o massimizzare.

In questo caso si è voluto minimizzare il tempo di attesa di ogni macchina.

Sono state effettuate millequattrocento simulazioni di cui è stato salvato il tempo totale di attesa delle macchine e l'elenco delle macchine assegnate alla prima operazione.

4.3.7 Output

Il risultato migliore è stato riprodotto individualmente per ottenere i risultati per il confronto finale.

Per ogni operazione è stato utile, al fine della valutazione globale, identificare, per ogni prodotto, lo slittamento tra l'inizio della prima operazione e la data di arrivo da fonderia, il margine tra la conclusione dell'ultima operazione e la deadline di consegna e, infine, il tempo trascorso tra l'inizio della prima operazione e la fine dell'ultima.

L'output complessivo è presentato in allegato.

4.4 RISULTATI

L'output è stato confrontato con i risultati sviluppati nel lavoro precedente sviluppato da Mattia Orlando (Orlando, Bruno e Lombardi 2020).

I risultati con cui è stato eseguito il confronto sono:

- La pianificazione delle attività effettuata all'interno dell'azienda dal direttore della produzione seguendo quegli iter consolidati all'interno dell'azienda;
- L'output fornito dal sistema MES di Octavic, implementato con l'algoritmo sviluppato dall'Università di Lund.

I diagrammi di Gantt prodotti nei tre casi sono riportati in allegato.

codice	CONFRONTO TEMPI DI PRODUZIONE CON EURODIES	CONFRONTO TEMPI D'ATTESA CON EURODIES	CONFRONTO TRA MARGINI DI CONSEGNA CON EURODIES	CONFRONTO TEMPI DI PRODUZIONE CON OCTAVIC	CONFRONTO TEMPI D'ATTESA CON OCTAVIC	CONFRONTO TRA MARGINI DI CONSEGNA CON OCTAVIC
4323 A 5010	-8	7	0	-3	2	0
4735 C 5010	-1	2	-1	-3	4	-1
4812 C 1510	-8	17	-9	10	-9	-1
4812 C 1520	-8	12	-3	5	-4	0
4812 C 1550	-14	15	0	0	1	0
4817 C 3015	-9	10	-1	0	0	0
4817 C 3020	-15	16	-1	-14	14	0
4821 C 3010	-27	27	0	-1	0	1
4821 C 3020	-20	24	-4	3	-2	-1
4821 C 3050	-21	24	-4	1	-2	0
4822 C 3015	-6	7	-1	3	-3	0
4828 D 1520	-15	19	-4	0	-1	1
4828 D 3017	5	-4	-1	7	-7	0
4828 S 1520	-6	7	-1	4	-4	0
4828 S 3017	-6	8	-2	0	1	-1
4829 A 3015	-3	6	-3	8	-8	0
4829 A 3020	-11	12	-1	-11	11	0
4830 A 3015	-4	7	-3	8	-7	-1
4830 A 3020	0	4	-4	-1	1	0
4831 A 3015	-6	7	-1	-2	3	-1
4831 A 5050	-12	13	-1	0	0	0
4835 S 3015	-16	21	-5	1	-1	0
4848 A 3015	-7	8	-1	0	-1	1
4848 A 3020	2	0	-2	-1	2	-1
4848 A 5010	-11	11	0	0	0	0
4848 A 5020	-5	4	0	-3	2	0
4848 A 5050	-5	5	0	1	-1	0
4851 D 3017	-3	11	-8	-3	3	0
4851 D 5010	-2	3	-2	-6	7	-2
4851 D 5021	0	0	0	-2	2	0
4851 D 5091	-12	13	-1	-14	15	-1
4851 S 3017	-7	4	3	0	-5	5
4851 S 3020	0	1	-1	0	6	-6
4851 S 5010	-12	13	-1	-15	15	0
4851 S 5021	-11	8	3	0	-3	3
4851 S 5091	-5	5	0	5	-5	0
4859 C 3010	-9	9	0	-8	8	0
4859 C 3020	-4	3	0	4	-5	0
4866 A 5010	-13	11	2	4	-6	2

Tabella 2: confronto tra i risultati dell' algoritmo sviluppato in contrapposizione con quelli prodotti dalla pianificazione di Eurodies e quelli ottenuti dal sistema MES di Octavic

Il confronto con la pianificazione di Eurodies evidenzia un miglioramento complessivo delle tempistiche in ogni aspetto, in particolare riguardo la differenza di tempo trascorso tra

l'arrivo del grezzo di fonderia e l'inizio della prima operazione nel quale si osserva un miglioramento nel 92% delle attività.

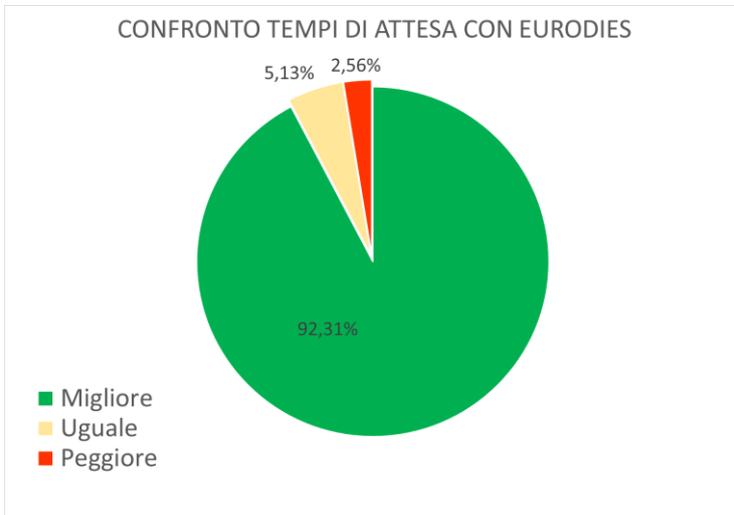


Figura 13: Confronto sulle tempistiche di avvio delle prime operazioni

Significativi sono anche le riduzioni dei tempi intercorsi tra le varie operazioni delle attività, dato nel quale si evidenzia un singolo elemento peggiore.

