



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale
Ingegneria Gestionale
A.a. 2025/2026

**Analisi e ottimizzazione delle prestazioni nei
sistemi Multi-Human Multi-Robot mediante
simulazione a eventi discreti**

Relatore:

Professor Federico Barravecchia

Candidato:

Spalluto Vittorio Claudio

ABSTRACT: L'integrazione crescente di robot collaborativi nei contesti industriali pone nuove sfide nella gestione e ottimizzazione dei sistemi Multi-Human Multi-Robot (MHMR), ossia sistemi produttivi nei quali più operatori umani e robot collaborano simultaneamente, condividendo spazi di lavoro, risorse e flussi informativi per il raggiungimento di un obiettivo comune. Uno degli aspetti critici è la stima accurata dei tempi di processamento per unità, tenendo conto sia delle sinergie derivanti dalla collaborazione tra operatori umani e robot, sia delle complessità introdotte dall'aumento del numero di agenti coinvolti.

La presente tesi si propone di analizzare il comportamento dei sistemi MHMR attraverso un approccio basato sulla simulazione. Utilizzando software di simulazione a eventi discreti, vengono modellati diversi scenari produttivi con configurazioni variabili di umani (H) e robot (R). Il focus è posto sull'analisi congiunta del tempo di processo del sistema e sull'identificazione della funzione di complessità di interazione più adatta a rappresentarne l'andamento, confrontando diverse ipotesi di crescita della complessità.

A tal fine viene selezionato e analizzato un caso di assemblaggio rappresentativo, selezionato in base alla sua architettura di prodotto e alla struttura del processo.

INDICE

Capitolo 1 – Introduzione e obiettivi della tesi	1
1.1 Contesto e motivazione dello studio	1
1.2 Importanza dell'integrazione di robot collaborativi nei sistemi produttivi.....	2
1.3 Definizione dei sistemi Multi-Human Multi-Robot come entità produttiva integrata.....	5
1.4 Applicazioni industriali dei sistemi Multi-Human Multi-Robot.....	7
1.5 Problema della stima dei tempi operativi e dell'impatto della complessità dell'interazione nei sistemi Multi-Human Multi-Robot.....	9
1.6 Obiettivo della tesi	11
Capitolo 2 – Modellazione dei tempi operativi in presenza di interazione uomo-robot.....	12
2.1 Definizione del modello di stima dei tempi operativi.....	12
2.2 Ruolo dell'efficienza relativa dei robot rispetto agli umani	13
2.3 Impatto della complessità di interazione.....	14
2.4 Differenti possibili funzioni di complessità implicazioni sul sistema	16
2.4.1 Funzione di elevamento a potenza.....	16
2.4.2 Funzione logaritmica	17
2.4.3 Funzione sigmoide.....	18
2.4.4 Funzione esponenziale.....	18
2.4.5 Funzione lineare.....	19
2.4.6 Funzione lineare con interazione	19
Capitolo 3 – Caso studio: assemblaggio di uno skateboard completo.....	22
3.1 La distinta base del prodotto (BOM)	22
3.2 I task di assemblaggio	23
3.3 Risorse impiegate nel processo di assemblaggio	25
Capitolo 4 – Approccio basato sulla simulazione	27
4.1 Motivazione della scelta della simulazione come metodo di analisi	27
4.2 Panoramica dello strumento di simulazione (Arena).....	29
4.3 Interfaccia del software Arena	30
4.3.1 Project bar.....	33
4.3.2 Barra superiore dei menu e degli strumenti	34
4.3.3 L'ambiente centrale di modellazione.....	35
4.3.4 Il pannello inferiore delle proprietà	38
4.4 Concetti base della simulazione a eventi discreti.....	39
4.4.1 Data definition	41
4.4.2 Discrete processing.....	42
4.4.2.1 Create	44

4.4.2.2	Assign.....	44
4.4.2.3	Process	44
4.4.2.4	Dispose.....	44
4.4.3	Decisions.....	45
4.4.3.1	Decide	46
4.4.4	Grouping	46
4.4.4.1	Separate e Batch.....	48
4.4.5	Record.....	48
4.4.6	Connessioni tra i moduli	49
Capitolo 5 - Implementazione del modello in Arena		54
5.1	Definizione degli elementi del modello	56
5.1.1	Definizione delle entità.....	56
5.1.2	Definizione delle risorse	56
5.1.3	Definizione degli attributi	58
5.1.3.1	Attributi di identificazione e classificazione	59
5.1.3.2	Attributi temporali.....	60
5.1.3.3	Attributi per la scelta delle risorse	60
5.1.3.4	Attributi legati alla qualità ed all'ergonomia	61
5.1.3.5	Attributi pesare i fattori influenti sul criterio decisionale	61
5.1.4	Struttura logica del flusso	62
5.1.4.1	Fasi principali del flusso	62
5.2	Dettaglio del modello simulativo	64
5.2.1	Generazione dello skateboard.....	64
5.2.2	Inizializzazione di attributi e parametri	67
5.2.3	Gestione delle differenze tra le tipologie di task	68
5.2.4	Criterio decisionale	70
5.2.5	Processamento di un task.....	75
5.2.6	Iterazione del ciclo.....	77
5.2.7	Gestione dei parallelismi	80
Capitolo 6 – Analisi dei dati		86
6.1	Contestualizzazione degli obiettivi della tesi nell'analisi dei risultati	86
6.2	Parametri di simulazione.....	87
6.2.1	Numero di repliche per ogni configurazione	87
6.2.2	Processo di arrivo delle entità nel sistema	88
6.2.2.1	Processo di arrivo delle entità nel sistema: numero di entità per simulazione	89

6.2.2.2	Processo di arrivo delle entità nel sistema: distribuzione statistica del tempo di inter-arrivo.....	90
6.2.3	Modellazione di distribuzioni e parametri dell'indice decisionale Q	93
6.2.4	Ipotesi sui costi orari delle risorse	95
6.3	Key Performance Indicators selezionati	96
6.3.1	Indicatori temporali	96
6.3.2	Indicatori di produttività	98
6.3.3	Indicatori economici	100
6.4	Risultati di simulazione.....	103
6.4.1	Impatto delle risorse sul tempo di processo.....	104
6.4.2	Impatto delle risorse sulla produttività	107
6.4.2.1	Impatto delle risorse sul numero di skateboard per ora per risorsa	107
6.4.2.2	Impatto delle risorse sul tempo impiegato per produrre uno skateboard per risorsa	112
6.4.3	Impatto delle risorse sui costi	115
6.5	Discussione complessiva dei risultati della simulazione	120
Capitolo 7	– Analisi della funzione di complessità di interazione.....	124
7.1	Procedura per l'individuazione della funzione	124
7.2	Forma funzionale migliore per la modellazione della complessità di interazione.....	126
7.3	Discussione complessiva dei risultati sulla ricerca della funzione di complessità di interazione	129
Capitolo 8	– Conclusioni, limiti dell'analisi e sviluppi futuri.....	132
8.1	Conclusioni	132
8.2	Limiti dell'analisi e sviluppi futuri	133
Appendice A	- Modello di simulazione.....	136
A.1	Elenco dei task elementari del processo di assemblaggio.....	136
A.2	Parametri delle distribuzioni triangolari	139
A.2.1	– Tempo di processo.....	139
A.2.2	– Tasso di errore	139
A.2.3	– Carico ergonomico.....	140
A.3	– Pesi dell'indice decisionale Q	140
A.4	– Procedura di determinazione dei costi orari delle risorse.....	140
A.5	– Modello di simulazione completo	141
Appendice B	– Risultati completi delle simulazioni.....	146
B.1	– Tempo di processo.....	146
B.2	– Indicatori di produttività basati sul numero di skateboard prodotti per unità di tempo	151

B.3 – Indicatori di produttività basati sul tempo medio per skateboard.....	155
B.4 – Costi unitari per skateboard attribuibili a una singola risorsa	160
B.5 – Tempo di processo.....	165
Appendice C – Stima della funzione di complessità di interazione	170
C.1 – Calcolo dei valori osservati della funzione di complessità.....	170
C.2 – Regressione lineare senza termine di interazione.....	174
C.3 – Regressione lineare con termine di interazione	175
Bibliografia	176

INDICE DELLE FIGURE

Figura 1 - Human-Robot collaboration (fonte: KUKA Global, 2026).....	2
Figura 2 - Automazione nell'industria automotive (fonte: Futurmotive, 2022)	3
Figura 3 - Human-Robot collaboration nell'industria automotive (fonte: Universal-robots, 2020)	4
Figura 4 - Robot impiegati in attività di carico e scarico (fonte: KUKA Global, 2026).....	8
Figura 5 - Robot impiegati nella movimentazione del materiale (fonte: Robotics24/7, 2022)	8
Figura 6 - Quality inspection (fonte: Star-ai, 2021).....	9
Figura 7 - Robot impiegati nel packaging (fonte: Universal-robots, 2026).....	9
Figura 8 - Interfaccia di Arena (fonte: Arena Simulation Software).....	32
Figura 9 - Project Bar (fonte: Arena Simulation Software)	33
Figura 10 - Barra degli strumenti (fonte: Arena Simulation Software)	35
Figura 11 - Ambiente centrale di modellazione (fonte: Arena Simulation Software).....	37
Figura 12 - Pannello inferiore delle proprietà (fonte: Arena Simulation Software)	39
Figura 13 - Data definition (fonte: Arena Simulation Software)	42
Figura 14 - Definizione delle risorse (fonte: Arena Simulation Software)	42
Figura 15 - Discrete processing (fonte: Arena Simulation Software).....	43
Figura 16 – Decisions (fonte: Arena Simulation Software).....	46
Figura 17 – Grouping (fonte: Arena Simulation Software)	48
Figura 18 - Modello elementare in Arena (fonte: Elaborazione propria)	50
Figura 19 - Create elementare (fonte: Elaborazione propria)	51
Figura 20 - Process elementare (fonte: Elaborazione propria)	52
Figura 21 - Decision elementare (fonte: Elaborazione propria)	53
Figura 22 - Definizione entità (fonte: Elaborazione propria)	56
Figura 23 - Definizione risorse (fonte: Elaborazione propria).....	57
Figura 24 - Definizione del set (fonte: Elaborazione propria)	58
Figura 25 - Membri del set (fonte: Elaborazione propria)	58
Figura 26 - Struttura del processo di assemblaggio (fonte: Elaborazione propria)	64
Figura 27 – Processo di arrivo degli skateboard nel sistema (fonte: Elaborazione propria)	65
Figura 28 - Struttura del primo ramo del processo (fonte: Elaborazione propria).....	66
Figura 29 - Inizializzazione (fonte: Elaborazione propria).....	67
Figura 30 – Inizializzazione 2 (fonte: Elaborazione propria)	67
Figura 31 - Assegnazione dei tempi ai task di identification and handling (fonte: Elaborazione propria).....	69

Figura 32 - Criterio decisionale (fonte: Elaborazione propria).....	72
Figura 33 - Confronto tra gli indici di prestazione (fonte: Elaborazione propria).....	73
Figura 34 - Assegnazione degli indici di risorsa: umani (fonte: Elaborazione propria)	74
Figura 35 - Assegnazione degli indici di risorsa: robot (fonte: Elaborazione propria).....	74
Figura 36 - Processamento di un task (fonte: Elaborazione propria).....	76
Figura 37 - Regola di selezione della risorsa per il processamento (fonte: Elaborazione propria) ...	77
Figura 38 - Incremento del TaskIndex (fonte: Elaborazione propria).....	78
Figura 39 - Gestione delle uscite delle entità dai rami (fonte: Elaborazione propria)	79
Figura 40 - Instradamento delle entità (fonte: Elaborazione propria).....	80
Figura 41 - Separate (fonte: Elaborazione propria)	82
Figura 42 - Batch (fonte: Elaborazione propria).....	83
Figura 43 - Gestione degli accoppiamenti tra entità separate (fonte: Elaborazione propria)	85
Figura 44 - Transitorio e stato stazionario (fonte: Elaborazione propria).....	90
Figura 45 - Fattori che influenzano il criterio decisionale (fonte: Elaborazione propria)	93
Figura 46 - Classificazione dei KPI introdotti (fonte: Elaborazione propria).....	96
Figura 47 - Modello di simulazione implementato in Arena (parte 1) (fonte: Elaborazione propria)	142
Figura 48 - Modello di simulazione implementato in Arena (parte 2) (fonte: Elaborazione propria)	143
Figura 49 - Modello di simulazione implementato in Arena (parte 3) (fonte: Elaborazione propria)	144
Figura 50 - Modello di simulazione implementato in Arena (parte 4) (fonte: Elaborazione propria)	145

INDICE DELLE TABELLE

Tabella 1 - Esempi generici di calcolo di $t_a(H,R)$ al variare del numero di umani e robot coinvolti.	14
Tabella 2 - Vantaggi e svantaggi associati a ciascuna forma funzionale.....	20
Tabella 3 - Bill of Materials dello skateboard.....	23
Tabella 4 - Comparazione delle caratteristiche considerate per operatori umani e robotici	26
Tabella 5 - Sintesi dei risultati di simulazione e delle conseguenti implicazioni manageriali.....	122
Tabella 6 - Sintesi delle evidenze emerse dall'analisi sulla funzione di complessità.....	131
Tabella 7 - Task di assemblaggio	138
Tabella 8 - Parametri delle distribuzioni triangolari del tempo di processo degli umani per tipologia di task	139
Tabella 9 - Parametri delle distribuzioni triangolari associate al tasso di errore	139
Tabella 10 - Parametri delle distribuzioni triangolari associate al carico ergonomico	140
Tabella 11 - Pesi associati all'indice decisionale Q	140
Tabella 12 - Determinazione del costo orario delle risorse umane	141
Tabella 13 - Determinazione del costo orario delle risorse robotiche.....	141
Tabella 14 - Tempo di processo per diverse configurazioni	150
Tabella 15 - Indicatori di produttività basati sul numero medio di skateboard prodotti nel tempo .	155
Tabella 16 - Indicatori di produttività basati sul tempo medio necessario alla produzione di uno skateboard	160
Tabella 17 - Indicatori di costo medio per skateboard escludendo i periodi di inattività delle risorse	165
Tabella 18 - Indicatore di costo reale complessivo per skateboard.....	169
Tabella 19 - $f_c(H,R)$	174
Tabella 20 - Regressione lineare senza termine di interazione	175
Tabella 21 - Regressione lineare con termine di interazione	175

INDICE DEI GRAFICI

Grafico 1 - Tempo di processo in funzione del numero di umani e del numero di robot	105
Grafico 2 - π_H in funzione del numero di umani e del numero di robot, vista prospettica 1	108
Grafico 3 - π_H in funzione del numero di umani e del numero di robot, vista prospettica 2	108
Grafico 4 - π_R in funzione del numero di umani e del numero di robot, vista prospettica 1	109
Grafico 5 - π_R in funzione del numero di umani e del numero di robot, vista prospettica 2	109
Grafico 6 - π_{tot} in funzione del numero di umani e del numero di robot	111
Grafico 7 - P_H in funzione del numero di umani e del numero di robot	112
Grafico 8 - P_R in funzione del numero di umani e del numero di robot	113
Grafico 9 - P_{tot} in funzione del numero di umani e del numero di robot	114
Grafico 10 - γ_H in funzione del numero di umani e del numero di robot.....	116
Grafico 11 - γ_R in funzione del numero di umani e del numero di robot.....	116
Grafico 12 - C_{work} in funzione del numero di umani e del numero di robot	118
Grafico 13 - C in funzione del numero di umani e del numero di robot.....	119

Capitolo 1 – Introduzione e obiettivi della tesi

1.1 Contesto e motivazione dello studio

Il rapido sviluppo delle tecnologie digitali e robotiche ha favorito, negli ultimi anni, una crescente integrazione dei robot nei sistemi produttivi. I robot industriali sono divenuti una componente essenziale nella ricerca di una maggiore efficienza produttiva da parte delle imprese.

La Federazione Internazionale di Robotica aveva previsto per il 2019 un tasso di crescita del 13% nel settore della produzione robotica. Inoltre, molte aziende stanno concentrando la propria attenzione sull'integrazione di funzionalità specifiche nei robot, come maggiore facilità d'uso, flessibilità e sicurezza. Cresce così la domanda di robot capaci di lavorare accanto all'uomo in sicurezza e di adattarsi a compiti e ambienti differenti: i robot collaborativi, o "cobot", stanno guadagnando sempre più popolarità, rendendo l'integrazione dei sistemi robotici nei processi produttivi accessibile anche a imprese di piccole e medie dimensioni. Ciò comporta un miglioramento dell'efficienza e della flessibilità in ambienti industriali (Yang, 2023).

“Dopo molti anni in cui l'automazione era confinata a processi rigidamente predefiniti, l'industria manifatturiera sta andando incontro ad una trasformazione significativa verso processi più flessibili ed intelligenti, la cosiddetta Industria 4.0” (S. Robla-Gomez, 2017). In questo contesto i robot stanno diventando parte integrante di ambienti di lavoro dinamici e condivisi con operatori umani. In particolare, l'attenzione della ricerca e dell'industria si sta progressivamente spostando verso sistemi collaborativi, nei quali esseri umani e robot interagiscono direttamente, suddividendo compiti, spazi e responsabilità operative (L. Wang, 2019).

La collaborazione uomo-macchina ha un ruolo molto importante nelle smart factories, ossia sistemi produttivi avanzati caratterizzati dall'integrazione di tecnologie digitali, automazione intelligente e sistemi cyber-fisici, contribuendo al raggiungimento di una maggiore produttività ed efficienza (S. Robla-Gomez, 2017). Collaborando, esseri umani e robot possono combinare le rispettive capacità e punti di forza per ottimizzare le operazioni manifatturiere (Figura 1). L'uso della collaborazione uomo-robot sta dunque trasformando l'approccio tradizionale all'impiego dei robot, e i ricercatori si stanno concentrando sempre più su nuove modalità per sfruttare questa collaborazione al fine di migliorare l'efficienza e la qualità produttiva (Yang, 2023).

Con questa evoluzione si sono sviluppati e diffusi sempre più i cosiddetti “sistemi Multi-Human Multi-Robot”, ossia strutture complesse in cui più agenti umani e robotici lavorano contemporaneamente per il raggiungimento di un obiettivo di produzione predefinito (Villani V., 2018). Tali sistemi hanno introdotto un livello di complessità del tutto nuovo, sia sotto il profilo tecnico-progettuale sia sotto quello gestionale. In particolare, la loro crescente diffusione ha reso necessaria l'introduzione di strumenti teorici e analitici in grado di analizzarne ed ottimizzarne le prestazioni.



Figura 1 - Human-Robot collaboration (fonte: KUKA Global, 2026)

La presente tesi si propone, in questo contesto, di affrontare il problema della definizione di un modello matematico in grado di stimare i tempi di processo e di supportare un'analisi economica di un sistema Multi-Human Multi-Robot, al variare del numero di operatori umani ed operatori robotici coinvolti in uno specifico processo produttivo con l'obiettivo di offrire una rappresentazione realistica e utile come supporto decisionale per la progettazione e l'applicazione di tali sistemi.

La motivazione del lavoro nasce dall'osservazione delle difficoltà attuali nella previsione dei tempi di esecuzione negli ambienti collaborativi descritti e dalla consapevolezza che un modello matematico strutturato possa offrire un contributo significativo sia alla ricerca sia alla pratica industriale.

1.2 Importanza dell'integrazione di robot collaborativi nei sistemi produttivi

L'integrazione dei robot collaborativi nel contesto industriale rappresenta una delle più significative trasformazioni che l'industria manifatturiera abbia mai attraversato. Studi

recenti hanno messo in evidenza i benefici della collaborazione uomo–robot nella produzione intelligente. Uno dei principali vantaggi è il miglioramento della produttività. Combinando i punti di forza di robot e persone, i produttori possono raggiungere tassi di produzione più elevati e cicli di lavoro più rapidi (Yang, 2023). L’approccio tradizionale all’automazione industriale, basato sull’integrazione di robot industriali rigidi e fisicamente separati dagli operatori umani, si è dimostrato efficace principalmente in contesti di realizzazione di prodotti caratterizzati da elevata standardizzazione e ripetitività, nonché per la produzione di massa (Villani V., 2018). L’esempio tipico è quello della catena di montaggio, dove i robot eseguono ripetutamente la stessa operazione (Figura 2).

Negli ultimi decenni, tuttavia, gli standard di mercato stanno evolvendo e mutando verso una più elevata personalizzazione, tempi di consegna ridotti ed elevata flessibilità. In particolare, si è reso necessario un adattamento continuo ai cambiamenti della domanda ed a una notevole riduzione di tempi e costi. Tali richieste hanno reso difficoltoso l’approccio adottato da sistemi produttivi rigidi, favorendo l’emergere di approcci più flessibili, adattabili e dinamici (Lasi H., 2014) (Figura 3).



Figura 2 - Automazione nell'industria automotive (fonte: Futurmotive, 2022)



Figura 3 - Human-Robot collaboration nell'industria automotive (fonte: Universal-robots, 2020)

I robot collaborativi, ossia robot progettati per interagire direttamente con l'essere umano senza la necessità di barriere fisiche, si sono progressivamente inseriti in questo nuovo scenario con l'obiettivo non solo di automatizzare i compiti più ripetitivi, gravosi e potenzialmente pericolosi, ma anche di incrementare la produttività e la qualità del lavoro, affiancando e supportando l'operatore umano nelle attività più complesse e in quelle che richiedono elevati livelli di precisione.

L'autonomia di questo tipo di robot "evoluti" deriva dalla capacità di elaborare le informazioni raccolte dai sensori e di pianificare la sequenza di azioni da intraprendere. Alcuni esempi di robot di questo tipo sono quelli impiegati nell'esplorazione spaziale, capaci di riconoscere ed evitare ostacoli in ambienti impervi e sconosciuti a grandissime distanze da qualsiasi operatore.

Utilizzare un robot in un ambiente naturale non modificato, condiviso con esseri umani, richiede precisi requisiti concernenti la percezione sensoriale, la mobilità e la destrezza nonché la capacità di pianificazione dei compiti, di prendere decisioni ed effettuare ragionamenti.

La scelta di integrare robot collaborativi nei sistemi produttivi è stata definita una "scelta strategica" dal National Science Foundation; di conseguenza, essa non costituisce soltanto un'evoluzione tecnologica, ma influenza profondamente la struttura e l'evoluzione dei processi industriali (Amedeo Cappelli, 2003).

L'introduzione di questi strumenti caratterizzati da elevata complessità rende pertanto necessario l'impiego di strumenti analitici adeguati alla comprensione del loro funzionamento.

1.3 Definizione dei sistemi Multi-Human Multi-Robot come entità produttiva integrata

Con il termine “sistema produttivo Multi-Human Multi-Robot” si intende una specifica tipologia di processo produttivo caratterizzato dalla presenza simultanea di un numero variabile di agenti umani e robotici che agiscono collaborativamente ed in modo organizzato per il raggiungimento di un obiettivo comune.

Questa tipologia di sistemi si distingue nettamente dai tradizionali sistemi automatizzati già ampiamente diffusi. In questo caso, infatti, gli operatori umani e i robot condividono lo stesso spazio operativo, le medesime risorse e gli stessi flussi informativi. La cooperazione risulta pertanto più diretta, coordinata e spesso simultanea (Villani V., 2018).

Al fine di comprendere meglio tale distinzione, è possibile classificare l’interazione uomo-robot in diverse macrocategorie: coesistenza, supervisione, cooperazione, supporto, collaborazione e collaborazione simbiotica.

- **Coesistenza:** si riferisce a relazioni in cui uomo e robot svolgono compiti differenti con obiettivi di lavoro distinti, pur condividendo lo stesso spazio fisico.
- **Supervisione:** in questo tipo di relazione, il robot dispone di un’autonomia limitata e richiede un input e una direzione costanti da parte dell’operatore umano. I compiti sono svolti simultaneamente e orientati verso lo stesso obiettivo, ma il robot mantiene un livello ridotto di indipendenza e l’adattabilità non rappresenta un requisito fondamentale.
- **Cooperazione:** si riferisce a uno sforzo coordinato tra esseri umani e robot finalizzato al raggiungimento di un obiettivo comune, in cui ciascuna parte opera su uno specifico compito o insieme di compiti. In questo senso, la cooperazione può essere definita come una modalità strutturata di lavoro congiunto, in cui ruoli e responsabilità sono chiaramente definiti e vi è una netta suddivisione delle attività.
- **Supporto:** robot o operatori umani possono agire in modalità di supporto, ovvero secondo una relazione di tipo master–slave. Nonostante la condivisione dell’obiettivo, delle risorse e dello spazio di lavoro, il soggetto che fornisce supporto non dispone di autonomia decisionale nell’assegnazione o nell’esecuzione del compito.
- **Collaborazione:** si riferisce a un processo in cui robot e operatori umani condividono compiti, informazioni e risorse al fine di raggiungere un obiettivo comune. Le

operazioni vengono svolte simultaneamente e in contatto diretto, e l'autonomia nell'esecuzione delle attività è ripartita in modo equilibrato tra gli agenti coinvolti.

- Collaborazione simbiotica: in questo tipo di relazione, uomo e robot sono reciprocamente dipendenti l'uno dall'altro. In tale forma di collaborazione, robot e operatore umano lavorano in modo complementare, integrando le rispettive capacità al fine di raggiungere l'obiettivo comune (F. Barravecchia, 2023).

Le interazioni possono essere sia fisiche sia senza contatto, tramite tecniche di comunicazione quali voce, gesti, riconoscimento delle intenzioni o movimenti oculari, analizzate mediante sistemi di visione artificiale e meccanismi di feedback.

Mentre i robot automatizzati possono far risparmiare energia, tempo e risorse ottimizzando i processi, i cobot sono in grado di trasformare le strutture sociali (R. Jahanmahin, 2022). Dato che la coesistenza tra esseri umani e robot è destinata a diventare inevitabile e considerando la potenziale esposizione degli utenti non sufficientemente addestrati, i robot devono essere in grado di eseguire compiti in modo consapevole.

Molti Paesi hanno avviato iniziative strategiche per l'implementazione dell'Industria 4.0 e numerose ricerche sono state compiute per svilupparne ulteriormente l'applicabilità, puntando ad una più efficiente autonomia attraverso l'uso di sistemi cyber-fisici e Internet of Things. L'Industria 5.0, che comprende le interazioni uomo-robot, si propone di sfruttare la creatività degli specialisti umani in collaborazione con macchine e robot intelligenti, precisi ed efficienti per completare tali sforzi di ricerca, il che sottolinea ulteriormente la necessità di interazioni uomo-robot appropriate, ottimali ed efficienti (R. Jahanmahin, 2022). I ricercatori stanno quindi ora esplorando modalità per integrare l'esperienza, le capacità decisionali e il pensiero critico umano con la forza, la ripetibilità e la precisione dei robot per affrontare compiti complessi (Yang, 2023).

Un sistema Multi-Human Multi-Robot non deve quindi essere inteso come la semplice somma di operatori umani e robotici, bensì come un'unica entità produttiva fortemente integrata. Tale approccio consente di combinare l'intuito, l'adattabilità e l'intelligenza contestuale degli esseri umani con la precisione, la ripetibilità e la forza operativa dei robot collaborativi. Questo tipo di collaborazione consente agli operatori umani di concentrarsi su operazioni ad alto valore aggiunto o che richiedono elevati livelli di destrezza, liberandoli così da attività ripetitive o potenzialmente rischiose (A. Hentout, 2019). Tuttavia, alcune operazioni possono essere troppo complesse per essere eseguite da robot, oppure troppo costose da automatizzare, in quanto richiederebbero lo sviluppo di strumenti e sistemi

specifici. Per questo motivo, creare un ambiente collaborativo in cui uomini e robot possano lavorare fianco a fianco e condividere i compiti in uno spazio aperto e privo di barriere è un obiettivo rilevante da raggiungere (I. Maurtua, 2017).

La sfida nell'applicazione industriale di questi sistemi risiede quindi nel coordinamento efficace delle attività tra gli agenti eterogenei, attraverso forme di interazione fluide, intelligenti e sicure.

1.4 Applicazioni industriali dei sistemi Multi-Human Multi-Robot

I sistemi di produzione basati sull'interazione uomo-macchina trovano applicazione in una vasta gamma di contesti industriali. Grazie alla loro flessibilità e capacità di adattarsi a diverse esigenze operative i cobot rappresentano un elemento chiave per l'ottimizzazione di molteplici fasi dei processi manifatturieri. Tra queste, una delle più diffuse risulta essere la fase di assemblaggio: in questo caso, i robot collaborativi possono essere impiegati, ad esempio, per selezionare i componenti corretti ed orientarli, per applicare eventuali sostanze adesive o per effettuare saldature, ma anche per verniciare l'assemblato, pulirlo e rifinirlo. Proprio questa fase costituirà il caso di studio approfondito nell'ambito della presente tesi. Tuttavia, l'assemblaggio non rappresenta l'unica fase produttiva in cui è possibile integrare i sistemi Multi-Human Multi-Robot. Questi sistemi trovano in realtà impiego anche in numerose altre fasi dei processi manifatturieri, tra cui:

1. Operazioni di carico, scarico e gestione delle macchine: si tratta spesso di attività ripetitive ed a basso valore aggiunto, in cui grazie alla presenza di operatori umani e robotici, i compiti possono essere distribuiti in modo più efficiente (Figura 4). I robot possono quindi occuparsi dell'avviare i cicli produttivi o movimentare pezzi già finiti, mentre gli umani si concentrano su operazioni a più elevato contenuto cognitivo, come la supervisione e la gestione delle anomalie. La capacità dei cobot di prevedere guasti o individuare criticità nei macchinari costituisce elemento importante nel loro impiego nel contesto specificato.



Figura 4 - Robot impiegati in attività di carico e scarico (fonte: KUKA Global, 2026)

2. Gestione del materiale: tra le attività a basso valore aggiunto che gravano maggiormente sugli operatori umani vi è la ricerca del materiale in magazzino e il trasporto di questo verso le postazioni operative (Figura 5). Si tratta di operazioni spesso ripetitive, dispendiose in termini di tempo e talvolta anche fisicamente impegnative. In questo contesto, i cobot possono contribuire a colmare l'inefficienza diffusa in molti contesti produttivi in modo da consentire agli operatori di concentrarsi su attività a maggiore contenuto decisionale o tecnico (Y. Cohen, 2021).

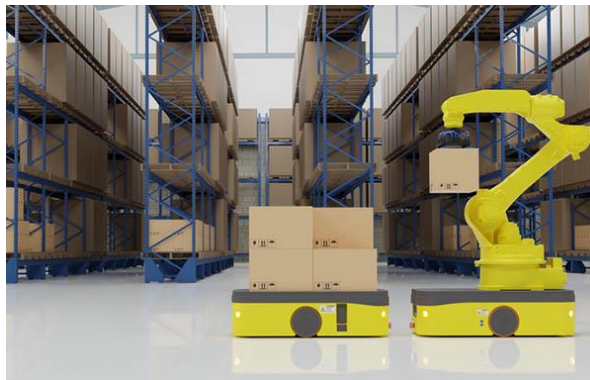


Figura 5 - Robot impiegati nella movimentazione del materiale (fonte: Robotics24/7, 2022)

3. Controllo qualità: anche nelle fasi di controllo qualità i robot collaborativi forniscono supporto nel carico e scarico delle macchine impiegate. In casi tecnologicamente più avanzati risulta possibile anche equipaggiarli con algoritmi di intelligenza artificiale e sensori in modo da eseguire direttamente l'ispezione di qualità e selezionare i difettosi con una accuratezza estremamente elevata. Ciò consente di migliorare sensibilmente le prestazioni e l'efficienza del processo, portando a una notevole riduzione dei difetti osservati in uscita (Figura 6).

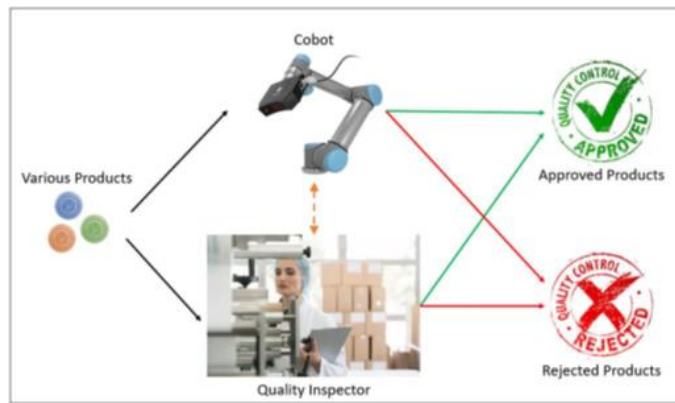


Figura 6 - Quality inspection (fonte: Star-ai, 2021)

4. Packaging e pallettizzazione: i cobot aiutano gli operatori umani eseguendo le fasi fisicamente pesanti, ripetitive o ergonomicamente pericolose. Il robot, in particolare, afferra il prodotto tramite un gripper e lo deposita sul pallet o nel contenitore secondo una disposizione ottimizzata seguendo un pattern. L'operatore si occupa invece di riempire i contenitori, cambiare i pallet o supervisionare più postazioni contemporaneamente (Figura 7). In questo modo l'ergonomia è migliorata, vi è una maggiore produttività grazie al fatto che il robot lavora senza pausa e l'operatore si concentra su attività a maggior valore aggiunto e si riducono infortuni, principalmente i disturbi muscoloscheletrici (S. El Zaatari, 2019).



Figura 7 - Robot impiegati nel packaging (fonte: Universal-robots, 2026)

1.5 Problema della stima dei tempi operativi e dell'impatto della complessità dell'interazione nei sistemi Multi-Human Multi-Robot

Nei sistemi produttivi dalla presenza congiunta di più agenti umani e robotici, risulta evidente come la complessità di realizzare un modello sufficientemente accurato per la stima dei tempi di processo sia molto elevata. Questa elevata complessità è dovuta al considerevole

numero di variabili coinvolte, generate dall'interazione tra agenti eterogenei, di cui è necessario tenere conto al fine di ottenere stime affidabili ed applicabili per l'ottimizzazione dei processi decisionali delle industrie (A. Dahiya, 2023).

In primo luogo, tener conto dell'efficienza operativa dei cobot impiegati risulta elemento fondamentale per ottenere risultati soddisfacenti. Tale efficienza è direttamente correlata allo stato attuale della tecnologia e della ricerca e presenta tuttora dei limiti fisiologici, soprattutto nei contesti produttivi ad alta variabilità. Inoltre, è necessario tener conto anche dell'entità dell'investimento che si è deciso di effettuare nel processo produttivo in esame: dispositivi rapidi e precisi nell'esecuzione dei compiti, con tempi di risposta brevi ed in grado di muoversi contribuiscono in modo più positivo alle performance globali del processo (Krüger, 2009).

Un ulteriore aspetto rilevante riguarda la capacità di adattamento dei robot ai continui cambiamenti del contesto operativo. In questo ambito, l'introduzione di tecnologie basate sull'intelligenza artificiale ha consentito un notevole passo in avanti in questa direzione, rendendo i robot sempre più flessibili e capaci di risolvere possibili problematiche autonomamente.

Un'altra variabile fondamentale da prendere in considerazione è quella legata all'interazione tra agenti umani e agenti robotici. In particolare, l'introduzione di un numero crescente di operatori (umani o robotici) in un sistema collaborativo non implica necessariamente una riduzione lineare del tempo di processo. Al contrario, all'aumentare della numerosità degli agenti coinvolti, possono subentrare fattori complessi come:

- Difficoltà di coordinamento: l'aumento del numero di agenti rende più complesso l'allineamento delle attività, aumentando il rischio di interferenze operative.
- Problemi di comunicazione e sincronizzazione: la crescita delle interazioni tra agenti può introdurre ritardi informativi tra le diverse fasi del processo.
- Uso non ottimale delle risorse e degli spazi condivisi: la condivisione di risorse e ambienti di lavoro da parte di più agenti può generare congestioni e ridurre l'efficienza complessiva del sistema.

Questi fattori possono generare conflitti operativi che porterebbero necessariamente a perdite in termini di efficienza.

Si osserva che si potrebbe giungere in alcuni casi ad una situazione controintuitiva, in cui un aumento del numero di agenti coinvolti porta ad un incremento del tempo necessario al completamento del processo e non una sua riduzione.

Tali dinamiche evidenziano come sia necessario tener conto di questi complessi fattori di interazione al fine di ottenere un modello predittivo che possa descrivere adeguatamente il sistema.

1.6 Obiettivo della tesi

L'obiettivo della tesi consiste nello sviluppare un modello preliminare per la stima dei costi complessivi di un sistema Multi-Human Multi-Robot e, in particolare, nella realizzazione di un'analisi comparativa tra le diverse configurazioni di uno specifico sistema, al fine di individuarne l'ottimo tecnico e l'ottimo economico. Per conseguire tale finalità sarà innanzitutto necessario costruire un modello matematico in grado di stimare il tempo di processo relativo a una specifica attività produttiva, in funzione del numero di operatori umani e robotici coinvolti.

Il modello non terrà conto esclusivamente della numerosità degli agenti, ma includerà ulteriori dimensioni di complessità, quali il livello tecnologico e dunque l'efficienza operativa dei robot impiegati, nonché l'impatto derivante dall'interazione tra agenti eterogenei, ossia tra operatori umani e cobot. Tali dimensioni saranno integrate esplicitamente nella formulazione, in quanto influenzano in modo significativo le prestazioni complessive del sistema.

Mentre l'efficienza operativa dei robot risulta relativamente agevole da modellare dal punto di vista matematico, la stima della funzione di complessità di interazione tra agenti umani e robotici appare più articolata. Un ulteriore obiettivo della tesi consiste pertanto nell'individuazione e nella definizione della forma funzionale più appropriata per rappresentarla.

Al fine di raggiungere questi risultati sarà condotta un'analisi basata sulla simulazione a eventi discreti, che consente di modellare con precisione scenari produttivi caratterizzati da configurazioni variabili di operatori umani e cobot. A supporto dell'analisi sarà approfondito un caso studio relativo a un processo di assemblaggio rappresentativo, così da consentire l'applicabilità del modello ottenuto a contesti industriali realistici.

Capitolo 2 – Modellazione dei tempi operativi in presenza di interazione uomo-robot

2.1 Definizione del modello di stima dei tempi operativi

Come già esplicitato nel paragrafo dedicato, l'obiettivo principale della tesi consiste nella stima dei costi complessivi di un processo nel quale sono coinvolti operatori umani e robot collaborativi. A tal fine è necessario creare un modello matematico che sia in grado di calcolare il tempo di processo $t_a(H,R)$ unitario, nonché il tempo necessario per ottenere una singola unità di prodotto finito in un contesto produttivo in cui intervengono, in maniera congiunta, operatori umani (indicati con H) e robot collaborativi (indicati con R) nello svolgimento delle operazioni previste.

La costruzione del modello richiede un approccio incrementale e strutturato, articolato nelle seguenti fasi:

1. Definizione del caso di riferimento

Si introduce come configurazione base il caso in cui l'intero processo produttivo venga svolto da un singolo operatore umano in assenza di robot collaborativi.

Il tempo di processo unitario di riferimento è quindi definito come:

$$t_a(1,0) \quad (2.1)$$

Dove $t_a(1,0)$ rappresenta il tempo necessario affinché un singolo operatore umano completi l'intero ciclo produttivo. Questo valore costituisce la base su cui si sviluppano le successive estensioni del modello.

2. Introduzione di un secondo operatore umano

Nel caso ideale in cui le operazioni siano perfettamente divisibili e non siano presenti inefficienze operative, l'aggiunta di un secondo operatore umano consente di suddividere il lavoro in modo equo, determinando una riduzione del tempo di processo unitario.

Si ottiene pertanto:

$$t_a(2,0) = \frac{t_a(1,0)}{2} \quad (2.2)$$

Generalizzando il risultato al caso di H operatori umani si ottiene:

$$t_a(H, 0) = \frac{t_a(1,0)}{H} \quad (2.3)$$

3. Introduzione dei robot collaborativi:

Si considera ora l'inserimento di robot collaborativi nel processo produttivo assumendo, in prima approssimazione, che robot e operatori umani abbiano la stessa efficienza operativa e che non siano presenti inefficienze derivanti dall'interazione tra gli agenti.

In tali condizioni, il tempo di processo unitario diventa:

$$t_a(H, R) = \frac{t_a(1,0)}{H + R} \quad (2.4)$$

2.2 Ruolo dell'efficienza relativa dei robot rispetto agli umani

Il caso illustrato finora risulta poco realistico, in quanto assume che robot e umani abbiano la stessa identica capacità di svolgere le operazioni previste, il che raramente corrisponde alla realtà: solitamente risulta tecnologicamente complesso e spesso poco utile far corrispondere esattamente l'efficienza dei robot a quella dell'uomo. È necessario quindi introdurre un fattore di efficienza associato ai cobot, indicato con il parametro α , che rappresenta l'efficienza relativa di un robot rispetto a un operatore umano.