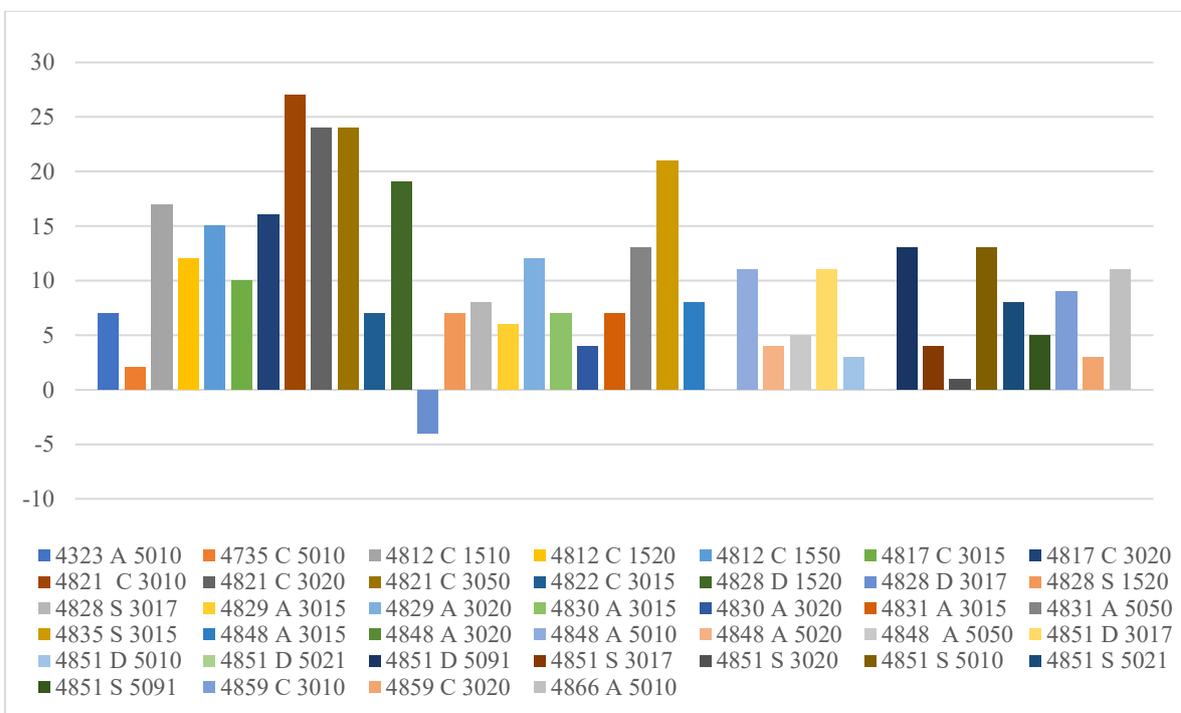


Figura 14: andamento delle durate complessive di lavorazione.

Positivo anche il miglioramento rispetto gli anticipi rispetto ai margini di consegna nel quale si evidenzia un miglioramento superiore al 65%.

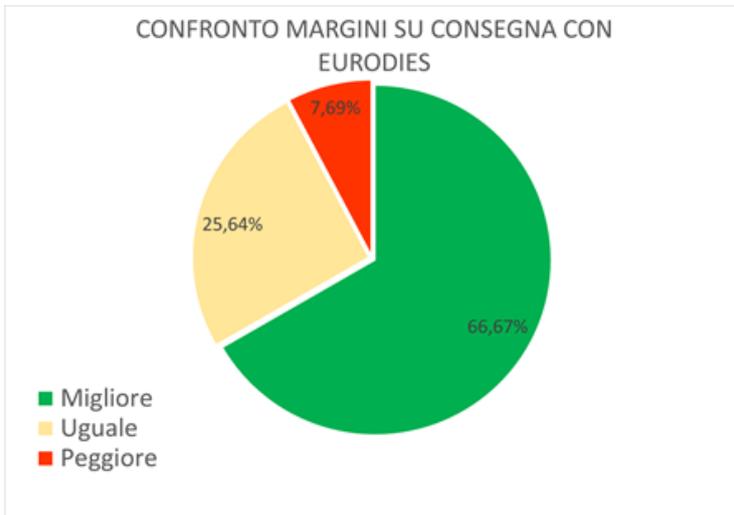


Figura 15: confronto sulle deadline rispetto alla pianificazione di Eurodies.

Per quanto riguarda il confronto con il sistema implementato da Octavic, possiamo notare che non vi è un netto miglioramento, ma piuttosto una condizione di equilibrio, nella quale il beneficio complessivo si può riscontrare in una riduzione complessiva del C_{MAX} di soli tre giorni.

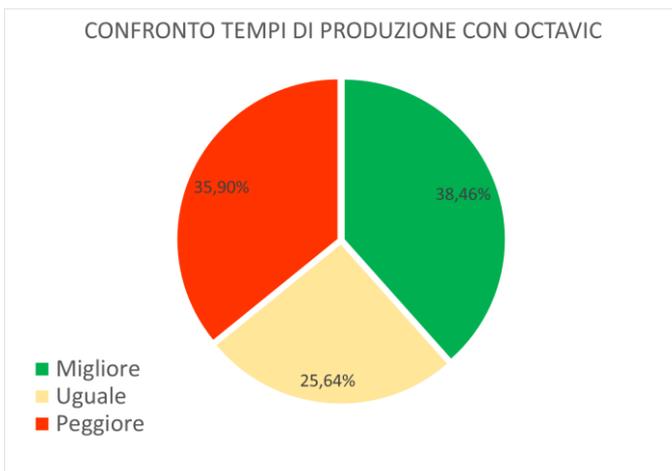


Figura 16: confronto tra le durate complessive ottenute su Matlab con quelle elaborate dal sistema MES

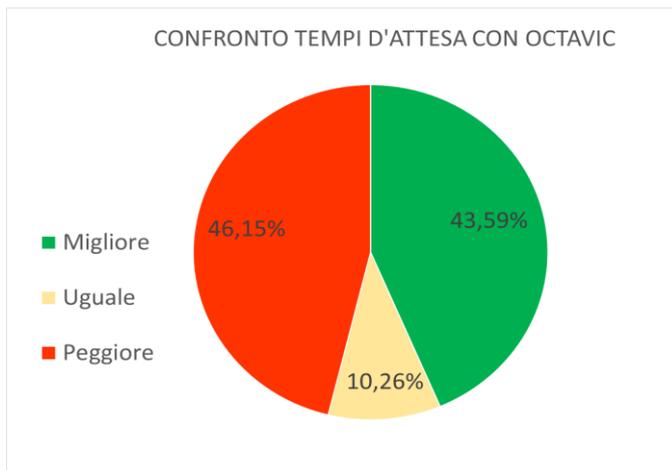


Figura 17: confronto tra i ritardi subiti nell'avvio di produzione tra l'algorithm sviluppato e il sistema MES

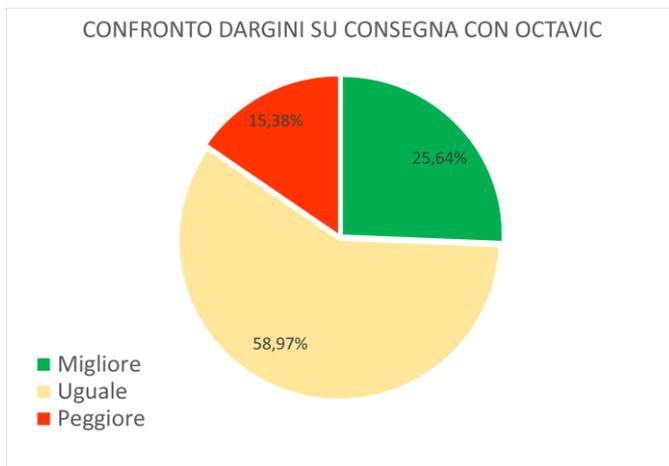


Figura 18: confronto tra i margini sulle deadline tra script Matlab e sistema MES

4.5 OSSERVAZIONI

La schedulazione ottenuta mediante l'algorithm sviluppato per questa tesi è ben lontano dal poter essere applicato nella realtà nel reparto di fresatura di Eurodies.