Il parametro consente di pesare in modo più realistico il contributo dei robot all'interno del sistema e vale:

- $\alpha > 1$ se il robot lavora più velocemente rispetto ad un umano.
- $\alpha = 1$ se il robot lavora alla stessa velocità dell'umano.
- $\alpha < 1$ se il robot lavora più lentamente rispetto ad un umano.

Il modello viene quindi modificato come segue:

$$t_a(H, R) = \frac{t_a(1,0)}{H + \alpha R} \quad (2.5)$$

Dove:

- $t_a(1,0)$ è il tempo di processo unitario necessario ad un singolo operatore umano.
- H è il numero di operatori umani.
- R è il numero di robot.
- α è il coefficiente che rappresenta l'efficienza dei robot relativa agli operatori umani.

Il valore di α dipende principalmente dalle caratteristiche tecnologiche dei cobot, quali velocità di esecuzione, precisione operativa, tempo di risposta e flessibilità nella riprogrammazione. Inoltre, anche l'organizzazione del processo produttivo e il livello di coordinamento tra operatori umani e robot possono influenzare significativamente tale parametro.

Alcuni esempi generici di confronto tra $t_a(I,0)$ e $t_a(H,R)$ al variare della configurazione delle risorse e del parametro di efficienza dei robot α sono riportati in Tabella 1.

H	R	α	$t_a(I,0)$ (sec)	$t_a(H,R)$ (sec)
2	0	-	100	50.0
3	0	-	100	33.33
1	1	0.5	100	66.67
1	1	1.5	100	40.0

Tabella 1 - Esempi generici di calcolo di $t_a(H,R)$ al variare del numero di umani e robot coinvolti

Come anticipato, tuttavia, non risulta realistico affermare che il tempo di processo diminuisce all'aumentare del numero di operatori coinvolti secondo la funzione definita in precedenza. La condivisione dello stesso ambiente operativo da parte di più operatori, per giunta di diversa natura, può infatti causare difficoltà di coordinamento e problemi di sincronizzazione, portando spesso gli agenti a ostacolarsi a vicenda.

2.3 Impatto della complessità di interazione

Nel proseguire con la definizione del modello, è fondamentale introdurre due elementi chiave che influenzano in modo significativo il tempo di processo complessivo e, di conseguenza, l'efficienza dell'intero sistema produttivo.

Questi due fattori rappresentano forze opposte che agiscono sull'evoluzione del sistema produttivo man mano che cresce il numero di operatori umani H e robotici R coinvolti.

1. Fattore di sinergia, che consiste in una riduzione del tempo di processo finale all'aumentare del numero di operatori umani H e robotici R , dovuta ad un miglioramento nell'efficienza di distribuzione delle operazioni e miglior bilanciamento del carico di lavoro.
2. Fattore di complessità, che consiste in un aumento del tempo di processo finale all'aumentare del numero di operatori umani H e robotici R , per via delle difficoltà di interazione fra operatori simili e fra operatori di natura differente.

Si può facilmente comprendere la natura di questi due fattori, in quanto risulta intuitivo aspettarsi che il tempo di processo inizialmente diminuisca rapidamente all'aumentare del numero di operatori coinvolti. Tuttavia, raggiunto un certo limite, il beneficio derivante dall'aggiunta di ulteriori operatori si riduce. Il problema del coordinamento diventa infatti più articolato, la comunicazione più complessa e la presenza di un numero elevato di operatori in uno spazio limitato aumenta il rischio di interferenze reciproche. Non risulta pertanto irrealistico ipotizzare anche un'inversione dell'andamento del tempo di processo, con conseguente aumento dello stesso all'aumentare del numero di operatori coinvolti oltre una determinata soglia. Questo fattore è amplificato a causa della diversa natura degli agenti coinvolti e risulta necessariamente rilevante per il problema in analisi (Barravecchia, 2025). Al fine di considerare all'interno del modello l'impatto delle difficoltà legate al coordinamento, alla sincronizzazione ed alla comunicazione tra agenti, si introduce un fattore moltiplicativo β della funzione di complessità di interazione, denominato *fattore di complessità di interazione*. Questo consente di rendere il tempo di processo stimato dal modello maggiormente rappresentativo di un contesto reale.

$$t_a(H, R) = \frac{t_a(1,0)}{H + \alpha R} * (1 + \beta f_c(H, R)) \quad (2.6)$$

Dove:

- $f_c(H, R)$ è la funzione di complessità di interazione, che può essere modellata in modi diversi e quindi assumere varie forme.
- β è un coefficiente che quantifica l'impatto della complessità di interazione sul tempo di processo. Cattura quindi la perdita di efficienza all'aumentare del numero di operatori umani e robotici coinvolti nelle operazioni e vale:
 - $\beta = 0$ nel caso ideale di assenza di complessità di interazione.
 - $\beta > 0$ in caso reale di presenza di difficoltà dovute all'interazione, in particolare più è elevato il termine, maggiore sarà il tempo di processo.

È importante considerare che il parametro β è influenzato in modo significativo dalla configurazione del sistema di produzione. Un flusso di lavoro mal progettato può infatti determinare un incremento di β , riflettendo un aumento della complessità dovuto a interazioni inefficaci o ridondanti tra le fasi del processo.

Risulta inoltre fondamentale evidenziare la rilevanza di entrambi i termini β e $f_c(H, R)$. In particolare β misura l'entità con cui la complessità di interazione ha impatto sul sistema,

mentre la funzione $f_c(H,R)$ esprime in che modo la complessità di interazione cresce all'aumentare del numero di agenti coinvolti. L'introduzione del solo parametro β implicherebbe l'assunzione di una complessità fissa, indipendente dal numero di operatori, rendendo impossibile la modellazione di diversi andamenti di crescita della complessità. La scelta della funzione di complessità risulta pertanto elemento fondamentale dell'analisi, senza la quale non sarebbe possibile ottenere una stima accurata del tempo di processo del sistema preso in considerazione.

2.4 Differenti possibili funzioni di complessità implicazioni sul sistema

Dopo aver introdotto la nozione di complessità di interazione come elemento che incide in maniera non trascurabile sul tempo di processo di un sistema produttivo, risulta necessario affrontare la problematica della sua formalizzazione all'interno del modello matematico proposto.

La criticità principale dello studio è rappresentata dall'attuale inesistenza di una forma univoca già universalmente accettata della funzione di complessità $f_c(H,R)$, in quanto questa dipende dal processo considerato e dalle modalità di interazione tra agenti diversi. Risulta quindi utile esplorare diverse possibili formulazioni di $f_c(H,R)$, ciascuna con propri vantaggi, svantaggi ed implicazioni matematiche (Barravecchia, 2025).

Il presente paragrafo si propone dunque di presentare e discutere alcune possibili formulazioni matematiche della funzione di complessità di interazione, osservando per ognuna eventuali caratteristiche modellistiche ed eventuali vantaggi e svantaggi rilevanti. Le proposte che seguono costituiranno la base per l'analisi successiva, in cui si cercherà, attraverso un approccio simulativo, di individuare la funzione che descrive con maggiore accuratezza il comportamento del sistema produttivo analizzato.

2.4.1 Funzione di elevamento a potenza

Una prima proposta per la formalizzazione della funzione di complessità è una funzione di elevamento a potenza, del tipo:

$$f(H, R) = (H + R - 1)^Y \quad (2.7)$$

Questa espressione rappresenta una delle scelte più semplici dal punto di vista analitico, ma al contempo presenta alcune caratteristiche interessanti.

Innanzitutto, il termine $H + R - 1$ rappresenta l'eccesso di operatori rispetto al caso base in cui è coinvolto un solo operatore. Risulta immediato di fatti comprendere che nel caso di un singolo operatore umano ($H=1, R=0$) si avrà una complessità di interazione nulla, in quanto l'unica unità coinvolta non dovrà interagire e condividere lo spazio destinato alla produzione con nessun'altra.

Il parametro $Y \geq 1$ rappresenta un coefficiente di crescita della complessità e regola l'intensità con cui l'incremento del numero di operatori umani e robotici incide sull'aumento della complessità del sistema.

Questa forma consente di introdurre in modo flessibile diverse dinamiche di crescita della complessità: si osserva ad esempio che nel caso più semplice in cui $Y = 1$, la complessità di interazione crescerà in modo lineare e ogni operatore aggiuntivo, di qualsiasi natura comporta un incremento costante del fattore penalizzante.

Un'ulteriore scelta possibile di Y risulta essere $Y = 2$: l'incremento quadratico rispetto al numero complessivo di agenti $H + R$ simula una crescita accelerata delle difficoltà di interazione, coerente con quanto osservato in ambienti produttivi ad alta densità operativa, dove i fenomeni di congestione, interferenza fisica e sovrapposizione comunicativa aumentano rapidamente. La forma quadratica permette di enfatizzare l'effetto negativo derivante da configurazioni con molti attori contemporaneamente presenti nel processo.

Infine, valori superiori a due possono essere impiegati in contesti estremi, dove la complessità aumenta in modo molto brusco con l'aggiunta di nuovi operatori (es. in ambienti fisicamente limitati o poco flessibili).

La scelta di Y assume un ruolo cruciale nella calibrazione del modello, poiché influenza direttamente la sensibilità del sistema agli effetti di coordinamento. Un valore troppo basso potrebbe sottostimare l'impatto delle interazioni multiple, mentre un valore troppo alto rischia di penalizzare eccessivamente configurazioni che nella realtà risultano ancora gestibili.

2.4.2 Funzione logaritmica

Una alternativa significativa è costituita dalla funzione logaritmica, la quale è formulata come:

$$f(H, R) = \log(1 + H + R) \quad (2.8)$$

Questa funzione consente di modellizzare in maniera opportuna l'ipotesi secondo cui la complessità dovuta all'interazione tra agenti non cresca in modo vertiginoso all'aumentare

del numero di agenti, ma tenda a saturarsi progressivamente. Questo comportamento potrebbe rispecchiare la realtà in un contesto in cui, oltre una certa soglia di operatori, i meccanismi organizzativi riescono a limitare l'effetto caotico dell'interazione.

La funzione logaritmica risulta adatta a descrivere scenari con crescita moderata della complessità, ma in sistemi meno stabili o con interazioni complesse può risultare teoricamente poco rappresentativa del processo osservato.

2.4.3 Funzione sigmoide

Un ulteriore approccio per modellare la complessità crescente all'aumentare del numero di operatori coinvolti prevede l'utilizzo di una funzione sigmoide, definita come:

$$f(H, R) = \frac{1}{(1 + e^{-k(H+R-T)})} \quad (2.9)$$

Questa espressione presenta un comportamento non lineare e limitato, e risulta particolarmente adatta a rappresentare fenomeni soglia nei quali la complessità rimane inizialmente contenuta, ma oltre un certo punto critico T cresce rapidamente, fino a saturarsi. Il parametro k determina la pendenza della curva, ovvero quanto rapidamente si passa dalla fase di bassa complessità a quella di congestione sistemica.

La funzione sigmoide risulta particolarmente adatta per descrivere sistemi soggetti a congestione oltre soglie critiche, dove i fenomeni di coordinamento collassano improvvisamente. La sua efficacia dipende fortemente dalla corretta identificazione dei parametri, che ne condiziona l'applicabilità e la precisione modellistica.

2.4.4 Funzione esponenziale

Proseguendo con l'analisi, tra le possibili rappresentazioni della complessità di interazione, la funzione esponenziale costituisce una delle formulazioni più estreme, in quanto prevede una crescita molto rapida del carico di coordinamento all'aumentare del numero totale di operatori. In particolare, la funzione:

$$f(H, R) = e^{\gamma(H+R)} \quad (2.10)$$

esprime un modello in cui anche piccoli incrementi nel numero di agenti producono un impatto significativo sulla complessità, soprattutto per valori elevati del parametro γ . Tale scelta può risultare coerente in ambienti operativi particolarmente sensibili a fenomeni di congestione o con una forte dipendenza da dinamiche interattive non lineari.

2.4.5 Funzione lineare

Una ulteriore formulazione particolarmente interessante per rappresentare la complessità di interazione in sistemi uomo-robot è quella lineare separata, espressa nella forma:

$$f(H, R) = \delta H + \varepsilon R \quad (2.11)$$

In questo caso, la complessità viene modellata come una somma pesata dei contributi individuali delle due categorie di operatori: gli umani, ponderati con il parametro δ , e i robot, ponderati con il parametro ε . Tale struttura consente di discriminare esplicitamente l'effetto che ciascun tipo di agente ha sulla complessità complessiva.

Nel complesso, questa funzione rappresenta una soluzione bilanciata e interpretativamente trasparente, particolarmente utile nei contesti in cui la distinzione tra operatori umani e robotici è cruciale per la corretta modellizzazione delle dinamiche di processo. Il suo utilizzo si presta bene anche come punto di partenza per raffinamenti successivi basati su osservazioni empiriche o simulazioni dettagliate.

2.4.6 Funzione lineare con interazione

Una estensione della funzione lineare prevede l'introduzione di un termine di interazione diretta tra operatori umani e robot, che consente di rappresentare in modo più realistico le complessità emergenti dalla loro collaborazione. La funzione assume così la forma:

$$f(H, R) = \delta H + \varepsilon R + \gamma HR \quad (2.12)$$

In questa configurazione, i parametri δ e ε mantengono il ruolo di pesi per il contributo lineare indipendente di ciascuna categoria di operatori, mentre il parametro γ misura l'effetto combinato e non lineare delle interazioni uomo-robot, tenendo conto della possibile insorgenza di difficoltà specifiche nella gestione congiunta del lavoro.

Nel complesso, la funzione lineare con termine di interazione rappresenta un buon compromesso tra espressività e controllo parametrico, ed è particolarmente indicata nei casi in cui si voglia modellare esplicitamente il contributo misto delle interazioni tra categorie differenti di operatori, senza rinunciare alla semplicità di base. La sua efficacia dipende dalla qualità del tuning dei parametri e dalla disponibilità di dati o simulazioni a supporto.

Al fine di rendere più immediato il confronto tra le diverse formulazioni proposte per la funzione di complessità di interazione, in Tabella 2 viene riportata una sintesi comparativa dei principali vantaggi e svantaggi associati a ciascun approccio modellistico.

Funzione	Vantaggi	Svantaggi
Elevamento a potenza	• Elevata flessibilità grazie al parametro γ • Semplicità formale e buona interpretabilità • Buona adattabilità empirica tramite calibrazione	• Non distingue tra operatori umani e robotici • Possibile sovrastima della complessità • Limitata capacità di rappresentare interazioni asimmetriche
Logaritmica	• Crescita moderata della complessità • Modellazione efficace di fenomeni di saturazione • Stabilità numerica e semplicità computazionale	• Non distingue tra tipologie di operatori • Possibile sottostima della complessità in sistemi altamente interattivi
Sigmoide	• Modellazione realistica di effetti soglia • Capacità di rappresentare fenomeni di congestione • Comportamento limitato e stabile	• Richiede calibrazione di più parametri • Possibile scarsa sensibilità nelle configurazioni con pochi operatori • Non distingue tra umani e robot
Esponenziale	• Elevata capacità di modellare scenari altamente instabili • Struttura matematica semplice	• Forte rischio di sovrastima della complessità • Possibile instabilità numerica in simulazione • Scarsa rappresentatività per sistemi reali moderatamente complessi
Lineare	• Permette distinzione tra contributo umano e robotico • Facilità interpretativa	• Non rappresenta interazioni tra agenti • Limitata capacità descrittiva in sistemi ad alta densità operativa
Lineare con interazione	• Introduce esplicitamente interazioni uomo-robot • Buon compromesso tra semplicità e capacità descrittiva • Elevata interpretabilità dei parametri	• Maggior numero di parametri da calibrare • Non modella direttamente effetti di saturazione o soglia

Tabella 2 - Vantaggi e svantaggi associati a ciascuna forma funzionale

In conclusione, le diverse formulazioni proposte per la funzione di complessità $f_c(H,R)$ riflettono approcci complementari alla modellizzazione delle difficoltà emergenti nei sistemi di collaborazione uomo-robot. Ogni forma funzionale presenta vantaggi e limiti specifici in termini di interpretabilità, capacità descrittiva, facilità di calibrazione e aderenza ai fenomeni reali, come l'effetto soglia, la saturazione, o la distinzione tra agenti di natura diversa. Le scelte modellistiche operate in questa fase costituiranno la base per le successive analisi sperimentali, dove un confronto tra le varie opzioni tramite simulazioni permetterà di individuare quale tra queste risulti più adatta a rappresentare la complessità di interazione nei diversi scenari produttivi analizzati.

Capitolo 3 – Caso studio: assemblaggio di uno skateboard completo

Il caso studio in esame riguarda il processo industriale di assemblaggio di uno skateboard finito, a partire dai singoli componenti elementari. La scelta di questo prodotto è determinata dalla relativa semplicità e linearità della sua struttura elementare e della sua componentistica, combinata con una sufficiente complessità che rende realistico il processo di assemblaggio da modellare. Il processo risulta di fatto caratterizzato da un numero considerevole di task elementari, vincoli di precedenza ed attività che possono essere svolte in parallelo.

3.1 La distinta base del prodotto (BOM)

Il punto di partenza per la modellazione del processo di assemblaggio è la definizione della distinta base (BOM, Bill of Materials) dello skateboard completo. La distinta base rappresenta la scomposizione gerarchica del prodotto finito nei suoi componenti elementari, con anche l'indicazione delle quantità di ciascuno di questi componenti necessarie per realizzare ciascuna unità di prodotto.

Nel caso in esame, la Bill of Materials mette in evidenza tre sottoinsiemi fondamentali:

- Il deck, che costituisce la piattaforma principale dello skateboard e la sua struttura portante, sulla quale l'utilizzatore si posiziona per effettuare lo spostamento.
- I truck, due per ogni skateboard, che rappresentano i complessi meccanici che collegano il deck alle ruote e ne determinano stabilità e manovrabilità. Ciascun truck è a sua volta costituito da diversi elementi, tra cui la base plate, il kingpin, le boccole e i dadi, che devono essere assemblati in modo sequenziale per ottenere un sottoinsieme funzionale.
- Le ruote, quattro per ogni skateboard, ciascuna montata sull'asse tramite i relativi dadi di fissaggio e i cuscinetti.

Si evidenzia come la presenza di due truck implichi la perfetta duplicazione di tutte le attività necessarie per l'assemblaggio di uno di questi ed anche di ciascuno dei sottocomponenti dei due complessi. Questo introduce naturalmente nel modello la possibilità di svolgere le due sequenze di sotto-assemblaggio contemporaneamente ed in parallelo, qualora il numero di risorse a disposizione lo permetta.

Il modello di distinta base costituisce quindi la base informativa essenziale per procedere con la modellazione del processo, chiarendo la struttura gerarchica del prodotto finito e determinando di conseguenza la numerosità delle operazioni da modellare.

Si riporta in Tabella 3 la Bill of Materials completa dello skateboard, che sintetizza i componenti, i codici identificativi e le quantità necessarie per la realizzazione di un'unità di prodotto completo, con lo scopo di fornire una visione immediata ed organica della struttura dello stesso.

Level	Code	Description	Quantity
1	B	Base plate	2
1	C1	Kingpin	2
1	RB	Top cup	2
1	GB	Bushing	2
1	RS	Lower cup	2
1	GS	Cone bushing	2
1	GN	Pivot bushing	2
1	A	Hanger	2
1	D1	Kingpin nut	2
1	C2	Deck bolt	8
1	D2	Deck nut	8
1	R	Wheel	4
1	D3	Axle nut	4
1	T	Skateboard deck	1
2	ST	Sub-assembled truck	2
3	CP	Complete product	1

Tabella 3 - Bill of Materials dello skateboard

3.2 I task di assemblaggio

La distinta base descrive in modo puntuale i componenti necessari alla realizzazione di uno skateboard completo, ma per costruire un modello di simulazione non è sufficiente conoscere gli elementi fisici che compongono il prodotto. Occorre tradurre quella struttura statica in una sequenza di attività operative che ne permettano l'assemblaggio. Da qui nasce la definizione dei task di assemblaggio, intesi come unità elementari di lavoro che, concatenandosi tra loro, consentono di ottenere progressivamente il prodotto finito.

Nel caso studio dello skateboard, il numero complessivo di task necessari per la sua costruzione è 108, i quali sono distribuiti lungo l'intero processo di montaggio ed il cui elenco completo è riportato in Appendice A.

Ciascun task è un'operazione necessaria per manipolare, allineare, fissare o verificare la correttezza delle operazioni effettuate in precedenza su un componente. e viene classificato in cinque categorie fondamentali individuate:

- *Identification and handling*: operazione di identificazione, manipolazione e movimentazione dei componenti.
- *Alignment*: in questa operazione i pezzi sono orientati in maniera precisa e preparati all'unione con altri elementi.
- *Joining*: è l'attività di fissaggio e assemblaggio vero e proprio.
- *Checking*: legato alla verifica della correttezza del montaggio effettuato in precedenza del componente
- *Adjustment*: operazione che comprende la regolazione finale e le operazioni di finitura.

Questa classificazione di tutti i 108 task fondamentali ha la funzione di semplificare notevolmente la descrizione del processo, riducendo l'eterogeneità delle operazioni di un contesto reale ad un insieme limitato di categorie.

È importante sottolineare che la riduzione dei 108 task elementari a cinque sole categorie operative non rappresenta una semplificazione eccessiva, né un indebolimento del modello. Al contrario questa scelta consente di eliminare dettagli che avrebbero appesantito la simulazione senza contribuire in modo significativo all'analisi. Le attività incluse in ognuna delle categorie presentate condividono infatti natura, obiettivi e modalità di esecuzione, tanto da poter essere considerate omogenee dal punto di vista dei tempi di lavorazione e dell'impatto sulle risorse. Grazie alla classificazione adottata è quindi possibile raggiungere un buon equilibrio tra fedeltà descrittiva rispetto ad un processo reale e gestibilità del modello di simulazione, permettendo di focalizzare maggiormente l'attenzione su aspetti molto più critici per l'analisi.

I task di assemblaggio non sono organizzati casualmente, ma si succedono in blocchi logici coerenti con la distinta base del prodotto. I primi task, in particolare, riguardano la costruzione dei due truck tramite i sottocomponenti elementari. Una volta completati, il processo prevede il fissaggio dei truck al deck mediante bulloni e dadi opportuni. Infine, si

procede con l'assemblaggio delle quattro ruote, due per ogni truck. Anche in questo caso, come per il montaggio dei due truck, qualora il numero di risorse coinvolte lo permetta, è possibile procedere in parallelo, in modo da garantire un tempo di processo inferiore.

Le attività di controllo e regolazione effettuate al termine dell'assemblaggio garantiscono la qualità del prodotto finale.

Un'ulteriore ipotesi assunta per l'implementazione del modello simulativo è che tutti i task appartenenti alla stessa categoria abbiano la stessa durata media, indipendentemente dal componente su cui vengono svolti. Anche in questo caso, la scelta è dovuta alla non necessità ai fini dell'analisi di introdurre un livello di dettaglio eccessivo, che avrebbe reso il modello troppo pesante senza apportare benefici sostanziali.

Tramite l'elenco dei 108 task si fornisce quindi, partendo dalla distinta base di prodotto, una trama logica del modello simulativo: grazie a questa scomposizione è possibile tradurre il processo reale in una sequenza di moduli in ambiente di simulazione.

3.3 Risorse impiegate nel processo di assemblaggio

Il modello costruito si è basato su due tipologie di risorse:

- Operatori umani o *HumanWorker*, caratterizzati da rapidità elevata e flessibilità, ma anche da minore precisione e conseguente maggiore probabilità di commettere errori ed a carichi ergonomici ben superiori.
- Robot collaborativi o *RobotWorker*, che operano meno rapidamente ma garantiscono ripetibilità e standardizzazione nello svolgimento dei task.

Entrambe le tipologie di risorse sono in grado di eseguire un unico task per volta.

La distinzione tra operatori umani e robot collaborativi non si limita al solo tempo di esecuzione dei task, ma coinvolge una pluralità di fattori che incidono sull'efficienza complessiva del sistema. Al fine di sintetizzare e confrontare in modo immediato le principali differenze operative tra operatori umani e robot collaborativi, si riporta di seguito una tabella comparativa delle caratteristiche considerate nella modellazione del sistema (Tabella 4).

Caratteristica	Operatori umani	Robot collaborativi
Rapidità di esecuzione	Generalmente più elevata grazie a destrezza manuale, esperienza e capacità di adattamento a situazioni variabili.	Generalmente inferiore a causa di traiettorie predefinite, vincoli di sicurezza e necessità di garantire elevata ripetibilità.
Tasso di errore	Maggiore probabilità di errore dovuta a fattori quali affaticamento, distrazione e condizioni ergonomiche variabili.	Minore probabilità di errore nelle attività ripetitive grazie alla costanza operativa e alla precisione esecutiva
Carico ergonomico	Sensibile a posture scomode, movimentazione di carichi e condizioni operative gravose, con possibili impatti su velocità e qualità del lavoro.	Poco influenzato dal carico ergonomico, risultando particolarmente adatto allo svolgimento di operazioni fisicamente gravose.

Tabella 4 - Comparazione delle caratteristiche considerate per operatori umani e robotici

Per modellare la differenza in termini di rapidità di esecuzione tra operatori umani e robot collaborativi, si è adottata l'ipotesi che il tempo di esecuzione di un robot sia pari al tempo umano moltiplicato per un fattore costante maggiore di uno, indipendentemente dalla tipologia di task.

Al fine di integrare i tre aspetti discussi, nella simulazione è stato introdotto un indice di performance Q che li combina opportunamente. Come verrà approfondito nelle sezioni successive, la scelta della risorsa da assegnare a ciascun task non dipende esclusivamente dal tempo di esecuzione, ma anche dalla probabilità di commettere errori di lavorazione e dal carico ergonomico. L'indice consente quindi di modellare in modo realistico i compromessi che emergono dalla collaborazione uomo-robot e di valutare scenari differenti in termini di velocità, stabilità e affidabilità del processo di produzione dello skateboard.

Un'ulteriore caratteristica del modello riguarda la possibilità di assegnazione dei task alle due distinte tipologie di risorse: i task di tipo *joining*, *checking* ed *adjustment* sono stati considerati eseguibili esclusivamente da operatori umani, in quanto richiedono una combinazione di sensibilità manuale, capacità di valutazione qualitativa e flessibilità operativa che i robot collaborativi non sono ancora in grado di replicare efficacemente. I task di tipo *handling* e *alignment*, invece, possono essere svolti sia da operatori umani sia da robot, con tempi differenziati secondo quanto definito sopra.

Capitolo 4 – Approccio basato sulla simulazione

4.1 Motivazione della scelta della simulazione come metodo di analisi

L'analisi dei sistemi produttivi complessi, come il caso studio di assemblaggio oggetto della presente tesi, richiede strumenti in grado di catturare l'elevato numero di variabili coinvolte e la dinamicità delle loro interazioni. La presenza di un numero elevato di task elementari, l'interazione tra risorse eterogenee di diversa natura e la necessità di coordinamento tra esse, unitamente alla gestione dei vincoli di precedenza e alla possibilità di parallelizzazione di alcune attività, rendono tali sistemi difficilmente trattabili mediante strumenti matematici semplificati e approcci puramente analitici.

Un primo limite degli approcci deterministici risiede nella loro incapacità di rappresentare la variabilità stocastica che caratterizza i processi reali. I tempi di esecuzione delle operazioni non sono mai costanti: variano in funzione delle condizioni ergonomiche, della tipologia di risorsa, di piccoli imprevisti o di fattori ambientali. Analogamente, il tasso di errore non è un parametro fisso, ma può dipendere dal carico di lavoro accumulato, dalla ripetitività dei compiti e dall'esperienza dell'operatore. Tentare di rappresentare queste dinamiche tramite formule deterministiche condurrebbe inevitabilmente a una semplificazione eccessiva, con il rischio di produrre stime poco affidabili.

Un secondo limite è legato alla complessità delle decisioni di allocazione. In ogni istante, ciascun task può teoricamente essere svolto da un operatore umano o da un robot, a seconda di temi, errori ed ergonomia. Questo comporta che il numero di possibili configurazioni cresce esponenzialmente con il numero di task, rendendo impraticabile la risoluzione mediante algoritmi di ottimizzazione classica, se non al prezzo di drastici compromessi modellistici.

Anche l'approccio sperimentale diretto, ovvero la realizzazione di prove fisiche di assemblaggio con differenti mix di risorse, non rappresenta una soluzione praticabile. Realizzare prototipi e organizzare sessioni di lavoro ripetute comporterebbe infatti costi elevati e tempi lunghi, senza garantire la ripetibilità necessaria per confronti sistematici. Inoltre, l'introduzione di variabilità controllata (come la modifica del numero di risorse o dei tempi medi di esecuzione) risulterebbe difficilmente realizzabile in un ambiente reale, mentre diventa naturale in un contesto simulativo.

Alla luce di queste considerazioni, risulta naturale adottare la simulazione a eventi discreti come strumento di analisi. Tale approccio consente di rappresentare in modo accurato i processi produttivi, caratterizzati da entità, risorse, code e attività, riproducendo la logica operativa con cui le operazioni vengono effettivamente eseguite in un contesto reale.

Un ulteriore vantaggio dell'applicazione di un approccio simulativo consiste nella possibilità di introdurre variabilità stocastica nei parametri in input: risulta di fatto possibile modellare i tempi di processo (ed anche i tempi di inter-arrivo tra un'entità ed un'altra) con distribuzioni probabilistiche note (come la normale o la triangolare) ed introdurle anche nella generazione di errori. I risultati ottenuti saranno di conseguenza più realistici e terranno conto non solo dei tempi medi, ma anche della loro variabilità.

Un ulteriore punto di forza risiede nella possibilità di ottenere rapidamente risultati in diversi scenari what-if. Una volta costruito il modello di riferimento, risulta infatti relativamente semplice variare il numero di operatori umani o robotici, in linea con l'obiettivo principale della tesi, nonché modificare i tempi di esecuzione o le politiche di allocazione delle risorse, al fine di osservare come tali fattori influenzino il tempo complessivo di assemblaggio.

La simulazione si rivela inoltre uno strumento ideale per collegare i modelli matematici a risultati concreti. Attraverso il modello è infatti possibile implementare logiche decisionali dinamiche, in cui ogni task seleziona la risorsa più idonea in base a fattori opportunamente definiti, consentendo di osservare l'impatto di tali scelte sull'evoluzione temporale del processo.

La simulazione, infine, consente di ottenere output sintetici e facilmente comparabili, come il tempo medio di attraversamento, i tempi medi di attesa in coda o il numero di task eseguiti per tipologia di operatore, riducendo drasticamente il tempo e le difficoltà che sarebbero necessarie per l'ottenimento di una raccolta dati comparabile effettuata in un contesto reale.

Il contesto selezionato per l'analisi risulta sufficientemente realistico rispetto agli obiettivi dello studio e, di conseguenza, caratterizzato da un elevato livello di complessità e variabilità, nonché dalla presenza di numerosi vincoli di precedenza. L'approccio simulativo risponde all'esigenza di modellare tale complessità e di fornire un adeguato potere predittivo, configurandosi come la scelta più naturale per comprendere le dinamiche del processo e dell'interazione tra risorse di diversa natura.

4.2 Panoramica dello strumento di simulazione (Arena)

Arena è uno dei principali software di simulazione a eventi discreti utilizzati in ambito manifatturiero, o più in generale per lo studio di sistemi complessi. Lo strumento offre elevata flessibilità, interfaccia grafica molto intuitiva e la possibilità di integrare in un unico ambiente la costruzione del modello, la raccolta delle statistiche e l'analisi dei risultati ottenuti.

Nella simulazione ad eventi discreti, caratteristica di *Arena*, l'evoluzione temporale del sistema non è continua, ma scandita dal verificarsi di eventi specifici puntuali, di conseguenza le variabili coinvolte modificano il loro valore solo in determinati istanti. In questo approccio, lo stato del sistema viene aggiornato solo quando si verifica un evento rilevante, come l'arrivo di una nuova entità, l'inizio di un'attività, il rilascio di una risorsa o il completamento di un task. Tra due eventi successivi, il sistema rimane invariato (Law, 2024).

A differenza delle simulazioni a tempo continuo, che richiedono la risoluzione numerica di equazioni differenziali, la simulazione a eventi discreti si concentra esclusivamente sugli istanti in cui lo stato cambia, riducendo così la complessità computazionale mantenendo la capacità descrittiva del modello (J. Banks, 2010).

Dal punto di vista operativo, *Arena* si presenta come un ambiente modulare in cui il modello viene costruito tramite blocchi funzionali predisposti, organizzati per categorie. Ciascun blocco corrisponde a un'operazione elementare del processo (creazione di entità, assegnazione di risorse, decisione logica, elaborazione) e può essere collegato agli altri in sequenza o in parallelo per riprodurre fedelmente la logica reale del sistema. Questa struttura a blocchi, pur essendo di immediata comprensione visiva, si basa su un motore di simulazione robusto, che consente di trattare variabilità stocastica, risorse limitate, logiche decisionali condizionate e flussi paralleli.

Uno dei punti di forza di *Arena* è la sua capacità di integrare la fase di modellazione con quella di analisi: i risultati della simulazione vengono prodotti sotto forma di report standardizzati, che includono misure di prestazione come tempi di attraversamento, lunghezza media delle code e grado di utilizzo delle risorse. A ciò si aggiunge la possibilità di definire statistiche personalizzate tramite appositi moduli, consentendo di adattare l'analisi alle esigenze specifiche del caso di studio.

Per queste caratteristiche, *Arena* è stato ritenuto lo strumento più idoneo a supportare l'analisi condotta in questa tesi, in quanto consente di mappare con precisione il processo di

assemblaggio oggetto di studio, introdurre logiche dinamiche di allocazione delle risorse e valutare in modo sistematico e controllato l'impatto di scenari alternativi.

4.3 Interfaccia del software Arena

Prima di introdurre i concetti fondamentali della simulazione a eventi discreti, è utile descrivere brevemente l'interfaccia di *Arena*, poiché essa rappresenta l'ambiente operativo attraverso cui il modellatore traduce in pratica le logiche del sistema da analizzare. L'interfaccia grafica di *Arena* costituisce il punto di contatto essenziale tra il modellatore e l'ambiente di simulazione, e rappresenta uno degli elementi che ne hanno decretato la diffusione sia in ambito accademico sia in ambito industriale. Essa è stata progettata per coniugare due esigenze apparentemente contrastanti: da un lato, la necessità di immediatezza visiva, che consente anche a chi non possiede conoscenze approfondite di linguaggi di programmazione di costruire modelli significativi attraverso un approccio grafico; dall'altro, l'esigenza di garantire un livello elevato di controllo e configurabilità, indispensabile per tradurre in modo fedele la complessità dei sistemi reali e per condurre analisi rigorose. Questa duplice natura si riflette nella struttura dell'interfaccia, semplice ed intuitiva nell'aspetto ma molto potente in quanto a possibilità di personalizzazione.

Per l'utente ne deriva la possibilità di procedimento incrementale, partendo da un numero minimo di semplici blocchi collegati tra loro con un successivo arricchimento del modello tramite l'introduzione di attributi, variabili, logiche decisionali e raccolta di statistiche. La curva di apprendimento è dunque relativamente accessibile, ma senza sacrificare la profondità di analisi, che rimane paragonabile a quella di ambienti di programmazione specialistici.

Un ulteriore punto di forza dell'interfaccia è la sua capacità di favorire la coerenza concettuale. La rappresentazione grafica del flusso non è infatti una semplice animazione, ma la vera e propria logica del modello: i collegamenti tra i moduli corrispondono alle relazioni causali tra gli eventi discreti, e il movimento delle entità durante la simulazione riflette il comportamento dinamico del sistema. L'approccio consente di ridurre notevolmente il rischio di discrepanze tra modello concettuale e modello implementato, fornendo all'interfaccia un'ulteriore funzione: quella di mezzo di comunicazione tra il modellatore ed i destinatari dell'analisi.

Per ogni modulo *Arena* offre la possibilità di accedere a pannelli di configurazione dettagliati e ciò rende il software adatto anche a modellatori esperti, che possono definire in modo preciso i parametri di ogni fase del processo. Ciò consente, ad esempio, di passare

rapidamente da un modello deterministico a uno stocastico introducendo distribuzioni di probabilità, oppure di implementare logiche condizionali complesse attraverso attributi e variabili. Di conseguenza, *Arena* non si limita a offrire un'interfaccia “user-friendly”, ma fornisce un ambiente che cresce con l'utente: accessibile per chi muove i primi passi nella simulazione, ma sufficientemente sofisticato da supportare studi avanzati e progetti di ricerca (Kelton, 2015).

L'interfaccia di *Arena* non è quindi un elemento secondario o esclusivamente estetico, ma il fulcro che permette di costruire, comprendere e comunicare il modello simulativo. La sua capacità di integrare semplicità operativa e profondità configurativa la rende uno strumento efficace per rappresentare sistemi caratterizzati da elevata complessità, come quello oggetto del presente studio (Figura 8).

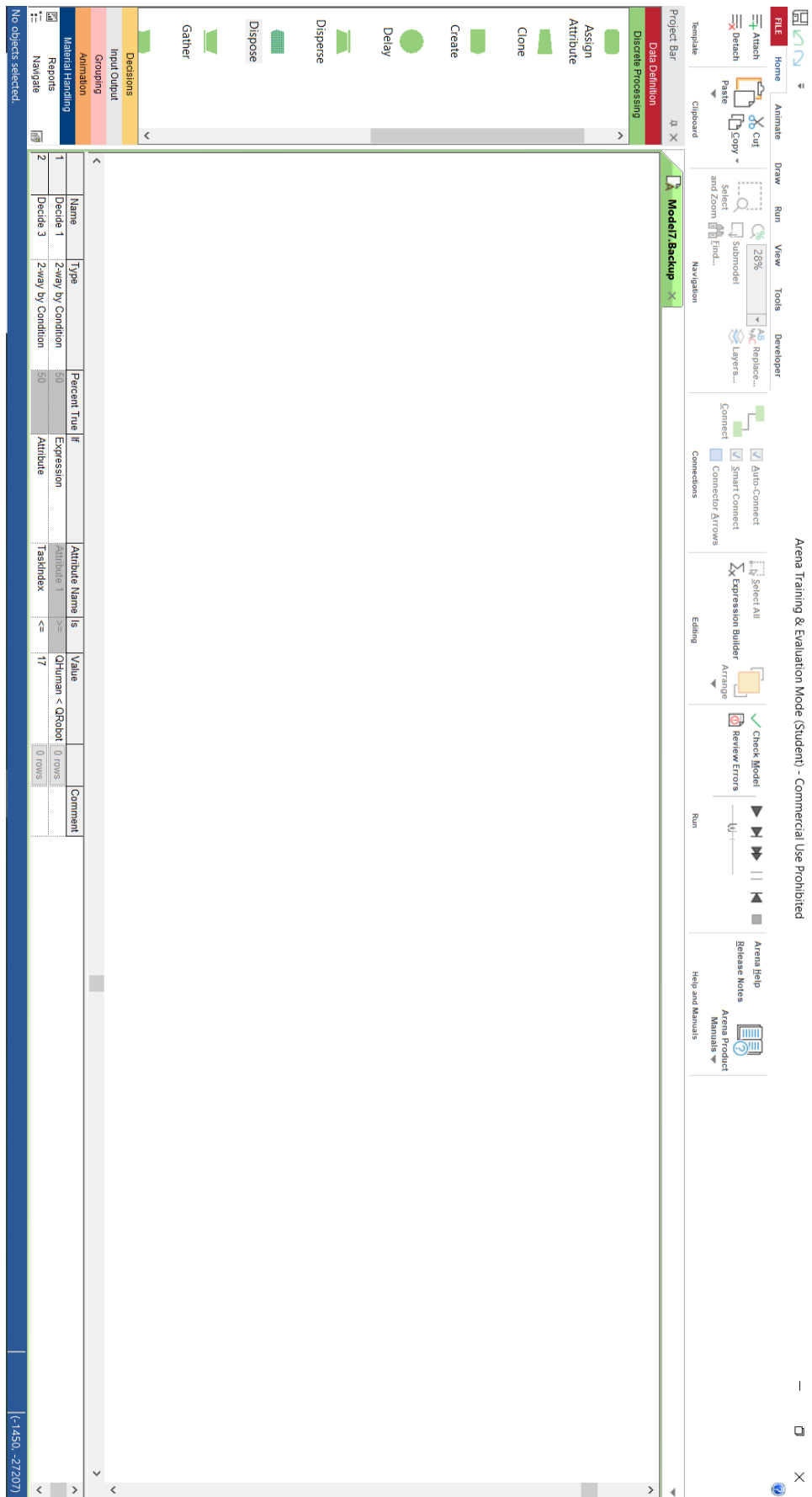


Figura 8 - Interfaccia di Arena (fonte: Arena Simulation Software)

Ci si concentrerà ora sull'approfondire ciascuno dei componenti in cui l'interfaccia visiva del software in oggetto è suddivisa, in maniera da poterne comprendere e descrivere le funzioni principali.