Questo perché nella realtà produttiva dell'azienda non tutte le macchine possono effettuare lavorazioni su ogni componente in quanto impossibilitate per questioni di dimensioni. Inoltre, effettuare lavorazioni su prodotti di piccole dimensioni occupando macchine che possono sopportare un carico maggiore riduce l'efficienza dell'azienda. Il software

sviluppato per il sistema MES, rispettando queste esigenze, pone un vincolo decisamente più stringente alla schedulazione.

La potenzialità del sistema MES implementato all'interno di Eurodies è quello di poter assegnare delle preferenze ad ogni prodotto rispetto a quali macchine verranno assegnate.

Il confronto effettuato permette di verificare l'efficacia dell'algorithmo di schedulazione, in quanto, nonostante i vincoli più stringenti, è in grado di ottenere un risultato più che ottimo, che in molti casi si rivela addirittura migliore rispetto al miglior risultato ottenuto escludendo i vincoli prima citati.

Riassumendo il sistema implementato da Octavic in collaborazione con l'Università di Lund permette di:

- Impostare un elevato numero di vincoli all'interno della schedulazione
- Selezionare le preferenze per l'assegnazione delle attività sulle varie macchine
- Ottenere un risultato di ottimo globale equiparabile al miglior risultato ottenuto in assenza di vincoli.

CAPITOLO 6

CONCLUSIONI

Ottimizzazione dei tempi, flessibilità di schedulazione, adattabilità agli imprevisti, riduzione degli sprechi, identificazione delle criticità sono solo alcuni dei benefici che l'integrazione di un sistema MES garantisce per l'organizzazione aziendale.

Il lavoro svolto evidenzia come la soluzione sviluppata da Octavic PET, con il contributo dell'Università di Lund, è prossima ad ottenere il miglior risultato di schedulazione possibile nonostante i limiti imposti dalla linea di produzione.

Per valutare maggiormente le potenzialità del sistema è necessario un maggiore sviluppo dell'algoritmo con l'inclusione dei vincoli aziendali in modo da validare con maggior precisione i risultati ottenuti.

ALLEGATI

codice	arrivo	consegna	op1	durata1	op2	durata2	op3	durata3
4817 C 3015	10/11/2020	03/12/2020	FRE_45	5,176			FRE_60	0,067
4817 C 3020	10/11/2020	04/12/2020	FRE_40	2,904	FRE_50	2,3076		
4822 C 3015	10/11/2020	04/12/2020	FRE_45	5,176			FRE_60	0,067
4828 D 1520	10/11/2020	18/12/2020	FRE_40	5,363	FRE_50	4,2521		
4828 S 1520	10/11/2020	16/12/2020	FRE_40	5,363	FRE_50	4,2521		
4828 S 3017	10/11/2020	11/12/2020	FRE_40	6,137	FRE_50	4,9194	FRE_60	0,112
4829 A 3015	10/11/2020	07/12/2020	FRE_40	2,862	FRE_50	2,3257	FRE_60	0,082
4829 A 3020	10/11/2020	03/12/2020	FRE_40	2,904	FRE_50	2,3257		
4821 C 3010	17/11/2020	23/12/2020	FRE_40	0,752	FRE_50	0,6194		
4821 C 3020	17/11/2020	23/12/2020	FRE_40	1,751	FRE_50	1,4042		
4821 C 3050	17/11/2020	23/12/2020	FRE_40	0,752	FRE_50	0,6049		
4828 D 3017	17/11/2020	16/12/2020	FRE_40	6,137	FRE_50	4,9194	FRE_60	0,112
4830 A 3020	17/11/2020	18/12/2020	FRE_40	2,255	FRE_50	1,7667		
4859 C 3010	17/11/2020	04/12/2020	FRE_45	1,341				
4859 C 3020	17/11/2020	09/12/2020	FRE_45	1,356				
4830 A 3015	24/11/2020	18/12/2020	FRE_40	2,222	FRE_50	1,7826	FRE_60	0,056
4831 A 3015	24/11/2020	11/12/2020	FRE_45	6,688			FRE_60	0,097
4848 A 3015	24/11/2020	21/12/2020	FRE_40	5,363	FRE_50	4,2521	FRE_60	0,097
4735 C 5010	27/11/2020	11/12/2020	FRE_40	1,375				
4812 C 1510	01/12/2020	15/01/2021	FRE_40	4,51	FRE_45	3,6076		
4848 A 5010	01/12/2020	18/12/2020	FRE_45	3,101				
4848 A 5020	01/12/2020	24/12/2020	FRE_45	2,213				
4848 A 5050	01/12/2020	15/12/2020	FRE_45	3,117				
4851 D 3017	01/12/2020	06/01/2021	FRE_40	7,004	FRE_50	5,616	FRE_60	0,142
4851 S 3017	01/12/2020	30/12/2020	FRE_40	7,004	FRE_50	5,5632	FRE_60	0,127
4851 S 3020	01/12/2020	23/12/2020	FRE_40	6,137	FRE_50	4,9604		
4812 C 1520	09/12/2020	11/01/2021	FRE_40	5,363	FRE_50	4,2521		
4812 C 1550	09/12/2020	11/01/2021	FRE_45	6,688				
4851 D 5010	09/12/2020	01/01/2021	FRE_40	6,201	FRE_50	4,9604		
4851 D 5091	09/12/2020	25/12/2020	FRE_45	1,326				
4851 S 5010	09/12/2020	07/01/2021	FRE_40	6,137	FRE_50	4,9604		
4851 S 5091	09/12/2020	23/12/2020	FRE_45	1,341				
4866 A 5010	09/12/2020	30/12/2020	FRE_45	3,101				
4848 A 3020	14/12/2020	24/12/2020	FRE_40	0,767	FRE_50	0,6194		
4835 S 3015	15/12/2020	18/01/2021	FRE_40	3,675	FRE_50	2,9472	FRE_60	0,067
4851 D 5021	15/12/2020	30/12/2020	FRE_45	2,213				
4851 S 5021	15/12/2020	06/01/2021	FRE_45	2,238				
4831 A 5050	22/12/2020	15/01/2021	FRE_40	2,238	FRE_50	1,7826		
4323 A 5010	24/12/2020	04/01/2021	FRE_45	1,326				