4.3.1 Project bar

Per comprendere in che modo *Arena* consenta di tradurre i concetti teorici della simulazione a eventi discreti in un modello operativo, è utile introdurre la *Project Bar*, ossia la barra laterale che raccoglie i diversi ambienti di lavoro e i moduli disponibili.

Ogni sezione della *Project Bar* contiene strumenti specializzati che rispondono a specifiche esigenze della costruzione del modello. La sua organizzazione modulare rende possibile affrontare sistemi di qualsiasi complessità senza perdere chiarezza, poiché i moduli sono raggruppati in categorie logiche coerenti.

La Figura 9 mostra un esempio della *Project Bar* nella configurazione adottata.

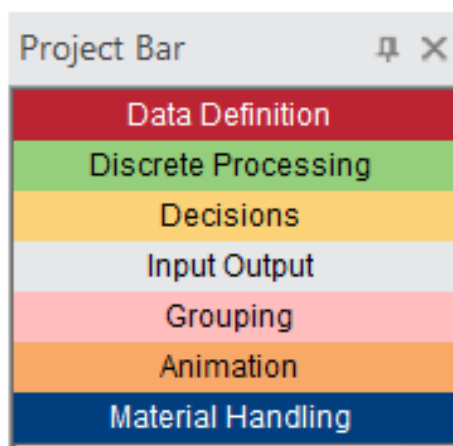


Figura 9 - Project Bar (fonte: Arena Simulation Software)

Si osserva come le diverse librerie siano immediatamente accessibili e come ciascuna possa essere espansa per rivelare i moduli in essa contenuti. La presenza di ambienti distinti riflette l'approccio di *Arena* alla modellazione: ogni aspetto del sistema, come dati di base, logica di processo, decisioni, raccolta dei risultati, animazione, è gestito in un contesto dedicato, ma coerente con gli altri.

Questa organizzazione consente al modellatore di mantenere una chiara separazione tra struttura statica (quindi definizione di entità e risorse) e dinamica di processo (quindi i moduli di flusso).

La *Project Bar* rappresenta quindi il punto di accesso a tutte le componenti della simulazione e costituisce la base pratica da cui prende forma il modello.

4.3.2 Barra superiore dei menu e degli strumenti

Collocata nella parte superiore della finestra di *Arena*, la *barra degli strumenti* rappresenta il centro di comando del software. Essa raccoglie tutte le funzioni di gestione, configurazione e monitoraggio necessarie per l'esecuzione corretta del modello simulativo, oltre che, più in generale, per il suo sviluppo. L'organizzazione visiva è molto simile a quella dei più comuni software di produttività, in modo da consentire un apprendimento più rapido ad un numero elevato di utenti. I due principali livelli in cui la barra è articolata sono il menu principale, che con le funzioni più generali *File*, *Home*, *Run*, *Tools*, consente l'accesso a varie opzioni fondamentali in fase di modellazione e la toolbar rapida, ovvero un insieme di icone grafiche che forniscono accesso diretto ai comandi più frequenti, riducendo i tempi operativi durante lo sviluppo.

Tra funzionalità più rilevanti offerte dalla sezione vi è la configurazione della simulazione, che consente di stabilire i parametri fondamentali di *Run*, come la durata della simulazione, il numero di repliche o le condizioni di terminazione. Tali impostazioni sono di primaria importanza perché incidono direttamente sulla significatività statistica dei risultati ottenuti: ad esempio, un numero insufficiente di repliche potrebbe condurre a conclusioni poco rappresentative.

È possibile inoltre avviare, sospendere o arrestare la simulazione, ma anche eseguirla passo passo, possibilità estremamente utile in fase di debug, poiché consente di controllare in modo specifico il comportamento delle entità coinvolte in ogni istante e di conseguenza individuare eventuali criticità del modello che ne impediscono il corretto funzionamento. *Arena* consente inoltre di variare la velocità di esecuzione: permettendo di valutare scenari di lungo periodo accelerandola, o rallentarla in modo da osservare nel dettaglio singoli eventi.

Sono disponibili anche strumenti grafici di disposizione e allineamento, che consentono di mantenere il modello leggibile anche quando è presente un numero molto elevato di moduli, rendendo la rappresentazione visibile anche nella successiva presentazione dei risultati.

La barra superiore svolge anche un ruolo cruciale nella fase di analisi dei risultati, poiché consente l'accesso diretto ai report statistici generati da *Arena* al termine di ciascun *Run*. Attraverso questi report è possibile esaminare indicatori chiave come tempi medi di attesa, utilizzo delle risorse o throughput del sistema. I dati possono inoltre essere esportati e ulteriormente elaborati in software esterni, così da integrare la simulazione con strumenti di analisi quantitativa avanzata.

Infine, un'ulteriore funzionalità implementata in *Arena* tramite la barra degli strumenti è la gestione delle animazioni, grazie alla quale è direttamente controllabile la modalità con cui entità e risorse sono rappresentate graficamente durante la simulazione. Attraverso semplici comandi è possibile attivare o disattivare indicatori animati o personalizzare la rappresentazione di operatori e macchine. Queste funzioni non influenzano il comportamento del modello ma risultano comunque importanti in fase di comunicazione dei risultati.

La struttura della barra degli strumenti la rende uno strumento che integra la dimensione grafica, quella logica e quella analitica della simulazione, configurandosi quindi come un punto di accesso centrale alle principali funzionalità di modellazione, esecuzione e analisi (Figura 10).

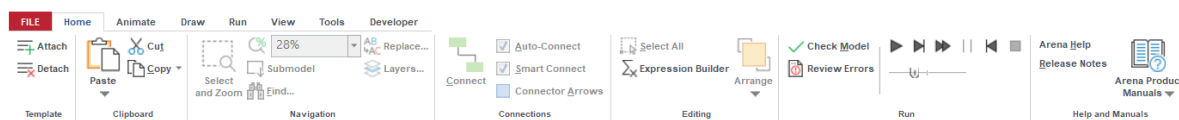


Figura 10 - Barra degli strumenti (fonte: Arena Simulation Software)

4.3.3 L'ambiente centrale di modellazione

Fulcro fondamentale dell'interfaccia di *Arena* è l'*ambiente centrale di modellazione*, nonché lo spazio di lavoro all'interno del quale il modello vero e proprio prende forma in maniera visiva e operativa. Il modellatore, in particolare, trascina i moduli necessari alla realizzazione del proprio progetto direttamente dalla *Project Bar* a questa area, disponendoli in sequenza grazie a connessioni e realizzando così un flusso ben preciso che riproduce le operazioni del sistema reale.

Arena, a differenza di molti altri software che separano in maniera molto netta la programmazione dalla rappresentazione grafica, integra le due cose, rendendo la costruzione visiva la vera e propria logica del modello: in particolare, ogni collegamento ed ogni blocco non sono semplici illustrazioni, ma elementi funzionali che definiscono concretamente il comportamento della simulazione.

L'ambiente centrale può essere concepito come un diagramma di flusso dinamico. Ogni modulo inserito corrisponde a un'azione o a un evento, mentre i collegamenti tra i moduli rappresentano le transizioni che le entità possono seguire.

Questa impostazione prettamente grafica rende il processo di costruzione del modello molto più intuitivo ed aiuta nel mantenimento di una corrispondenza il più chiara possibile tra il modello concettuale e quello implementato. L'utente può infatti osservare come il percorso

logico delle entità venga rappresentato in modo esplicito: dall'ingresso nel sistema, attraverso i diversi nodi di elaborazione e decisione, fino all'uscita finale.

Uno dei punti di forza principali dell'ambiente centrale è la possibilità di adottare un approccio incrementale e modulare. Il modellatore può iniziare costruendo un semplice schema lineare, come ad esempio *Create*, *Process*, *Dispose*, moduli utilizzati per rappresentare la nascita, l'elaborazione e la terminazione delle entità che saranno approfonditi più avanti. Successivamente, può arricchire il modello con ulteriori elementi, come condizioni logiche tramite il modulo *Decide* o rami paralleli utilizzando *Separate* e *Batch*. Questa logica progressiva permette di gestire anche sistemi molto complessi senza perdere il controllo della struttura complessiva, perché ogni aggiunta è costruita a partire da un nucleo semplice e chiaro.

Un ulteriore vantaggio della sezione centrale dell'interfaccia è rappresentato dalla disponibilità di numerose funzioni di supporto grafico, che contribuiscono a migliorare la leggibilità del modello. I moduli possono essere spostati, riallineati e raggruppati in modo da organizzare in maniera chiara la struttura del processo. È inoltre possibile inserire etichette, note testuali o aree evidenziate per distinguere visivamente le diverse fasi del flusso operativo.

L'ambiente centrale risulta essere anche il luogo in cui le entità vengono animate durante l'esecuzione della simulazione: queste appaiono, si muovono lungo i collegamenti, attendono in coda e vengono processate nei moduli opportuni. In questo modo si è in grado di osservare ed analizzare l'andamento temporale del sistema, al fine di individuarne colli di bottiglia, eventuali congestioni o risultati mostrati nella reportistica apparentemente inspiegabili.

Infine, si osserva come l'ambiente centrale dell'interfaccia grafica del software sia un punto di convergenza tra le diverse altre sezioni: i moduli inseriti provengono dalla *Project Bar*, mentre le loro proprietà dettagliate sono configurate nel pannello inferiore. Tutte le parti dell'interfaccia svolgono un ruolo distinto, ma ognuna si integra nell'ambiente centrale, dove la logica del sistema prende effettivamente forma in modo visivo e coerente. In questa sezione i concetti teorici della simulazione a eventi discreti si traducono in sequenze concrete di operazioni, consentendo al modellatore di modificare e perfezionare il proprio modello (Figura 11).

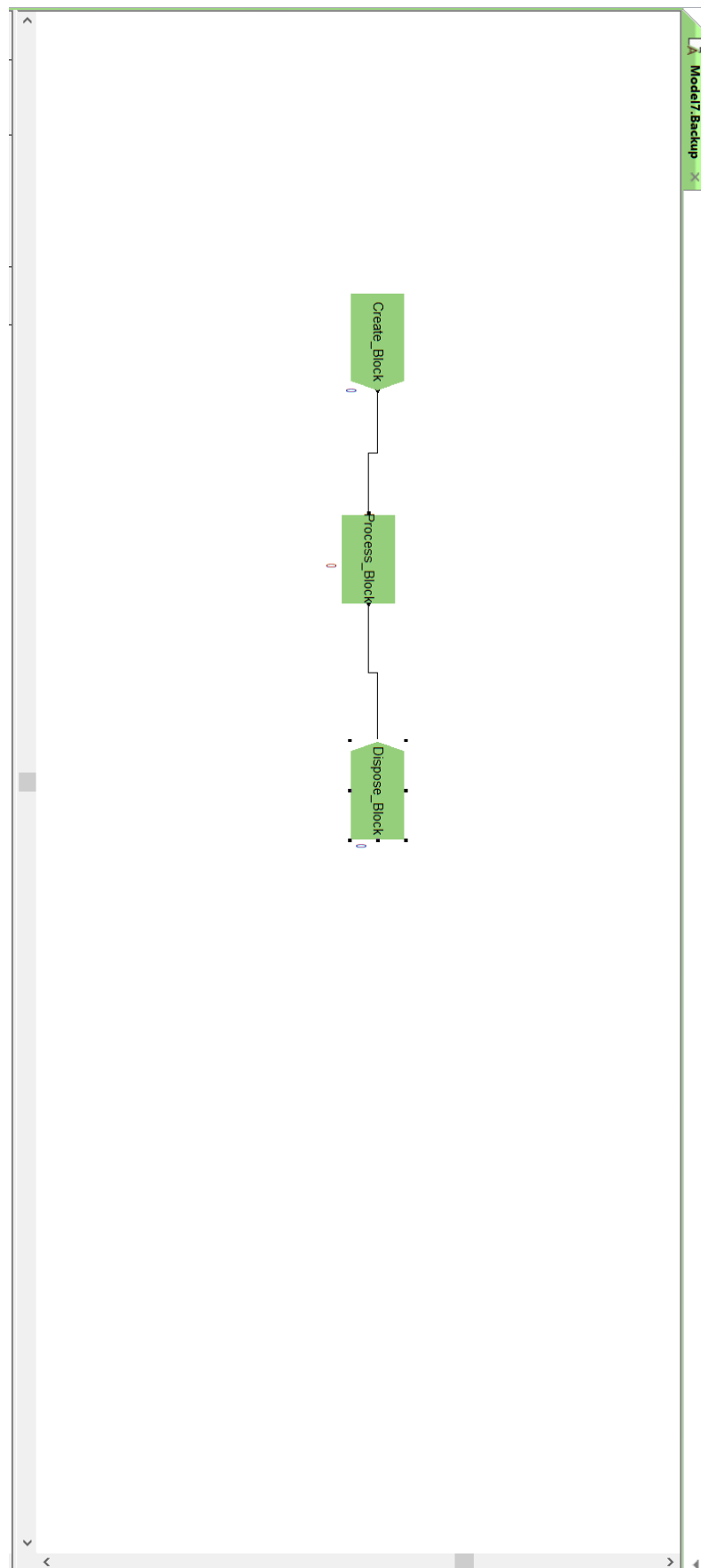


Figura 11 - Ambiente centrale di modellazione (fonte: Arena Simulation Software)

4.3.4 Il pannello inferiore delle proprietà

Il *pannello inferiore delle proprietà* è la sezione dell'interfaccia di *Arena* in cui il modellatore entra effettivamente nel dettaglio dei moduli posizionati nello spazio centrale, rappresentando il luogo in cui ciascuno di questi moduli è analizzabile e personalizzabile in modo approfondito in relazione alle esigenze del sistema che si vuole simulare (Figura 12). In questa sezione, selezionando un blocco, vengono visualizzati i campi ad esso associati, compilabili con valori, condizioni logiche o distribuzioni statistiche che determinano in modo concreto il comportamento del blocco durante la simulazione.

Le specifiche proprietà che caratterizzano un modulo dipendono anche dalla tipologia del modulo stesso. Ad esempio, in un blocco *Process* è necessario indicare il tempo di lavorazione come parametro, in modo che la logica *seize-delay-release* possa operare correttamente; inoltre, risulta necessario specificare anche quali siano le risorse che saranno impiegate. In un modulo *Decide*, invece, risulta necessario, specificare le condizioni logiche o probabilistiche che guidano l'instradamento delle entità.

Risulta evidente come il pannello inferiore non rappresenti una componente accessoria, bensì un vero e proprio strumento di modellazione. Per attribuire un significato logico e operativo a ciascun modulo, è infatti necessario definire un'adeguata configurazione attraverso il pannello inferiore delle proprietà.

Il pannello delle proprietà è quindi l'ambiente di *Arena* dove le ipotesi teoriche sul sistema vengono tradotte in regole logiche e matematiche. Le scelte fatte a livello concettuale sono qui formalizzate ed esplicitate adeguatamente. È essenziale che tale passaggio da modello concettuale a modello logico sia effettuato con precisione, in quanto ciò ha un impatto molto diretto sulla qualità dei risultati.

Infine, il pannello inferiore svolge anche un ruolo importante nella fase di validazione e nella presentazione del modello. Mostrare i parametri inseriti rende trasparenti le assunzioni di partenza: questo ambiente consente, insieme al pannello centrale, di ottenere un modello pienamente leggibile e comunicabile, perché esplicita non solo la struttura logica del processo, ma anche i criteri numerici e statistici che lo governano.

Il pannello inferiore delle proprietà, quindi, completa le altre parti dell'interfaccia: è in questa sezione che il modello acquisisce la precisione necessaria per riflettere fedelmente il sistema reale.

	Name	Entity Type	Type	Value	Units	Entities per Arrival	Max Arrivals	First Creation	Comment
1	Start	Entity 1	Constant	1	Minutes	1	10	0.0	
2 ▶	Create_	Entity 1	Random (Expo)	1	Hours	1	Infinite	0.0	

Figura 12 - Pannello inferiore delle proprietà (fonte: Arena Simulation Software)

4.4 Concetti base della simulazione a eventi discreti

Il funzionamento di un modello ad eventi discreti, e in particolare di un modello realizzato in *Arena* si fonda su dei concetti teorici fondamentali, che costituiscono i mattoni di tutte le simulazioni realizzabili. In particolare, in *Arena* è fattibile creare sistemi anche molto articolati e variabili, ciò è reso possibile grazie alla combinazione opportuna di questi concetti elementari.

- **Entità:** gli elementi dinamici che “percorrono” il modello, nonché gli oggetti che si muovono nel sistema, subiscono trasformazioni, attendono in coda ed occupano le risorse.

Nel caso di studio analizzato, le entità principali sono gli skateboard in fase di assemblaggio, tuttavia, durante le fasi del processo è possibile distinguere anche delle sotto-entità temporanee che vengono generate tramite operazioni di *Batch* o di *Merge*, le quali rappresentano i sottocomponenti principali del prodotto finale, come i due truck o le quattro ruote.

- **Attributi:** ogni entità può essere caratterizzata da attributi, ossia variabili locali che accompagnano l’entità lungo l’intero percorso. Gli attributi permettono di differenziare il comportamento delle entità in base a proprietà specifiche (ad esempio il tipo di task, il tempo base di esecuzione, l’indice Q per l’allocazione della risorsa). In *Arena* gli attributi sono essenziali per introdurre logiche di decisione dinamica, poiché consentono a ciascuna entità di portare con sé le informazioni necessarie a scegliere la via successiva o la risorsa più adeguata.
- **Risorse:** rappresentano gli elementi che compiono un’operazione durante il processo. La loro caratteristica fondamentale è la capacità, di fatto ogni risorsa è in grado di servire un numero limitato di entità alla volta. Nel caso studio, le risorse sono i lavoratori umani (*HumanWorker*) e i robot collaborativi (*RobotWorker*), ciascuno con capacità unitaria, potendosi dedicare ad una sola operazione per volta.

La dinamica di allocazione delle risorse è governata dalle operazioni di *seize* (impegno della risorsa), *delay* (tempo di utilizzo) e *release* (rilascio), che costituiscono il nucleo operativo della simulazione a eventi discreti.

- *Code*: punti di accumulo delle entità in attesa di una risorsa non immediatamente disponibile. Esse svolgono un ruolo cruciale nella rappresentazione realistica di un sistema, perché modellano fenomeni tipici come la congestione, l'attesa e lo sbilanciamento tra capacità e domanda. La lunghezza media delle code e i tempi medi di attesa costituiscono indicatori diretti dell'efficienza del sistema e rappresentano parametri di output chiave nei report di simulazione.
- *Eventi*: una modifica di stato che si verifica in un istante puntuale del tempo simulato, come l'arrivo di una nuova entità, l'inizio o la fine di un processo, o il rilascio di una risorsa. Come anticipato, la simulazione a eventi discreti passa da un evento rilevante al successivo, senza rappresentare gli intervalli temporali in maniera continua. Lo stato del sistema viene infatti aggiornato solo quando è necessario, riducendo l'elaborazione superflua e consentendo di simulare anche processi molto complessi con tempi computazionali contenuti.
- *Stati del sistema*: descritti in ogni istante dalla combinazione di entità, risorse ed eventi. In particolare, lo stato è definito dal numero di entità presenti in ciascun nodo, dal livello di occupazione delle risorse, dalla lunghezza delle code e dai valori degli attributi. Il passaggio da uno stato all'altro è governato dagli eventi discreti, che ne determinano l'evoluzione nel tempo.
- *Variabilità*: come definito in precedenza, è possibile introdurre variabilità nel sistema attraverso distribuzioni di probabilità. I tempi di processo, ad esempio, possono essere modellati come variabili casuali piuttosto che come valori deterministici, in modo da catturare la natura incerta e variabile dei sistemi reali. Allo stesso modo, le decisioni possono essere influenzate da probabilità, introducendo biforcazioni stocastiche nel percorso delle entità.

Ora che si sono introdotti i principali concetti necessari alla comprensione approfondita della modellazione in *Arena*, è opportuno soffermarsi ulteriormente su uno degli strumenti più importanti messi a disposizione dal software: la *Project Bar*. Situata, come specificato in precedenza, sul lato sinistro dell'interfaccia, essa rappresenta la libreria dei moduli, ovvero l'insieme degli strumenti operativi attraverso cui il modellatore costruisce la logica del sistema. La sua importanza è determinata dall'organizzazione della struttura concettuale

della simulazione. Ognuna delle librerie contenute nella *Project bar* corrisponde di fatto ad un diverso ambito del processo di modellazione, come la definizione dei dati o la costruzione del flusso (Kelton, 2015).

4.4.1 Data definition

Nella costruzione di un modello a eventi discreti con *Arena*, un ruolo centrale è svolto dall'ambiente di *Data Definition*, considerabile la base di dati del modello. Se i concetti teorici descritti in precedenza di entità, risorse, code, eventi e variabilità costituiscono la base concettuale della simulazione, la Data Definition rappresenta lo strumento attraverso cui questi concetti vengono formalizzati e resi disponibili per la successiva modellazione operativa.

In questa specifica sezione del software vengono definiti gli elementi chiave del sistema, quindi le entità, le risorse, gli attributi, le variabili ed altri concetti.

Una configurazione accurata in questa fase è essenziale, perché permette di tradurre i concetti astratti in elementi concreti e gestibili, ponendo le basi per la costruzione del flusso logico vero e proprio. Grazie a questa distinzione, *Arena* separa le definizioni statiche dalle dinamiche di processo che verranno introdotte in seguito tramite moduli (Figura 13).

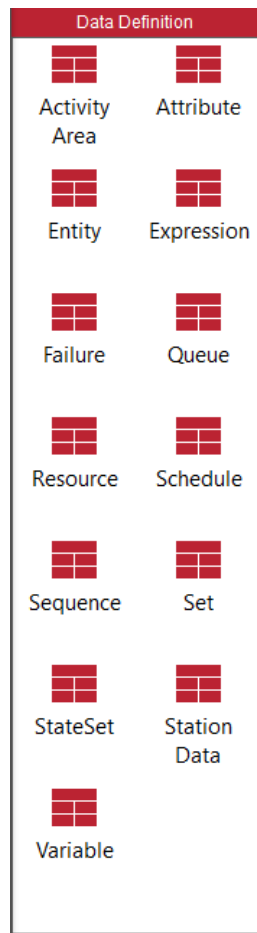


Figura 13 - Data definition (fonte: Arena Simulation Software)

A titolo esemplificativo, la Figura 14 riporta l'ambiente di definizione delle risorse utilizzato nel modello. Nello specifico, è mostrata la configurazione della risorsa denominata *HumanWorker*, impostata come risorsa a capacità fissa (*Fixed Capacity*) con valore pari a 1. Questa scelta riflette la presenza di un unico operatore umano. L'alternativa disponibile in *Arena*, ovvero la definizione *Based on Schedule*, non è stata adottata, poiché non si è ritenuto opportuno introdurre una pianificazione oraria della disponibilità delle risorse. La schermata evidenzia inoltre i valori predefiniti relativi a *Busy per hour*, *Idle per hour* e *Per use*, impostati a zero in quanto non pertinenti agli obiettivi del caso studio.

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics
1	HumanWorker	Fixed Capacity	1	0.0	0.0	0.0		0 rows	☒

Figura 14 - Definizione delle risorse (fonte: Arena Simulation Software)

4.4.2 Discrete processing

In *Arena*, la costruzione di un modello avviene attraverso l'impiego di moduli, ossia blocchi funzionali predefiniti che rappresentano operazioni elementari di un sistema a eventi discreti. Ciascun modulo corrisponde a una specifica azione o logica: dalla creazione di nuove entità,

alla loro movimentazione, al sequestro di risorse, fino alla registrazione di dati statistici. I moduli possono essere collegati in sequenza o in parallelo, e la loro combinazione consente di costruire flussi logici di qualsiasi complessità, pur mantenendo leggibilità e coerenza con i concetti teorici della simulazione.

L'approccio modulare ha un duplice vantaggio: da un lato permette di tradurre in forma grafica i concetti della simulazione a eventi discreti (entità, risorse, code, eventi), dall'altro garantisce flessibilità, poiché ogni modello può essere esteso o modificato semplicemente aggiungendo o riorganizzando moduli. I moduli risultano quindi delle unità elementari che combinate opportunamente permettono di riprodurre sistemi complessi e sono disponibili nella libreria *Discrete Processing* della *Project Bar* (Figura 15).

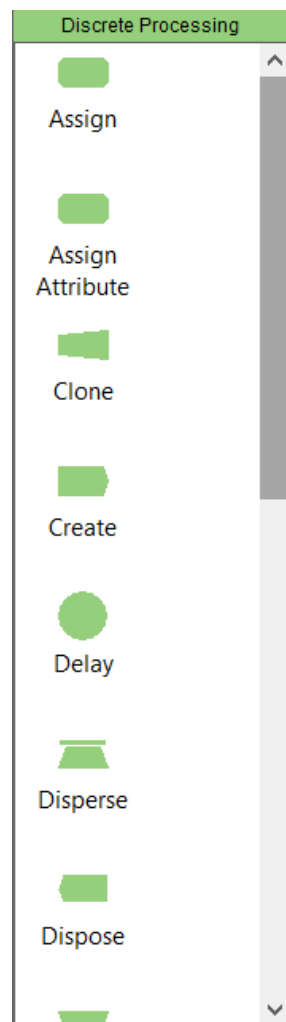


Figura 15 - Discrete processing (fonte: Arena Simulation Software)

In seguito, è riportata una descrizione dei moduli principali utilizzati nella simulazione di interesse, con riferimento diretto alla loro funzione nel modello.

4.4.2.1 Create

Il modulo *Create* consente la generazione delle entità all'interno del sistema produttivo modellato e, nel caso studio considerato, viene utilizzato per creare le entità corrispondenti agli skateboard da assemblare. La frequenza e il numero di entità create possono essere variati al fine di simulare differenti volumi produttivi o scenari di domanda.

4.4.2.2 Assign

Il modulo *Assign* ha il compito di arricchire le entità con informazioni aggiuntive che ne determinano il comportamento all'interno del modello. In altre parole, esso non modifica il flusso fisico delle entità, ma ne definisce le proprietà logiche e quantitative, consentendo anche di personalizzare le scelte di instradamento. Grazie a questo meccanismo, ogni entità è accompagnata da tutte le informazioni necessarie ad attraversare il modello in modo coerente con gli obiettivi di modellazione, selezionando di volta in volta il percorso e la risorsa più adeguati. In questa tesi il modulo *Assign* è stato impiegato per gestire tutti gli attributi associati all'entità skateboard, come il tipo di task o il tempo base di esecuzione.

4.4.2.3 Process

Il modulo *Process* costituisce il nucleo operativo del modello, rappresentando l'esecuzione di un task elementare secondo diverse possibili logiche di prelievo delle risorse. Nel caso oggetto di studio è stata adottata in particolare la logica *seize-delay-release*. Secondo tale logica, l'entità coinvolta, dopo aver impegnato la specifica risorsa assegnata, è soggetta a un tempo di lavorazione pari al tempo di processo associato al task e, successivamente, rilascia la risorsa, rendendola nuovamente disponibile per l'elaborazione di altre entità. Il tempo di lavorazione viene differenziato tra operatori umani e robot mediante l'applicazione del fattore moltiplicativo definito nelle assunzioni iniziali di modellazione.

4.4.2.4 Dispose

Infine, il modulo *Dispose* rappresenta la conclusione del ciclo vita dell'entità all'interno del processo modellato. Ogni singola entità raggiunge il modulo *Dispose* al termine del processo di assemblaggio, consentendo di calcolare il numero totale di prodotti completati e di raccogliere i tempi di attraversamento globali.

4.4.3 Decisions

Avendo già descritto la parte di *Data Definition* ed i moduli più importanti del *Discrete Processing*, risulta ora utile soffermarsi sulla sezione *Decisions* della *Project Bar*, la quale raccoglie le logiche di instradamento delle entità del modello, quindi la sua componente di dinamica condizionale.

Grazie a questa sezione, il modello non si limita più a rappresentare una semplice sequenza lineare di operazioni, ma consente di modellare biforcazioni e controlli dinamici, riflettendo in modo più accurato la complessità del processo reale (Figura 16).

Accanto al *Decide*, modulo approfondito nel paragrafo seguente, la sezione *Decisions* comprende altri moduli che svolgono funzioni complementari, come *Hold* che consente di trattenere un'entità in un punto del modello fino al verificarsi di una specifica condizione, logica o legata ad un evento; il *Signal* che genera, appunto, segnali che consentono di sbloccare entità in attesa in un punto del processo bloccate da moduli *Hold*; *Remove* consente invece, di eliminare entità da determinati punti del processo, risultando uno strumento utile nel caso in cui si debbano rimuovere elementi che non hanno più ragione di rimanere nel sistema; il modulo *Search*, infine, permette di individuare entità nel sistema secondo condizioni prestabilite, aprendo alla possibilità di implementazione di logiche gestionali più avanzate e flessibili.

Grazie all'insieme di questi moduli, ed in particolare grazie al modulo *Decide*, il modellatore è in grado di tradurre in regole operative scelte che, in un contesto reale, richiederebbero l'interesse di operatori umani o di sistemi di controllo opportuni. Senza questi moduli, il modello resterebbe vincolato a una logica puramente sequenziale, mentre gran parte dei sistemi reali si distingue proprio per la presenza di alternative, priorità e incertezze. La capacità di gestire situazioni simili rappresenta uno degli aspetti che rendono la simulazione a eventi discreti uno strumento in grado di rappresentare coerentemente processi produttivi reali.

Risulta importante evidenziare che spesso, le condizioni che regolano l'instradamento effettuato tramite un modulo *Decide* includono condizioni basate su attributi o variabili configurate in precedenza: questo sottolinea una forte interdipendenza fra le diverse librerie della *Project Bar*: le entità non sono semplici oggetti passivi che percorrono un flusso determinato, ma portano con sé caratteristiche specifiche che, grazie ai moduli decisionali, ne determinano i percorsi e la destinazione finale all'interno del processo simulato.

La sezione *Decisions* introduce quindi nel software di simulazione gli strumenti che consentono di passare da un flusso semplice e lineare, in cui le entità seguono tutte lo stesso percorso, ad una rappresentazione ramificata, guidata da regole condizionali, che simulano la variabilità e l'incertezza tipiche dei sistemi complessi.

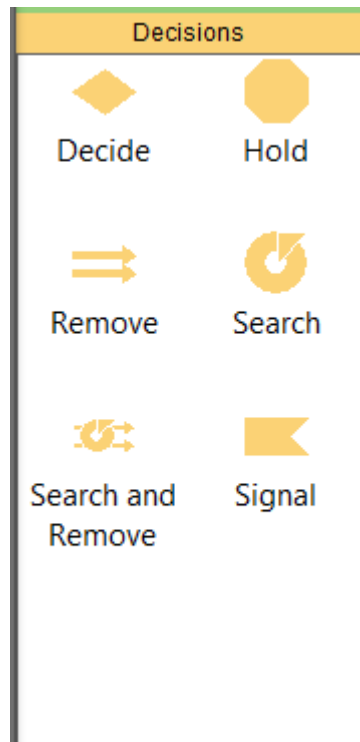


Figura 16 – Decisions (fonte: Arena Simulation Software)

4.4.3.1 Decide

Questo modulo ha la funzione di instradare le entità lungo percorsi alternativi all'interno del modello, sulla base di condizioni logiche o probabilistiche. Nel processo modellato, esso è stato utilizzato per implementare la logica dinamica di allocazione delle risorse, instradando l'entità verso l'operatore umano o verso il robot in funzione di uno specifico indice calcolato. Inoltre, il modulo *Decide* è stato utilizzato per implementare logiche iterative, in particolare per la ripetizione del ciclo nel caso in cui task di tipologia simile vengano eseguiti all'interno dello stesso ramo del processo.

4.4.4 Grouping

Un'ulteriore sezione della *Project Bar* che risulta fondamentale approfondire per una comprensione più profonda del software *Arena* è quella di *Grouping*, che raccoglie gli strumenti utilizzati per gestire le entità non come elementi isolati, ma come insiemi che

possono essere separati, aggregati o coordinati in funzione della logica di processo. Se le librerie di *Data Definition*, *Discrete Processing* e *Decisions* permettono di definire le caratteristiche del sistema e le condizioni che muovono le entità lungo il flusso, la sezione *Grouping* opera un livello superiore, consentendo, ad esempio, di trattare più entità come unità logiche (Figura 17).

Si introduce in questo modo la possibilità in *Arena* di non gestire più un singolo oggetto per volta e ciò rappresenta un aspetto molto diffuso in contesti industriali reali: le operazioni, nell'industria e nei servizi, sono spesso effettuate su gruppi di elementi che devono essere trattati insieme o, viceversa, i cui componenti devono essere trattati singolarmente. Si pensi, ad esempio, all'assemblaggio di più componenti diversi che costituiscono la distinta base di uno specifico prodotto finito, alla spedizione in lotti di articoli o alla sincronizzazione di flussi paralleli in un punto comune. L'uso dei moduli di *Grouping* consente di tradurre queste situazioni in logiche di simulazione, rendendo il modello più aderente al comportamento reale del sistema.

Grazie ai moduli disponibili in questa sezione, è possibile definire logiche di raggruppamento delle entità, scegliendo se formare gruppi di dimensioni fisse, accorparle in lotti o prodotti unici sulla base di condizioni specifiche, oppure consentire che i gruppi si adattino dinamicamente allo stato del sistema. L'impiego dei moduli di *Grouping* permette di tradurre tali situazioni in logiche di simulazione, rendendo il modello più aderente al comportamento reale del sistema.

Queste caratteristiche arricchiscono notevolmente le possibilità di modellazione e consentono inoltre di affrontare domande di ricerca più complesse, legate all'efficienza dei processi di aggregazione o al bilanciamento delle risorse disponibili quando più entità devono essere trattate congiuntamente.

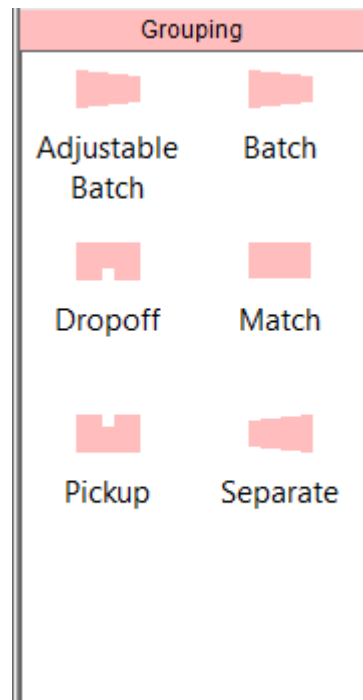


Figura 17 – Grouping (fonte: Arena Simulation Software)

Sarà di seguito approfondito il gruppo di moduli più rappresentativo della sezione, nonché il modulo *Separate* ed il modulo *Batch*, utilizzati nel modello in esame per la gestione dei componenti della distinta base dello skateboard e del loro assemblaggio.

4.4.4.1 Separate e Batch

I moduli *Separate* e *Batch* sono stati utilizzati per modellare la struttura gerarchica e parallela della distinta base del prodotto finito.

Il modulo *Separate* consente di duplicare un'entità, permettendo così di generare flussi paralleli indipendenti (ad esempio per l'assemblaggio dei due truck o delle quattro ruote). Il modulo *Batch*, invece, permette di ricongiungere più entità in un'unica unità, come accade quando i sotto-assiemi vengono uniti al deck per comporre lo skateboard completo. L'uso combinato di questi due moduli è essenziale per garantire la coerenza del flusso con la logica definita dalla distinta base dello skateboard.

4.4.5 Record

Il modulo *Record* è impiegabile per raccogliere statistiche personalizzate e per tenere traccia di grandezze non immediatamente disponibili tra quelle standard fornite dal software. *Arena* mette infatti a disposizione, al termine di ogni simulazione, un report predefinito contenente un insieme di indicatori utili alla descrizione delle principali caratteristiche del processo, quali l'utilizzo delle risorse o la lunghezza media delle code. In molti casi studio risulta

tuttavia necessario monitorare variabili più specifiche, strettamente legate agli obiettivi dell'analisi, rendendo in tali contesti il modulo *Record* uno strumento fondamentale.

In particolare, esso può essere utilizzato per raccogliere dati relativi a tempi parziali, come la durata di una specifica fase produttiva o il tempo di attesa in un determinato nodo della rete di processo. Il modulo si caratterizza inoltre per un'elevata flessibilità, consentendo al modellatore di registrare singoli valori, differenze temporali o altre variabili di interesse. Ciò permette non solo di ottenere statistiche di carattere generale, ma anche di costruire indicatori mirati, ad esempio il tempo medio di utilizzo di una determinata risorsa da parte di uno specifico tipo di entità. In questo modo, il modulo *Record* consente di superare un'analisi puramente macroscopica e di ricavare informazioni utili a rispondere a quesiti più puntuali.

L'uso del *Record* è strettamente connesso alle attività di validazione e confronto: la possibilità di ottenere report su grandezze definite ad hoc permette infatti di verificare la coerenza dei risultati del modello con le ipotesi di partenza, nonché di confrontare scenari alternativi simulati a parità di parametri, rendendo il modulo parte integrante del processo di analisi ed interpretazione dei risultati ottenuti.

In conclusione, il modulo *Record* amplia le potenzialità di *Arena* in modo significativo, consentendo all'utente di estrarre informazioni specifiche, costruite su misura per il problema che si vuole analizzare. Questa capacità di personalizzazione rende il *Record* un elemento essenziale in qualunque progetto di simulazione che miri non solo a riprodurre un processo, ma anche a trarne conoscenze utili per decisioni operative o strategiche.

4.4.6 Connessioni tra i moduli

Un aspetto fondamentale nella costruzione di un modello in *Arena* è rappresentato dalle *connessioni* tra i moduli. Se i moduli possono essere considerati come le unità elementari della simulazione, le connessioni costituiscono i leganti che danno forma al flusso logico del processo. Esse determinano non solo la sequenza con cui le entità attraversano il modello, ma anche le possibili ramificazioni, i percorsi alternativi e le ricongiunzioni di flussi paralleli.

Collegare due moduli differenti implica per il simulatore che una volta terminate le operazioni previste in un modulo, l'entità deve spostarsi nel successivo.

La struttura complessiva del modello dipende direttamente dalla correttezza e dalla coerenza di queste connessioni, che devono rispecchiare fedelmente la logica del processo reale.

Si fornisce in seguito un esempio elementare finalizzato a chiarire ulteriormente il significato ed il funzionamento delle connessioni nel software di simulazione utilizzato.

Si vuole modellare un'attività semplice in cui un paio di occhiali arriva nel sistema, viene sottoposto ad una fase di controllo e successivamente scartato in caso di difettosità, oppure inviato alla fase di assemblaggio se conforme, dove viene processato.

In *Arena*, tale flusso può essere rappresentato collegando in sequenza un modulo *Create* deputato alla generazione dell'entità prodotto, un modulo *Process* che esegue l'operazione di controllo, e un modulo *Decide*, che distingue tra esito positivo e negativo. Le connessioni tra i moduli hanno il compito di trasferire l'entità da un blocco al successivo: il collegamento tra *Create* e *Process* rappresenta l'arrivo del prodotto alla fase di controllo, mentre dal modulo *Decide* si diramano due rami alternativi che instradano l'entità verso un modulo *Dispose* in caso di difettosità, oppure verso un secondo *Process* qualora il prodotto risulti conforme.

Il processo è rappresentato in *Arena* dal modello in Figura 18.

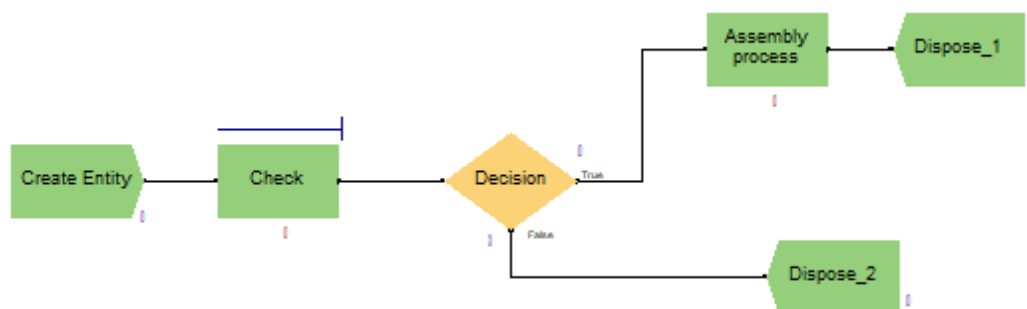


Figura 18 - Modello elementare in Arena (fonte: Elaborazione propria)

Nel flusso rappresentato, si sono in particolare definite le configurazioni seguenti per i principali moduli.

Nel modulo *Create* si sono definite la funzione che descrive il tempo di inter-arrivo delle entità (in questo specifico caso si è scelta la forma esponenziale), la dimensione ogni lotto che giunge nel sistema ed il numero massimo di arrivi oltre il quale la simulazione si arresta, in questo caso infinito (Figura 19).

The image shows a 'Create' dialog box with the following fields and values:

- Name:** Create Entity (dropdown menu)
- Entity Type:** Glasses (dropdown menu)
- Time Between Arrivals:**
 - Type:** Random (Expo) (dropdown menu)
 - Value:** 1 (text input)
 - Units:** Hours (dropdown menu)
- Entities per Arrival:** 1 (text input)
- Max Arrivals:** Infinite (text input)
- First Creation:** 0.0 (text input)
- Comment:** (empty text area)
- Buttons:** OK, Cancel, Help

Figura 19 - Create elementare (fonte: Elaborazione propria)

Nel modulo *Process*, dopo aver impostato la tipologia di azione su *seize-delay-release*, che come precedentemente definito consente di impegnare una risorsa libera, farle processare l'entità giunta nel blocco e successivamente liberarla, è possibile selezionare le risorse che, qualora libere, possono eseguire la lavorazione necessaria. In questo caso si è selezionata l'unica risorsa definita nel processo elementare, nonché lo *HumanWorker* (Figura 20).

Infine, è possibile definire anche secondo distribuzioni probabilistiche la funzione di tempo necessario per il processamento, in questo caso, per semplicità, si è scelta una funzione costante avente valore di 1 ora, rendendo il tempo di processo deterministico.

Process

×

Name:

Check

▼

Type:

Standard

▼

Logic

Action:

Seize Delay Release

▼

Priority:

Medium(2)

▼

Resources:

Resource, HumanWorker, 1

<End of list>

Add...

Edit...

Delete

Delay Type:

Constant

▼

Units:

Hours

▼

Allocation:

Value Added

▼

Value:

1

☒ Report Statistics

Comment:

OK

Cancel

Help

Figura 20 - Process elementare (fonte: Elaborazione propria)

Nell'esempio, si è scelto di assumere che solo il 10% dei prodotti giunti nel sistema sia difettoso, di conseguenza durante la configurazione del modulo *Decision* si è selezionata una tipologia di instradamento definita da probabilità: in questo caso 90% di probabilità di avere un prodotto conforme e proseguire con l'assemblaggio (Figura 21).

Decide

Name: Type:

Percent True (0-100): %

Comment:

OK Cancel Help

Figura 21 - Decision elementare (fonte: Elaborazione propria)

Si osserva che il *Process* rappresenta il controllo, nonché l'attività fisica di ispezione, con un certo tempo di lavorazione, mentre il *decide* rappresenta l'esito del controllo, che può essere conforme o difettoso. Si sta quindi separando il tempo di lavorazione dal suo esito logico, in quanto il controllo può sempre richiedere tempo, anche se il pezzo alla fine risulta difettoso.

Il modello di assemblaggio di uno skateboard completo si fonda quindi su una combinazione di moduli semplici, la cui orchestrazione consente di riprodurre fedelmente un processo produttivo molto articolato, costituito da oltre cento task. Questa architettura modulare, tipica di *Arena*, ha permesso di implementare le logiche decisionali dinamiche, di gestire i rami paralleli della distinta base e di raccogliere informazioni quantitative per l'analisi comparativa degli scenari.

Capitolo 5 - Implementazione del modello in Arena

Dopo aver analizzato la struttura del prodotto oggetto del caso studio, rappresentato dallo skateboard attraverso la relativa distinta base, aver definito i 108 task necessari all'assemblaggio e le tipologie di risorse coinvolte nel processo produttivo, nonché aver esplorato il funzionamento del software di simulazione *Arena*, approfondendone le principali funzionalità e i moduli rilevanti, il passo successivo consiste nel tradurre tali informazioni in un modello di simulazione completo.

L'obiettivo non è quello di riprodurre in modo meccanico l'elenco completo dei task, ma costruire una rappresentazione logica complessa, che rifletta le caratteristiche reali del processo, come decisioni, code ed assegnazione dei compiti, che permetta di analizzare le prestazioni del sistema in funzione delle risorse disponibili.

La definizione della distinta base e la scomposizione in task elementari hanno fornito una visione dettagliata del processo di assemblaggio dello skateboard, ma per poterlo rappresentare all'interno di *Arena* è stato necessario un passaggio ulteriore: tradurre tale descrizione in una logica simulativa. Questo lavoro di traduzione consiste in un processo di astrazione e ricomposizione che consente di individuare una sequenza strutturata secondo la quale procedere allo svolgimento delle operazioni, determinando le loro interdipendenze e i punti in cui è conveniente prevedere parallelizzazioni. Solo con una simile riorganizzazione è possibile ottenere un modello rappresentativo del reale processo di assemblaggio, ma allo stesso tempo utilizzabile come strumento analitico.

Il primo passo necessario alla strutturazione del modello in *Arena* è stato quello di organizzare l'elenco dei task necessari in sottoinsiemi coerenti, ciascuno dei quali corrisponde a una macro-fase del processo. La suddivisione adottata riflette la struttura gerarchica dello skateboard e i vincoli tecnici dell'assemblaggio:

- Assemblaggio dei due truck: Ogni skateboard richiede due truck, ciascuno composto da numerosi componenti elementari (base plate, kingpin, hanger, boccole, dadi, ecc.). Il montaggio di un singolo truck implica una sequenza ordinata di operazioni, che devono essere completate integralmente prima di poter considerare il sottoinsieme completo. Poiché i due truck sono indipendenti, il modello proposto prevede la parallelizzazione dei due sottoinsiemi di task necessari alla costruzione di ognuno.
- Fissaggio dei due truck al deck: Una volta completati i due sottoinsiemi, essi vengono montati sul deck mediante otto bulloni e otto dadi. Questa fase non può iniziare

finché entrambi i truck non sono stati completati, di conseguenza è posizionata in una fase immediatamente successiva alla parallelizzazione effettuata in precedenza. Solo dopo il ricongiungimento dei due flussi paralleli è possibile proseguire con il fissaggio.

- Montaggio delle ruote: Ogni skateboard prevede quattro ruote, ciascuna assemblata sul relativo asse e fissata con un dado. Anche in questo caso, come per il montaggio dei due truck, le operazioni sono indipendenti tra loro e quindi modellate come flussi paralleli. Nel processo, il montaggio delle quattro ruote è rappresentato come altrettanti rami paralleli identici, contenenti ognuno il sottoinsieme dei task necessari al montaggio della corrispondente ruota.
- Controlli e regolazioni finali: Il processo si conclude con attività di *checking* e *adjustment*, volte a garantire che l'assemblaggio sia corretto e che il prodotto risponda agli standard di qualità e sicurezza. Queste operazioni, per loro natura, non sono parallelizzabili e devono essere eseguite in sequenza una volta che l'assemblaggio fisico è stato completato, di conseguenza si posizionano dopo il ricongiungimento dei quattro rami che rappresentano il fissaggio delle ruote, nonché dopo l'ultima fase del processo meccanico.

Questa articolazione in blocchi logici costituisce la spina dorsale del modello, perché evidenzia quali task devono essere gestiti in sequenza e quali, invece, possono procedere in parallelo.

A questo punto, è necessario considerare che la traduzione in logica simulativa richiede particolare attenzione ai parallelismi. Nel caso dei truck e delle ruote, non sarebbe realistico modellare le operazioni come sequenziali, poiché nella realtà possono essere eseguite contemporaneamente da operatori diversi o da più robot. Per riprodurre questo comportamento, in *Arena* è stato necessario utilizzare moduli che permettessero la duplicazione delle entità, così che il flusso lineare di operazioni potesse ramificarsi in più sottoprocessi paralleli. In seguito, grazie a moduli di ricongiungimento, le entità sono state riunificate in un flusso unico, pronto a proseguire verso la fase successiva.

Accanto ai parallelismi, un altro elemento cruciale è rappresentato dalle relazioni di precedenza: come evidente, non è possibile montare i truck sul deck prima di averli assemblati, così come non si possono fissare le ruote prima che i truck siano collegati. Tali vincoli sono traducibili nel modello simulativo tramite la semplice sequenziazione delle

operazioni: in *Arena*, di fatto, non è possibile per un'entità spostarsi in un blocco successivo senza prima completare l'elaborazione del blocco subito precedente.

La traduzione in logica simulativa ha quindi previsto non solo la ricostruzione delle sequenze operative, ma anche l'integrazione di parallelizzazioni, criteri di scelta delle risorse e vincoli di precedenza, aspetti fondamentali per la costruzione di un modello realistico finalizzato all'analisi dei tempi di processo con diverse configurazioni.

5.1 Definizione degli elementi del modello

5.1.1 Definizione delle entità

Si procede, a questo punto, con la definizione dell'entità fondamentale che percorrerà il flusso definito per rappresentare il processo di assemblaggio desiderato: lo skateboard. È possibile effettuare tale definizione iniziale all'interno della sezione *Entity* della libreria *Data definition* della *Project Bar*. Come si osserva nell'immagine corrispondente, si è scelto di non assegnare alcuna tipologia di costo aggiuntivo per l'entità, in quanto non necessario ai fini della simulazione (Figura 22).

	Entity Type	Initial Picture	Holding Cost / Hour	Initial VA Cost	Initial NVA Cost	Initial Waiting Cost	Initial Tran Cost	Initial Other Cost	Report Statistics	Comment
1 ▶	Skateboard	Picture Report ▼	0.0	0.0	0.0	0.0	0.0	0.0	☒	

Figura 22 - Definizione entità (fonte: Elaborazione propria)

L'entità principale è stata quindi definita come il prodotto stesso in fase di costruzione. In altri termini l'entità rappresenta l'intero skateboard che attraversa le varie fasi del montaggio.

È necessario fare un'importante precisazione: al fine di gestire correttamente i parallelismi, ad esempio nella fase di costruzione separata dei due truck identici, sarà necessario, durante la modellazione del flusso, duplicare temporaneamente l'entità mediante un corretto utilizzo del modulo apposito di *Separate*, così da poter seguire contemporaneamente più rami del processo. Una volta completati i task eseguiti nei rami paralleli, le copie ottenute delle entità sono nuovamente ricongiunte, ricostituendo l'entità inizialmente dichiarata dello skateboard grazie al modulo opportuno *Batch*.

5.1.2 Definizione delle risorse

Le risorse disponibili, come le entità lavorate, devono essere definite in partenza, nella sezione *Resource* della libreria *Data definition* e costituiscono, come anticipato, le unità operative che eseguono i task (Figura 23).

Nel modello costruito sono state individuate due tipologie principali di risorse: gli operatori umani (*HumanWorker*) e i robot (*RobotWorker*). La capacità delle risorse, richiesta come parametro iniziale di configurazione durante la definizione delle stesse, corrisponde al numero di unità introdotte nel sistema della tipologia di risorsa corrispondente. Al fine di perseguire l'obiettivo della tesi, si sono analizzate diverse combinazioni di capacità.

	Name	Type	Capacity	Busy / Hour	Idle / Hour	Per Use	StateSet Name	Failures	Report Statistics	Comment
1 ▶	HumanWorker	Fixed Capacity ▼	1	0.0	0.0	0.0		0 rows	☒	
2	RobotWorker	Fixed Capacity	3	0.0	0.0	0.0		0 rows	☒	

Figura 23 - Definizione risorse (fonte: Elaborazione propria)

È inoltre fondamentale osservare che le singole unità operative, una volta impegnate nello svolgimento di un'attività, non sono disponibili per l'esecuzione di altri task. Analogamente, le risorse non possono interrompere un'operazione in corso per eseguirne un'altra appartenente a un ramo parallelo del processo. Tale vincolo logico riveste un ruolo centrale per una corretta e approfondita comprensione del modello simulativo.

Un ulteriore vincolo che si è scelto di introdurre al fine di rispecchiare al meglio una realtà operativa concreta consiste nella limitazione dei task eseguibili da operatori robotici: attività come *handling* e *alignment* possono essere affidate sia a umani sia a robot, mentre *joining*, *checking* e *adjustment* sono state riservate esclusivamente agli operatori umani. Questa distinzione deriva da considerazioni pratiche: i task che richiedono capacità di giudizio o particolare manualità non sono ancora automatizzabili in modo realistico, di conseguenza non sarebbe stato sensato modellare la possibilità di delegarli ad un robot. In questo modo il modello mantiene un equilibrio tra aderenza alla realtà e semplificazione operativa.

Si introduce a questo punto un ulteriore costrutto logico tipico di *Arena*, il set, definito come un contenitore in grado di raggruppare un insieme di risorse appartenenti alla stessa tipologia.

L'utilizzo dei set consente di selezionare dinamicamente la risorsa da impiegare al momento dell'esecuzione di un determinato task. In tal modo, anziché collegare un modulo Process a una risorsa specifica, è possibile associarlo a un set contenente le risorse candidate all'esecuzione dell'operazione, scegliendo l'allocazione più opportuna sulla base di criteri prestabiliti.

Nel contesto dell'assemblaggio sono state introdotte nel set definito e denominato *WorkSet* (Figura 24) le due tipologie di risorse scelte, quindi *HumanWorker* e *RobotWorker* (Figura 25). Quando l'entità giunge a un determinato task, il modello valuta in base ad uno specifico indice quale sia la risorsa più adatta tra quelle disponibili e procede a sequestrarla,

riproducendo il comportamento reale di un sistema ibrido, in cui operatori umani e operatori robotici possono condividere determinate attività, ma con tempi ed efficienze differenti.

	Name	Type	Member Definition Method	Members	Comment
1 ►	WorkSet	Resource	Manual List	2 rows	

Figura 24 - Definizione del set (fonte: Elaborazione propria)

Members		
	Member Type	Resource Name
1	Single Element	HumanWorker
2	Single Element	RobotWorker

Figura 25 - Membri del set (fonte: Elaborazione propria)

Una corretta definizione iniziale sia delle entità che delle risorse risulta importante ai fini della raccolta di dati. Le statistiche ottenibili relative ai tempi di attraversamento sono infatti determinate tenendo conto del percorso dell'entità specifica all'interno del processo, mentre le misure di utilizzo delle risorse derivano da un corretto tracciamento della loro operatività. Grazie a queste definizioni iniziali, il modello è in grado di fornire, a fine simulazione, indicatori di performance complessiva del sistema e di carico di ciascuna delle risorse impiegate.

L'interazione delle diverse risorse coinvolte nel sistema attraverso la logica di sequestro, utilizzo e rilascio consente alla simulazione di riprodurre fedelmente il comportamento di un processo produttivo misto, offrendo al contempo gli strumenti per analizzarne le prestazioni al variare della combinazione di risorse impiegate.

5.1.3 Definizione degli attributi

Un importante pilastro concettuale del modello sviluppato in *Arena* è rappresentato dall'uso degli attributi, definiti inizialmente nella libreria di *Data Definition*. Un attributo è definibile in *Arena* come un'informazione locale di un'entità specifica, ossia un'informazione che un'entità porta con sé lungo tutto il percorso simulativo.

A differenza delle variabili globali, che rappresentano elementi di memoria condivisi e possiedono un unico valore valido per l'intero modello, gli attributi operano secondo una logica completamente diversa: ogni entità mantiene una propria copia indipendente e aggiorna i valori senza influenzare quelli delle altre.

Questa caratteristica si è rivelata particolarmente utile nel modello, nonostante la presenza di un'unica entità principale, lo skateboard da assemblare. Per rappresentare la parallelizzazione di alcune operazioni identiche, infatti, è stato impiegato il modulo *Separate*, che duplica temporaneamente l'entità originaria generandone più copie, una per ciascun ramo logico del processo. Se in questa situazione si fossero utilizzate variabili globali come contenitori informativi, si sarebbe verificato un problema evidente: lo stesso valore sarebbe stato aggiornato contemporaneamente nei diversi rami, perdendo così qualunque significato ai fini della simulazione. Con gli attributi, invece, tale difficoltà non si pone, poiché ogni entità duplicata conserva un set di valori autonomo. Questo consente di tracciare con precisione lo stato e l'evoluzione di ciascun sottocomponente dell'entità primaria, garantendo la corretta rappresentazione degli eventi paralleli e preservando la coerenza del modello complessivo.

Ogni copia dell'entità skateboard porterà con sé la propria configurazione di attributi, garantendo che le decisioni prese (come quale risorsa usare) siano specifiche di quella particolare istanza.

Gli attributi principali che sono stati fondamentali per modellare il comportamento del sistema di assemblaggio considerato, sono suddivisibili in categorie principali ed alcuni di essi svolgono la semplice funzione di parametri costanti, mentre altri si aggiornano durante l'evoluzione del sistema produttivo simulato.

5.1.3.1 Attributi di identificazione e classificazione

La categoria raccoglie gli attributi che hanno un ruolo principale di etichettatura e consentono di distinguere i task tra loro, di tracciare l'avanzamento delle operazioni e rendere esplicita l'appartenenza di ognuno dei task a una delle cinque famiglie operative individuate nel modello.

- *TaskIndex*: identifica univocamente il task in esecuzione, al fine di distinguere ognuno dei diversi step del processo e tracciare il percorso logico delle entità.
- *TaskType*: assegna ad ognuno dei 108 task individuati la sua tipologia, appartenente all'insieme *handling*, *alignment*, *joining*, *checking* e *adjustment*. È il punto di partenza per determinare i tempi e le modalità di esecuzione e per la logica di assegnazione delle risorse implementata.
- *TruckID* e *WheelID*: consentono di identificare, per ogni parallelismo creato tramite il modulo *Separate*, rispettivamente quale truck e quale ruota si stanno gestendo nello specifico ramo.

5.1.3.2 Attributi temporali

Sono gli attributi che definiscono la dimensione temporale del task. Partono da un valore standard, che riflette la durata media di un'operazione eseguita da un umano, e vengono poi modificati per adattarsi alle caratteristiche della risorsa effettivamente impiegata. Sono quindi la base per simulare i tempi di attraversamento.

- *BaseTime*: indica il tempo di processo standard del task in esecuzione, qualora svolto da un operatore umano e dipende direttamente dalla famiglia a cui l'operazione specifica appartiene.
- *TimeHuman* e *TimeRobot*: rappresentano la durata effettiva dell'operazione, determinata a partire dal *BaseTime* e corretta con i coefficienti associati alla risorsa selezionata, come nel caso di aumento del tempo di processo dovuto all'impiego di una risorsa di tipo robot.
- *TaskTime*: questo è il valore di tempo che è effettivamente utilizzato nel modulo Process, quindi durante l'esecuzione effettiva dei task all'interno della simulazione, nonché l'attributo finale all'interno del quale si copia il tempo di esecuzione per operatore umano oppure robotico, a seconda dell'esito del criterio decisionale applicato.

5.1.3.3 Attributi per la scelta delle risorse

Questa categoria raccoglie gli attributi introdotti al fine di modellare la logica decisionale del modello, in particolare necessaria per la scelta dinamica di assegnazione della risorsa da assegnare ad ogni task.

- *SetIndex*: è uno degli attributi più importanti, poiché determina quale risorsa all'interno del set *WorkSet* debba essere impiegata per il task corrente. Viene calcolato dinamicamente sulla base degli indici di prestazione Q , assumendo valore 1 se l'operazione viene assegnata all'umano e 2 se viene assegnata al robot.
- Q_{Human} e Q_{Robot} : rappresentano i valori degli indici di prestazione calcolati per le due tipologie di risorse. Questi indici combinano in maniera ponderata tre dimensioni fondamentali: il tempo, il carico ergonomico e la probabilità di errore. Il confronto tra i due valori consente di determinare quale risorsa risulti più vantaggiosa per l'esecuzione del task.

5.1.3.4 Attributi legati alla qualità ed all'ergonomia

Si tratta di attributi che non influenzano direttamente la durata del task, ma che introducono ulteriori dimensioni di valutazione, quali la probabilità di errore e il carico ergonomico. Essi sono utilizzati anche in questo caso per supportare il processo di selezione della risorsa incaricata dell'esecuzione di un determinato task e consentono di rappresentare aspetti qualitativi che, pur non incidendo sempre sui tempi, condizionano la qualità e la sostenibilità complessiva del processo.

- *ErrorHuman*: probabilità che un operatore umano commetta un errore durante l'esecuzione di un task.
- *ErrorRobot*: probabilità che un robot commetta un errore durante l'esecuzione di un task.
- *LoadHuman*: livello di carico ergonomico (fisico o cognitivo) associato al compito per un operatore umano.
- *LoadRobot*: misura equivalente per i robot, legata a vincoli tecnici e operativi.

5.1.3.5 Attributi pesare i fattori influenti sul criterio decisionale

Questa categoria raccoglie i coefficienti che consentono di pesare in maniera differente i fattori considerati (tempo, errore, carico) per l'applicazione del criterio decisionale proposto. Essi rappresentano il livello di flessibilità del modello: modificandoli, è possibile simulare scenari alternativi e analizzare la sensibilità dei risultati rispetto alla rilevanza attribuita ai diversi criteri.

- *AHuman*, *BHuman*, *GHuman* e *ARobot*, *BRobot*, *GRobot*: rappresentano i coefficienti utilizzati per ponderare rispettivamente le tre dimensioni considerate (tempo, ergonomia, errori) nel calcolo degli indici Q . La presenza di questi attributi consente di calibrare il modello e di effettuare analisi di sensibilità, variando il peso relativo dei diversi fattori.

La logica operativa del modello si basa proprio sulla manipolazione di questi attributi. Grazie alla capacità di immagazzinare informazioni e di aggiornarle quando richiesto, gli attributi consentono di tradurre in un modello simulativo il comportamento di un sistema reale.

L'architettura proposta garantisce, da un lato, il tracciamento del comportamento delle entità all'interno del processo e la loro modifica coerente lungo il percorso, e dall'altro assicura un'elevata flessibilità del modello, poiché eventuali variazioni possono essere implementate

agevolmente intervenendo su parametri specifici, senza stravolgerne la struttura logica. Come verrà mostrato più avanti, il processo elementare relativo all'esecuzione di un singolo task viene infatti iterato per sottogruppi di task che operano sul medesimo sottocomponente della distinta base del prodotto.

In conclusione, gli attributi rappresentano un elemento centrale dell'architettura informativa del modello. In loro assenza, non sarebbe possibile gestire in modo appropriato le differenze tra entità parallele, implementare logiche decisionali dinamiche né raccogliere statistiche personalizzate. La loro introduzione ha consentito di superare le limitazioni iniziali associate all'utilizzo di variabili globali, garantendo al modello un elevato livello di robustezza e una maggiore capacità di adattamento.

5.1.4 Struttura logica del flusso

Definite le entità, le risorse e gli attributi necessari alla costruzione del modello in *Arena*, il passo successivo è quello di organizzare la sequenza di task all'interno di una struttura logica di flusso. In particolare, questa fase non costituisce esclusivamente una trascrizione della distinta base, ma corrisponde ad una rappresentazione ricostruita che tiene conto di sequenze, parallelismi e punti di ricongiungimento allo scopo di ottenere una rappresentazione reale del processo produttivo, mantenendo il modello leggibile e facilmente organizzabile.

5.1.4.1 Fasi principali del flusso

Si può ora descrivere la macrostruttura logica del modello implementato in *Arena*, che rappresenta la spina dorsale della simulazione e riproduce in modo fedele le fasi principali del processo di assemblaggio dello skateboard. Subito dopo l'introduzione dell'entità principale, corrispondente al prodotto in via di costruzione, il flusso viene parallelizzato per consentire la costruzione dei due truck in contemporanea. In questa prima fase (task 1–17 e task 18–34) i due rami paralleli riproducono le stesse operazioni elementari, relative al montaggio sequenziale dei singoli componenti che costituiscono ciascun truck. Le risorse dichiarate nel modello sono quindi disponibili in maniera condivisa e possono essere allocate indifferentemente a uno dei due rami: questo consente di riprodurre la possibilità reale che più operatori o robot lavorino su sottoinsiemi indipendenti dello stesso prodotto. *Arena* gestisce in modo automatico l'allocazione, basandosi sulla disponibilità istantanea delle risorse e seguendo le regole di priorità definite tramite i moduli relativi. Una volta completati i due truck, i due flussi paralleli vengono nuovamente ricongiunti, così da consentire l'avvio

della seconda macro-fase del processo. In questo intervallo di task (35–88) la modellazione è dedicata al fissaggio dei truck al deck, ovvero al collegamento dei sottoinsiemi precedentemente costruiti con l'elemento portante dello skateboard. Questa fase è particolarmente articolata poiché ogni truck deve essere fissato tramite otto bulloni e otto dadi, operazioni che vengono modellate in maniera esplicita per rispettare la scomposizione prevista dalla distinta base. Completato il fissaggio dei truck al deck, il flusso logico prevede una nuova suddivisione in rami paralleli per consentire il montaggio delle quattro ruote. Attraverso un'ulteriore operazione di separazione, l'entità viene suddivisa in quattro copie, ciascuna indirizzata a un ramo che rappresenta l'insieme di operazioni necessarie al montaggio di una singola ruota sull'asse corrispondente. I task compresi tra 89–104 coprono quindi il posizionamento della ruota e il fissaggio tramite dado, con operazioni strutturalmente analoghe ma replicate quattro volte. Questa scelta consente di mantenere coerenza con la logica della distinta base e, allo stesso tempo, di modellare correttamente la possibilità di svolgere tali operazioni in parallelo, sfruttando la disponibilità simultanea di più risorse.

Il flusso si conclude con l'ultima macro-fase, corrispondente alle attività di *checking* e *adjustment* (task 105–108). In questa sezione il prodotto, ormai assemblato, viene sottoposto a controlli di correttezza e a eventuali regolazioni finali. A differenza di alcune fasi precedenti, che hanno potuto beneficiare di parallelismi, queste operazioni devono essere eseguite in sequenza, poiché si applicano al prodotto finito nella sua interezza. Dal punto di vista modellativo questa fase assicura che la simulazione rappresenti fedelmente il flusso reale fino alla consegna del prodotto completo. La struttura macroscopica del flusso è rappresentata in Figura 26.

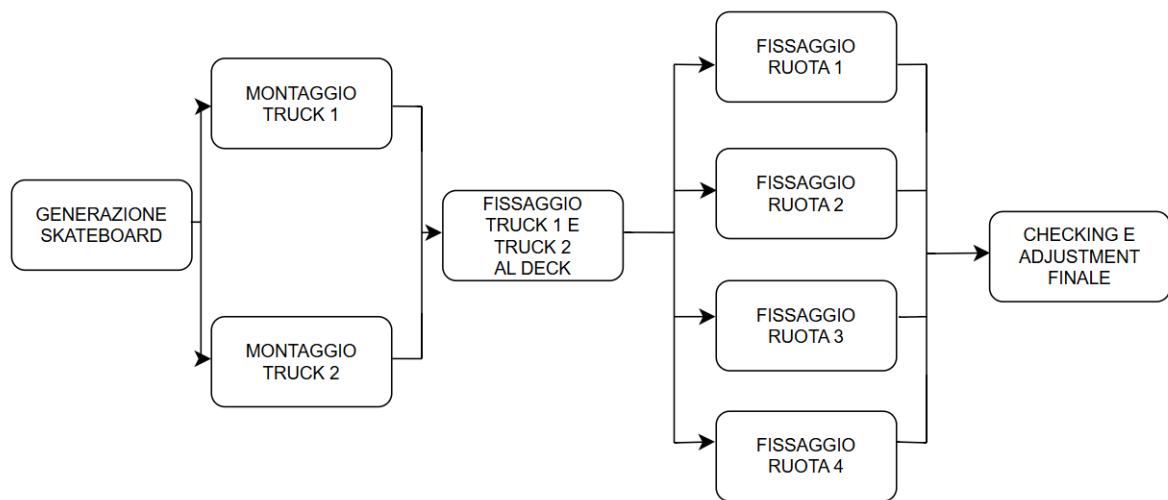


Figura 26 - Struttura del processo di assemblaggio (fonte: Elaborazione propria)

Si passa ora a un livello di analisi più dettagliato, dedicato alla descrizione puntuale del modello sviluppato in *Arena*, che riproduce il processo di assemblaggio dello skateboard. I dettagli numerici e quantitativi del modello sono invece approfonditi nel capitolo dedicato all'analisi dei dati e riportati in Appendice.

5.2 Dettaglio del modello simulativo

5.2.1 Generazione dello skateboard

Il primo elemento inserito all'interno dell'ambiente di simulazione è il modulo *Create*, utilizzato per la generazione dell'entità principale: lo skateboard in fase di costruzione che rappresenta l'oggetto fisico sul quale vengono applicate tutte le operazioni del flusso. Questa entità percorre l'intero modello, attraversando in sequenza le diverse fasi operative che ne determinano la progressiva trasformazione in prodotto finito.

Per semplificare la struttura logica e concentrarsi sul comportamento dinamico del flusso piuttosto che sulla gestione di lotti multipli, gli skateboard vengono generati singolarmente, e non in gruppi. Questa impostazione consente di osservare con maggiore chiarezza le interazioni tra le risorse e il singolo prodotto, evitando sovrapposizioni che avrebbero complicato l'interpretazione dei risultati (Figura 27). La scelta della distribuzione probabilistica che caratterizza il tempo di inter-arrivo è approfondita in un paragrafo dedicato.

The image shows a 'Create' dialog box with the following fields and values:

- Name:** Create
- Entity Type:** Skateboard
- Time Between Arrivals:**
 - Type: Random (Expo)
 - Value: 3.56
 - Units: Minutes
- Entities per Arrival:** 1
- Max Arrivals:** 35
- First Creation:** 0.0
- Comment:** (empty text area)
- Buttons:** OK, Cancel, Help

Figura 27 – Processo di arrivo degli skateboard nel sistema (fonte: Elaborazione propria)

Dopo la generazione dell'entità principale, il modello procede con la definizione di un percorso logico che dovrà essere seguito dallo skateboard lungo l'assemblaggio. Per una più chiara comprensione della struttura interna, si è scelto di analizzare nel dettaglio la configurazione di un singolo ramo del flusso, rappresentativo di ognuna delle sezioni parallele che compongono il processo. Ogni ramo segue la stessa logica di funzionamento e differisce dagli altri esclusivamente per il sottoinsieme di task specifici eseguiti al suo interno.

Si prosegue quindi, dopo la generazione dell'entità corrispondente allo skateboard, con la prima parallelizzazione del flusso, necessaria a rappresentare la costruzione simultanea dei due truck. Tale sdoppiamento è realizzato mediante un modulo *Separate*, che consente di generare due istanze indipendenti dell'entità principale, ognuna destinata a un ramo specifico del processo. Il funzionamento di questo modulo, centrale per la gestione delle attività parallele, sarà analizzato in modo più approfondito in un paragrafo dedicato.

Si entra ora nel dettaglio della struttura logica di uno dei due rami, che può essere considerato rappresentativo dell'intero schema di assemblaggio: si prende in esame in particolare il primo ramo (Figura 28).

5.2.2 Inizializzazione di attributi e parametri

La prima operazione eseguita all'interno di uno specifico ramo consiste nell'inizializzazione degli attributi e dei parametri necessari alle successive fasi decisionali e all'effettiva esecuzione del processo. Poiché tale operazione si limita all'assegnazione di valori iniziali agli attributi, il modulo impiegato è il modulo *Assign*, disponibile nella libreria di *Discrete Processing*. I valori assegnati sono riportati nella Figura 29 e nella Figura 30.

The 'Assign' dialog box has a title bar with a close button (X). It contains the following elements:

- Name:** A dropdown menu with 'SetValues 1' selected.
- Assignments:** A list box containing the following items:
 - Attribute, TaskIndex, 1
 - Attribute, TruckID, 1
 - Attribute, LoadHuman, TRIA(0.3,0.35,0.4)
 - Attribute, LoadRobot, TRIA(0.0125,0.025,0.0375)
 - Attribute, AHuman, 0.3
 - Attribute, ARobot, 0.45
 - Attribute, BHuman, 0.4
 - Attribute, BRobot, 0.5
- Buttons:** 'Add...', 'Edit...', and 'Delete' buttons are located to the right of the list box.
- Comment:** An empty text input field.
- Footer:** 'OK', 'Cancel', and 'Help' buttons.

Figura 29 - Inizializzazione (fonte: Elaborazione propria)

The 'Assign' dialog box has a title bar with a close button (X). It contains the following elements:

- Name:** A dropdown menu with 'SetValues 1' selected.
- Assignments:** A list box containing the following items:
 - Attribute, BHuman, 0.4
 - Attribute, BRobot, 0.5
 - Attribute, GHuman, 0.3
 - Attribute, GRobot, 0.05
 - Attribute, ErrorRobot, TRIA(0.001,0.002,0.003)
 - Attribute, ErrorHuman, TRIA(0.09,0.10,0.11)
 - <End of list>
- Buttons:** 'Add...', 'Edit...', and 'Delete' buttons are located to the right of the list box.
- Comment:** An empty text input field.
- Footer:** 'OK', 'Cancel', and 'Help' buttons.

Figura 30 – Inizializzazione 2 (fonte: Elaborazione propria)

Il parametro *TruckID* consente di tracciare in modo univoco il sottoinsieme di assemblaggio relativo a ciascun truck all'interno del flusso parallelo. Nel ramo dedicato alla costruzione del primo truck tale parametro assume il valore 1, mentre nel ramo corrispondente al secondo

truck assume valore pari a 2. Un principio analogo è stato applicato alla fase di montaggio delle ruote, per la quale il parametro *WheelID* è stato introdotto al fine di distinguere le quattro linee parallele; esso assumerà pertanto i valori 1, 2, 3 e 4, consentendo di tracciare con precisione il percorso di ciascuna ruota nel processo.

Nel modello costruito in *Arena*, è stata utilizzata la distribuzione triangolare, implementata attraverso la funzione *TRIA(min, mode, max)*, che consente di introdurre una variabilità controllata nei parametri di input. La funzione è stata utilizzata all'interno del modulo *Assign* di inizializzazione, in particolare per il carico ergonomico (*Load*) e per il tasso di errore (*Error*), sia per gli operatori umani sia per quelli robotici. Questi attributi possono essere intrinsecamente soggetti a fluttuazioni a causa di condizioni ambientali, fatica o precisione dei sistemi di controllo.

Per ciascun attributo sono stati definiti tre valori caratteristici: un estremo inferiore, che rappresenta la condizione di minima intensità o probabilità; un valore più probabile; e un estremo superiore, associato a scenari meno frequenti.

In questo modo, ad ogni nuova entità generata nel modello *Arena* estrae casualmente un valore compreso all'interno dell'intervallo definito, con maggiore probabilità attorno al valore modale. Tale impostazione permette di simulare in modo realistico le differenze operative che possono manifestarsi tra un task e l'altro, o tra un operatore e l'altro, evitando che i risultati del modello assumano un comportamento deterministico.

AHuman, *ARobot*, *BHuman*, *BRobot*, *Ghuman*, *GRobot* sono dei parametri necessari al criterio decisionale applicato per la scelta della risorsa da assegnare ad ogni task, di conseguenza la loro interpretazione sarà maggiormente dettagliata in seguito.

5.2.3 Gestione delle differenze tra le tipologie di task

Analizzando la struttura logica del processo, è possibile osservare che, tralasciando la fase conclusiva di finitura e controllo le operazioni principali di assemblaggio possono essere ricondotte a tre categorie fondamentali: *identification and handling*, *alignment* e *joining*. A livello logico e modellativo. Queste tre famiglie racchiudono l'insieme dei task necessari alla costruzione dei diversi sottoassiemi e, dal punto di vista modellativo, si differenziano unicamente per il tempo di processo associato.

Nel modello realizzato in *Arena*, questa caratteristica è stata gestita attraverso tre moduli *Assign* distinti denominati rispettivamente *Type1*, *Type2* e *Type3*, ai quali ciascun task viene indirizzato in base alla propria categoria di appartenenza. In questo modo ogni gruppo di operazioni è associato al relativo tempo di processo, definito secondo le logiche descritte in

precedenza. Come evidenziato dall'elenco dei task, in ciascun ramo del modello la prima operazione corrisponde sempre a un'attività di *identification and handling*. Per tale motivo, dal modulo di inizializzazione degli attributi il flusso si dirige direttamente verso il blocco *Type1*, che rappresenta questa tipologia di task. Il collegamento diretto tra il modulo di inizializzazione ed il modulo *Type1*, si ripete pertanto in tutti gli altri rami del flusso modellato.

L'assegnazione delle operazioni successive alle rispettive famiglie è invece gestita in una fase più avanzata del ramo, mediante un modulo *Decide* che consente di realizzare un comportamento iterativo. Attraverso questo meccanismo, il flusso viene riportato ciclicamente ai moduli corrispondenti alle diverse tipologie di task, garantendo che ogni operazione sia eseguita nella sequenza corretta fino al completamento del sottoassieme. In tal modo, il modello riproduce la logica reale di avanzamento del processo, mantenendo una struttura compatta ma sufficientemente flessibile da adattarsi alla complessità dei 108 task complessivi.

Le assegnazioni effettuate nei moduli in analisi, sono osservabili in Figura 31, la quale riporta a titolo di esempio la struttura del modulo specifico *Type1*.

Figura 31 - Assegnazione dei tempi ai task di *identification and handling* (fonte: Elaborazione propria)

Come descritto in precedenza, a ciascun task è stato associato un tempo base di esecuzione, memorizzato in un attributo denominato *BaseTime*, che rappresenta la durata di riferimento per l'operatore umano. Successivamente, tale valore viene copiato nell'attributo *TimeHuman* e utilizzato come parametro di confronto per la definizione del tempo di processamento da parte del robot. Quest'ultimo è calcolato moltiplicando il *BaseTime* per una costante

predefinita, memorizzata nell'attributo *TimeRobot*, così da riprodurre la maggiore durata tipica delle operazioni automatizzate rispetto a quelle eseguite manualmente.

5.2.4 Criterio decisionale

La fase successiva del modello riguarda l'assegnazione della risorsa specifica incaricata di eseguire il task identificato dal valore corrente dell'attributo *TaskID*. In questa fase, la logica decisionale deve stabilire se l'operazione debba essere affidata a un operatore umano o a un robot, sulla base delle condizioni operative istantanee e delle prestazioni attese.

Quando almeno una delle unità appartenenti a ciascuna tipologia di risorsa risulta disponibile, la scelta non può essere ridotta al semplice confronto dei tempi di esecuzione. Una selezione basata esclusivamente sulla rapidità porterebbe, infatti, a un criterio eccessivamente semplicistico: la risorsa mediamente più veloce verrebbe sistematicamente preferita ogniquale volta fosse libera, relegando l'altra a un ruolo marginale, attivandosi solo in caso di saturazione della prima. Tale comportamento non rappresenterebbe in modo realistico la dinamica di un sistema di produzione ibrido, in cui l'interazione tra umani e robot è governata da una pluralità di fattori.

In un contesto operativo reale, la scelta della risorsa più adeguata non dipende unicamente dalla rapidità, ma deve tener conto anche di altri aspetti critici per la qualità e la sostenibilità del processo. Tra questi, il tasso di errore e il carico ergonomico assumono un ruolo determinante. Sebbene gli operatori umani tendano a essere più rapidi grazie alla loro capacità di adattamento e alla flessibilità cognitiva, le risorse robotiche garantiscono livelli di precisione e ripetibilità nettamente superiori, riducendo al minimo la probabilità di difetti. Parallelamente, il fattore ergonomico non può essere trascurato: i robot sono pressoché insensibili alla fatica e possono mantenere performance costanti nel tempo, mentre gli operatori umani risentono inevitabilmente di condizioni fisiche e psicologiche variabili, come stress, stanchezza o fattori ambientali.

Di conseguenza, il modello integra nella decisione non solo il tempo di esecuzione, ma anche indicatori di accuratezza e di affaticamento, così da riprodurre in modo più fedele la complessità delle scelte operative in un contesto collaborativo uomo-robot. Questa logica consente di bilanciare in modo dinamico efficienza, qualità e benessere operativo, fornendo una rappresentazione più aderente al comportamento reale del sistema produttivo.