Allegato 1: Input per la schedulazione

codice	arrivo	consegna	macchina1	inizio1	fine1	macchina2	inizio2	fine2	macchina3	inizio3	fine3	ritardo da arrivo	margini su deadline	tempo totale di produzione
4829 A 3020	'10-Nov-2020'	'03-Dec-2020'	3	'10-Nov-2020 00:00:00'	'12-Nov-2020 21:42:00'	6	'12-Nov-2020 21:42:00'	'15-Nov-2020 05:30:59'	2	'15-Nov-2020 05:30:59'	'15-Nov-2020 05:30:59'	0	18	5
4817 C 3015	'10-Nov-2020'	'03-Dec-2020'	2	'10-Nov-2020 00:00:00'	'15-Nov-2020 04:14:00'	7	'15-Nov-2020 04:14:00'	'15-Nov-2020 04:14:00'	2	'15-Nov-2020 04:15:00'	'15-Nov-2020 05:52:00'	0	18	5
4817 C 3020	'10-Nov-2020'	'04-Dec-2020'	1	'10-Nov-2020 00:00:00'	'12-Nov-2020 21:41:59'	5	'12-Nov-2020 21:41:59'	'15-Nov-2020 05:04:59'	2	'15-Nov-2020 05:04:59'	'15-Nov-2020 05:04:59'	0	19	5
4822 C 3015	'10-Nov-2020'	'04-Dec-2020'	3	'12-Nov-2020 21:42:00'	'18-Nov-2020 01:56:00'	6	'18-Nov-2020 01:56:00'	'18-Nov-2020 01:56:00'	4	'18-Nov-2020 01:57:00'	'18-Nov-2020 03:34:03'	3	16	5
4829 A 3015	'10-Nov-2020'	'07-Dec-2020'	3	'18-Nov-2020 01:56:00'	'20-Nov-2020 22:37:00'	7	'20-Nov-2020 22:37:00'	'23-Nov-2020 06:26:00'	2	'23-Nov-2020 06:27:00'	'23-Nov-2020 08:24:59'	8	14	5
4828 S 3017	'10-Nov-2020'	'11-Dec-2020'	1	'12-Nov-2020 21:41:59'	'19-Nov-2020 00:58:59'	6	'19-Nov-2020 00:58:59'	'23-Nov-2020 23:02:59'	2	'23-Nov-2020 23:03:00'	'24-Nov-2020 01:44:00'	3	17	11
4828 S 1520	'10-Nov-2020'	'16-Dec-2020'	1	'19-Nov-2020 00:58:59'	'24-Nov-2020 09:42:00'	7	'24-Nov-2020 09:42:00'	'28-Nov-2020 15:45:00'	3	'28-Nov-2020 15:45:00'	'28-Nov-2020 15:45:00'	9	17	10
4828 D 1520	'10-Nov-2020'	'18-Dec-2020'	4	'10-Nov-2020 00:00:00'	'15-Nov-2020 08:43:00'	7	'15-Nov-2020 08:43:00'	'19-Nov-2020 14:46:01'	2	'19-Nov-2020 14:46:01'	'19-Nov-2020 14:46:01'	0	28	10
4859 C 3010	'17-Nov-2020'	'04-Dec-2020'	2	'17-Nov-2020 00:00:00'	'18-Nov-2020 08:11:02'	6	'18-Nov-2020 08:11:02'	'18-Nov-2020 08:11:02'	4	'18-Nov-2020 08:11:02'	'18-Nov-2020 08:11:02'	0	16	1
4859 C 3020	'17-Nov-2020'	'09-Dec-2020'	1	'24-Nov-2020 09:42:00'	'25-Nov-2020 18:14:55'	6	'25-Nov-2020 18:14:55'	'25-Nov-2020 18:14:55'	3	'25-Nov-2020 18:14:55'	'25-Nov-2020 18:14:55'	7	13	1
4828 D 3017	'17-Nov-2020'	'16-Dec-2020'	1	'25-Nov-2020 18:14:55'	'01-Dec-2020 21:31:55'	7	'01-Dec-2020 21:31:55'	'06-Dec-2020 19:35:55'	1	'06-Dec-2020 19:36:00'	'06-Dec-2020 22:17:00'	9	9	11
4830 A 3020	'17-Nov-2020'	'18-Dec-2020'	1	'01-Dec-2020 21:31:55'	'04-Dec-2020 03:38:59'	5	'04-Dec-2020 03:38:59'	'05-Dec-2020 22:03:01'	1	'05-Dec-2020 22:03:01'	'05-Dec-2020 22:03:01'	15	12	4
4821 C 3010	'17-Nov-2020'	'23-Dec-2020'	4	'17-Nov-2020 00:00:00'	'17-Nov-2020 18:02:59'	5	'17-Nov-2020 18:02:59'	'18-Nov-2020 08:55:00'	4	'18-Nov-2020 08:55:00'	'18-Nov-2020 08:55:00'	0	35	1
4821 C 3050	'17-Nov-2020'	'23-Dec-2020'	2	'18-Nov-2020 08:11:02'	'19-Nov-2020 02:14:03'	5	'19-Nov-2020 02:14:03'	'19-Nov-2020 16:45:07'	2	'19-Nov-2020 16:45:07'	'19-Nov-2020 16:45:07'	1	33	1
4821 C 3020	'17-Nov-2020'	'23-Dec-2020'	3	'20-Nov-2020 22:37:00'	'22-Nov-2020 16:39:00'	5	'22-Nov-2020 16:39:00'	'24-Nov-2020 02:21:00'	3	'24-Nov-2020 02:21:00'	'24-Nov-2020 02:21:00'	4	29	3
4831 A 3015	'24-Nov-2020'	'11-Dec-2020'	2	'24-Nov-2020 00:00:00'	'30-Nov-2020 16:31:00'	6	'30-Nov-2020 16:31:00'	'30-Nov-2020 16:31:00'	3	'30-Nov-2020 16:33:00'	'30-Nov-2020 18:52:58'	0	10	7
4830 A 3015	'24-Nov-2020'	'18-Dec-2020'	1	'04-Dec-2020 03:38:59'	'06-Dec-2020 08:57:56'	6	'06-Dec-2020 08:57:56'	'08-Dec-2020 03:44:53'	1	'08-Dec-2020 03:45:00'	'08-Dec-2020 05:05:03'	10	10	4
4848 A 3015	'24-Nov-2020'	'21-Dec-2020'	4	'24-Nov-2020 00:00:00'	'29-Nov-2020 08:43:00'	5	'29-Nov-2020 08:43:00'	'03-Dec-2020 14:46:01'	1	'06-Dec-2020 09:06:00'	'06-Dec-2020 11:25:58'	0	15	12
4735 C 5010	'27-Nov-2020'	'11-Dec-2020'	2	'30-Nov-2020 16:31:00'	'02-Dec-2020 01:31:00'	6	'02-Dec-2020 01:31:00'	'02-Dec-2020 01:31:00'	1	'02-Dec-2020 01:31:00'	'02-Dec-2020 01:31:00'	4	9	1
4848 A 5050	'01-Dec-2020'	'15-Dec-2020'	2	'02-Dec-2020 01:31:00'	'05-Dec-2020 04:19:03'	6	'05-Dec-2020 04:19:03'	'05-Dec-2020 04:19:03'	1	'05-Dec-2020 04:19:03'	'05-Dec-2020 04:19:03'	1	10	3
4848 A 5010	'01-Dec-2020'	'18-Dec-2020'	3	'01-Dec-2020 00:00:00'	'04-Dec-2020 02:25:00'	6	'04-Dec-2020 02:25:00'	'04-Dec-2020 02:25:00'	1	'04-Dec-2020 02:25:00'	'04-Dec-2020 02:25:00'	0	14	3
4851 S 3020	'01-Dec-2020'	'23-Dec-2020'	2	'05-Dec-2020 04:19:03'	'11-Dec-2020 07:36:02'	7	'11-Dec-2020 07:36:02'	'16-Dec-2020 06:39:01'	3	'16-Dec-2020 06:39:01'	'16-Dec-2020 06:39:01'	4	7	11
4848 A 5020	'01-Dec-2020'	'24-Dec-2020'	2	'11-Dec-2020 07:36:02'	'13-Dec-2020 12:43:03'	5	'13-Dec-2020 14:52:04'	'13-Dec-2020 14:52:04'	2	'13-Dec-2020 14:52:04'	'13-Dec-2020 14:52:04'	10	10	2
4851 S 3017	'01-Dec-2020'	'30-Dec-2020'	3	'04-Dec-2020 02:25:00'	'11-Dec-2020 02:30:02'	6	'11-Dec-2020 02:30:02'	'16-Dec-2020 16:01:03'	3	'21-Dec-2020 18:06:00'	'21-Dec-2020 21:09:01'	3	8	18
4851 D 3017	'01-Dec-2020'	'06-Jan-2021'	4	'01-Dec-2020 00:00:00'	'08-Dec-2020 00:05:02'	5	'08-Dec-2020 00:05:02'	'13-Dec-2020 14:52:04'	2	'21-Dec-2020 15:45:00'	'21-Dec-2020 19:09:02'	0	15	21
4812 C 1510	'01-Dec-2020'	'15-Jan-2021'	3	'11-Dec-2020 02:30:02'	'15-Dec-2020 14:44:02'	7	'16-Dec-2020 06:39:01'	'19-Dec-2020 21:14:01'	2	'19-Dec-2020 21:14:01'	'19-Dec-2020 21:14:01'	10	26	9
4851 S 5091	'09-Dec-2020'	'23-Dec-2020'	2	'13-Dec-2020 12:43:03'	'14-Dec-2020 20:54:05'	5	'14-Dec-2020 20:54:05'	'14-Dec-2020 20:54:05'	3	'14-Dec-2020 20:54:05'	'14-Dec-2020 20:54:05'	5	8	1
4851 D 5091	'09-Dec-2020'	'25-Dec-2020'	1	'09-Dec-2020 00:00:00'	'10-Dec-2020 07:50:00'	7	'10-Dec-2020 07:50:00'	'10-Dec-2020 07:50:00'	2	'10-Dec-2020 07:50:00'	'10-Dec-2020 07:50:00'	0	15	1
4866 A 5010	'09-Dec-2020'	'30-Dec-2020'	2	'14-Dec-2020 20:54:05'	'17-Dec-2020 23:19:06'	7	'19-Dec-2020 21:14:01'	'19-Dec-2020 21:14:01'	2	'19-Dec-2020 21:14:01'	'19-Dec-2020 21:14:01'	6	10	5
4851 D 5010	'09-Dec-2020'	'01-Jan-2021'	1	'10-Dec-2020 07:50:00'	'16-Dec-2020 12:39:01'	6	'16-Dec-2020 16:01:03'	'21-Dec-2020 15:04:01'	2	'21-Dec-2020 15:04:01'	'21-Dec-2020 15:04:01'	1	10	11
4851 S 5010	'09-Dec-2020'	'07-Jan-2021'	4	'09-Dec-2020 00:00:00'	'15-Dec-2020 03:16:59'	5	'15-Dec-2020 03:16:59'	'20-Dec-2020 02:19:58'	2	'20-Dec-2020 02:19:58'	'20-Dec-2020 02:19:58'	0	18	11
4812 C 1520	'09-Dec-2020'	'11-Jan-2021'	3	'15-Dec-2020 14:44:02'	'20-Dec-2020 23:27:02'	7	'20-Dec-2020 23:27:02'	'25-Dec-2020 05:30:02'	2	'25-Dec-2020 05:30:02'	'25-Dec-2020 05:30:02'	7	17	10
4812 C 1550	'09-Dec-2020'	'11-Jan-2021'	1	'16-Dec-2020 12:39:01'	'23-Dec-2020 05:10:01'	6	'23-Dec-2020 05:10:01'	'23-Dec-2020 05:10:01'	2	'23-Dec-2020 05:10:01'	'23-Dec-2020 05:10:01'	8	19	7
4848 A 3020	'14-Dec-2020'	'24-Dec-2020'	3	'20-Dec-2020 23:27:02'	'21-Dec-2020 17:51:05'	6	'21-Dec-2020 17:51:05'	'22-Dec-2020 08:43:01'	2	'22-Dec-2020 08:43:01'	'22-Dec-2020 08:43:01'	7	2	1
4851 D 5021	'15-Dec-2020'	'30-Dec-2020'	1	'23-Dec-2020 05:10:01'	'25-Dec-2020 10:17:01'	5	'25-Dec-2020 10:17:01'	'25-Dec-2020 10:17:01'	1	'25-Dec-2020 10:17:01'	'25-Dec-2020 10:17:01'	8	5	2
4851 S 5021	'15-Dec-2020'	'06-Jan-2021'	4	'15-Dec-2020 03:16:59'	'17-Dec-2020 09:00:00'	7	'19-Dec-2020 21:14:01'	'17-Dec-2020 21:14:01'	2	'19-Dec-2020 21:14:01'	'19-Dec-2020 21:14:01'	0	17	5
4835 S 3015	'15-Dec-2020'	'18-Jan-2021'	2	'17-Dec-2020 23:19:06'	'21-Dec-2020 15:31:06'	5	'21-Dec-2020 15:31:06'	'24-Dec-2020 14:18:00'	2	'24-Dec-2020 14:18:00'	'24-Dec-2020 15:55:03'	3	24	7