Al fine di tener conto di ognuno dei tre fattori individuati per l'applicazione del criterio decisionale dinamico finalizzato alla scelta della risorsa da assegnare ad ogni task da eseguire, si è sviluppato un indice, convenzionalmente denominato *indice Q* che calcola la

performance di ognuna delle due tipologie di risorse per lo specifico task da eseguire. L'espressione formale dell'indice Q è riportata di seguito:

$$Q_r = UNIF(-C, C) + A_r * \frac{T_{max,r} - T_r}{T_{max,r} - T_{min,r}} + B_r * \frac{E_{max,r} - E_r}{E_{max,r} - E_{min,r}} + G_r * \frac{L_{max,r} - L_r}{L_{max,r} - L_{min,r}} \quad (5.1)$$

Dove:

- $UNIF(-C, C)$ rappresenta il termine che introduce del rumore stocastico nel sistema, simulando la variabilità intrinseca delle prestazioni reali, non spiegabili dai tre fattori individuati (tempo, errore ed ergonomia). Sono in questo modo modellate anche piccole fluttuazioni di valutazione istantanea, come differenze ambientali o micro-errori.
- r rappresenta la tipologia di risorsa a cui il parametro o l'attributo si riferisce e può assumere unicamente i valori H per l'operatore umano e R per il robot.
- A_r, B_r, G_r rappresentano i pesi relativi associati alla risorsa, assegnati rispettivamente al tempo di processo, al tasso di errore e al carico ergonomico, in funzione dell'impatto che ciascun criterio ha sulla specifica risorsa considerata. Ad esempio, nel caso di un robot, l'influenza della fatica sull'efficienza risulta trascurabile; di conseguenza il coefficiente G_r assume un valore ridotto.
Si osserva che $A_r + B_r + G_r = 1$.
- $T_{max,r}, E_{max,r}, G_{max,r}$, sono i valori massimi assumibili rispettivamente dal tempo di processo, dal tasso di errore e dal coefficiente di carico ergonomico all'interno del modello.
- $T_{min,r}, E_{min,r}, G_{min,r}$, sono i valori minimi assumibili rispettivamente dal tempo di processo, dal tasso di errore e dal coefficiente di carico ergonomico all'interno del modello.
- T_r, E_r, L_r sono i valori di *Time*, *Error* e *Load* per le due tipologie di risorse disponibili, determinati nei moduli precedenti.

Nel modello è stata introdotta una fase di normalizzazione dei parametri impiegati nel calcolo dell'indice di performance Q . Tale operazione si è resa necessaria poiché i tre criteri decisionali utilizzati possiedono natura e scala di misura differenti, e non sarebbero pertanto confrontabili in forma diretta.

Il tempo di esecuzione, espresso in secondi, rappresenta una grandezza fisica e può assumere valori anche di ordine superiore all'unità. Al contrario, il tasso di errore e il carico ergonomico sono grandezze adimensionali, definite in modo tale da assumere valori compresi tra 0 e 1. Il tasso di errore, infatti, è determinato come il rapporto tra il numero di errori e il numero totale di tentativi, mentre il carico ergonomico rappresenta un indice relativo di affaticamento o di sforzo richiesto alla risorsa.

Per consentire una valutazione coerente, ciascun parametro è stato quindi normalizzato rispetto al proprio intervallo di variazione. In questo modo, le tre grandezze possono contribuire all'indice Q su una scala omogenea e confrontabile, evitando che un singolo fattore domini il calcolo per effetto della sua unità di misura o del suo ordine di grandezza.

Dal punto di vista concettuale, la normalizzazione consente anche di garantire una coerenza direzionale tra i fattori: in tutti i casi, un incremento del valore del parametro (sia esso tempo, errore o carico) determina una riduzione dell'indice di performance complessivo, e dunque una minore probabilità che la risorsa associata venga selezionata. Questo accorgimento rende il criterio decisionale più intuitivo e coerente con la logica del processo produttivo, in cui tempi più lunghi, maggiore affaticamento o più alti tassi di errore rappresentano condizioni indesiderabili per la scelta operativa.

Come riportato in Figura 32, il modulo *CalcQ1.0* è caratterizzato esclusivamente dal calcolo dei due indici Q_{Human} e Q_{Robot} .

Assign

Name: CalcQ1.0 2

Assignments:

- Attribute, QHuman, UNIF(-0.2,0.2)+AHuman*(16-TimeHuman)/(16-5)+BHuman*(1-ErrorHuman)/(1-0)+GHun
- Attribute, QRobot, UNIF(-0.2,0.2)+ARobot*(16*1.1-TimeRobot)/(16*1.1-5*1.1)+BRobot*(1-ErrorRobot)/(1-0)+
- <End of list>

Buttons: Add..., Edit..., Delete

Comment:

Buttons: OK, Cancel, Help

Figura 32 - Criterio decisionale (fonte: Elaborazione propria)

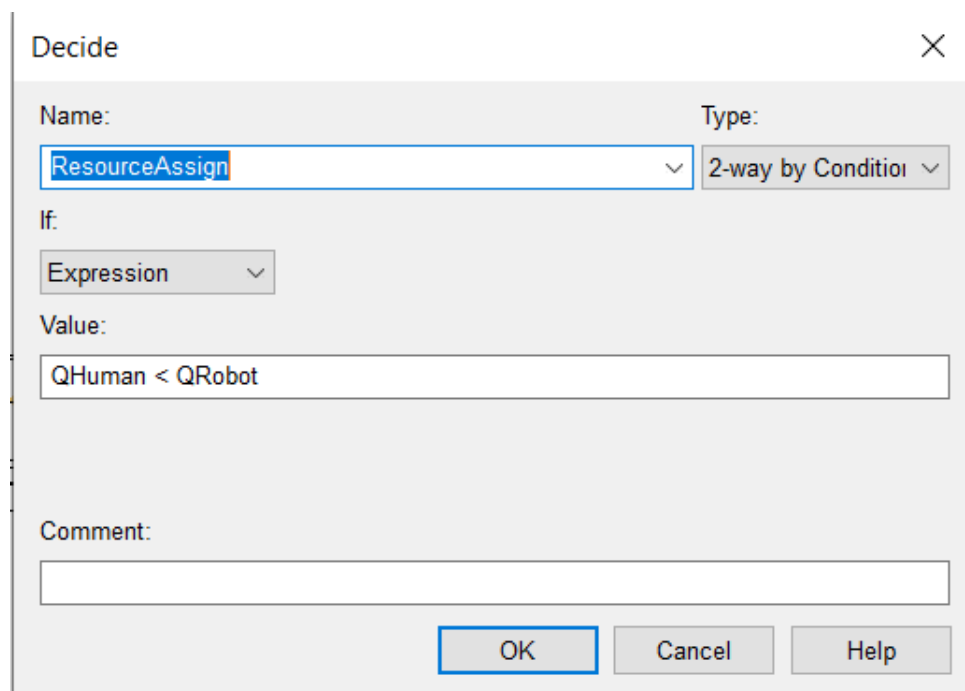
La determinazione dell'indice decisionale Q è effettuata in ognuna delle 108 iterazioni rappresentative dei task di assemblaggio sia per la risorsa di tipo umano, sia per la risorsa di tipo robotico.

Una volta calcolati i due valori Q_{Human} e Q_{Robot} , essi vengono confrontati mediante il modulo *Decide* denominato *ResourceAssign*, posizionato immediatamente a valle del modulo *Assign* in cui il calcolo viene effettuato.

Il modulo *Decide* può essere configurato secondo due principali tipologie di criteri di instradamento: criteri probabilistici e criteri condizionali. Nel modello sviluppato, si è definito un criterio decisionale apposito basato sul calcolo di un indice di performance per ciascuna tipologia di risorsa; di conseguenza, il modulo *Decide* sarà impostato secondo una logica condizionale.

In particolare, la risorsa assegnata al task è quella caratterizzata dal valore dell'indicatore di performance Q elevato, purché risulti disponibile nell'istante in cui l'assegnazione viene effettuata.

Si riporta in Figura 33 la configurazione del modulo *Decide* per l'assegnazione della tipologia di operatore al task in esecuzione.



The image shows a software dialog box titled "Decide". It contains the following fields and controls:

- Name:** A dropdown menu with "ResourceAssign" selected.
- Type:** A dropdown menu with "2-way by Condition" selected.
- If:** A dropdown menu with "Expression" selected.
- Value:** A text input field containing the expression "QHuman < QRobot".
- Comment:** An empty text input field.
- Buttons:** "OK", "Cancel", and "Help" buttons at the bottom right.

Figura 33 - Confronto tra gli indici di prestazione (fonte: Elaborazione propria)

Nel momento in cui la condizione di confronto tra gli indici di prestazione risulta verificata, ovvero quando $Q_{Human} < Q_{Robot}$, l'entità segue il ramo *True* in uscita dal modulo *Decide* e raggiunge il modulo *Assign* denominato *Robot*. In questo punto del modello viene attribuito

il tempo di lavorazione corrispondente a quello di un operatore robotico, che verrà poi utilizzato nei moduli successivi per determinare la durata effettiva del task in corso.

Se invece la condizione non risulta soddisfatta, il flusso prosegue lungo il ramo *False* e lo skateboard viene indirizzato verso il modulo *Assign* denominato *Human*. In tal caso, il tempo di processo associato risulta essere quello caratteristico dell'operatore umano.

Per tenere traccia della tipologia di risorsa effettivamente impiegata, viene utilizzato l'attributo *SetIndex*, che assume convenzionalmente il valore 1 nel caso di assegnazione a una risorsa umana e 2 nel caso di assegnazione a un robot. L'aggiornamento di tale attributo avviene all'interno dei due moduli *Assign* denominati *Human* e *Robot* (Figura 34 e Figura 35).

The screenshot shows a dialog box titled "Assign" with a close button (X) in the top right corner. It contains the following elements:

- Name:** A dropdown menu with "Human" selected.
- Assignments:** A list box containing three items: "Attribute, SetIndex, 1" (highlighted in blue), "Attribute, TaskTime, TimeHuman", and "<End of list>". To the right of the list box are three buttons: "Add...", "Edit...", and "Delete".
- Comment:** A text input field.
- Buttons:** "OK", "Cancel", and "Help" buttons at the bottom right.

Figura 34 - Assegnazione degli indici di risorsa: umani (fonte: Elaborazione propria)

The screenshot shows a dialog box titled "Assign" with a close button (X) in the top right corner. It contains the following elements:

- Name:** A dropdown menu with "Robot" selected.
- Assignments:** A list box containing three items: "Attribute, SetIndex, 2" (highlighted in blue), "Attribute, TaskTime, TimeRobot", and "<End of list>". To the right of the list box are three buttons: "Add...", "Edit...", and "Delete".
- Comment:** A text input field.
- Buttons:** "OK", "Cancel", and "Help" buttons at the bottom right.

Figura 35 - Assegnazione degli indici di risorsa: robot (fonte: Elaborazione propria)

È opportuno chiarire che, nel modello di simulazione adottato, una volta che il criterio decisionale individua la tipologia di risorsa più appropriata per l'esecuzione di un determinato task, l'assegnazione non viene riconsiderata dinamicamente in funzione della disponibilità istantanea di risorse alternative. In caso di indisponibilità temporanea della risorsa selezionata, il task resta in attesa fino alla sua liberazione.

Tale scelta rappresenta una semplificazione modellistica consapevole, coerente con contesti produttivi in cui l'assegnazione dei compiti segue logiche organizzative strutturate e non viene continuamente rinegoziata nel breve periodo. Inoltre, questa impostazione consente di separare in modo chiaro la fase decisionale da quella operativa, garantendo una maggiore interpretabilità dei risultati.

5.2.5 Processamento di un task

A questo punto, determinata la risorsa che prende in carico il task in corso, il flusso logico procede verso il modulo di *Process* denominato *DoTask*, che rappresenta la fase di effettiva esecuzione dell'operazione sull'entità. In *Arena*, infatti, il modulo *Process* è il nucleo operativo del sistema di simulazione, in quanto consente di modellare in modo realistico la fase di lavorazione attraverso l'impiego delle risorse coinvolte e già assegnate nelle fasi precedenti (Figura 36).

Nel modello sviluppato, il modulo *DoTask* è configurato secondo la logica *seize-delay-release*, che riproduce fedelmente il comportamento di un'attività produttiva reale. Questa configurazione permette, in primo luogo, di sequestrare la risorsa assegnata rendendola temporaneamente non disponibile per altri moduli *Process*; successivamente, di sommare al tempo di processo dell'entità la durata della lavorazione, determinata in un passaggio precedente del flusso, e infine di rilasciare la risorsa, rendendola di nuovo disponibile per ulteriori operazioni di assemblaggio.

Per garantire che a ciascun task venga attribuito il tempo corretto di esecuzione, è necessario specificare nel campo *Expression* del modulo *Process* l'attributo *TaskTime*. In questo modo, *Arena* utilizza direttamente il valore di tempo di processo, calcolato nei moduli antecedenti per ciascun task, assicurando la coerenza tra la logica decisionale di assegnazione delle risorse e la durata effettiva delle lavorazioni simulate.

Process

Name: DoTask Type: Standard

Logic

Action: Seize Delay Release Priority: Medium(2)

Resources:

Set, WorkSet, 1, Specific Member, SetIndex
<End of list>

Add... Edit... Delete

Delay Type: Expression Units: Minutes Allocation: Value Added

Expression: TaskTime

☒ Report Statistics

Comment:

OK Cancel Help

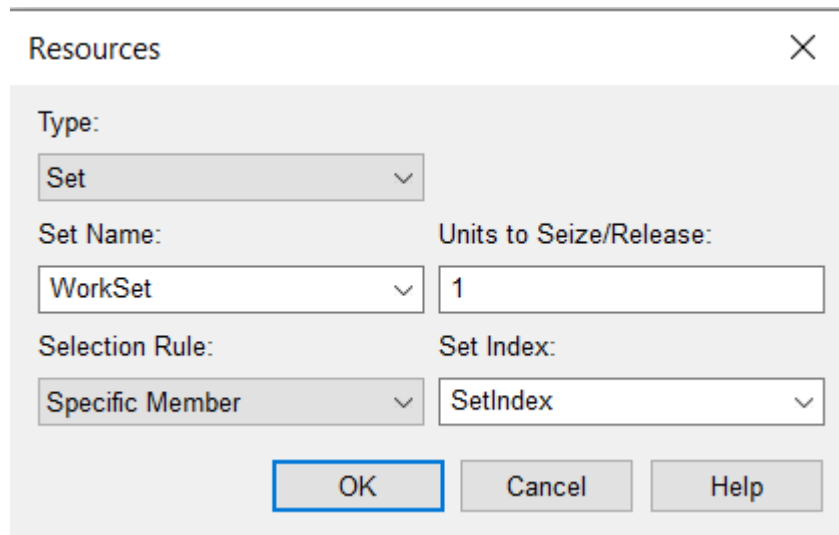
Figura 36 - Processamento di un task (fonte: Elaborazione propria)

Un ulteriore elemento da configurare all'interno del modulo *Process* riguarda la selezione del set di risorse da cui prelevare l'unità effettivamente impiegata per l'esecuzione dell'operazione. Nel modello sviluppato, tale funzione è svolta dal set denominato *WorkSet*, definito nelle fasi iniziali di costruzione del sistema e contenente entrambe le tipologie di risorse disponibili: gli operatori umani e i cobot.

La numerosità di ciascuna categoria di risorsa è determinata in fase di definizione all'interno dell'ambiente *Data Definition*, dove vengono specificati i parametri di capacità di ogni elemento. Ai fini dell'analisi condotta, è particolarmente rilevante poter modificare questi valori, in modo da eseguire diverse configurazioni sperimentali e osservare come varia il tempo complessivo di assemblaggio al variare del numero di risorse umane e robotiche. Tale approccio consente di individuare valutare l'impatto delle diverse caratteristiche delle due

tipologie di operatori in termini di rapidità, precisione ed ergonomia sul comportamento complessivo del sistema.

Per garantire coerenza con la logica decisionale sviluppata, è stata adottata come regola di selezione delle risorse quella denominata *Specific Member* (Figura 37). Tale impostazione consente di individuare, all'interno del set di risorse definito, l'elemento il cui indice corrisponde al valore specificato nel campo *Set Index*.



The image shows a software dialog box titled "Resources". It has a close button (X) in the top right corner. The dialog contains several input fields and buttons. The "Type:" field is a dropdown menu with "Set" selected. The "Set Name:" field is a dropdown menu with "WorkSet" selected. The "Units to Seize/Release:" field is a text box containing the number "1". The "Selection Rule:" field is a dropdown menu with "Specific Member" selected. The "Set Index:" field is a dropdown menu with "SetIndex" selected. At the bottom of the dialog, there are three buttons: "OK", "Cancel", and "Help". The "OK" button is highlighted with a blue border.

Figura 37 - Regola di selezione della risorsa per il processamento (fonte: Elaborazione propria)

In questo modello, al fine di rendere la scelta della risorsa dinamica e dipendente dall'esito della valutazione prestazionale, il valore di *Set Index* non è impostato manualmente ma viene assegnato in fase precedente tramite l'attributo omonimo *SetIndex*. Quest'ultimo viene aggiornato nel modulo *Decide* denominato *ResourceAssign*, che stabilisce quale tipologia di risorsa, umana o robotica, debba essere impiegata per l'esecuzione del task corrente.

Una volta conclusa la fase di processamento, il modello deve consentire il ritorno al ciclo operativo per l'esecuzione del task successivo appartenente allo stesso sottoinsieme di operazioni, oppure permettere il passaggio alla fase seguente dell'assemblaggio. A tal fine, vengono introdotti i moduli necessari a realizzare la logica iterativa del processo, assicurando che l'entità possa percorrere ripetutamente la porzione di flusso dedicata fino al completamento di tutte le attività previste per quel ramo.

5.2.6 Iterazione del ciclo

La prima operazione necessaria al termine della fase di processamento di ciascun task consiste nell'aggiornamento dell'attributo *TaskIndex*. Questo parametro, come precedentemente illustrato, svolge un ruolo centrale nel controllo del flusso, poiché consente

al modello di riconoscere quale operazione del sottoinsieme sia attualmente in esecuzione e di determinare se il ciclo di lavorazione debba proseguire o terminare.

L'incremento di *TaskIndex* viene effettuato attraverso un modulo *Assign*, posizionato immediatamente dopo il *Process* relativo al task in corso. In tale modulo viene impostata l'operazione aritmetica che incrementa il valore dell'attributo di un'unità, secondo la sintassi riportata in Figura 38.

The image shows a software dialog box titled "Assign". At the top right is a close button (X). The dialog contains several fields and buttons:

- Name:** A dropdown menu showing "TaskIndex increase".
- Assignments:** A list box containing "Attribute, TaskIndex, TaskIndex+ 1" and "<End of list>". To the right of this list are three buttons: "Add...", "Edit...", and "Delete".
- Comment:** A text input field.
- At the bottom right are three buttons: "OK", "Cancel", and "Help".

Figura 38 - Incremento del *TaskIndex* (fonte: Elaborazione propria)

In questo modo, a ogni completamento di un task, l'entità aggiorna automaticamente il proprio stato, segnalando al sistema di passare all'operazione successiva.

Una volta aggiornato l'attributo *TaskIndex*, il modello deve verificare se il valore assunto corrisponde ancora a un'operazione appartenente alla sottofase di assemblaggio rappresentata dal ramo corrente. Tale controllo è implementato attraverso un modulo *Decide*. In particolare, qualora il nuovo valore di *TaskIndex* indichi che tutte le operazioni previste in quella sezione del processo sono state completate, l'entità abbandona il ciclo e procede verso la fase successiva dell'assemblaggio. Viceversa, se l'attributo risulta ancora compreso nell'intervallo dei task appartenenti alla sottofase considerata, il flusso prosegue. Il modulo *Decide* che consente di modellare tale instradamento è riportato in Figura 39.

The image shows a 'Decide' dialog box with the following fields and values:

- Name:** Keep in this branch
- Type:** 2-way by Condition
- If:** Attribute
- Named:** TaskIndex
- Is:** <=
- Value:** 17
- Comment:** (empty)

Buttons at the bottom: OK, Cancel, Help.

Figura 39 - Gestione delle uscite delle entità dai rami (fonte: Elaborazione propria)

Nel caso in cui i task previsti per il sottoassemblaggio rappresentato dal ramo corrente non siano ancora stati completati, l'entità prosegue verso il successivo modulo *Decide*, denominato *Task Type Direction* (Figura 40). Questo modulo ha la funzione di identificare la famiglia di operazioni a cui appartiene il task individuato dal valore corrente di *TaskIndex*, così da orientare correttamente il flusso. In base all'esito della condizione logica impostata, l'entità viene quindi instradata verso il modulo *Assign* corrispondente alla tipologia di attività da eseguire (*handling*, *alignment* o *joining*) avviando così una nuova iterazione del ciclo. In tale passaggio, vengono ridefiniti i parametri fondamentali del task, tra cui *BaseTime* e *TaskType*, permettendo al modello di riprodurre in modo coerente la sequenza operativa fino al completamento dell'intero sottoassemblaggio.

Figura 40 - Instradamento delle entità (fonte: Elaborazione propria)

In questa fase, a differenza di quanto avviene nei moduli *Decide* precedentemente introdotti, non risulta più sufficiente applicare una logica binaria basata sulle due possibilità *True* e *False*.

Poiché il ramo deve distinguere tra più famiglie di operazioni, ciascuna associata a un intervallo specifico di *TaskIndex*, è necessario prevedere un numero di percorsi in uscita pari al numero di tipologie di task rappresentate nella sottofase dell'assemblaggio modellata dal ramo corrente. Per tale ragione, nel campo *Type* del modulo *Decide* è stata selezionata l'opzione *N-way by Condition*, che consente di definire più condizioni logiche tra loro indipendenti. Ognuna di queste condizioni logiche verifica a quale categoria di operazioni appartiene il valore corrente di *TaskIndex* e indirizza l'entità verso il ramo appropriato del flusso, nel quale vengono nuovamente assegnati il *TaskType* ed i tempi di esecuzione coerenti con la famiglia di task individuata.

5.2.7 Gestione dei parallelismi

A questo punto, per completare la descrizione del modello creato in *Arena*, occorre approfondire l'aspetto di gestione di parallelismi di flussi che possono essere eseguiti contemporaneamente dalle risorse impiegate, indispensabile per riprodurre coerentemente le caratteristiche del sistema di produzione reale.

Nella produzione dello skateboard, come mostrato in fase preliminare di descrizione del processo, diverse sottofasi possono svolgersi simultaneamente: la costruzione dei due truck

iniziali, avviene in parallelo, così come l'assemblaggio delle quattro ruote, in cui le quattro unità possono essere installate in modo indipendente, sfruttando le stesse risorse di tipologia umana o robotica ma su rami di flusso separati.

La capacità di *Arena* di gestire questa tipologia di fenomeni è resa possibile grazie all'impiego congiunto dei moduli *Separate* e *Batch*, appartenenti alla libreria *Grouping* della *Project Bar*.

Questi costituiscono il nucleo logico attraverso cui il software permette di scomporre temporaneamente un'entità in più sottounità, seguirne l'evoluzione indipendente lungo percorsi paralleli e, infine, ricongiungerle coerentemente una volta completate le rispettive operazioni. In termini concettuali, tale meccanismo consente di tradurre in linguaggio simulativo la realtà di un sistema produttivo in cui le attività non procedono in sequenza rigida, ma in una combinazione di fasi parallele, restituendo così un modello capace di rappresentare accuratamente la complessità temporale e logica del processo.

Il funzionamento del modulo *Separate* risulta concettualmente semplice, ma molto efficace: esso consente di duplicare un'entità entrante in più copie, generando dei flussi indipendenti, i quali possono procedere separatamente all'interno del modello. In questo modo, una singola entità logica, nel caso in esame lo skateboard, può essere temporaneamente scomposta in più sottoentità, ciascuna incaricata di modellare un sottocomponente distinto del prodotto ottenibile dall'assemblaggio. La struttura del modulo è rappresentata in Figura 41.

Separate

Name: Separate 1 Type: Duplicate Original

Percent Cost to Duplicate(0-100): 0.0 % # of Duplicates: 1

Comment:

OK Cancel Help

Figura 41 - Separate (fonte: Elaborazione propria)

Come configurazione, si seleziona la tipologia, in questo caso duplicazione dell'entità principale, il numero di entità uscenti in cui duplicare l'originale ed il costo percentuale della duplicazione, in questo caso nullo per semplificare il modello.

Ogni copia conserva gli stessi attributi dell'entità originale, che vengono trasferiti integralmente nel momento della duplicazione. Tuttavia, a partire da tale istante, ciascuna entità evolve autonomamente, aggiornando i propri attributi in funzione delle operazioni e dei tempi di processo che incontrerà lungo il proprio percorso. Questa indipendenza è un aspetto chiave: consente infatti di tracciare con precisione l'evoluzione di ogni componente, evitando, anche grazie all'utilizzo degli attributi e non delle variabili, interferenze tra i diversi rami e garantendo la coerenza dei risultati statistici.

Nel modello sviluppato, il modulo *Separate* trova applicazione immediata subito dopo la creazione dell'entità principale, permettendo di rappresentare la costruzione simultanea dei due truck. Il numero di copie è in questo caso una (che si somma all'entità originale, per ottenere due sottocomponenti in totale). In una fase successiva, la stessa logica è stata riutilizzata per generare quattro flussi distinti destinati al montaggio indipendente delle ruote. In particolare, si è scelto di annidare tre moduli di *Separate* per la generazione delle quattro entità fittizie rappresentanti le quattro ruote assemblate sullo skateboard.

Ciascun flusso così creato dispone dei propri moduli di *Assign*, *Decide* e *Process*, replicando la logica di base del ramo principale ma con parametri e attributi specifici, come *TruckID* o *WheelID*, utili a mantenere la tracciabilità del componente a cui ogni entità si riferisce.

Le entità dal *Separate* procedono quindi contemporaneamente, di conseguenza la risorse umane e robotiche possono essere impiegate su più sottofasi di assemblaggio nello stesso intervallo temporale, sempre sotto il vincolo che sia impossibile per una singola risorsa effettuare più task simultaneamente.

L'utilizzo del modulo *Separate* richiede un successivo ricongiungimento delle entità generate. Ciascuna copia deve infatti essere riunita alle altre nel momento appropriato, una volta completato il proprio percorso all'interno del modello. Tale funzione è affidata al modulo *Batch*, che rappresenta il complemento naturale del *Separate* e ne assicura la coerenza complessiva nel flusso simulativo. Attraverso il modulo *Batch* è infatti possibile ricomporre le entità precedentemente suddivise in un'unica entità composita, restituendo al modello una struttura coerente con l'avanzamento fisico del prodotto (Figura 42).

Dal punto di vista logico, il modulo *Batch* svolge una funzione di sincronizzazione del flusso, permettendo di far proseguire l'elaborazione solo quando tutte le sottoentità generate in precedenza hanno completato le rispettive attività. In tal modo, si garantisce che la fase successiva dell'assemblaggio inizi solo quando ciascun sottocomponente necessario risulta disponibile.

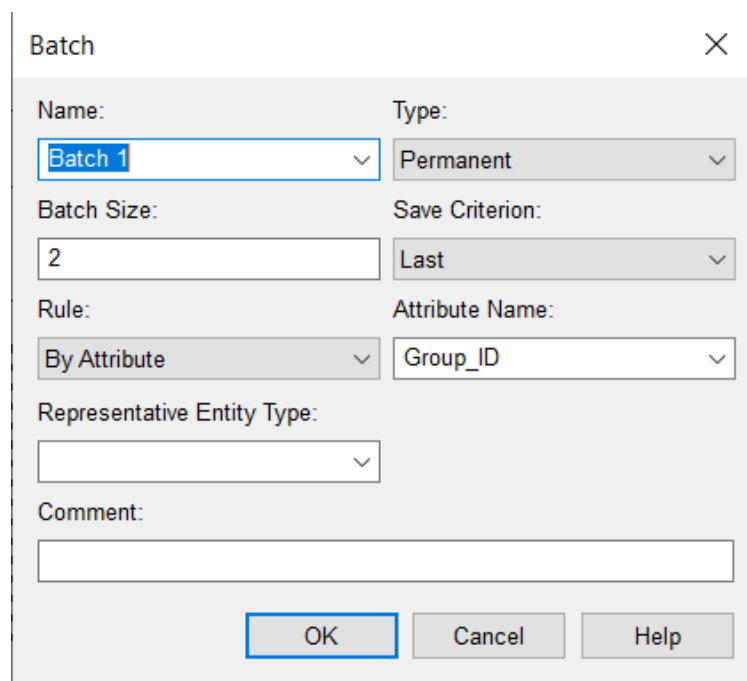


Figura 42 - Batch (fonte: Elaborazione propria)

I parametri richiesti nella configurazione del modulo sono innanzitutto il numero di entità in ingresso, nonché il *Batch Size*, impostato pari a due per la fase di assemblaggio dei truck e pari a quattro per la fase di montaggio delle ruote. Il parametro *Type* è stato invece configurato come *Permanent*, in quanto il ricongiungimento delle entità rappresenta una vera e propria fusione dei componenti, che non vengono più trattati come entità indipendenti. In tal modo, i due truck o le quattro ruote assemblate in parallelo confluiscono definitivamente in un'unica entità che rappresenta il prodotto parzialmente completato.

Il criterio di salvataggio è stato invece impostato su *Last*, così da mantenere nell'entità risultante dai rami paralleli gli attributi associati all'ultimo componente ad aver completato la lavorazione. Tale scelta consente al prodotto aggregato di ereditare lo stato informativo più aggiornato tra quelli disponibili.

Per garantire che i due sottocomponenti generati dal modulo *Separate* vengano successivamente ricongiunti in modo corretto, evitando accoppiamenti errati dovuti a differenze nei tempi di percorrenza dei due rami del modello, è necessario assegnare a ogni entità in ingresso un identificativo univoco. A questo scopo si introduce, nel modulo *Assign* posto all'inizio del modello, una variabile globale (*Global_ID*) che viene incrementata a ogni nuova entità creata e che rappresenta un codice identificativo crescente.

Tale informazione deve poi essere propagata alle sottoentità generate dal modulo *Separate*. Per questo motivo il valore della variabile globale viene copiato in un attributo associato a ciascuna entità in ingresso, denominato *Group_ID*, che rimane legato a tutte le sottoentità costituenti lo skateboard lungo l'intero percorso all'interno del sistema.

La presenza dell'attributo *Group_ID* consente di configurare il modulo *Batch* con la regola di accoppiamento *By Attribute*, specificando proprio *Group_ID* come chiave di aggregazione. In questo modo, il modulo *Batch* riunisce esclusivamente le sottoentità che condividono lo stesso identificativo, garantendo che ciascuna coppia sia formata dalle due metà appartenenti allo stesso skateboard originale, indipendentemente dai tempi di attraversamento dei rami paralleli (Figura 43).

Figura 43 - Gestione degli accoppiamenti tra entità separate (fonte: Elaborazione propria)

La corretta configurazione dei moduli *Separate* e *Batch* rappresenta un aspetto fondamentale per garantire stabilità del modello: un'impostazione errata genererebbe blocchi o perdita di informazioni. I due moduli risultano pertanto tra i più significativi nella modellazione, in quanto consentono di tradurre in termini simulativi la separazione e convergenza delle fasi di assemblaggio dei sottocomponenti dello skateboard completo.

In appendice A è riportata la rappresentazione integrale del modello di simulazione implementato in *Arena*.

Capitolo 6 – Analisi dei dati

6.1 Contestualizzazione degli obiettivi della tesi nell'analisi dei risultati

Nel capitolo precedente è stata descritta in modo dettagliato la costruzione del modello di simulazione a eventi discreti, illustrando le logiche di funzionamento del sistema di assemblaggio e il ruolo svolto dai diversi moduli implementati nell'ambiente *Arena*. A partire da tale modello, il presente capitolo è dedicato all'analisi e all'interpretazione dei risultati ottenuti dalle simulazioni condotte.

Prima di procedere con l'esposizione dettagliata dei risultati, risulta opportuno richiamare in modo sintetico ma strutturato gli obiettivi che hanno orientato l'intero lavoro e che costituiscono la chiave di lettura dell'analisi che segue.

Il modello di simulazione sviluppato nel capitolo precedente non è stato concepito esclusivamente come strumento per la stima dei tempi di processo, ma come supporto all'analisi comparativa di configurazioni operative diverse di un sistema di assemblaggio caratterizzato dalla presenza congiunta di operatori umani e robot collaborativi. In tale prospettiva, la simulazione a eventi discreti è stata utilizzata per osservare in modo analitico l'impatto che variazioni nella composizione e nella numerosità delle risorse impiegate esercitano sulle prestazioni complessive del sistema.

Il primo obiettivo dell'analisi consiste quindi nella valutazione delle prestazioni del sistema dal punto di vista puramente operativo, al variare delle diverse configurazioni di risorse considerate.

Si pone particolare attenzione al fatto che l'aumento delle risorse coinvolte non comporta necessariamente un miglioramento proporzionale dei tempi di processo, a causa di fenomeni quali rendimenti decrescenti, congestioni operative e interazioni tra agenti eterogenei.

La simulazione consente di mettere in evidenza tali dinamiche emergenti, difficilmente coglibili attraverso modelli analitici semplificati, offrendo una rappresentazione più realistica del comportamento del sistema.

Accanto alla dimensione temporale e prestazionale, un secondo obiettivo fondamentale riguarda la valutazione economica delle diverse configurazioni di sistema. I risultati della simulazione vengono pertanto letti anche in termini di costi, considerando l'impatto che il numero di risorse umane e robotiche ha sui costi complessivi di produzione. In questo contesto, l'analisi mira a distinguere tra un possibile ottimo tecnico, associato alla

minimizzazione del tempo di processo, e un ottimo economico, legato invece alla minimizzazione dei costi complessivi. La distinzione risulta fondamentale in un sistema Multi-Human Multi-Robot come quello in analisi, nei quali le risorse coinvolte presentano caratteristiche e costi notevolmente differenti e per le quali risulta fondamentale individuare una regione di soluzioni preferibili.

Inoltre, si vuole individuare la forma funzionale più coerente per la rappresentazione matematica della funzione di complessità di interazione tra unità operative umane e robotiche all'interno del modello considerato. Come ampiamente esposto difatti, tale interazione risulta un fattore fondamentale per la determinazione del tempo di processo in un sistema produttivo Multi-Human Multi-Robot.

Infine, l'obiettivo complessivo del capitolo è quello di fornire una lettura integrata dei risultati ottenuti, mettendo in relazione indicatori di tempo, produttività e costo, al fine di individuare configurazioni che rappresentino un compromesso efficace tra efficienza operativa ed efficienza economica.

L'analisi, come anticipato, non si limita pertanto all'individuazione di un singolo punto ottimale, ma mira a delineare un insieme di soluzioni preferibili, all'interno delle quali le prestazioni del sistema risultano soddisfacenti sotto entrambi i profili considerati. In questo senso, i risultati presentati costituiscono una base quantitativa utile a supportare decisioni progettuali e organizzative in contesti industriali reali, nei quali la scelta della configurazione delle risorse rappresenta un elemento critico per il raggiungimento di elevati livelli di competitività e sostenibilità del sistema produttivo.

6.2 Parametri di simulazione

Al fine di analizzare i risultati delle simulazioni, è fondamentale definire in modo chiaro e coerente i parametri che caratterizzano il sistema e le condizioni sperimentali.

6.2.1 Numero di repliche per ogni configurazione

In primo luogo, è stata adottata la scelta di effettuare cento repliche per ciascuna configurazione del numero di risorse umane e robotiche impiegate nel modello. Questa decisione nasce dalla necessità di ottenere stime statisticamente affidabili delle principali grandezze di interesse, in un contesto caratterizzato da elevata variabilità stocastica, tipica dei modelli di simulazione a eventi discreti.

Ognuna delle repliche rappresenta una singola realizzazione indipendente del processo simulato e produce una osservazione altrettanto indipendente dei parametri di output, quali tempo medio di processo, produttività e costi associati.

L'esecuzione di un numero abbastanza elevato di repliche consente di stimare con maggiore precisione il valore atteso di tali grandezze, riducendo la varianza campionaria della media stimata. In particolare, al crescere del numero di repliche, l'errore statistico associato alla stima decresce all'aumentare del numero di repliche in modo proporzionale all'inverso della radice quadrata del numero di osservazioni:

$$SE(\bar{X}) \propto \frac{1}{\sqrt{n}} \quad (6.1)$$

L'adozione di un numero inferiore di repliche avrebbe potuto condurre a risultati maggiormente influenzati da eventi casuali, rendendo meno robuste le analisi comparative e potenzialmente fuorvianti le conclusioni sull'esistenza di configurazioni ottimali. Al contrario, l'impiego di cento repliche rappresenta un compromesso adeguato tra accuratezza statistica dei risultati ed onere computazionale, risultando coerente con gli obiettivi della tesi.

6.2.2 Processo di arrivo delle entità nel sistema

In questa sezione vengono analizzate le scelte relative al processo di arrivo delle singole entità nel sistema di assemblaggio simulato. In particolare, l'attenzione è rivolta sia al numero complessivo di entità introdotte nel modello in ciascuna configurazione, sia alla caratterizzazione statistica del tempo di inter-arrivo tra entità successive.

La corretta modellazione degli arrivi riveste un ruolo centrale nell'analisi delle prestazioni del sistema, in quanto influisce direttamente sul livello di congestione, sull'utilizzo delle risorse e, di conseguenza, sui tempi di processo osservati. Per questo motivo, le scelte effettuate sono state guidate dall'obiettivo di rappresentare un sistema produttivo che operi in condizioni realistiche e stabili, evitando comportamenti poco coerenti legati alla scorretta definizione degli input.

Nel presente paragrafo si definirà quindi il criterio adottato per la determinazione del numero di entità in ingresso, evidenziando le considerazioni legate alla stabilità del sistema, e verrà descritta la distribuzione di probabilità assunta per il tempo di inter-arrivo e la determinazione del valore medio della stessa.

6.2.2.1 Processo di arrivo delle entità nel sistema: numero di entità per simulazione

Al fine di giustificare le scelte effettuate riguardo il numero massimo di entità in arrivo (parametro di input *Max Arrivals*) per ciascuna simulazione, si deve prima esplicitare il concetto di stabilità di un sistema produttivo e le relative implicazioni.

La stabilità di un sistema produttivo è definibile come la condizione in cui le principali grandezze di interesse (tempo di processo, utilizzo delle risorse, entità presenti in coda) raggiungono valori medi poco sensibili alle fluttuazioni casuali e non dipendenti dal tempo (G. L. Curry, 2011).

In questo contesto, gli effetti transitori dovuti alle condizioni iniziali e alla variabilità degli arrivi non influenzano in maniera significativa i risultati complessivi delle simulazioni effettuate. In tale condizione scompaiono le dipendenze temporali dei valori medi delle misure di interesse.

Sebbene le caratteristiche di un sistema produttivo reale, come il tempo di ciclo medio o il numero di entità in sistema, possano variare nel tempo, nella maggior parte delle analisi pratiche si fa riferimento alle cosiddette misure *steady-state*, ovvero ai valori limite assunti dalle grandezze nel lungo periodo, indipendenti dalle condizioni iniziali del sistema e rappresentativi del comportamento medio osservabile (D. Gross, 1998).

Di conseguenza, il numero di entità introdotte in ogni simulazione deve essere sufficientemente elevato da consentire il raggiungimento di uno stato stazionario, nel quale le grandezze di interesse si stabilizzano e gli effetti transitori iniziali risultano trascurabili (Figura 44).

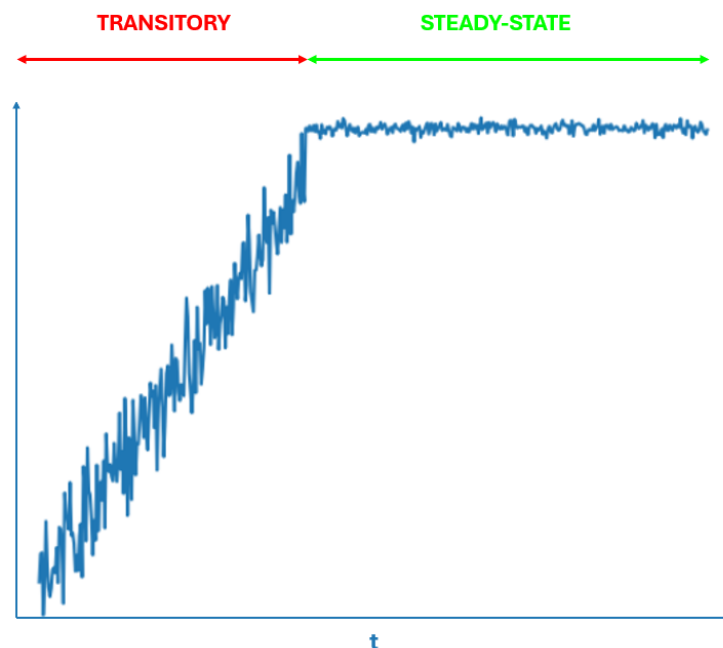


Figura 44 - Transitorio e stato stazionario (fonte: Elaborazione propria)

La scelta di introdurre nel sistema un numero pari a 35 entità per ciascuna simulazione è il risultato di un compromesso tra accuratezza dei risultati e sostenibilità computazionale del modello. Tale valore si è dimostrato sufficientemente elevato da consentire al sistema di raggiungere un regime di funzionamento stabile, in cui le principali grandezze di interesse assumono valori medi rappresentativi e poco sensibili alle condizioni iniziali, garantendo al contempo stime affidabili delle prestazioni del processo produttivo. Allo stesso tempo, un numero maggiore di entità non avrebbe apportato benefici significativi in termini di qualità dei risultati, ma avrebbe comportato un aumento del carico computazionale. Per queste ragioni, il parametro *Max Arrivals* è stato fissato pari a 35 in tutte le simulazioni condotte.

6.2.2.2 Processo di arrivo delle entità nel sistema: distribuzione statistica del tempo di inter-arrivo

Un ulteriore parametro di input rilevante per la costruzione del processo di arrivo simulato delle entità nel sistema è rappresentato dal tempo di inter-arrivo (in *Arena Time Between Arrivals*). La corretta scelta della distribuzione associata a tale variabile influisce in maniera significativa sulla dinamica del sistema, determinando il livello di congestione, la formazione delle code e, di conseguenza, l'utilizzo delle risorse e le prestazioni complessive osservate.

Si assume che gli arrivi siano generati da molteplici sorgenti indipendenti e che non siano soggetti a una sincronizzazione temporale rigida. In particolare, si ipotizza che il verificarsi

di un arrivo non influenzi in modo significativo il tempo di occorrenza degli arrivi successivi e che non esistano meccanismi di accumulo o di anticipazione degli ingressi nel sistema.

In un contesto reale, infatti, gli ordini dei clienti sono spesso costituiti da quantità multiple di prodotto; tuttavia, il flusso di entità che alimenta il sistema produttivo risulta generalmente distribuito nel tempo a causa di fattori esterni al sistema considerato, quali vincoli organizzativi, ritardi informativi o modalità operative non esplicitamente modellate. Di conseguenza, anche a fronte di ordini aggregati, il processo di ingresso può essere efficacemente rappresentato come una sequenza di arrivi unitari indipendenti.

Nel contesto considerato, si è scelto di modellare il tempo di inter-arrivo tramite una distribuzione esponenziale, in quanto essa permette di rappresentare in modo coerente un processo di arrivo privo di memoria, nel quale la probabilità di osservare un nuovo ingresso è indipendente dal tempo trascorso dall'ultimo evento. L'utilizzo della distribuzione esponenziale consente pertanto di catturare il comportamento medio del flusso in arrivo, introducendo al contempo una componente di aleatorietà che riflette l'incertezza e l'eterogeneità tipiche dei sistemi produttivi reali, in assenza di dati che consentano di identificare una distribuzione empirica degli arrivi.

Per poter implementare la distribuzione esponenziale nel modello simulativo, è necessario specificarne un parametro caratteristico: λ , che indica il numero medio di arrivi per unità di tempo. Più nello specifico, il software *Arena* richiede in input la media della distribuzione, che è dimostrabile essere pari a:

$$E[X] = \frac{1}{\lambda} \quad (6.2)$$

In particolare, nel processo modellato, λ risulta essere il tasso medio di arrivo delle entità nel sistema, e di conseguenza $E[X]$ rappresenta il tempo medio di inter-arrivo.

Con l'obiettivo di individuare un tempo di inter-arrivo coerente è stato necessario determinare un valore realistico del flusso di ingresso delle entità nel sistema. A tal fine, è stata scelta una configurazione di riferimento caratterizzata da un numero intermedio di risorse, costituito da 5 operatori umani e 5 robot, e dall'immissione nel sistema di una singola entità. La simulazione è stata condotta in queste condizioni al fine di osservare il tempo di completamento medio di un'unità di prodotto (*cycle time*, *CT*), che è risultato pari a 14.24 min.

Successivamente si è scelto un valore desiderabile per il work-in-progress medio (*WIP*) del sistema.

Come riportato in letteratura, il numero medio di job presenti nel sistema durante un intervallo di tempo può essere definito come la media nel tempo dell'occupazione del sistema osservata in tale intervallo. Nel contesto considerato, tale valore è stato fissato pari a 4 unità.

È possibile a questo punto applicare la *Legge di Little*, che mette in relazione il *work-in-progress* medio, il *throughput* medio del sistema (numero di entità completate che lasciano il sistema nell'unità di tempo) e il tempo medio di permanenza delle entità secondo la relazione:

$$WIP = TH * CT \quad (6.3)$$

Per il *throughput* medio del sistema, si è ottenuto un valore pari a 0.281 u/min.

È opportuno sottolineare che la *legge di Little* è formalmente valida per sistemi stabili osservati in regime stazionario. Nel presente lavoro, tuttavia, tale relazione non viene utilizzata per descrivere direttamente il comportamento del sistema in steady state, bensì come strumento di calibrazione preliminare del flusso in ingresso. In particolare, il tempo di processo osservato nella configurazione di riferimento, ottenuto simulando l'ingresso di una singola entità in condizioni di assenza di congestione, è assunto come tempo medio di permanenza di un'unità nel sistema. A partire da tale valore e da un *WIP* medio desiderabile, la *legge di Little* viene impiegata per determinare un throughput coerente con la capacità del sistema.

Una volta determinato il *throughput* medio del sistema nella configurazione di riferimento, si assume che il sistema operi in condizioni di stabilità. In tale regime vale:

$$TH = \lambda \quad (6.4)$$

In quanto il numero di entità in ingresso nel sistema deve esser mediamente pari al numero di entità in uscita.

Si ottiene quindi:

$$\lambda = 0.281 \text{ entità/min}$$

E, di conseguenza, il valore del parametro richiesto in input per la simulazione corretta del processo di arrivo delle entità nel sistema modellato tramite distribuzione esponenziale si ricava da:

$$E[X] = \frac{1}{\lambda}$$

$$E[X] = 3.56 \text{ min}$$

6.2.3 Modellazione di distribuzioni e parametri dell'indice decisionale Q

Come descritto nel capitolo precedente, l'assegnazione dei task alle risorse disponibili avviene sulla base di un indice decisionale Q , concepito per sintetizzare in un'unica grandezza le principali dimensioni rilevanti del processo, ovvero il tempo di esecuzione, il livello di errore e il carico ergonomico associato alla risorsa (Figura 45).

Nel seguito non si ripercorrono nel dettaglio le logiche di funzionamento del criterio di assegnazione, già ampiamente discusse, ma ci si concentra esclusivamente sulle ipotesi adottate per la parametrizzazione dell'indice e sulla modellazione delle variabili che lo compongono, così come implementate all'interno del modello di simulazione.

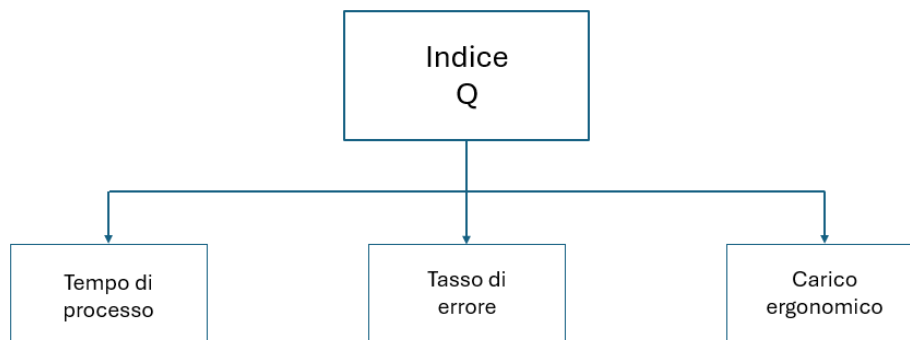


Figura 45 - Fattori che influenzano il criterio decisionale (fonte: Elaborazione propria)

Come già illustrato, i parametri che governano il criterio decisionale adottato per l'assegnazione delle risorse ai task e le distribuzioni utilizzate per rappresentare la variabilità dei principali fattori operativi sono integrati nel modello attraverso specifici attributi associati alle entità e concorrono al calcolo dell'indice decisionale Q , impiegato per la selezione della risorsa più appropriata.

Per quanto riguarda la modellazione della variabilità, si è scelto di rappresentare il tempo di lavorazione, il carico ergonomico e il tasso di errore mediante distribuzioni triangolari, definite da un valore minimo, uno massimo e un valore modale. Questa scelta consente di descrivere in modo semplice ma efficace fenomeni caratterizzati da un comportamento prevalentemente concentrato attorno a una condizione più probabile, pur ammettendo la

possibilità di scostamenti verso scenari meno frequenti. I parametri delle distribuzioni sono stati differenziati tra risorse umane e robotiche, al fine di riflettere le diverse caratteristiche operative delle due tipologie di agenti. In Appendice A sono riportati schematicamente tutti i valori di min, mode e max utilizzati per ciascuna delle distribuzioni definite.

I valori numerici assegnati ai parametri delle distribuzioni sono stati definiti in modo da riflettere una configurazione operativa plausibile e coerente con contesti produttivi reali: innanzitutto è stata associata agli operatori umani una maggiore rapidità media di esecuzione dei task. In particolare, ai robot è stato associato un tempo di processo superiore del 10% rispetto a quello necessario ad una risorsa umana per compiere la medesima operazione. Al contrario, per gli operatori umani si è simulata una più elevata esposizione a carico ergonomico e a una maggiore variabilità del tasso di errore, rispetto alle risorse robotiche, in linea con la maggiore ripetibilità e quasi assenza di difficoltà dovute a fatica, come tipicamente accade nei sistemi automatizzati.

I parametri A , B e G , introdotti nel calcolo dell'indice Q , rappresentano i pesi associati rispettivamente al tempo di esecuzione, al tasso di errore e al carico ergonomico. Essi permettono di modulare l'importanza relativa delle diverse dimensioni prestazionali nella scelta della risorsa, rendendo il criterio decisionale flessibile e adattabile agli obiettivi dell'analisi. In particolare, l'assegnazione di pesi differenti consente di esplorare configurazioni in cui si privilegia l'efficienza temporale rispetto alla qualità, oppure scenari in cui assumono maggiore rilevanza aspetti legati all'affidabilità o alla sostenibilità ergonomica del lavoro umano. Nel presente studio, tali parametri sono stati mantenuti costanti al variare delle configurazioni simulate, così da garantire la confrontabilità dei risultati ottenuti e consentire un'analisi coerente dell'impatto del numero di risorse umane e robotiche sulle prestazioni del sistema.

È importante sottolineare che le scelte effettuate in merito ai parametri delle distribuzioni e ai pesi dell'indice non mirano a rappresentare un caso industriale specifico in senso stretto, bensì a definire uno scenario di riferimento plausibile e sufficientemente generale. In questo modo, il modello può essere utilizzato per analizzare le tendenze e le relazioni tra variabili chiave del sistema produttivo, più che per fornire stime puntuali assolute.

I pesi associati all'indice decisionale rispondono a una logica distinta e complementare rispetto alle distribuzioni associate alle tre variabili considerate. Essi non hanno la funzione di modificare le prestazioni fisiche delle risorse, già descritte dalle distribuzioni, ma servono a modellare le priorità decisionali del sistema nell'assegnazione dei task. In questo senso, i pesi esprimono la rilevanza relativa dei diversi criteri di valutazione, distinta e normalizzata

separatamente per operatori umani e robotici. Nel caso delle risorse robotiche, ad esempio, il tempo di esecuzione e il tasso di errore assumono un ruolo particolarmente rilevante. I robot sono infatti caratterizzati da un comportamento fortemente standardizzato e da un contributo temporale più prevedibile; tuttavia, eventuali errori risultano più critici, in quanto possono generare arresti o interruzioni del processo produttivo.

I valori numerici dettagliati adottati per tali parametri sono riportati in Appendice A.

6.2.4 Ipotesi sui costi orari delle risorse

Per completare la definizione dei parametri iniziali di simulazione, è stato infine necessario introdurre un insieme di ipotesi relative ai costi orari associati alle risorse umane e robotiche, al fine di tradurre le prestazioni temporali del sistema in indicatori economici confrontabili. Il costo orario delle risorse umane è stato modellato come un costo unitario comprensivo della retribuzione diretta dell'operatore e degli oneri indiretti associati alla sua presenza nel sistema produttivo. Tale costo è stato assunto costante nel corso delle simulazioni, ipotizzando un regime operativo stabile e trascurando effetti straordinari che esulando dagli obiettivi dello studio.

Il costo unitario considerato per gli operatori umani è stato successivamente diviso per il numero di ore lavorative annue (assumendo 40 ore lavorative settimanali e 52 settimane annue) ed ha portato ad un valore effettivo pari a:

$$C_h = 20 \frac{\text{€}}{h}$$

Per quanto riguarda invece la determinazione del costo orario di ogni singolo cobot, si è considerato il costo iniziale di acquisto ammortizzato linearmente su un periodo di 5 anni, il costo di manutenzione ed il consumo energetico. Quest'ultimo è stato stimato assumendo un consumo effettivo costante in kWh e adottando un costo medio dell'energia fisso ed espresso in €/KWh.

Sommando i tre fattori di costo per un robot ed assumendo in questo caso un periodo di operatività di 52 settimane annue, si ottiene un costo orario pari a:

$$C_r = 7,4 \frac{\text{€}}{h}$$

Il dettaglio numerico delle componenti considerate nella determinazione dei costi è riportato in Appendice A, mentre nel seguito dell'analisi i valori vengono impiegati esclusivamente

come parametri di input per la valutazione comparativa delle prestazioni economiche del sistema.

6.3 Key Performance Indicators selezionati

Una volta definiti i parametri iniziali della simulazione, si rende necessario individuare un insieme coerente di indicatori di prestazione (Key Performance Indicators, KPI) in grado di descrivere in modo completo ed efficace il comportamento del sistema produttivo al variare della configurazione delle risorse ed alla luce degli obiettivi dello studio.

Gli indicatori selezionati sono stati scelti con l'obiettivo di cogliere sia gli aspetti tecnici legati alle prestazioni operative del sistema, sia le implicazioni economiche connesse all'impiego congiunto di risorse umane e robotiche. In particolare, i KPI selezionati per l'analisi dati, sono riconducibili a tre macrocategorie principali: indicatori temporali, indicatori di produttività e indicatori di costo (Figura 46).

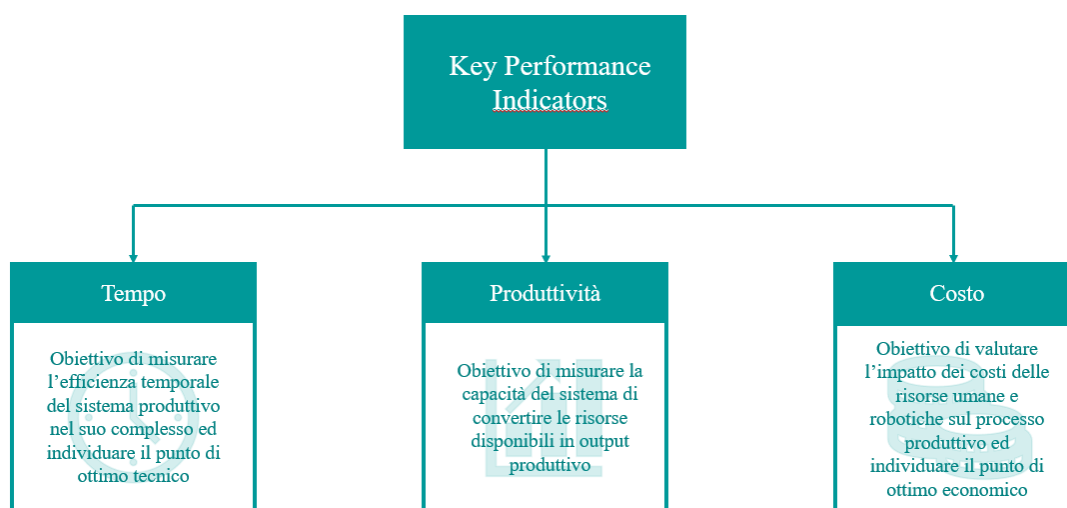


Figura 46 - Classificazione dei KPI introdotti (fonte: Elaborazione propria)

6.3.1 Indicatori temporali

Per quanto riguarda gli indicatori temporali, il principale riferimento è costituito dal tempo di processo complessivo del sistema. Tale grandezza rappresenta il tempo totale necessario affinché tutte le entità previste in ingresso completino l'intero flusso produttivo e lascino definitivamente il sistema. In ciascuna delle simulazioni effettuate, indipendentemente dal numero di risorse umane e robotiche coinvolte, il numero di entità in ingresso è mantenuto

costante ed è pari a 35; di conseguenza, il tempo di processo può essere interpretato come una misura diretta della rapidità con cui il sistema di assemblaggio è in grado di evadere un carico produttivo prefissato.

Dal punto di vista operativo, il tempo di processo è misurato a partire dall'istante di ingresso della prima entità nel sistema fino all'istante in cui l'ultima entità completa tutte le operazioni previste ed esce dal modello. Esso include quindi non solo i tempi di lavorazione effettiva dei singoli task, ma anche i tempi di attesa dovuti alla competizione per le risorse, alle code che si formano nei diversi stadi del processo, nonché agli effetti introdotti dalle logiche di assegnazione delle risorse e dall'interazione tra operatori umani e robotici.

La scelta di utilizzare un numero fisso di entità in ingresso consente di confrontare in modo coerente le diverse configurazioni simulate. Poiché il volume di lavoro complessivo è identico in tutte le simulazioni, eventuali differenze nel tempo di processo sono imputabili esclusivamente alla variazione del numero e della tipologia delle risorse disponibili, nonché alle conseguenti dinamiche di interazione e di utilizzo delle stesse, oltre che alla variabilità, la quale impatta in modo molto ridotto sui risultati, in quanto come evidenziato, si sono effettuate cento repliche distinte per ciascuna delle configurazioni considerate.

Il tempo di processo risulta quindi un indicatore di efficienza tecnica fondamentale per la descrizione del sistema, intesa come capacità di trasformare un insieme determinato di input in output nel minore tempo possibile.

Dal punto di vista implementativo, il tempo di processo è restituito automaticamente dall'ambiente di simulazione *Arena* all'interno del report dei risultati. In particolare, per ciascuna delle simulazioni effettuate, *Arena* genera un foglio di calcolo riepilogativo, nel quale sono riportate le principali statistiche di output del modello. Uno dei fogli generati, cioè quello denominato *Across Replication Summary* contiene dati aggregati su tutte le repliche effettuate per ciascuna simulazione, tra cui il tempo medio di processo ottenuto effettuando la media aritmetica dei tempi di processo ottenuti per ciascuna delle repliche eseguite nella simulazione. All'interno di tale report, il tempo di processo è identificato dalla voce *Total Time*, che rappresenta il tempo medio necessario affinché l'ultima entità lasci il sistema, calcolato sull'insieme delle repliche effettuate.

Infine, il tempo di processo costituisce la base di riferimento per la definizione di ulteriori indicatori di prestazione, in particolare quelli legati alla produttività e ai costi. La sua interpretazione corretta risulta quindi essenziale per comprendere le relazioni tra configurazione delle risorse, prestazioni operative ed efficienza economica del sistema.

6.3.2 Indicatori di produttività

Accanto agli indicatori temporali ed economici, l'analisi dei risultati della simulazione si basa su un insieme di indicatori di produttività, introdotti con l'obiettivo di valutare l'efficienza con cui le risorse disponibili vengono impiegate nel processo produttivo. Tali indicatori consentono di analizzare non solo le prestazioni complessive del sistema, ma anche il contributo medio delle diverse tipologie di risorse, distinguendo tra operatori umani e robotici.

Nel contesto dell'analisi condotta, si è scelto di approfondire in modo sistematico il tema della produttività, introducendo indici che devono essere interpretati correttamente al fine di fornire un quadro completo del sistema di assemblaggio analizzato.

Un primo indicatore considerato è il numero medio di skateboard prodotti da una singola risorsa di una determinata tipologia nell'unità di tempo, all'interno di una specifica configurazione del sistema. Tale indicatore può essere definito come:

$$\pi_r = \frac{S}{\#r * T} \quad (6.5)$$

Dove:

- S è il numero totale di entità introdotte nel processo di assemblaggio in ciascuna replica della simulazione.
- $\#r$ è il numero di risorse di una certa tipologia introdotte, $r = (H, R)$.
- T è il tempo di processo totale.

Questa grandezza fornisce una misura della produttività media di una singola risorsa umana o robotica, ed è interpretabile come output generato per unità di tempo tenendo conto della dotazione complessiva di operatori della specifica tipologia presenti nella simulazione. L'indicatore risulta particolarmente utile per analizzare come varia l'efficienza individuale al crescere o diminuire del numero di risorse umane, anche in presenza di risorse robotiche e viceversa.

È importante sottolineare che questo indicatore non rappresenta la produttività intrinseca della singola risorsa, bensì la produttività media di una risorsa di una determinata tipologia all'interno di una specifica configurazione del sistema. Il valore dell'indicatore risente quindi implicitamente della presenza e del numero delle altre risorse, umane o robotiche, con cui la risorsa considerata interagisce nel processo produttivo.

In modo analogo, è possibile definire un indicatore di produttività complessiva del sistema che tenga conto simultaneamente di tutte le risorse disponibili, umane e robotiche.

Tale indicatore assume la forma:

$$\pi_{tot} = \frac{S}{(\#H + \#R) * T} \quad (6.6)$$

Questa seconda grandezza esprime la produttività media per risorsa, indipendentemente dalla tipologia, e consente di valutare l'efficienza globale del sistema produttivo in funzione del mix di risorse adottato. A differenza degli indicatori temporali, che misurano esclusivamente la durata complessiva del processo, questa grandezza permette di comprendere se un aumento delle risorse comporti un reale miglioramento dell'efficienza o, al contrario, una diluizione del contributo medio di ciascun agente.

Per tutti gli indicatori di produttività introdotti finora, valori numericamente più elevati sono associati a prestazioni migliori del sistema, in quanto indicano una maggiore capacità di generare output a parità di tempo complessivo di processo e di risorse impiegate. In altri termini, una produttività più alta riflette un utilizzo più efficiente delle risorse disponibili, che si traduce in un numero maggiore di skateboard completati nelle stesse condizioni operative.

Accanto a queste misure basate sull'output per unità di tempo, è stato inoltre introdotto un indicatore complementare che consente di interpretare la produttività in termini temporali, ovvero il tempo medio imputabile a una singola risorsa definito per ciascuna tipologia, necessario al completamento di uno skateboard nella configurazione in analisi.

Tale indicatore è definito come:

$$P_r = \frac{T}{\#r * S} \quad (6.7)$$

Dove i simboli mantengono il significato precedentemente introdotto.

Questa grandezza consente una lettura complementare rispetto all'indicatore di produttività espresso in termini di output per unità di tempo e costituisce la base per la successiva valutazione dei costi unitari per skateboard associati alle diverse tipologie di risorse.

Come per la misura complementare della produttività precedentemente introdotta, anche in questo caso è possibile definire un indicatore di produttività complessiva, espresso come:

$$P_{tot} = \frac{T}{(\#H + \#R) * S} \quad (6.8)$$

È possibile interpretare l'indicatore come produttività media globale, ossia tempo medio imputabile a ciascuna risorsa per la realizzazione di un singolo skateboard, senza distinzione tra operatori umani e robot collaborativi.

È opportuno chiarire che l'indicatore di produttività complessiva P_{tot} non coincide con la somma delle produttività calcolate separatamente per le risorse umane e robotiche.

In particolare, non vale in generale la relazione:

$$P_{tot} = P_H + P_R \quad (6.9)$$

poiché ciascun indicatore è normalizzato rispetto a un diverso denominatore. Le produttività P_H e P_R misurano infatti l'output medio per unità di tempo attribuibile a una singola risorsa di una specifica tipologia, mentre P_{tot} rappresenta un valore medio riferito all'insieme complessivo delle risorse presenti nel sistema.

Al contrario, risulta significativa la relazione che lega il tempo medio di processo per skateboard al contributo medio delle risorse complessive.

Vale infatti:

$$\frac{T}{S} = P_{tot} * (\#H + \#R) \quad (6.10)$$

Risulta di conseguenza evidente che il tempo medio per unità di prodotto può essere interpretato come la somma dei contributi medi di tutte le risorse coinvolte nel processo produttivo nella configurazione analizzata.

Con questo secondo gruppo di indicatori di produttività, basati sul tempo medio necessario a ciascuna risorsa per realizzare uno skateboard, a differenza del gruppo precedente, risultano preferibili valori bassi. Infatti, valori più bassi indicano che, a parità di risorse impiegate e di skateboard completati, il tempo richiesto per il loro assemblaggio è minore.

Gli indicatori di produttività definiti costituiscono un passaggio intermedio fondamentale tra l'analisi puramente temporale e la successiva valutazione economica, consentendo di interpretare come le diverse configurazioni di risorse incidano sull'efficienza operativa prima ancora di tradursi in effetti sui costi.

6.3.3 Indicatori economici

Accanto agli indicatori temporali e di produttività, l'analisi dei risultati della simulazione si fonda su un insieme di indicatori economici, introdotti con l'obiettivo di valutare l'impatto dei costi delle risorse umane e robotiche sulle prestazioni complessive del sistema

produttivo. Tali indicatori consentono di leggere i risultati non solo in termini di efficienza operativa, ma anche di sostenibilità economica delle diverse configurazioni analizzate, oltre a consentire l'individuazione dell'ottimo economico.

Il primo indicatore di costo considerato è il costo unitario per skateboard prodotto attribuibile a una singola risorsa di una determinata tipologia, definibile come:

$$\gamma_r = c_r * \frac{T}{\#r * S} \quad (6.11)$$

Dove c_r è il costo orario di una singola risorsa di tipologia umana (H) o robotica (R) e gli altri simboli mantengono il significato definito per le misure precedenti.

Osservando che il termine:

$$\frac{T}{\#r * S} \quad (6.12)$$

Coincide con la misura di produttività P_r definita in precedenza come tempo medio imputabile ad una risorsa di una certa tipologia per produrre uno skateboard, risulta semplice esprimere il costo unitario per risorsa come:

$$\gamma_r = c_r * P_r \quad (6.13)$$

Utilizzando gli indicatori introdotti risulta quindi possibile confrontare l'efficienza economica relativa di operatori umani e robotici, tenendo conto sia delle differenze di costo orario sia delle differenti prestazioni operative.

Questi indicatori, tuttavia, non rappresentano un costo reale, in quanto si basano su una ripartizione media del tempo di processo tra le risorse disponibili e non tengono conto del costo generato dalle risorse inattive; quindi, del fatto che le risorse in realtà non sono pagate solo per il loro tempo medio di attività (T/r), ma per tutto il tempo di processo T , quindi anche se inutilizzate. Essi forniscono pertanto un'informazione utile dal punto di vista operativo e comparativo, ma non sono idonei a supportare decisioni economiche, per le quali sarà necessario introdurre un indicatore che consideri il costo orario complessivo del sistema lungo l'intera durata del processo.

È stato introdotto un ulteriore indicatore che, come i due precedenti, si colloca in una posizione intermedia tra l'analisi tecnica e quella economica. Tale indicatore non rappresenta il costo reale sostenuto dall'azienda, ma fornisce una misura del costo medio del lavoro

effettivamente impiegato nella produzione di uno skateboard, consentendo di isolare l'effetto dell'efficienza operativa delle risorse dalla loro disponibilità numerica.

$$C_{work} = \#H * \gamma_H + \#R * \gamma_R \quad (6.14)$$

Dove, come definito in precedenza, γ_H e γ_R rappresentano il costo unitario attribuibile ad una singola risorsa umana e robotica per la produzione di uno skateboard.

Tale grandezza misura il costo complessivo imputabile alla produzione di uno skateboard considerando esclusivamente il contributo medio delle risorse che hanno effettivamente partecipato al processo. Questo indicatore non penalizza la presenza di risorse inutilizzate o parzialmente utilizzate e tende pertanto a stabilizzarsi all'aumentare del numero di risorse, manifestando un andamento asintotico.

L'indicatore non viene utilizzato per l'individuazione della configurazione economicamente ottimale del sistema, ma esclusivamente come strumento di supporto per l'interpretazione dei risultati e per la comprensione delle dinamiche di efficienza del lavoro all'interno del processo produttivo.

Infine, è stato introdotto un terzo indicatore economico, di particolare rilevanza manageriale, che rappresenta il costo reale per skateboard sostenuto dall'azienda. Tale indicatore è definito come:

$$C = \frac{T}{S} * (\#H * c_H + \#R * c_R) \quad (6.15)$$

A differenza dell'indicatore precedente, questo tiene conto del fatto che le risorse disponibili per il processo, generano un onere economico anche nei periodi di inattività.

Questo indicatore rappresenta quindi il costo che l'azienda sostiene realmente per la produzione di un singolo skateboard data la configurazione del sistema ed è il costo che assume maggior rilevanza decisionale.

L'introduzione di una risorsa aggiuntiva, indipendentemente dalla tipologia, genera due effetti tra loro contrapposti. Da un lato si osserva un incremento lineare dei costi, dovuto all'aumento del numero di risorse da remunerare a parità di tempo di processo. Dall'altro l'incremento delle risorse disponibili contribuisce a ridurre il tempo complessivo necessario alla produzione di uno skateboard, determinando una potenziale riduzione dei costi complessivi.

Come sarà evidenziato dai risultati sperimentali, tuttavia, la riduzione del tempo di processo associata all'impiego di risorse aggiuntive presenta rendimenti marginali decrescenti. Di

conseguenza, poiché il costo per unità di tempo cresce linearmente con il numero di risorse impiegate, sarà individuabile un minimo interno della funzione di costo, risultato del compromesso tra questi due effetti opposti. Per tale motivo, il costo reale per skateboard rappresenta la grandezza di riferimento per l'individuazione dell'ottimo economico del sistema.

L'insieme degli indicatori introdotti consente di valutare il sistema produttivo secondo diverse prospettive complementari, consentendo una lettura profonda dei risultati di simulazione e preparando il confronto tra ottimo tecnico e ottimo economico nelle diverse configurazioni di risorse analizzate.

6.4 Risultati di simulazione

Si introduce ora l'analisi ed interpretazione dei risultati ottenuti attraverso il modello di simulazione sviluppato e descritto nei capitoli precedenti. In particolare, si analizza il comportamento del sistema produttivo al variare della configurazione di risorse disponibili, in termini di numero di operatori umani e robotici impiegati nel processo.

Si ricorda che per ciascuna configurazione analizzata, il sistema è stato simulato mantenendo costante il numero di entità in ingresso, pari a 35 skateboard per ogni replica di simulazione. Tale scelta consente di confrontare in modo coerente le diverse configurazioni, attribuendo le variazioni osservate nelle prestazioni del sistema esclusivamente alla diversa dotazione di risorse e non a fluttuazioni nel carico di lavoro complessivo.

Le configurazioni analizzate sono state definite variando in modo sistematico il numero di risorse umane (H) e robotiche (R). In una prima fase, entrambe le tipologie sono state incrementate di un'unità alla volta, analizzando tutte le configurazioni comprese tra 1 e 10 umani e 1 e 10 robot.

Questa scelta consente di osservare con elevata granularità il comportamento del sistema nelle configurazioni a bassa e media dotazione di risorse, nonché dove l'impatto marginale dell'aggiunta di una singola risorsa risulta più significativo.

Successivamente, per valori superiori a 10 unità per tipologia, le configurazioni sono state esplorate mediante incrementi di 5 risorse alla volta. Tale scelta è legata al fatto che, come emergerà dall'analisi dei risultati, il sistema di assemblaggio tende a stabilizzarsi, rendendo meno rilevanti variazioni puntuali del numero di risorse. L'adozione di passi più ampi rende quindi possibile estendere l'analisi a configurazioni ad elevata disponibilità di risorse, mantenendo sostenibile il numero complessivo di simulazioni e focalizzando l'attenzione sulle tendenze generali del sistema.

I risultati della simulazione vengono analizzati attraverso l'insieme di indicatori di prestazione (KPI) introdotti nel paragrafo precedente, articolati in indicatori temporali, di produttività ed economici. Ciascun indicatore fornisce una prospettiva complementare sul funzionamento del sistema, consentendo di valutare non solo l'efficienza operativa, ma anche la sostenibilità economica delle diverse configurazioni e il contributo medio delle risorse impiegate.

L'intento è quello di comprendere le tendenze che caratterizzano il comportamento del sistema al variare della dotazione di risorse, mettendo in luce i compromessi tra efficienza temporale, produttività e costo.

I risultati saranno pertanto presentati in questa sezione prevalentemente in forma grafica, al fine di rendere immediatamente leggibili le relazioni tra le variabili in gioco e di facilitare l'interpretazione delle dinamiche emergenti. I risultati in completi in forma numerica saranno invece riportati in Appendice B.

6.4.1 Impatto delle risorse sul tempo di processo

La prima dimensione analizzata riguarda l'impatto della configurazione delle risorse sul tempo di processo complessivo del sistema produttivo. Tale indicatore rappresenta una misura diretta dell'efficienza tecnica del sistema e costituisce il punto di partenza per l'analisi dei risultati di simulazione.

Come definito, il tempo di processo T è inteso come l'intervallo temporale che intercorre tra l'istante di ingresso della prima entità nel sistema e l'istante di uscita dell'ultima entità. Il valore di T sintetizza il comportamento globale del sistema sotto una determinata configurazione di risorse, includendo tempi di lavorazione, attese, eventuali congestioni e interazioni tra operatori umani e robotici.

L'attenzione di questa sezione è rivolta a comprendere se ed in che modo l'introduzione di risorse aggiuntive si traduca in una riduzione significativa del tempo di processo complessivo.

Per visualizzare tale relazione, i risultati sono rappresentati attraverso un grafico tridimensionale, nel quale sul piano XY sono riportate le due tipologie di risorse disponibili, ossia il numero di operatori umani H ed il numero di robot collaborativi R . Sull'asse Z è invece rappresentato il tempo di processo T .

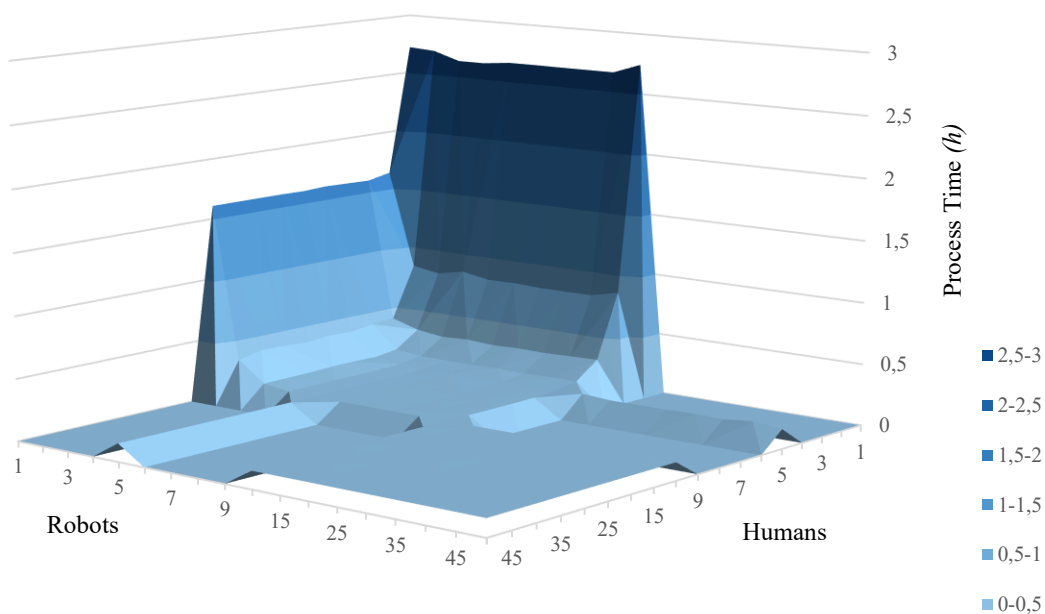


Grafico 1 - Tempo di processo in funzione del numero di umani e del numero di robot

Osservando i risultati, emerge immediatamente un andamento decrescente del tempo di processo all'aumentare del numero complessivo di risorse. Il comportamento evidenziato risulta coerente con le aspettative teoriche: una maggiore disponibilità di risorse consente di ridurre le attese e aumentare le capacità di lavorazione del sistema produttivo. Tuttavia, la riduzione del tempo di processo non risulta uniforme lungo tutto l'intervallo analizzato.

La superficie ottenuta è chiaramente caratterizzata da una rapida discesa iniziale e da una successiva regione di saturazione. Nelle configurazioni caratterizzate da un numero limitato di risorse, infatti, l'introduzione di un'unità aggiuntiva comporta una riduzione significativa del tempo di processo. In questa fase, il sistema opera in condizioni di sottodimensionamento, e l'incremento della capacità produttiva consente di alleviare i colli di bottiglia. Al crescere del numero di risorse, invece, l'effetto marginale associato all'introduzione di ulteriori operatori umani o robot tende progressivamente a ridursi.

Il comportamento osservato è riconducibile al fenomeno dei rendimenti marginali decrescenti: il tempo di processo complessivo del sistema beneficia in misura via via minore dell'incremento di un'unità di risorse disponibili al crescere della dotazione di risorse.

Superata una certa soglia di risorse, l'effetto dei rendimenti marginali decrescenti diviene evidente e il tempo medio di processo tende a stabilizzarsi attorno a un valore asintotico, pari a circa 0,142 h. Oltre tale soglia, ulteriori incrementi del numero di risorse non determinano riduzioni apprezzabili del tempo di completamento del processo.

Questo comportamento risulta coerente con le aspettative teoriche ed è riconducibile alla struttura stessa del sistema produttivo modellato. Nelle fasi iniziali, l'aumento delle risorse consente di ridurre efficacemente i colli di bottiglia, migliorando il grado di parallelizzazione delle attività di assemblaggio. Tuttavia, una volta superato un certo livello di dotazione, il sistema non è più limitato dalla disponibilità di risorse, bensì dalla sequenzialità intrinseca di alcune operazioni e dalla limitata parallelizzabilità del processo.

In tali condizioni, l'aggiunta di ulteriori risorse non può tradursi in un aumento della capacità produttiva, poiché il numero di task simultaneamente eseguibili è già prossimo al massimo consentito dalla logica del processo. Di conseguenza, le risorse aggiuntive risultano parzialmente o totalmente inattive, contribuendo esclusivamente ad accrescere la dotazione disponibile senza generare benefici in termini di tempo di processo.

Un ulteriore elemento che emerge con chiarezza dal grafico tridimensionale riguarda la diversa efficacia marginale delle due tipologie di risorse. L'incremento del numero di operatori umani produce, soprattutto nelle configurazioni inizialmente sottodimensionate, una riduzione più significativa del tempo di processo rispetto a un analogo incremento del numero di robot. Questo suggerisce che, nell'ambito del processo analizzato, l'aggiunta di risorse umane risulti più benefica in termini di efficienza temporale rispetto all'aggiunta di risorse robotiche, a parità di incremento unitario. Tale comportamento è direttamente riconducibile alle ipotesi di modellazione adottate: i robot presentano infatti tempi di esecuzione sistematicamente superiori per ciascuna delle operazioni previste. In particolare, per costruzione del modello, i robot richiedono un tempo di lavorazione pari a circa il 10% in più rispetto agli operatori umani per ognuna delle 108 attività che compongono il processo di assemblaggio.

Per un maggiore dettaglio sui risultati numerici esatti, si rimanda all'Appendice B, ove sono riportati puntualmente i valori esatti ottenuti per il tempo di processo di ogni simulazione.

L'analisi del tempo di processo evidenzia come l'efficienza tecnica del sistema dipenda in modo non lineare dalla configurazione delle risorse umane e robotiche. L'esistenza di rendimenti marginali decrescenti e di una regione di saturazione rende evidente che la presenza di un'ampia area di saturazione suggerisce che la minimizzazione del tempo di processo non costituisce un criterio sufficiente per la scelta della configurazione ottimale del sistema. Questo risultato introduce naturalmente la necessità di affiancare all'analisi temporale considerazioni di produttività e di costo, che verranno sviluppate nelle sezioni successive.

6.4.2 Impatto delle risorse sulla produttività

Dopo aver analizzato l'impatto delle configurazioni di risorse sul tempo medio di processo, l'attenzione viene ora posta sugli indicatori di produttività, introdotti nel capitolo precedente, con l'obiettivo di comprendere come il contributo delle risorse umane e robotiche vari al variare della loro numerosità e del mix complessivo del sistema.

Come già descritto, la produttività è stata analizzata sotto due prospettive differenti: da un lato si è considerato il numero di skateboard prodotti da una singola risorsa di una certa tipologia specifica ed in una determinata configurazione del sistema (π), dall'altro, si è osservato il tempo medio impiegato da una singola risorsa di una certa tipologia per la realizzazione di uno skateboard (P). Entrambe le analisi sono state effettuate sia distinguendo per tipologia di risorsa (π_r e P_r), sia non effettuando questa separazione (π_{tot} e P_{tot})

Anche in questo caso i grafici realizzati sono rappresentati mediante superfici tridimensionali, nelle quali il numero di risorse umane e di robot collaborativi è riportato rispettivamente sull'asse X e Y , mentre lo specifico indicatore è rappresentato sull'asse Z .