Allegato 2: output di schedulazione

codice	CONFRONTO TEMPI DI PRODUZIONE CON EURODIES	CONFRONTO TEMPI D'ATTESA CON EURODIES	CONFRONTO TRA MARGINI DI CONSEGNA CON EURODIES	CONFRONTO TEMPI DI PRODUZIONE CON OCTAVIC	CONFRONTO TEMPI D'ATTESA CON OCTAVIC	CONFRONTO TRA MARGINI DI CONSEGNA CON OCTAVIC	tempo tra arrivo e deadline	tempo d'attesa SIMULAZIONE	marginie su consegna SIMULAZIONE	tempo di produzione SIMULAZIONE	tempo d'attesa EURODIES	marginie su consegna EURODIES	tempo di produzione EURODIES	tempo d'attesa OCTAVIC	marginie su consegna OCTAVIC	tempo di produzione OCTAVIC
4323 A 5010	-8	7	0	-3	2	0	11	0	9	1	8	2	1	3	7	1
4735 C 5010	-1	2	-1	-3	4	-1	14	4	9	1	5	7	2	7	5	2
4812 C 1510	-8	17	-9	10	-9	-1	45	10	26	9	18	9	18	0	35	10
4812 C 1520	-8	12	-3	5	-4	0	33	7	17	10	15	5	13	2	21	10
4812 C 1550	-14	15	0	0	1	0	33	8	19	7	22	4	7	8	18	7
4817 C 3015	-9	10	-1	0	0	0	23	0	18	5	9	8	6	0	18	5
4817 C 3020	-15	16	-1	-14	14	0	24	0	19	5	15	3	6	14	5	5
4821 C 3010	-27	27	0	-1	0	1	36	0	35	1	27	8	1	1	35	0
4821 C 3020	-20	24	-4	3	-2	-1	36	4	29	3	24	5	7	1	31	4
4821 C 3050	-21	24	-4	1	-2	0	36	1	33	1	22	9	5	0	35	1
4822 C 3015	-6	7	-1	3	-3	0	24	3	16	5	9	9	6	0	19	5
4828 D 1520	-15	19	-4	0	-1	1	38	0	28	10	15	9	14	0	29	9
4828 D 3017	5	-4	-1	7	-7	0	29	9	9	11	4	13	12	2	16	11
4828 S 1520	-6	7	-1	4	-4	0	36	9	17	10	15	10	11	5	21	10
4828 S 3017	-6	8	-2	0	1	-1	31	3	17	11	9	9	13	3	16	12
4829 A 3015	-3	6	-3	8	-8	0	27	8	14	5	11	8	8	0	22	5
4829 A 3020	-11	12	-1	-11	11	0	23	0	18	5	11	6	6	11	7	5
4830 A 3015	-4	7	-3	8	-7	-1	24	10	10	4	14	3	7	2	17	5
4830 A 3020	0	4	-4	-1	1	0	31	15	12	4	15	8	8	16	11	4
4831 A 3015	-6	7	-1	-2	3	-1	17	0	10	7	6	3	8	2	7	8
4831 A 5050	-12	13	-1	0	0	0	24	0	20	4	12	7	5	0	20	4
4835 S 3015	-16	21	-5	1	-1	0	34	3	24	7	19	3	12	2	25	7
4848 A 3015	-7	8	-1	0	-1	1	27	0	15	12	7	7	13	0	16	11
4848 A 3020	2	0	-2	-1	2	-1	10	7	2	1	5	2	3	8	0	2
4848 A 5010	-11	11	0	0	0	0	17	0	14	3	11	3	3	0	14	3
4848 A 5020	-5	4	0	-3	2	0	23	10	10	2	15	6	2	13	8	2
4848 A 5050	-5	5	0	1	-1	0	14	1	10	3	6	5	3	0	11	3
4851 D 3017	-3	11	-8	-3	3	0	36	0	15	21	3	4	29	3	12	21
4851 D 5010	-2	3	-2	-6	7	-2	23	1	10	11	3	7	13	7	3	13
4851 D 5021	0	0	0	-2	2	0	15	8	5	2	8	5	2	10	3	2
4851 D 5091	-12	13	-1	-14	15	-1	16	0	15	1	12	2	2	14	0	2
4851 S 3017	-7	4	3	0	-5	5	29	3	8	18	10	4	15	3	13	13
4851 S 3020	0	1	-1	0	6	-6	22	4	7	11	4	6	12	4	1	17
4851 S 5010	-12	13	-1	-15	15	0	29	0	18	11	12	5	12	15	3	11
4851 S 5021	-11	8	3	0	-3	3	22	0	17	5	11	9	2	0	20	2
4851 S 5091	-5	5	0	5	-5	0	14	5	8	1	10	3	1	0	13	1
4859 C 3010	-9	9	0	-8	8	0	17	0	16	1	9	7	1	8	8	1
4859 C 3020	-4	3	0	4	-5	0	22	7	13	1	11	10	1	3	18	1
4866 A 5010	-13	11	2	4	-6	2	21	6	10	5	19	-1	3	2	16	3
percentuali																
Migliore	87,18%	92,31%	66,67%	38,46%	43,59%	25,64%										
Uguale	7,69%	5,13%	25,64%	25,64%	10,26%	58,97%										
Peggior	5,13%	2,56%	7,69%	35,90%	46,15%	15,38%										

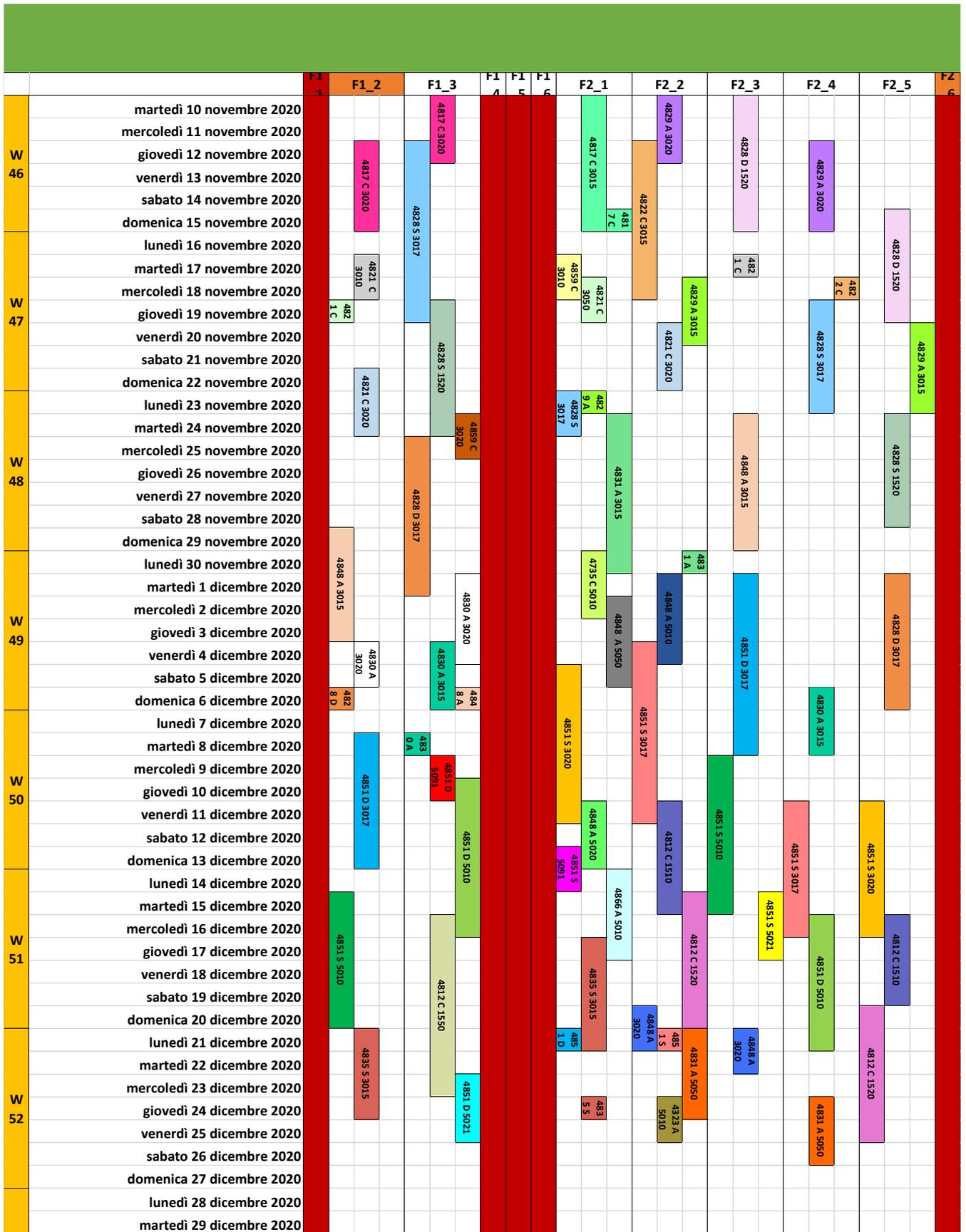
Allegato 3: confronto tra pianificazioni