6.4.2.1 Impatto delle risorse sul numero di skateboard per ora per risorsa

Si riportano in seguito le superfici tridimensionali ottenute per il numero medio di skateboard prodotti rispettivamente per singolo umano e per robot nell'unità di tempo al variare del mix di risorse disponibili per lo svolgimento delle operazioni. I risultati numerici completi sono riportati in Appendice B. π

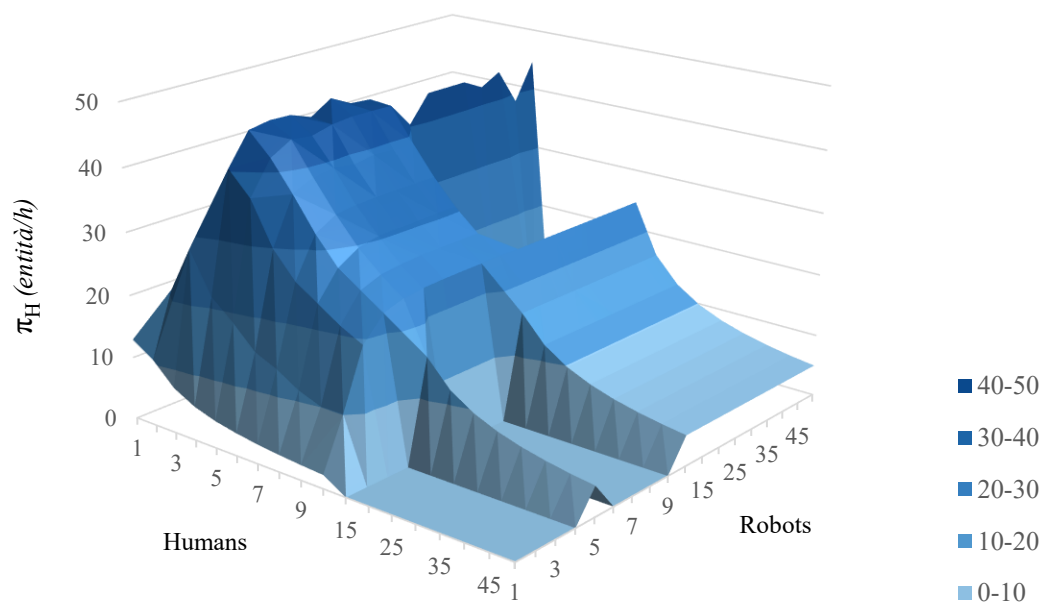


Grafico 2 - π_H in funzione del numero di umani e del numero di robot, vista prospettica 1

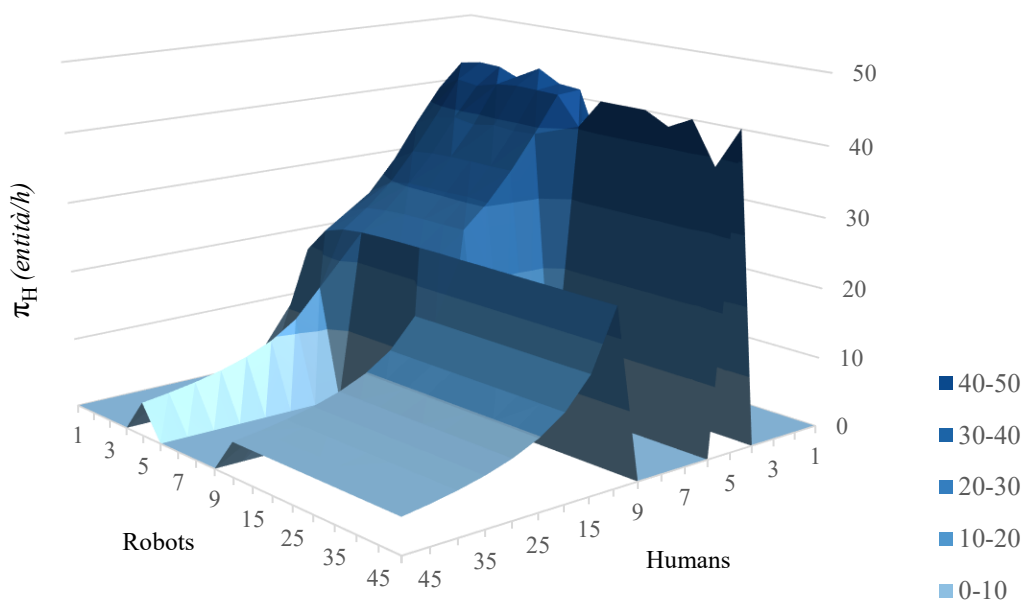


Grafico 3 - π_H in funzione del numero di umani e del numero di robot, vista prospettica 2

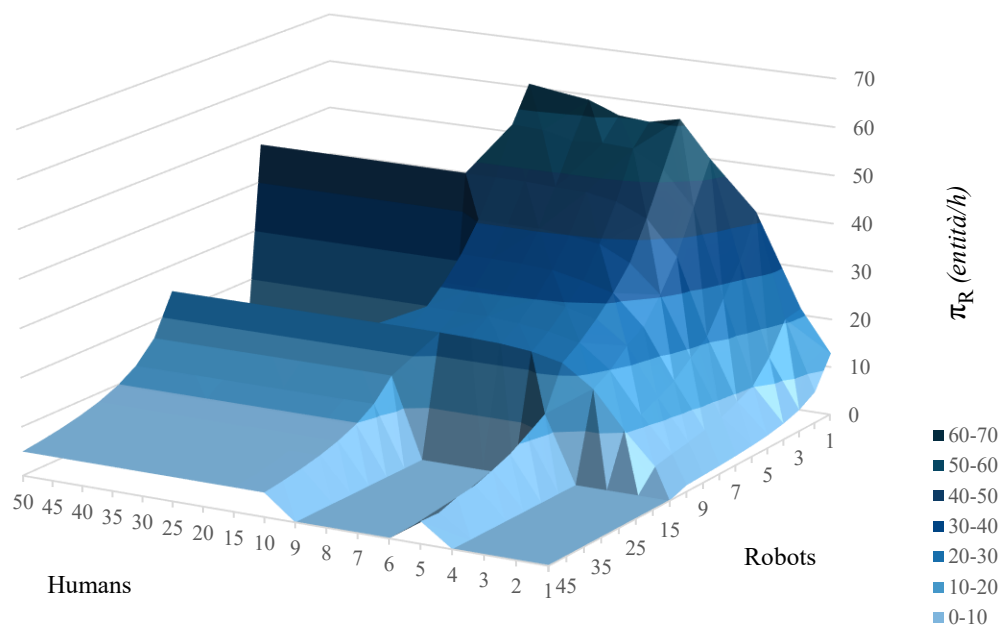


Grafico 4 - π_R in funzione del numero di umani e del numero di robot, vista prospettica 1

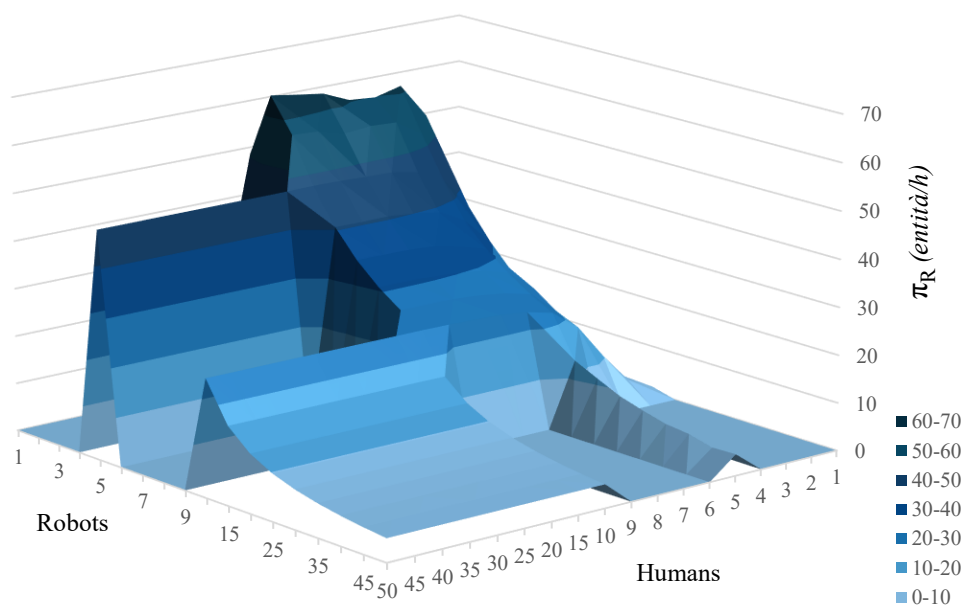


Grafico 5 - π_R in funzione del numero di umani e del numero di robot, vista prospettica 2

Risulta immediato osservare come per questi indicatori, l'andamento assuma caratteristiche meno regolari rispetto a quelle riscontrate per gli indicatori temporali. In particolare, per entrambe le tipologie di risorse si osserva un valore iniziale di produttività relativamente basso, che cresce rapidamente al crescere del numero di risorse disponibili, fino a

raggiungere un punto di massimo per configurazioni intermedie. Oltre tale soglia, la produttività tende progressivamente a diminuire e successivamente ad appiattirsi, stabilizzandosi per valori elevati di entrambe le tipologie di risorse.

Tale comportamento è spiegabile osservando che, quando il numero di risorse è molto ridotto, il sistema risulta fortemente sottodimensionato: il tempo di processo è elevato e le risorse disponibili non riescono a sfruttare appieno il potenziale di parallelizzazione del sistema. Di conseguenza, la produzione normalizzata per risorsa e per unità di tempo risulta contenuta. All'aumentare delle risorse, il tempo di processo si riduce in modo significativo e ciascuna risorsa riesce a contribuire in maniera più efficace alla produzione complessiva, determinando l'incremento della produttività media.

Superata una certa soglia, tuttavia, entrano in gioco i rendimenti marginali decrescenti già evidenziati nell'analisi temporale. In questa regione, l'ulteriore incremento delle risorse non comporta una riduzione significativa del tempo di processo impiegato per la produzione di un'unità di skateboard, mentre il numero di risorse al denominatore dell'indicatore continua ad aumentare. Ne consegue una progressiva diluizione del contributo medio di ciascuna risorsa, che porta alla riduzione della produttività individuale e al successivo stabilizzarsi dell'indicatore su valori pressoché costanti.

Si riporta a questo punto la superficie tridimensionale che rappresenta la produttività totale π_{tot} relativa all'indicatore analizzato finora, in questo caso quindi non si distingue tra risorse umane e robot collaborativi. Esso consente di valutare quanto efficacemente il sistema nel suo complesso trasformi la disponibilità di risorse in output produttivo, integrando simultaneamente gli effetti della riduzione del tempo di processo e dell'aumento del numero totale di risorse impiegate.

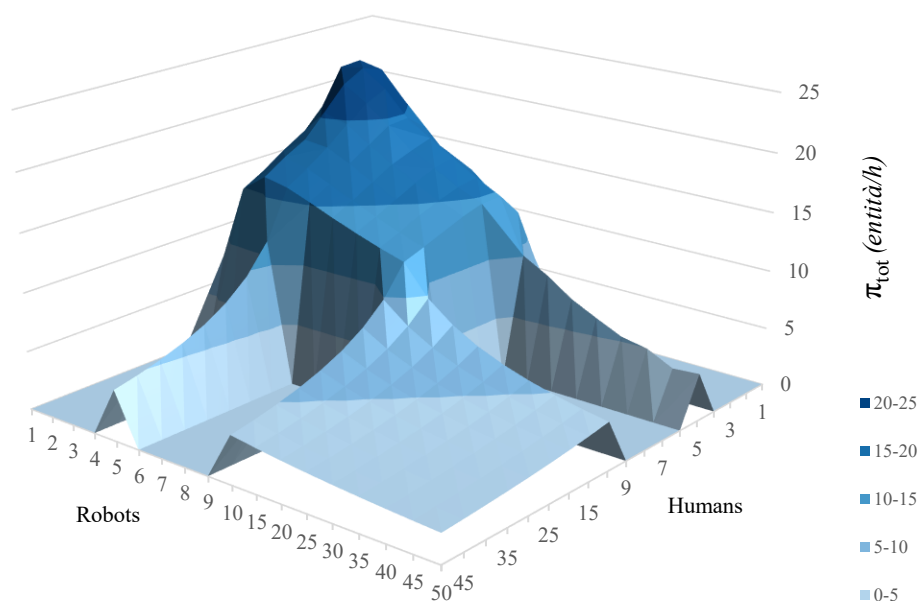


Grafico 6 - π_{tot} in funzione del numero di umani e del numero di robot

L'analisi dei risultati anche in questo caso evidenzia come la produttività totale presenti valori elevati per configurazioni caratterizzate da un numero contenuto ma bilanciato di risorse umane e robotiche. In particolare, come nei contesti precedenti si osserva che i valori massimi dell'indicatore si collocano in configurazioni intermedie, dove la produttività totale supera significativamente quella ottenuta sia in sistemi fortemente sottodimensionati sia in sistemi fortemente sovradimensionati.

Anche in questo caso, l'andamento ottenuto è spiegabile da un iniziale sfruttamento incompleto della capacità di parallelizzazione di alcuni task e da un successivo sovradimensionamento eccessivo: aggiungere risorse in più porta ad un decremento della produttività quando il beneficio marginale ottenuto sul tempo di processo per l'aggiunta di un'unità di risorsa è minimo. In questa regione, il tempo di processo tende a stabilizzarsi attorno a un valore asintotico, determinando un appiattimento della superficie osservata.

Un ulteriore elemento rilevante riguarda la forma della superficie che rappresenta l'indicatore. A differenza di quanto osservato per gli indicatori di produttività calcolati separatamente per operatori umani e robot, in questo caso il grafico presenta un grado di simmetria sensibilmente maggiore rispetto alle due variabili indipendenti. Tale simmetria risulta particolarmente evidente nelle regioni in cui il mix di risorse è fortemente sbilanciato. Nei casi precedenti, infatti, era possibile osservare una marcata asimmetria: fissando a uno il numero di risorse della tipologia oggetto di analisi e incrementando l'altra tipologia, la

produttività mostrava un andamento significativamente diverso rispetto alla configurazione opposta. Nel caso dell'indicatore di produttività totale, invece, queste differenze risultano fortemente attenuate e le curve corrispondenti a configurazioni speculari appaiono visibilmente più simmetriche.

Questo comportamento è coerente con la definizione stessa dell'indicatore, che normalizza la produzione rispetto al numero complessivo di risorse impiegate, senza distinguere tra operatori umani e robot. Di conseguenza, l'effetto delle differenze prestazionali tra le due tipologie viene mediato a livello di sistema, facendo emergere una risposta più regolare e simmetrica rispetto a quella osservata negli indicatori di produttività specifica.

6.4.2.2 Impatto delle risorse sul tempo impiegato per produrre uno skateboard per risorsa

Prosegue a questo punto l'analisi della produttività del sistema e si sposta in questo caso l'attenzione sulla tipologia di indicatore utilizzato: finora si è ricercato il numero di skate prodotti per risorsa nell'unità di tempo in una determinata configurazione. Al fine di ottenere una prospettiva più completa sull'effettiva produttività del sistema si analizzerà ora l'indicatore di produttività duale introdotto, che rappresenta il tempo medio imputabile rispettivamente a un operatore umano e a un robot per la realizzazione di un singolo skateboard.

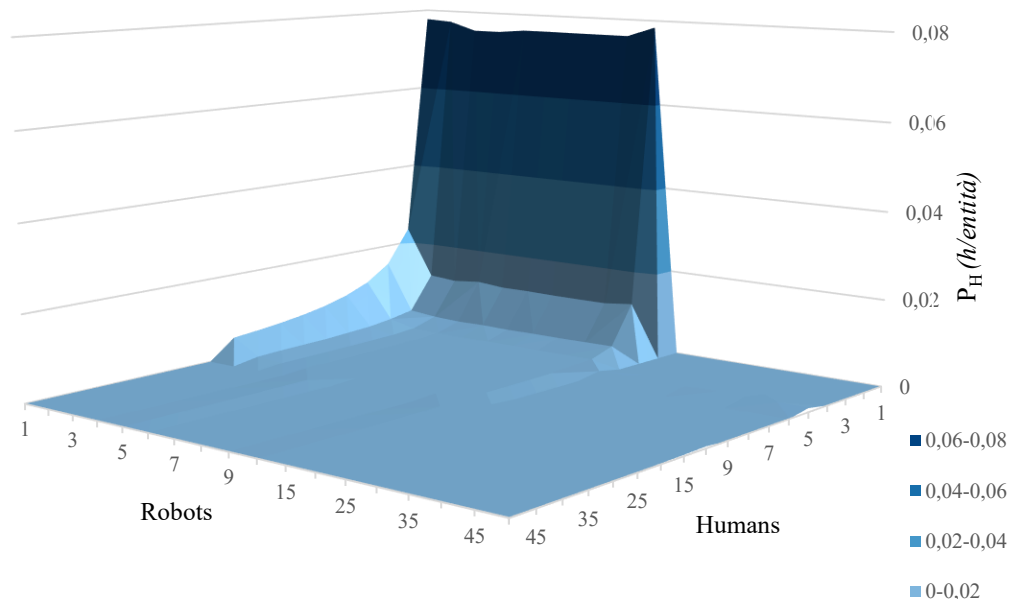


Grafico 7 - P_H in funzione del numero di umani e del numero di robot

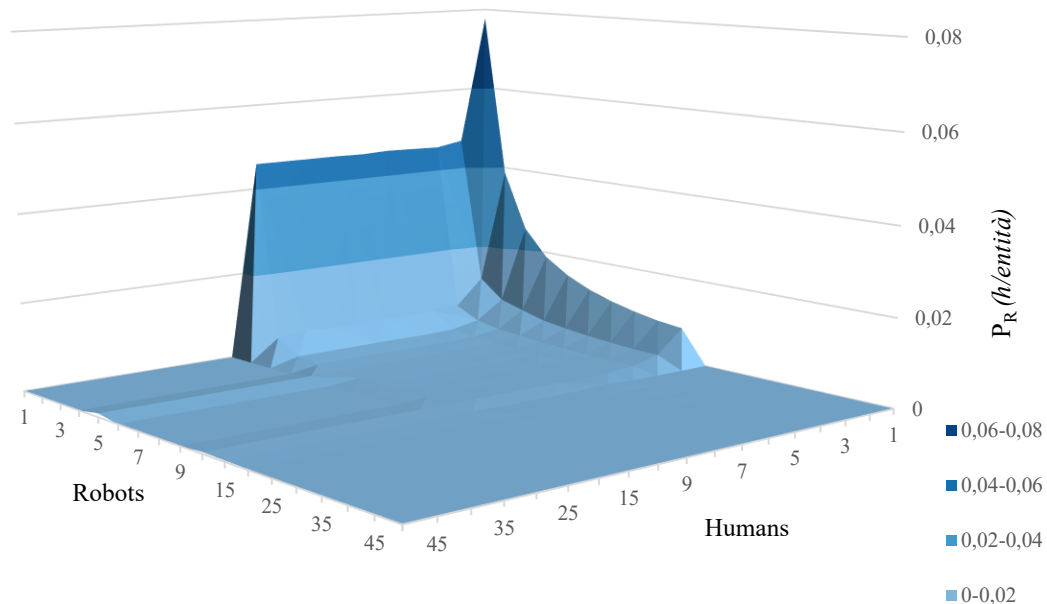


Grafico 8 - P_R in funzione del numero di umani e del numero di robot

Le superfici tridimensionali ottenute mostrano andamenti coerenti con quelli osservati per la produttività espressa in termini di output per unità di risorsa, ma consentono una lettura più immediata in termini di carico di lavoro effettivo delle singole risorse.

In condizioni di sottodimensionamento del sistema, caratterizzate da un numero ridotto di risorse, il tempo imputabile a ciascuna risorsa risulta elevato. In tali configurazioni, ogni operatore o robot è fortemente sollecitato, e il tempo di attraversamento del processo rimane alto a causa dei vincoli strutturali e della limitata parallelizzazione dei task.

All'aumentare del numero di risorse, si osserva una rapida riduzione del tempo medio per skateboard, dovuta alla maggiore disponibilità di capacità produttiva e alla conseguente riduzione dei tempi di attesa, in linea con quanto emerso nell'analisi del tempo di processo al variare delle configurazioni.

Tuttavia, rispetto al grafico relativo al solo tempo di processo, nelle superfici qui considerate si osserva una diminuzione ancora più ripida dell'indicatore, in particolare lungo la direzione associata alla risorsa che compare al denominatore.

Tale comportamento è spiegabile osservando la struttura dell'indicatore stesso: oltre alla riduzione del numeratore T , comune a tutte le configurazioni con aumento delle risorse complessive, il denominatore include esplicitamente il numero di risorse della tipologia analizzata. Di conseguenza, quando l'aumento delle risorse riguarda la stessa tipologia

considerata nell'indicatore, la riduzione del valore è amplificata dall'incremento diretto del denominatore. Al contrario, l'aumento delle risorse dell'altra tipologia influisce sull'indicatore solo in modo indiretto, attraverso la riduzione del tempo di processo complessivo, mentre il denominatore rimane invariato. In configurazioni fortemente sbilanciate verso una sola tipologia di risorsa quindi, si osservano comportamenti marcatamente asimmetrici negli indicatori di produttività normalizzati.

Si riporta infine il grafico tridimensionale relativo alla produttività totale P_{tot} calcolata secondo la logica introdotta in questo sottogruppo di indicatori di produttività, rappresentante il tempo medio associato ad una singola risorsa, senza distinzione tra tipologie, relativo alla produzione di uno skateboard per una determinata configurazione.

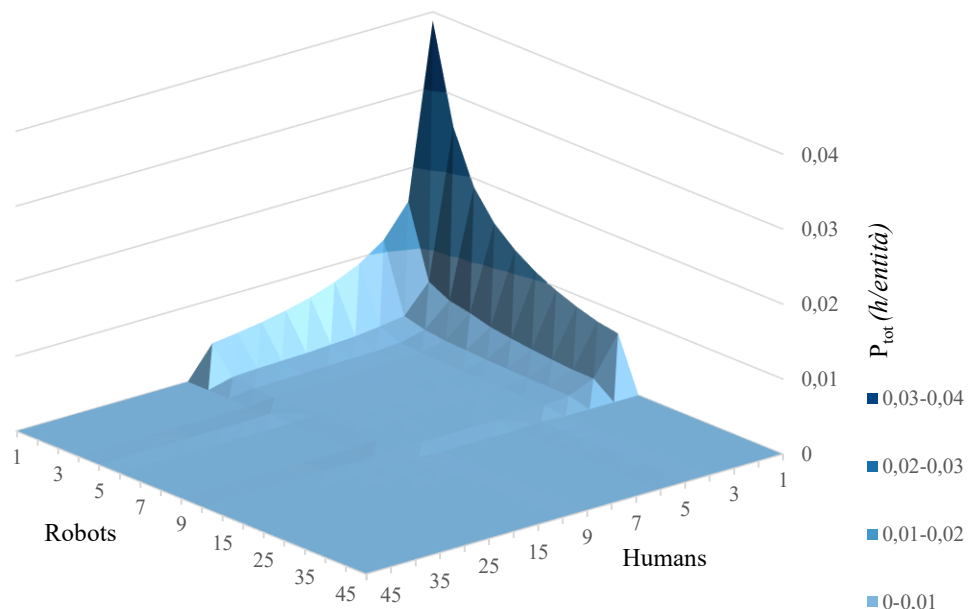


Grafico 9 - P_{tot} in funzione del numero di umani e del numero di robot

Dai dati emerge come, rispetto agli indicatori normalizzati su una singola tipologia di risorsa, le superfici associate a questo indicatore mostrino un andamento sensibilmente più simmetrico rispetto agli assi H e R . Tale simmetrizzazione è riconducibile anche alla struttura del denominatore, che in questo caso include simultaneamente entrambe le tipologie di risorsa.

Infatti, poiché il denominatore cresce con l'aumento sia di H sia di R , l'effetto di normalizzazione agisce in modo analogo lungo entrambe le direzioni del piano delle configurazioni. A differenza degli indicatori precedentemente analizzati, nei quali l'aumento

di una sola tipologia di risorsa produceva un effetto diretto sul denominatore e l'altra agiva esclusivamente attraverso la riduzione del tempo di processo, in questo caso l'incremento di una qualunque risorsa contribuisce direttamente alla riduzione dell'indicatore. Ne consegue una superficie più regolare e meno asimmetrica, nella quale configurazioni caratterizzate dallo stesso numero totale di risorse tendono a presentare valori simili dell'indicatore, indipendentemente dalla specifica ripartizione tra operatori umani e robot.

È infine utile osservare che, a differenza della produttività espressa come numero di skateboard prodotti per unità di tempo e per risorsa, la quale presenta un massimo interno, l'indicatore basato sul tempo medio per skateboard imputabile a una singola risorsa non presenta un minimo interno, ma mostra un progressivo appiattimento all'aumentare del numero di risorse. Ciò è dovuto alla struttura dell'indicatore, in cui l'aumento delle risorse agisce esclusivamente al denominatore, mentre il tempo di processo, una volta raggiunta la saturazione del sistema, tende a stabilizzarsi. Al contrario, nella produttività espressa in termini di output per risorsa, la riduzione del tempo di processo e l'aumento del numero di risorse agiscono in direzioni opposte, dando luogo a un massimo interno.

Nel complesso, l'analisi degli indicatori di produttività, espressi sia in termini di output per risorsa sia come tempo medio imputabile a una singola risorsa per skateboard, consente di mettere in evidenza con chiarezza le principali dinamiche operative del sistema. In particolare, emerge come l'aumento della capacità produttiva conduca inizialmente a significativi miglioramenti delle prestazioni, mentre oltre una certa soglia tali benefici si riducono progressivamente, a causa dei vincoli strutturali del processo e della conseguente presenza di risorse parzialmente o totalmente inutilizzate.

Gli indicatori considerati risultano pertanto efficaci nel descrivere il grado di saturazione delle risorse e nel confronto tra diverse configurazioni dal punto di vista puramente operativo. Tuttavia, valori elevati di produttività o tempi medi particolarmente ridotti non sono di per sé sufficienti per individuare configurazioni preferibili in un contesto decisionale reale. Configurazioni che appaiono efficienti dal punto di vista tecnico possono infatti risultare caratterizzate da un impiego non ottimale delle risorse e da un conseguente aumento dei costi complessivi. Al fine di completare l'analisi, pertanto, nella sezione successiva l'attenzione viene spostata sugli indicatori economici.

6.4.3 Impatto delle risorse sui costi

Per la rappresentazione coerente degli indicatori economici, si è optato nuovamente, come per produttività e tempo di processo per rappresentazioni grafiche tridimensionali in quanto

anche gli indicatori scelti sono funzioni di due variabili: il numero di operatori umani ed il numero di robot collaborativi che operano nel processo di assemblaggio.

Si riportano inizialmente i grafici relativi agli indicatori γ_H e γ_R , i quali sono stati introdotti nel capitolo precedente e rappresentano il costo unitario relativo ad una singola risorsa specifica per produrre un'unità di skateboard.

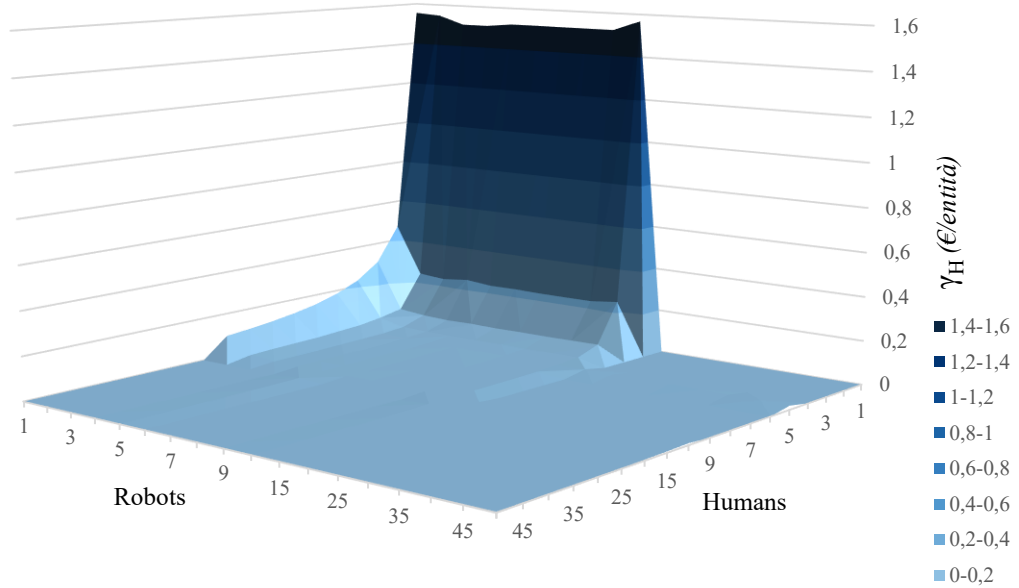


Grafico 10 - γ_H in funzione del numero di umani e del numero di robot

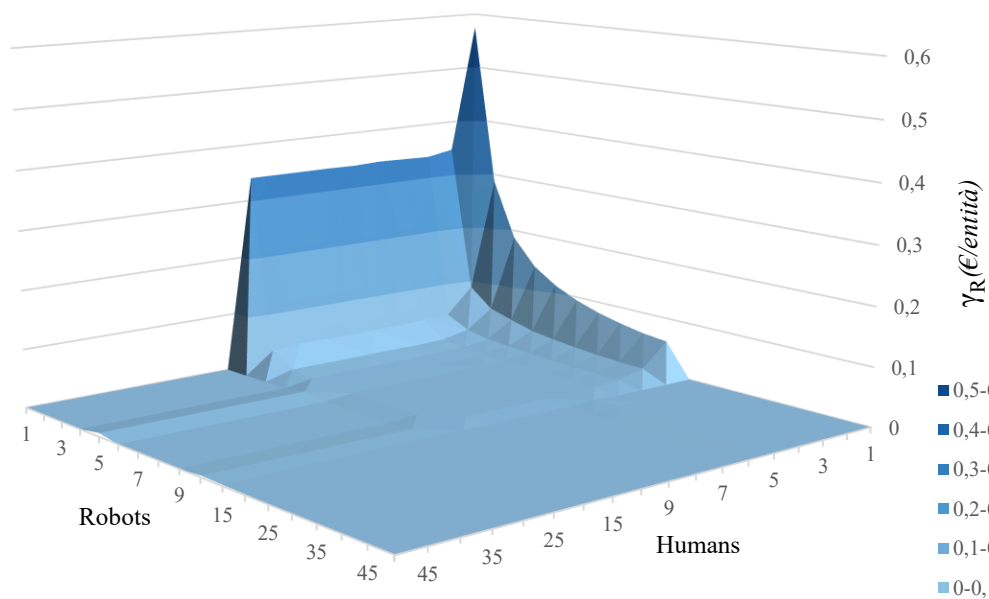


Grafico 11 - γ_R in funzione del numero di umani e del numero di robot

Risulta immediatamente evidente dalla definizione degli indicatori γ_H e γ_R come questi costituiscano una diretta trasformazione degli indicatori di produttività precedentemente analizzati ed in particolare di quelli relativi al tempo medio necessario ad una singola specifica risorsa per la produzione di uno skateboard, ottenuta attraverso una moltiplicazione per il costo orario della risorsa, che per ipotesi risulta costante.

Le superfici tridimensionali relative a γ_H e γ_R presentano infatti un andamento formalmente identico a quello osservato per gli indicatori di tempo medio per skateboard imputabile a una singola risorsa. Tale risultato è atteso, in quanto la struttura matematica degli indicatori di costo ricalca esattamente quella degli indicatori di produttività, differendo unicamente per un fattore di scala. Di conseguenza, tutte le considerazioni relative alla ripidità delle superfici, alla direzione lungo la quale la riduzione dell'indicatore è più marcata e agli effetti di normalizzazione rimangono valide anche nel contesto economico.

Anche in questo contesto è inoltre evidenziabile una maggiore simmetria delle superfici per valori elevati di H e R . Ne deriva che, per configurazioni caratterizzate da un elevato numero totale di risorse, il costo unitario per skateboard attribuibile a una singola risorsa tende a dipendere principalmente dalla capacità complessiva del sistema piuttosto che dal dettaglio della ripartizione tra operatori umani e robot.

È tuttavia importante sottolineare che, valori particolarmente bassi di γ_H e γ_R non devono in questo caso essere interpretati come un segnale di elevata efficienza economica del sistema nel suo complesso. In molte configurazioni, tali valori sono infatti il risultato di un forte sovradimensionamento e della presenza di risorse inutilizzate, che riducono il costo imputabile alla singola risorsa ma non riflettono il costo effettivamente sostenuto dal sistema per la produzione di uno skateboard. In particolare, γ_H e γ_R non rappresentano il costo reale sostenuto dal sistema. Tali espressioni forniscono infatti un costo imputato alle singole risorse sulla base del tempo medio a esse attribuibile T/r , ma non tengono conto del fatto che operatori umani e robot generano un costo per l'intera durata del processo, indipendentemente dal loro effettivo grado di utilizzo.

Il valore informativo degli indicatori analizzati non risiede nella loro capacità di rappresentare il costo reale sostenuto dal sistema, bensì nel fatto che essi consentono di isolare il costo economico medio di una singola risorsa quando viene effettivamente utilizzata alla produzione di uno skateboard, attribuendo a ciascuna risorsa un tempo medio di attività pari a T/r .

Tali indicatori forniscono quindi una misura dell'efficienza operativa relativa media delle risorse umane e robotiche, depurata dall'effetto del costo delle risorse inutilizzate.

Infine, questi indicatori costituiscono un passaggio intermedio necessario verso la definizione del costo reale per skateboard. Essi permettono di comprendere perché configurazioni che appaiono tecnicamente efficienti o presentano bassi costi imputati per risorsa possano risultare economicamente sfavorevoli una volta considerato il costo complessivo del sistema.

Per una valutazione economicamente corretta delle diverse configurazioni risulterà pertanto necessario introdurre un indicatore di costo che consideri il costo orario complessivo del sistema lungo l'intero tempo di produzione.

Per completare l'analisi effettuata in termini di costi associati alla produzione solo per quando le risorse sono effettivamente utilizzate, si riporta la rappresentazione grafica dell'indice C_{work} , cioè dell'indice di costo totale per skateboard relativo al solo utilizzo effettivo delle risorse.

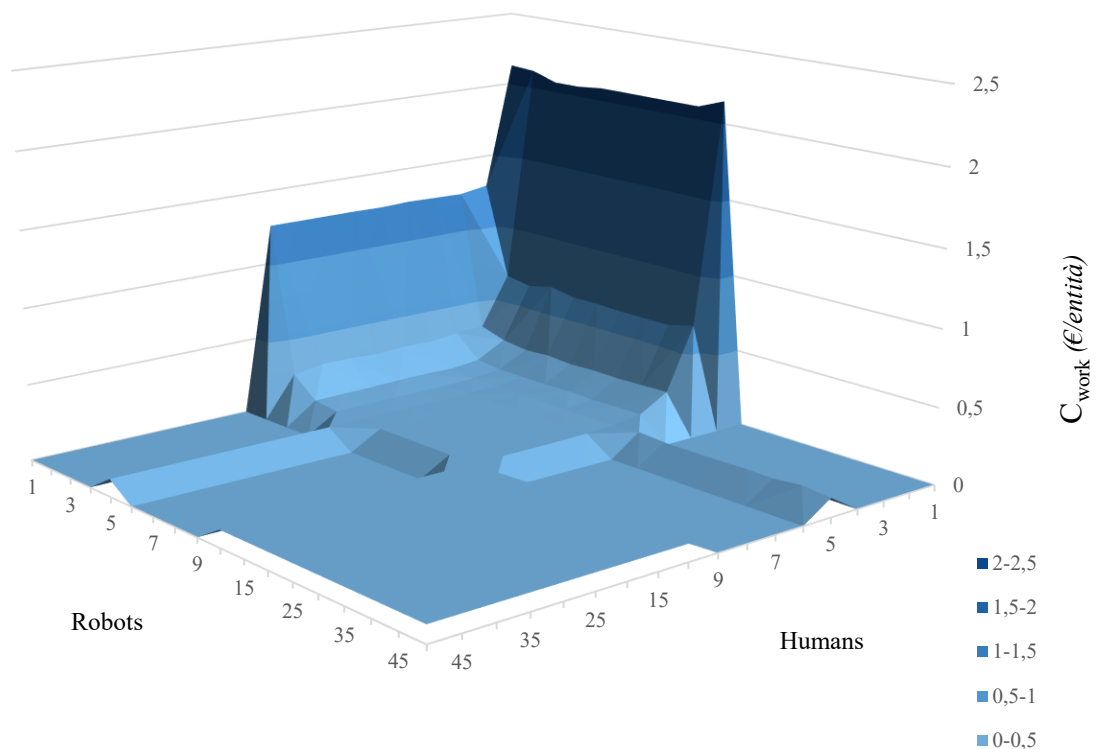


Grafico 12 - C_{work} in funzione del numero di umani e del numero di robot

L'analisi di questa superficie assume un ruolo intermedio nel percorso di valutazione economica, consentendo di comprendere come i costi imputabili alle singole risorse solo quando utilizzate si combinino nelle diverse configurazioni e preparando l'introduzione

dell'indicatore di costo reale per skateboard, che terrà conto del costo orario complessivo del sistema lungo l'intera durata del processo produttivo.

Per completare l'analisi dal punto di vista economico e superare i limiti degli indicatori di costo imputato introdotti nelle sezioni precedenti, si considera ora il costo reale per skateboard C , che rappresenta l'indicatore di riferimento per una valutazione corretta delle diverse configurazioni del sistema. Tale indicatore tiene conto del costo orario complessivo delle risorse umane e robotiche impiegate e dell'intera durata del processo produttivo, includendo quindi anche i periodi di inattività delle risorse.

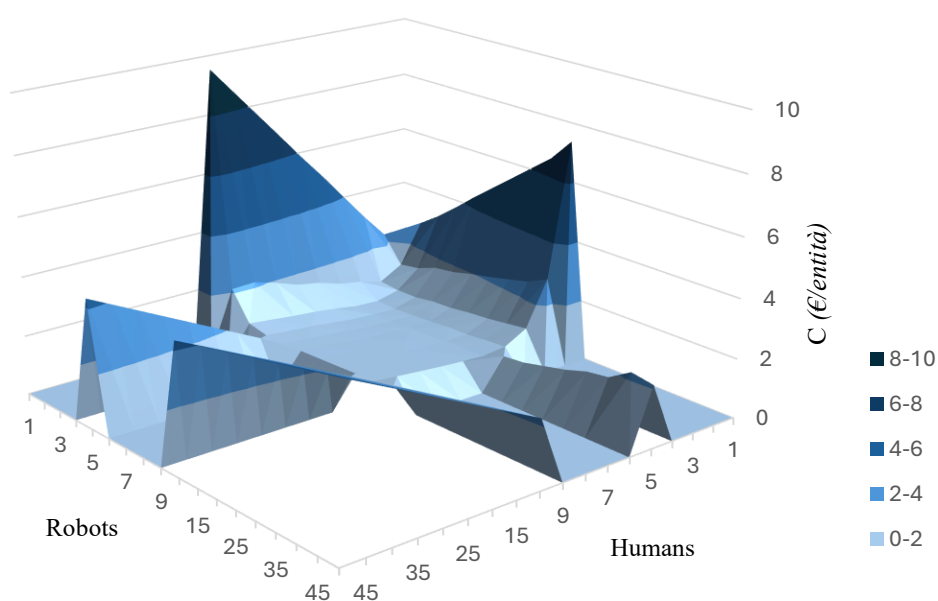


Grafico 13 - C in funzione del numero di umani e del numero di robot

L'andamento della superficie tridimensionale associata al costo reale per skateboard evidenzia chiaramente un comportamento non monotono al variare del numero di operatori umani e robot. Per configurazioni con un numero ridotto di risorse, il costo unitario risulta elevato principalmente a causa dei lunghi tempi di processo, che amplificano il costo orario sostenuto dal sistema. In questa fase, l'incremento delle risorse porta a una riduzione significativa del costo per skateboard, grazie alla diminuzione del tempo di completamento del processo. Si osserva inoltre che il costo unitario reale tende a crescere sensibilmente nelle regioni caratterizzate da forte sbilanciamento del mix di risorse. In tali configurazioni, la tipologia presente in numero estremamente ridotto costituisce il principale collo di bottiglia del sistema, limitando la riduzione del tempo di processo. L'incremento della sola risorsa già abbondante, come evidenziabile osservando il grafico che rappresenta il tempo di processo,

non produce benefici temporali apprezzabili, mentre il costo complessivo continua a crescere linearmente, determinando un peggioramento del costo unitario reale.