PIANIFICAZIONE OCTAVIC													
		F1_1	F1_2	F1_3	F1_4	F1_5	F1_6	F2_1	F2_2	F2_3	F2_4	F2_5	F2_6
W 46	martedì 10 novembre 2020	M				M	M						M
	mercoledì 11 novembre 2020	A				N	N						A
	giovedì 12 novembre 2020	N		4817 C 3015 F_PFE_40		U	U	4829 S 3015 F_PFE_40		4829 D 1020 L_PFE_40	4829 C 3015 F_PFE_40		N
	venerdì 13 novembre 2020	T				T	T					4829 A 3017 F_PFE_50	T
	sabato 14 novembre 2020	E			4817 C 3015 F_PFE_40	4823 C 3015 F_PFE_40	E	E			4829 S 3015 F_PFE_40		E
W 47	domenica 15 novembre 2020	N				N	N	4828 S 3017 L_PFE_40					N
	lunedì 16 novembre 2020	Z				Z	Z						Z
	martedì 17 novembre 2020	I				I	I				4829 D 1020 L_PFE_50		I
	mercoledì 18 novembre 2020	O	4821 C 3020 B_PFE_50			O	O		4828 S 1020 L_PFE_40	4821 C 3020 B_PFE_40	4821 C 3010 B_PFE_40		4821 C 3020 B_PFE_50
	giovedì 19 novembre 2020	N				N	N						N
W 48	venerdì 20 novembre 2020	E				E	E						E
	sabato 21 novembre 2020		4828 S 3017 L_PFE_50					4828 D 3017 L_PFE_40		4829 A 3020 F_PFE_40			4821 C 3020 D_PFE_50
	domenica 22 novembre 2020												
	lunedì 23 novembre 2020										4828 S 1020 L_PFE_50		
	martedì 24 novembre 2020												
W 49	mercoledì 25 novembre 2020							4819 C 3010 B_PFE_40	4828 S 3017 L_PFE_50	4817 C 3020 F_PFE_40			4829 A 3020 F_PFE_50
	giovedì 26 novembre 2020												
	venerdì 27 novembre 2020		4817 C 3020 F_PFE_50		4818 A 3015 L_PFE_40			4828 S 3015 L_PFE_40				4828 D 3017 L_PFE_50	
	sabato 28 novembre 2020												
	domenica 29 novembre 2020												
W 50	lunedì 30 novembre 2020		4828 A 3015 L_PFE_50					4828 D 3017 L_PFE_40		4828 S 3015 L_PFE_40			
	martedì 1 dicembre 2020												
	mercoledì 2 dicembre 2020				4818 A 3020 D_PFE_40			4818 C 1020 L_PFE_40			4818 A 5020 D_PFE_40		4818 A 3015 L_PFE_50
	giovedì 3 dicembre 2020												
	venerdì 4 dicembre 2020								4828 S 1020 L_PFE_40				
W 51	sabato 5 dicembre 2020												
	domenica 6 dicembre 2020												
	lunedì 7 dicembre 2020					4815 S 3017 M_PFE_40							
	martedì 8 dicembre 2020												
	mercoledì 9 dicembre 2020												
W 52	giovedì 10 dicembre 2020							4815 S 3011 B_PFE_45					4818 C 1020 L_PFE_50
	venerdì 11 dicembre 2020												
	sabato 12 dicembre 2020												
	domenica 13 dicembre 2020												
	lunedì 14 dicembre 2020												
W 53	martedì 15 dicembre 2020												
	mercoledì 16 dicembre 2020												
	giovedì 17 dicembre 2020							4815 S 3017 M_PFE_40					
	venerdì 18 dicembre 2020												
	sabato 19 dicembre 2020												
W 54	domenica 20 dicembre 2020												
	lunedì 21 dicembre 2020												
	martedì 22 dicembre 2020												
	mercoledì 23 dicembre 2020												
	giovedì 24 dicembre 2020												
W 55	venerdì 25 dicembre 2020												
	sabato 26 dicembre 2020												
	domenica 27 dicembre 2020												
	lunedì 28 dicembre 2020												
	martedì 29 dicembre 2020												
W 56	mercoledì 30 dicembre 2020												
	giovedì 31 dicembre 2020												
	venerdì 1 gennaio 2021												
	sabato 2 gennaio 2021												
	domenica 3 gennaio 2021												
W 57	lunedì 4 gennaio 2021												
	martedì 5 gennaio 2021												

Allegato 4: pianificazione Octavic

PIANIFICAZIONE EURODIES												
	F1_1	F1_2	F1_3	F1_4	F1_5	F1_6	F2_1	F2_2	F2_3	F2_4	F2_5	F2_6
W 46	martedì 10 novembre 2020	M			M	M						M
	mercoledì 11 novembre 2020	A			A	A						A
	giovedì 12 novembre 2020	N			N	N						N
	venerdì 13 novembre 2020	T			T	T						T
	sabato 14 novembre 2020	E			E	E						E
W 47	domenica 15 novembre 2020	Z			Z	Z						Z
	martedì 17 novembre 2020	I			I	I						I
	mercoledì 18 novembre 2020	O			O	O						O
	giovedì 19 novembre 2020	N			N	N						N
	venerdì 20 novembre 2020	E			E	E						E
W 48	domenica 22 novembre 2020											
	martedì 24 novembre 2020			4823 C 3001 F_Fin_40	4823 C 3001 F_Fin_40							4823 A 3001 F_Fin_50
	mercoledì 25 novembre 2020				4823 C 3000 R_Fin_40							4823 A 3001 F_Fin_50
	giovedì 26 novembre 2020	4823 S 3002 L_Fin_50										4823 A 3001 F_Fin_50
	venerdì 27 novembre 2020								4823 S 3000 L_Fin_40			
W 49	sabato 28 novembre 2020					4823 S 3002 L_Fin_40						
	domenica 29 novembre 2020											
	martedì 1 dicembre 2020											
	mercoledì 2 dicembre 2020						4823 S 3007 L_Fin_40					
	giovedì 3 dicembre 2020						4823 C 3007 L_Fin_40		4823 A 3000 F_Fin_40			
W 50	venerdì 4 dicembre 2020											
	sabato 5 dicembre 2020											
	domenica 6 dicembre 2020											
	martedì 8 dicembre 2020						4823 A 3001 G_Fin_40					
	mercoledì 9 dicembre 2020											
W 51	giovedì 10 dicembre 2020											
	venerdì 11 dicembre 2020											
	sabato 12 dicembre 2020											
	domenica 13 dicembre 2020											
	martedì 15 dicembre 2020											
W 52	mercoledì 16 dicembre 2020											
	giovedì 17 dicembre 2020											
	venerdì 18 dicembre 2020											
	sabato 19 dicembre 2020											
	domenica 20 dicembre 2020											
W 53	martedì 22 dicembre 2020											
	mercoledì 23 dicembre 2020											
	giovedì 24 dicembre 2020											
	venerdì 25 dicembre 2020											
	sabato 26 dicembre 2020											
W 54	domenica 27 dicembre 2020											
	martedì 29 dicembre 2020											
	mercoledì 30 dicembre 2020											
	giovedì 31 dicembre 2020											
	venerdì 1 gennaio 2021											
W 55	sabato 2 gennaio 2021											
	domenica 3 gennaio 2021											
	martedì 5 gennaio 2021											
	mercoledì 6 gennaio 2021											
	giovedì 7 gennaio 2021											
W 56	venerdì 8 gennaio 2021											
	sabato 9 gennaio 2021											
	domenica 10 gennaio 2021											

Allegato 5: pianificazione Eurodies



Allegato 6: pianificazione Matlab

BIBLIOGRAFIA

- Deepalakshmi, P, e K Shankar. «Role and Impacts of Ant Colony Optimization in Job Shop Scheduling.» In *Evolutionary Computation in Scheduling*, di Problem: A Detailed Analysis, 11-37. Wiley, 2020.
- Dorigo, M., e V. Stutzle. «Ant Colony Optimization.» MIT Press: London, 2004.
- Dorigo, M., V. Maniezzo, e A. Colorni. «Positive Feedback as a Research Strategy.» Milan: Politecnico di Milano, 1991.
- Dorigo, Maniezzo, e Colorni. «Ant system: optimization by a colony of cooperating agents.» *IEEE Trans. Sys. Man Cybernet. B*, 26, 1996: 29 - 41.
- Enciclopedia Treccani*. s.d.
<https://www.treccani.it/vocabolario/attivita/#:~:text=tardo%20attiv%C4%ADtas%20%2Datis%5D.,.%2C%20di%20un'a.&text=della%20mente%2C%20del%20pensiero%3B%20essere,gli%20altiforni%20sono%20in%20a>.
- Holland, J. H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. 1975.
- Jacobs, R. F., R B Chase, A Grando, e A Sianesi. *Operation Management - nella produzione e nei servizi 4/ed*. McGraw-Hill Education, 2020.
- Jain, A. S., e S. Meeran. «A State-of-the-Art Review of Job-Shop Scheduling Techniques.» *European Journal of Operation Research*, 113, 1999: 390-434.
- Jia, Wenyong, Zhibin Jiang, e You Li. «Scheduling to Minimize the Makespan in Large Piece One-Of-A-Kind Production with Machines Availability Constraints.» *Expert Systems with Application Vol 42*, 2015: 9174-9182.
- Kennedy, J., e R.C. Eberhart. «Particle Swarm Optimization.» *IEEE International Conference on Neural Network*, 1995: 1942-1948.
- Kuchcinski, Krzysztof, e Szymanek Radoslaw. «JaCoP - Java Constraint Programming Solver.» *CP Solvers: Modeling, Applications, Integration, and Standardization, co-located with the 19th International Conference on Principles and Practice of Constraint Programming*. Uppsala, Sweden, 2013.
- Kumar, Sandeep, Sanjay Jain, e Harish Sharma. «Genetic Algorithms.» In *Advances in Swarm Intelligence for Optimizing Problems in Computer Science*, di Anand Nayyar, Dac-Nhuong Le e Nhu Gia Nguyen. 2019.
- Kuznetsov, A. S., e E. E. Noskova. «Assessment of planning methods at job-order manufacturing facilities.» *IOP Conf. Series: Materials Science and Engineering* 862. IOP Publishing, 2020.

- Laganà, Francesca. «Il Manufacturing Execution System. L'inserimento in azienda per la tracciabilità dei semilavorati.» Tesi di laurea magistrale, Torino, 2019.
- M.R. Gary, D.S. Johnson. *Computer and intractability: a guide to the theory of NP-completeness*. Freeman, 1979.
- Metropolis, N., A. Rosenbluth, M. Rosenbluth, A. Teller, e E. Teller. «Equation of state calculations by fast computing machines.» *Journal of chemical physics*, vol. 21, 1953: 1087-1092, 1953.
- Orlando, Mattia, Giulia Bruno, e Franco Lombardi. *Real time scheduling and activity control in one-of-a-kind production*. Tesi di Laurea Magistrale, Torino: Politecnico di Torino, 2020.
- Pinedo, Michael L. *Planning and Scheduling in Manufacturing and Services*. Springer, 2005.
- Qian, B., e et al. «Scheduling multi-objective job shops using a memetic algorithm based on differential evolution.» *International Journal of Advanced Manufacturing and Technology*, 35 (9-10), 2008: 1014 - 1027.
- Sculli, D., e K. K. Tsang. «Priority dispatching rules in a fabrication/assembly shop.» *Mathematical and Computer Modelling*, vol 13, issue 3, 1990: 73-79.
- Slowik, Adam. *Swarm Intelligence Algorithms, A Tutorial*. Taylor & Francis Group, 2020.
- Storn, Reiner, e Kenneth Price. «Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Space.» *Journal of Global Optimization*, 1997: 341 - 359.
- Villa. *Introduction to Production Planning*. 2006.
- Wisittipanich, W., e V. Kachitvichyanukul. «Two enhanced differential evolution algorithms.» *International Journal of Production Research*, Vol. 50, No. 10, 2012: 2757-2773.
- Yalaoui, A, H Chehade, F Yalaoui, e L Amodeo. *Optimization of Logistics*. New York: John Wiley & Sons, 2020.
- Yang, Xin-She. *Optimization Techniques and Applications with examples*. Wiley, 2019.