Tuttavia, all'aumentare ulteriore del numero di risorse, la riduzione del tempo di processo diventa progressivamente meno rilevante, mentre il costo orario complessivo continua a crescere in modo lineare. Di conseguenza, oltre una certa soglia, a causa dei rendimenti marginali decrescenti del tempo di processo, l'effetto dominante diventa l'aumento dei costi di capacità, e il costo reale per skateboard inizia nuovamente a crescere. Questo comportamento porta all'emergere di una regione di minimo interno, che identifica le configurazioni economicamente più convenienti.

La regione di minimo interno corrisponde a valori intermedi di risorse impiegate, in particolare il punto di minimo assoluto corrisponde all'impiego di 4 umani e 4 robot e consente di produrre skateboard ad un costo delle risorse unitario di 0,60 €/entità.

Dall'analisi dei dati emerge chiaramente che il minimo del costo reale per skateboard non coincide con le configurazioni caratterizzate dal tempo di processo minimo. In particolare, mentre l'ottimo tecnico si colloca in regioni con un numero elevato di risorse, dove il tempo di attraversamento del processo è minimo e fortemente saturato, l'ottimo economico si concentra in configurazioni intermedie, nelle quali il compromesso tra riduzione del tempo e contenimento del costo orario complessivo risulta più favorevole.

Questo risultato evidenzia come l'incremento delle risorse oltre tale regione, pur migliorando marginalmente le prestazioni temporali del sistema, non sia economicamente giustificato alla luce delle ipotesi adottate nel modello. L'analisi conferma quindi che il criterio decisionale finale per la configurazione del sistema non può basarsi esclusivamente sull'ottimizzazione del tempo di processo, ma deve necessariamente tenere conto del trade-off tra prestazioni operative e costi, rendendo esplicita la distinzione tra ottimo tecnico e ottimo economico.

6.5 Discussione complessiva dei risultati della simulazione

L'analisi integrata degli indicatori temporali, di produttività ed economici consente di delineare in modo chiaro il comportamento del sistema di assemblaggio al variare delle configurazioni di risorse umane e robotiche. I risultati evidenziano una marcata non linearità delle prestazioni del sistema e la presenza di rendimenti marginali decrescenti delle performance operative all'aumentare del numero complessivo di risorse impiegate.

Nelle configurazioni inizialmente sottodimensionate, l'incremento delle risorse produce benefici significativi in termini di riduzione del tempo di processo e di miglioramento della

produttività. In questa regione, le risorse rappresentano il principale collo di bottiglia del sistema e la loro aggiunta consente di sfruttare in modo più efficace le possibilità di parallelizzazione offerte dal processo. Tuttavia, superata una certa soglia, il sistema entra in una zona di saturazione in cui ulteriori incrementi delle risorse non determinano miglioramenti apprezzabili delle prestazioni temporali. In tale regime, la struttura del processo e la presenza di task non parallelizzabili limitano l'efficacia dell'aggiunta di nuove risorse, che risultano progressivamente sottoutilizzate.

Questo comportamento evidenzia chiaramente il rischio di sovradimensionamento del sistema: se da un punto di vista tecnico il tempo di processo può risultare prossimo al minimo già per configurazioni intermedie, dal punto di vista economico l'aumento del numero di risorse comporta un incremento dei costi che non è compensato da una corrispondente riduzione dei tempi. Ne consegue che l'ottimo tecnico, identificato come la configurazione che minimizza il tempo di processo, non coincide necessariamente con l'ottimo economico, definito come la configurazione che minimizza il costo reale per unità prodotta.

Le simulazioni consentono inoltre di osservare come, in un sistema Multi-Human Multi-Robot, le prestazioni complessive non siano determinate unicamente dal numero totale di risorse disponibili, ma dipendano in misura significativa anche dalla loro composizione. La scelta del mix tra risorse umane e robotiche influisce infatti sia sull'efficienza temporale raggiungibile nelle fasi iniziali di incremento della capacità produttiva, sia sul comportamento del sistema in prossimità delle regioni di saturazione.

Dal punto di vista manageriale, l'analisi mette in evidenza l'esistenza di una regione di soluzioni robuste dal punto di vista economico, caratterizzata da configurazioni in cui le prestazioni del sistema risultano relativamente poco sensibili a piccole variazioni del numero di risorse. Tale area è individuabile per un mix di risorse bilanciato e intermedio, attorno al valore di minimo assoluto di costo, per il quale si raggiungono tempi di processo già prossimi ai valori asintotici ottenibili con configurazioni caratterizzate da un numero molto elevato di risorse.

Operare in prossimità di tale regione consente di mitigare il rischio associato a incertezze operative e variazioni della domanda, evitando sia condizioni di sottodimensionamento, che comportano tempi elevati e bassa produttività, sia condizioni di sovradimensionamento, che generano costi non giustificati dai benefici ottenuti.

Al fine di sintetizzare in modo organico i risultati emersi dall'analisi simulativa e facilitarne la lettura complessiva, si riporta in Tabella 5 una sintesi dei principali comportamenti osservati nel sistema al variare delle configurazioni di risorse.

Dimensione analizzata	Evidenze emerse	Implicazioni manageriali
Tempo di processo	Il tempo medio di processo diminuisce rapidamente all'aumentare del numero delle risorse nelle configurazioni inizialmente sottodimensionate. Oltre una certa soglia emerge un comportamento asintotico dovuto alla presenza di task non parallelizzabili e alla saturazione del sistema.	L'incremento indiscriminato delle risorse non garantisce benefici temporali proporzionali. È necessario individuare la regione oltre la quale l'aumento della capacità produttiva genera benefici marginali trascurabili.
Produttività del sistema	La produttività media cresce inizialmente con l'aumento delle risorse, raggiunge valori massimi in configurazioni intermedie e successivamente tende a ridursi a causa della diluizione del contributo medio delle singole risorse.	Il dimensionamento ottimale del sistema deve considerare non solo l'aumento dell'output totale, ma anche l'efficienza di impiego delle risorse, evitando configurazioni con elevati livelli di sottoutilizzo.
Costi di produzione	Il costo reale per unità prodotta presenta un minimo interno derivante dal compromesso tra riduzione del tempo di processo e incremento lineare dei costi delle risorse.	L'ottimo economico non coincide con l'ottimo tecnico e le decisioni di investimento devono basarsi su valutazioni integrate tempo-costo. La minimizzazione del solo tempo di produzione conduce a soluzioni economicamente inefficienti
Composizione del mix umano-robot	Le prestazioni del sistema dipendono non solo dal numero totale di risorse, ma anche dal bilanciamento tra operatori umani e robot. Sono state osservate in generale prestazioni migliori per composizioni bilanciate delle due tipologie di risorse.	La progettazione dei sistemi MHMR deve considerare strategie di bilanciamento tra automazione e lavoro umano. Un mix equilibrato consente di sfruttare le complementarità tra le due tipologie di risorse e ridurre inefficienze organizzative.
Regione di soluzioni robuste	È individuabile un'area di configurazioni in cui il costo unitario risulta relativamente stabile rispetto a variazioni moderate del numero di risorse e in cui il tempo di processo è già sufficientemente prossimo al valore di minimo asintotico ed valore di costo reale risulta vicino al minimo interno.	Operare all'interno di tale regione consente di ridurre il rischio operativo e garantire maggiore flessibilità.

Tabella 5 - Sintesi dei risultati di simulazione e delle conseguenti implicazioni manageriali

I risultati confermano quindi l'importanza di un approccio integrato alla progettazione dei sistemi produttivi, nel quale l'analisi del tempo di processo sia affiancata da valutazioni di produttività e di costo. Solo attraverso questa lettura congiunta è possibile individuare configurazioni che rappresentino un compromesso efficace tra efficienza operativa e sostenibilità economica, fornendo indicazioni utili per il supporto alle decisioni in contesti produttivi reali.

Capitolo 7 – Analisi della funzione di complessità di interazione

Come già descritto nei capitoli precedenti, sul tempo di processo del sistema agiscono due effetti contrapposti: da un lato il beneficio derivante dall'aumento delle risorse disponibili, dall'altro l'incremento della complessità organizzativa legata all'interazione tra operatori umani e robotici. Le possibili forme funzionali analizzate atte a rappresentare la complessità di interazione sono state introdotte in precedenza; in questa sezione si descrive invece la metodologia adottata per la determinazione della tipologia di funzione scelta per la descrizione dell'effetto a partire dai risultati di simulazione e si motiva la scelta finale della forma ritenuta più adeguata.

7.1 Procedura per l'individuazione della funzione

La procedura seguita si basa sull'ipotesi che il tempo di processo osservato possa essere scomposto in una componente puramente capacitiva, legata al numero di risorse disponibili, e in una componente correttiva riconducibile alla complessità di interazione.

La forma funzionale generale atta a rappresentare la scomposizione del tempo di processo nei due effetti previsti è stata assunta essere:

$$t_a(H, R) = \frac{t_a(1,0)}{H + \alpha R} * (1 + \beta f_c(H, R)) \quad (7.1)$$

Dove $t_a(1,0)$ rappresenta il tempo di processo ottenuto rendendo operativa una sola risorsa umana, α esprime l'efficienza relativa dei robot rispetto agli operatori umani, β rappresenta un coefficiente di intensità con effetto moltiplicativo sulla funzione $f_c(H,R)$ di complessità di interazione da determinare.

Poiché l'obiettivo dell'analisi non è la stima assoluta dell'intensità della complessità, ma la sua forma funzionale, il coefficiente β è stato normalizzato a 1. Tale scelta consente di assorbire l'effetto di scala all'interno della funzione stessa e di semplificare l'analisi, evitando problemi di identificabilità tra β e $f_c(H,R)$. In questo modo, la funzione stimata rappresenta direttamente la complessità relativa del sistema, coerentemente con l'impostazione comparativa adottata nella tesi.

A partire dai dati di simulazione, la funzione è stata quindi ricavata in forma inversa come:

$$f_c(H, R) = \frac{t_a(H, R) * (H + \alpha R)}{t(1,0)} - 1 \quad (7.2)$$

Considerando inoltre l'assunzione di modello secondo cui indipendentemente dalla tipologia di operazione, gli operatori robotici impiegano il 10% del tempo in più rispetto agli umani, si è definito il parametro α come:

$$\alpha = \frac{1}{0.1}$$

Ottenendo:

$$\alpha = 0.909$$

Questa trasformazione consente di ottenere, per ciascuna configurazione simulata, un valore osservato della complessità di interazione, che può essere utilizzato come variabile dipendente in un modello di regressione. I risultati della $f_c(H, R)$ per ogni configurazione analizzata sono riportati in Appendice C.

Prima di procedere alla stima, alcune forme funzionali sono state escluse a priori.

Le funzioni a elevamento a potenza, pur risultando frequentemente utilizzate per descrivere fenomeni di scala e relazioni non lineari, sono state escluse in quanto implicano una crescita o decrescita monotona della complessità al variare del numero di risorse. Tale struttura non risulta coerente con il comportamento osservato nel sistema simulato, in cui l'aumento delle risorse produce inizialmente benefici organizzativi (riduzione della congestione e migliore distribuzione del carico), seguiti da effetti di saturazione e da un incremento della complessità dovuto al coordinamento.

Le funzioni esponenziali sono state escluse in quanto descrivono dinamiche caratterizzate da una crescita estremamente rapida della variabile dipendente al crescere delle variabili indipendenti. Sebbene tale comportamento possa essere appropriato in contesti di instabilità o di fenomeni cumulativi, esso risulta poco plausibile nel caso in esame. Nel sistema produttivo analizzato, infatti, la complessità di interazione cresce in modo progressivo e controllato, senza evidenze di esplosioni rapide. L'adozione di una funzione esponenziale porterebbe pertanto a una sovrastima della complessità per configurazioni con un numero medio-alto di risorse, in contrasto con i risultati di simulazione.

Le funzioni sigmoidi, pur essendo in grado di rappresentare fenomeni di saturazione, sono state escluse in quanto presuppongono l'esistenza di una soglia critica ben definita oltre la

quale il comportamento del sistema cambia radicalmente, inoltre possono risultare poco sensibili nelle fasi iniziali del processo, sottovalutando la complessità fino al raggiungimento della soglia. Nel caso in esame, invece, la transizione tra regime di sottodimensionamento, zona di miglioramento e area di saturazione avviene in modo graduale e continuo, senza la presenza di un punto di flesso netto riconducibile a un valore specifico delle risorse. L'introduzione di una funzione sigmoide avrebbe pertanto imposto una struttura eccessivamente rigida e poco giustificata dai dati disponibili, oltre a complicare inutilmente l'interpretazione dei parametri stimati.

Le funzioni logaritmiche sono state infine escluse in quanto descrivono una relazione in cui l'effetto marginale delle variabili indipendenti decresce rapidamente già per valori contenuti delle stesse. Tale comportamento implica che l'aggiunta di risorse produca benefici organizzativi rilevanti solo nelle configurazioni iniziali, per poi diventare quasi trascurabile al crescere del sistema. Nel caso analizzato, invece, i risultati di simulazione mostrano che gli effetti legati all'interazione tra risorse umane e robotiche emergono in modo più articolato e non sono riconducibili a una semplice dipendenza dal numero complessivo di risorse. Inoltre, la forma logaritmica non consente di rappresentare in modo esplicito e interpretabile l'interazione tra le due tipologie di risorse, risultando quindi poco adatta a descrivere la complessità di coordinamento uomo-robot che costituisce l'oggetto dell'analisi.

Alla luce di queste considerazioni, tali forme funzionali sono state scartate a priori, concentrando l'analisi su specificazioni in grado di combinare semplicità strutturale, interpretabilità economica e coerenza con le dinamiche osservate nel modello di simulazione.

La stima è stata quindi condotta mediante regressione ai minimi quadrati ordinari, confrontando le specificazioni funzionali sulla base di criteri sia quantitativi sia interpretativi. In particolare, sono stati valutati: la capacità della funzione di riprodurre l'andamento osservato nei dati, la significatività e il segno dei coefficienti stimati, la coerenza economica del modello e il valore del coefficiente di determinazione corretto R^2_{adj} . I risultati relativi alle forme funzionali lineare e lineare con interazione sono riportati in Appendice C.

7.2 Forma funzionale migliore per la modellazione della complessità di interazione

Tra le diverse forme funzionali considerate, quella che ha mostrato il miglior compromesso tra capacità descrittiva, coerenza con la struttura del modello e interpretabilità economica è

risultata essere una forma lineare comprensiva di un termine di interazione tra risorse umane e robotiche, espressa come:

$$\beta f_c(H, R) = c + X_0H + X_1R + X_2HR \quad (7.3)$$

Dove $f_c(H, R)$ rappresenta la funzione di complessità di interazione, H e R indicano rispettivamente il numero di operatori umani e robotici impiegati nel sistema, e i parametri c , X_0 , X_1 , X_2 sono stati stimati tramite regressione ai minimi quadrati ordinari.

È opportuno sottolineare che, in fase di modellazione concettuale, la complessità di interazione tra risorse umane e robotiche era inizialmente interpretata come un fattore prevalentemente penalizzante, associato a un incremento dei tempi di processo dovuto a esigenze di coordinamento, sincronizzazione e gestione di agenti eterogenei. In questa prospettiva, la complessità di interazione era attesa crescere all'aumentare del numero di risorse coinvolte.

L'analisi empirica dei risultati di simulazione suggerisce tuttavia una lettura più articolata. In particolare, per configurazioni caratterizzate da un numero ridotto di risorse, l'introduzione congiunta di operatori umani e robotici non comporta un aumento della complessità di interazione, ma contribuisce inizialmente a una sua riduzione. In questa fase, l'effetto dominante è legato al superamento di condizioni di sottodimensionamento del sistema: l'aumento delle risorse migliora la fluidità del processo, riduce le attese e attenua rigidità decisionali, con un impatto complessivamente positivo anche sulla componente di complessità.

Solo al crescere della scala del sistema emerge progressivamente l'effetto associato alla coordinazione tra risorse eterogenee, che si manifesta come una riduzione dei benefici marginali ottenibili dall'incremento congiunto di operatori umani e robotici. La complessità di interazione, pertanto, non si configura come un fattore puramente crescente, ma come una componente non lineare che inizialmente viene compensata dai benefici organizzativi dell'aumento di capacità e che diventa rilevante solo oltre determinate soglie dimensionali. È opportuno inoltre chiarire fin da subito che la capacità esplicativa del modello risulta limitata: il valore di R^2_{adj} , pari a circa 0,21, indica che la funzione stimata spiega solo una quota contenuta della variabilità osservata. Questo risultato non rappresenta una criticità inattesa, ma riflette la natura dello studio effettuato. La funzione di complessità non è infatti concepita per spiegare l'intero andamento del tempo di processo del sistema, che nel modello è già fortemente determinato dal termine principale legato alla capacità produttiva complessiva, espresso dalla combinazione $H + \alpha R$. Al contrario, l'obiettivo della sua

introduzione è quello di verificare se, a parità di capacità produttiva, esista una relazione sistematica, interpretabile e coerente tra il numero di risorse umane e robotiche e la complessità di interazione emergente dal sistema simulato. In questo senso, la funzione di complessità agisce su una componente residuale del comportamento del sistema, catturando effetti legati al coordinamento, alla gestione delle risorse eterogenee e alle rigidità operative che non sono esplicitamente modellate nel termine principale del tempo di processo. La variabilità osservata nei dati, e riflessa nei valori della funzione stimata, è pertanto influenzata non solo dalla struttura funzionale adottata, ma anche dalla natura stocastica della simulazione, dalle dinamiche di coda e dalle decisioni locali di assegnazione delle risorse, che introducono effetti emergenti difficilmente riconducibili a una relazione analitica semplice.

L'obiettivo della regressione non è una previsione puntuale del valore della complessità per ogni configurazione possibile, bensì l'identificazione di una struttura funzionale in grado di cogliere le tendenze principali e i meccanismi dominanti che governano l'evoluzione della complessità di interazione al variare del numero di operatori umani e robotici.

Il valore di R^2_{adj} ottenuto, pur modesto, risulta il più elevato tra le forme funzionali testate e suggerisce la presenza di una struttura debole ma non puramente casuale nei dati.

L'intercetta positiva c rappresenta un livello di complessità di base del sistema, presente anche per configurazioni con un numero ridotto di risorse, e riflette l'inevitabile complessità intrinseca legata all'organizzazione del processo e alla logica di assegnazione dei task, indipendentemente dalla scala del sistema.

I coefficienti associati ai termini lineari in H e R risultano negativi, indicando che l'aumento di una singola tipologia di risorsa, a parità dell'altra, tende, almeno per valori contenuti della seconda risorsa, a ridurre l'incidenza relativa della complessità di interazione. In termini operativi, l'incremento di risorse di una sola tipologia consente di assorbire più agevolmente rigidità decisionali e tempi di attesa legati alla disponibilità degli operatori.

Al contrario, il coefficiente positivo associato al termine di interazione $H*R$ evidenzia come la crescita congiunta di risorse umane e robotiche introduca una complessità aggiuntiva, legata alla necessità di coordinare agenti eterogenei caratterizzati da tempi di esecuzione differenti. Tale termine cattura il costo organizzativo e decisionale legato al coordinamento tra agenti eterogenei.

L'interpretazione è ulteriormente chiarita osservando le derivate parziali della funzione rispetto alle due variabili:

$$\frac{\partial f_c}{\partial H} = X_0 + X_2 R \quad (7.4)$$

$$\frac{\partial f_c}{\partial R} = X_1 + X_2 H \quad (7.5)$$

Per valori contenuti dell'altra tipologia di risorsa, sostituendo i valori numerici dei coefficienti ottenuti, si osserva che l'effetto marginale di un incremento di H o R risulta negativo, indicando una riduzione complessiva della complessità di interazione. All'aumentare dell'altra tipologia, tuttavia, il termine di interazione rende tale riduzione via via meno marcata. In altre parole, l'aggiunta congiunta di risorse continua a ridurre la complessità complessiva del sistema, ma con rendimenti marginali decrescenti. Oltre determinate soglie l'effetto complessivo può annullarsi e, in media, tendere a invertire di segno, portando ad un contributo positivo del tempo di processo dovuto alla complessità di interazione.

Questo risultato non va interpretato come una soglia netta oltre la quale la complessità aumenta in modo deterministico, ma piuttosto come una tendenza media coerente con il comportamento osservato nelle simulazioni e con la natura approssimata della funzione stimata.

Nel complesso, la forma lineare con interazione si dimostra adeguata dal punto di vista interpretativo. Essa consente di rappresentare in modo parsimonioso l'effetto combinato delle risorse umane e robotiche, mantenendo una chiara leggibilità economica dei parametri e garantendo coerenza con la struttura teorica del modello di simulazione.

7.3 Discussione complessiva dei risultati sulla ricerca della funzione di complessità di interazione

I risultati dell'analisi simulativa hanno evidenziato un comportamento fortemente non lineare delle prestazioni del sistema al variare della numerosità e della composizione delle risorse coinvolte.

Un utile strumento di interpretazione dei risultati ottenuti è rappresentato dall'analisi della forma funzionale adottata per descrivere la complessità di interazione tra risorse umane e robotiche. L'introduzione esplicita di un termine di interazione consente infatti di esplorare in modo strutturato come la compresenza delle due tipologie di risorse influenzi il comportamento complessivo del sistema, contribuendo a spiegare parte della non linearità osservata nei risultati di simulazione.

In particolare, la funzione stimata suggerisce che l'aumento congiunto di risorse umane e robotiche non determini necessariamente, nelle fasi iniziali, un incremento della complessità di interazione. Al contrario, per configurazioni caratterizzate da un numero contenuto di risorse, l'integrazione tra agenti eterogenei può favorire una maggiore flessibilità operativa e una migliore distribuzione dei carichi decisionali, traducendosi in una riduzione della complessità relativa. Tuttavia, al crescere della scala del sistema, l'effetto positivo tende progressivamente ad attenuarsi: la necessità di coordinare un numero crescente di risorse con caratteristiche operative differenti introduce rigidità organizzative che limitano i benefici marginali ottenibili.

La forma funzionale selezionata non deve pertanto essere interpretata come una descrizione esaustiva o deterministica della complessità del sistema, bensì come uno strumento di sintesi interpretativa, in grado di cogliere tendenze medie e meccanismi emergenti coerenti con il comportamento osservato nelle simulazioni. In questo senso, essa fornisce un supporto analitico complementare agli indicatori temporali, di produttività ed economici, contribuendo a una comprensione più articolata delle dinamiche di integrazione uomo-robot senza pretendere di esaurirne la complessità reale.

Al fine di sintetizzare le principali evidenze emerse dall'analisi empirica della complessità di interazione tra risorse umane e robotiche, si riporta in Tabella 6 una sintesi dei risultati interpretativi ottenuti.

Aspetto analizzato	Evidenza emersa	Interpretazione
Natura della complessità di interazione	La complessità non cresce in modo monotono con il numero di risorse	La relazione tra dimensione del sistema e complessità organizzativa è non lineare
Effetto nelle configurazioni sottodimensionate	L'aumento congiunto delle risorse riduce la complessità relativa	Il superamento di colli di bottiglia migliora la fluidità operativa e la distribuzione dei carichi
Effetto nelle configurazioni ad alta scala	L'aumento simultaneo di risorse umane e robotiche introduce rigidità di coordinamento	Emergono inefficienze organizzative legate alla gestione di agenti eterogenei
Ruolo del coordinamento uomo-robot	La compresenza delle due tipologie di risorse genera effetti specifici non riconducibili alla sola numerosità totale	Il coordinamento tra agenti con caratteristiche operative differenti rappresenta una componente autonoma della complessità
Struttura funzionale identificata	La funzione lineare con termine di interazione rappresenta il miglior compromesso interpretativo	La complessità emerge dall'effetto combinato delle due tipologie di risorse
Capacità esplicativa del modello	La funzione spiega una quota limitata della variabilità osservata	La complessità rappresenta una componente residuale influenzata anche da dinamiche stocastiche e decisionali
Significato economico dei parametri	I termini lineari riflettono benefici organizzativi iniziali, mentre il termine di interazione cattura i costi di coordinamento	Il modello consente una lettura economico-organizzativa dei fenomeni emergenti

Tabella 6 - Sintesi delle evidenze emerse dall'analisi sulla funzione di complessità

Capitolo 8 – Conclusioni, limiti dell'analisi e sviluppi futuri

8.1 Conclusioni

La presente tesi ha affrontato il problema della dell'analisi di sistemi produttivi caratterizzati dalla coesistenza di risorse umane e robotiche, con l'obiettivo di comprendere in che modo il numero e il mix di tali risorse influenzino le prestazioni complessive del sistema in termini di tempo di processo, produttività e costi. In particolare, l'attenzione è stata rivolta allo studio di configurazioni Multi-Human Multi-Robot applicate a un processo di assemblaggio rappresentativo, analizzato mediante la simulazione a eventi discreti.

Attraverso la costruzione di un modello di simulazione flessibile, generato utilizzando come supporto il software *Arena*, è stato possibile valutare un ampio insieme di configurazioni, variando il numero di operatori umani e robotici e osservandone l'impatto sui principali indicatori di prestazione introdotti. L'approccio adottato ha consentito di superare una visione puramente deterministica del processo produttivo, integrando la variabilità operativa, le logiche di assegnazione delle risorse e le interazioni tra agenti eterogenei.

L'analisi dei risultati ha evidenziato come il comportamento del sistema sia fortemente non lineare e caratterizzato da rendimenti marginali decrescenti. Nelle configurazioni inizialmente sottodimensionate, l'incremento delle risorse produce benefici significativi in termini di riduzione del tempo di processo e aumento della produttività, confermando il ruolo delle risorse come principale collo di bottiglia del sistema. Tuttavia, superata una certa soglia, l'aggiunta di ulteriori risorse non determina miglioramenti temporali apprezzabili, portando il sistema in una regione di saturazione in cui le prestazioni risultano limitate dalla struttura del processo e dalla presenza di task non completamente parallelizzabili.

Questo risultato mette in luce una distinzione fondamentale tra ottimo tecnico e ottimo economico. Se da un lato l'ottimo tecnico è individuabile come la configurazione che minimizza il tempo di processo, dall'altro l'ottimo economico emerge come il punto di equilibrio tra la riduzione dei tempi e l'aumento dei costi legato alla disponibilità delle risorse. L'analisi economica ha mostrato chiaramente come l'incremento del numero di risorse comporti un aumento lineare dei costi, a fronte di benefici temporali progressivamente ridotti, rendendo individuabile un minimo interno della funzione di costo reale per unità prodotta.

La tesi ha inoltre affrontato il tema della complessità di interazione tra risorse umane e robotiche, proponendo una metodologia per la stima della funzione che ne descrive l'andamento in relazione al numero di agenti coinvolti. È stato possibile valutare diverse forme funzionali candidate e selezionare una rappresentazione coerente dal punto di vista sia statistico sia interpretativo. In particolare, la funzione lineare con termine di interazione si è dimostrata in grado di catturare efficacemente gli effetti incrementali e combinati derivanti dalla presenza simultanea di più risorse, mantenendo al contempo una struttura semplice e facilmente interpretabile.

Dal punto di vista gestionale, i risultati suggeriscono che le configurazioni più interessanti non coincidono necessariamente con quelle estreme, ma si collocano in una regione intermedia caratterizzata da un giusto equilibrio tra benefici derivanti dall'introduzione di nuove risorse e costi legati all'aumento delle stesse. Nel caso studiato il risultato economicamente più conveniente è una regione intermedia caratterizzata da un mix bilanciato di risorse. Operare in prossimità di tale regione consente di evitare sia condizioni di sottodimensionamento, associate a tempi elevati e bassa produttività, sia condizioni di sovradimensionamento, che comportano costi non giustificati dai benefici ottenuti.

Nel complesso, il lavoro svolto conferma l'importanza di un approccio integrato alla progettazione dei sistemi produttivi, in cui l'analisi temporale sia affiancata da valutazioni di produttività, costo e complessità di interazione. La simulazione a eventi discreti si dimostra uno strumento particolarmente efficace per supportare questo tipo di analisi, consentendo di esplorare scenari complessi e di fornire indicazioni utili per il supporto alle decisioni in contesti industriali reali.

8.2 Limiti dell'analisi e sviluppi futuri

Nonostante i risultati ottenuti forniscano indicazioni significative sul comportamento del sistema analizzato, è opportuno evidenziare alcuni limiti dell'approccio adottato, che delimitano il campo di validità delle conclusioni e suggeriscono possibili direzioni di sviluppo futuro.

Un primo limite riguarda la modellazione economica del sistema. Nell'analisi svolta, la valutazione delle configurazioni è stata condotta principalmente sulla base del costo reale per unità prodotta, definito come costo di disponibilità delle risorse durante il tempo di processo. Tale indicatore consente di individuare un ottimo economico interno, ma non coincide necessariamente con una valutazione completa della redditività del sistema, non esaurendo la complessità delle decisioni economiche reali. In particolare, un costo unitario

minore non implica automaticamente un profitto maggiore, poiché il modello non considera esplicitamente il lato dei ricavi né eventuali vincoli di servizio legati ai tempi di consegna. In contesti produttivi reali, infatti, il rispetto di tempi massimi di consegna può rappresentare un vincolo cruciale. Configurazioni caratterizzate da costi inferiori ma tempi di processo elevati potrebbero infatti generare penalità contrattuali, perdita di competitività, riduzione della fidelizzazione del cliente o danni reputazionali, con effetti indiretti che il modello non è in grado di catturare. In questo senso, l'ottimo economico individuato nella presente analisi deve essere interpretato come un ottimo condizionato all'assenza di vincoli di servizio, e non come una soluzione universalmente dominante.

Un secondo limite riguarda la rappresentazione semplificata della struttura decisionale di assegnazione delle risorse. Nel modello implementato in *Arena*, una volta determinata la tipologia di risorsa più appropriata per l'esecuzione di un task sulla base del criterio decisionale adottato, il sistema attende la disponibilità di una risorsa di tale tipologia, anche nel caso in cui risorse di tipologia alternativa siano immediatamente disponibili. Questa logica di prelievo, sebbene coerente con l'obiettivo di mantenere separati i criteri di valutazione delle risorse, non tiene conto di possibili strategie decisionali più sofisticate.

In particolare, il modello non contempla l'opportunità di assegnare il task a una risorsa alternativa qualora la risorsa preferita risulti occupata. In contesti reali, una decisione di questo tipo potrebbe essere influenzata da informazioni aggiuntive, quali il tempo residuo di occupazione della risorsa preferita, la durata attesa del task, o il livello di congestione del sistema. Pur nell'ottica di una modellazione volutamente semplificata, l'introduzione di una logica decisionale più articolata, in grado di considerare ulteriori fattori rilevanti, potrebbe costituire un'estensione significativa del presente studio.

Un'ulteriore limitazione è legata alla parametrizzazione delle risorse. I parametri adottati per i tempi di lavorazione, il carico ergonomico e il tasso di errore sono stati scelti per rappresentare uno scenario operativo plausibile, ma non derivano da una calibrazione su dati industriali reali. Di conseguenza, i risultati ottenuti devono essere interpretati come indicativi delle tendenze del sistema piuttosto che come stime puntuali immediatamente trasferibili a un contesto reale senza un'ulteriore fase di validazione.

Infine, per quanto riguarda la modellazione della complessità di interazione, la funzione stimata fornisce una rappresentazione empirica coerente con i dati di simulazione, ma rimane dipendente dalla struttura del modello e dalle ipotesi adottate. Il valore relativamente contenuto del coefficiente R^2_{adj} evidenzia come una parte della variabilità del tempo di

processo sia riconducibile a fattori non esplicitamente modellati, suggerendo la possibilità di arricchire ulteriormente la descrizione delle interazioni tra agenti.

Alla luce di tali considerazioni, diversi sviluppi futuri risultano particolarmente rilevanti. Un'estensione naturale del lavoro consisterebbe nell'introduzione di politiche di assegnazione delle risorse maggiormente dinamiche e adattive rispetto a quelle adottate, in grado di confrontare in tempo reale l'attesa per una risorsa preferita con l'assegnazione a una risorsa alternativa sulla base di criteri di convenienza temporale o economica. Ulteriori sviluppi potrebbero includere l'integrazione di funzioni di profitto che tengano conto di ricavi e penalità di servizio, nonché la validazione del modello su dati industriali reali, al fine di rafforzarne l'applicabilità pratica come strumento di supporto alle decisioni in sistemi produttivi complessi.

Nel suo complesso, il lavoro svolto mostra come l'impiego congiunto della simulazione a eventi discreti e di indicatori sintetici di prestazione consenta di analizzare in modo strutturato sistemi produttivi caratterizzati dalla compresenza di risorse umane e robotiche. L'approccio adottato non è finalizzato alla definizione di soluzioni puntuali valide in senso assoluto, ma alla costruzione di un quadro interpretativo in grado di supportare il confronto tra configurazioni alternative e di mettere in luce le relazioni tra capacità produttiva, produttività e costi.

In questo senso, il valore principale del modello risiede nella sua flessibilità e nella possibilità di adattamento a contesti differenti, sia in termini di caratteristiche delle risorse sia di struttura del processo. La separazione concettuale tra prestazioni tecniche, indicatori di produttività ed elementi economici consente inoltre di leggere i risultati da prospettive complementari, evitando valutazioni parziali basate su un unico criterio.

Il lavoro fornisce quindi una base metodologica su cui costruire analisi più approfondite e mirate, ponendo l'accento sull'importanza di considerare congiuntamente aspetti operativi ed economici nella progettazione e nella gestione di sistemi produttivi complessi.

Appendice A - Modello di simulazione

La presente Appendice raccoglie in forma sistematica i principali dati di input, i parametri di modellazione e ipotesi quantitative adottate per la costruzione del modello di simulazione. Lo scopo dell'Appendice è duplice: da un lato garantire trasparenza e replicabilità dell'analisi condotta, dall'altro alleggerire il corpo principale della tesi, evitando di appesantire la trattazione con informazioni numeriche di dettaglio.

A.1 – Elenco dei task elementari del processo di assemblaggio

In questa sezione è riportato l'elenco completo dei task elementari che costituiscono il processo di assemblaggio modellato. Ciascun task rappresenta una operazione indivisibile dal punto di vista del modello, alla quale viene assegnata una singola risorsa (umana o robotica) secondo il criterio decisionale descritto nel corpo della tesi.

Per ogni task sono indicati il suo identificativo, il codice del componente sul quale viene effettuato, la tipologia di task e la descrizione del componente (Tabella 7).

Order	Component code	Task type	Component description
1	B	Identification and handling	Base plate
2	C1	Identification and handling	Kingpin
3	C1	Alignment	Kingpin
4	RB	Identification and handling	Top cup
5	RB	Alignment	Top cup
6	GB	Identification and handling	Bushing
7	GB	Alignment	Bushing
8	GN	Identification and handling	Pivot bushing
9	GN	Alignment	Pivot bushing
10	A	Identification and handling	Hanger
11	A	Alignment	Hanger
12	GS	Identification and handling	Cone bushing
13	GS	Alignment	Cone bushing
14	RS	Identification and handling	Lower cup
15	RS	Alignment	Lower cup
16	D1	Identification and handling	Kingpin nut
17	D1	Joining	Kingpin nut
18	ST	Identification and handling	Sub-assembled truck
19	B	Identification and handling	Base plate
20	C1	Identification and handling	Kingpin
21	C1	Alignment	Kingpin
22	RB	Identification and handling	Top cup
23	RB	Alignment	Top cup
24	GB	Identification and handling	Bushing
25	GB	Alignment	Bushing

Order	Component code	Task type	Component description
26	GN	Identification and handling	Pivot bushing
27	GN	Alignment	Pivot bushing
28	A	Identification and handling	Hanger
29	A	Alignment	Hanger
30	GS	Identification and handling	Cone bushing
31	GS	Alignment	Cone bushing
32	RS	Identification and handling	Lower cup
33	RS	Alignment	Lower cup
34	D1	Identification and handling	Kingpin nut
35	D1	Joining	Kingpin nut
36	ST	Identification and handling	Sub-assembled truck
37	T	Identification and handling	Skateboard deck
38	ST	Identification and handling	Sub-assembled truck
39	ST	Alignment	Sub-assembled truck
40	C2	Identification and handling	Deck bolt
41	C2	Alignment	Deck bolt
42	C2	Joining	Deck bolt
43	D2	Identification and handling	Deck nut
44	D2	Alignment	Deck nut
45	C2	Joining	Deck bolt
46	C2	Identification and handling	Deck bolt
47	C2	Alignment	Deck bolt
48	C2	Joining	Deck bolt
49	D2	Identification and handling	Deck nut
50	D2	Alignment	Deck nut
51	C2	Joining	Deck bolt
52	C2	Identification and handling	Deck bolt
53	C2	Alignment	Deck bolt
54	C2	Joining	Deck bolt
55	D2	Identification and handling	Deck nut
56	D2	Alignment	Deck nut
57	C2	Joining	Deck bolt
58	C2	Identification and handling	Deck bolt
59	C2	Alignment	Deck bolt
60	C2	Joining	Deck bolt
61	D2	Identification and handling	Deck nut
62	D2	Alignment	Deck nut
63	C2	Joining	Deck bolt
64	ST	Identification and handling	Sub-assembled truck
65	ST	Alignment	Sub-assembled truck
66	C2	Identification and handling	Deck bolt
67	C2	Alignment	Deck bolt
68	C2	Joining	Deck bolt
69	D2	Identification and handling	Deck nut
70	D2	Alignment	Deck nut

Order	Component code	Task type	Component description
71	C2	Joining	Deck bolt
72	C2	Identification and handling	Deck bolt
73	C2	Alignment	Deck bolt
74	C2	Joining	Deck bolt
75	D2	Identification and handling	Deck nut
76	D2	Alignment	Deck nut
77	C2	Joining	Deck bolt
78	C2	Identification and handling	Deck bolt
79	C2	Alignment	Deck bolt
80	C2	Joining	Deck bolt
81	D2	Identification and handling	Deck nut
82	D2	Alignment	Deck nut
83	C2	Joining	Deck bolt
84	C2	Identification and handling	Deck bolt
85	C2	Alignment	Deck bolt
86	C2	Joining	Deck bolt
87	D2	Identification and handling	Deck nut
88	D2	Alignment	Deck nut
89	C2	Joining	Deck bolt
90	R	Identification and handling	Wheel
91	R	Alignment	Wheel
92	D3	Identification and handling	Axle nut
93	D3	Joining	Axle nut
94	R	Identification and handling	Wheel
95	R	Alignment	Wheel
96	D3	Identification and handling	Axle nut
97	D3	Joining	Axle nut
98	R	Identification and handling	Wheel
99	R	Alignment	Wheel
100	D3	Identification and handling	Axle nut
101	D3	Joining	Axle nut
102	R	Identification and handling	Wheel
103	R	Alignment	Wheel
104	D3	Identification and handling	Axle nut
105	D3	Joining	Axle nut
106	D3	Checking	Axle nut
107	D3	Adjustment	Axle nut
108	CP	Identification and handling	Complete product

Tabella 7 - Task di assemblaggio

A.2 – Parametri delle distribuzioni triangolari

La variabilità dei principali fattori operativi è stata modellata attraverso distribuzioni triangolari, definite da un valore minimo, uno massimo e un valore modale.

Tale scelta consente di rappresentare in modo semplice ma realistico fenomeni caratterizzati da una concentrazione delle osservazioni attorno a una condizione più probabile.

Per ciascuna variabile sono stati definiti parametri distinti per risorse umane e robotiche, al fine di riflettere le differenti caratteristiche operative delle due tipologie di agenti.

Per il tempo di processo dei robot, data la scelta di assegnazione effettuata nel modello, si riporta esclusivamente il coefficiente moltiplicativo di efficienza, il quale è utilizzato moltiplicandolo per il tempo corrispondente calcolato per l'operatore umano.

A.2.1 – Tempo di processo

Sono riportati in Tabella 8 i parametri delle distribuzioni triangolari del tempo di processo degli operatori umani, distinguendo per tipologia di task eseguito. Il coefficiente moltiplicativo di efficienza relativa dei robot rispetto agli umani è pari a 1.1: i robot impiegano mediamente il 10% in più rispetto agli umani per lo svolgimento delle operazioni. Si ricorda inoltre che il robot non sono stati considerati in grado di eseguire task di tipologia *joining*, *checking* e *adjustment*.

TimeHuman (min)	Min	Mode	Max
Identification and handling	5	6	7
Alignment	9	10	11
Joining	12	14	16
Checking	7	8	9
Adjustment	5	6	7

Tabella 8 - Parametri delle distribuzioni triangolari del tempo di processo degli umani per tipologia di task

A.2.2 – Tasso di errore

In modo analogo, sono riportati in Tabella 9 i parametri delle distribuzioni triangolari associate al tasso di errore, differenziati in questo caso unicamente tra risorse umane e robotiche.

Il tasso di errore rappresenta la probabilità che un task sia eseguito in modo non corretto.

(%)	Min	Mode	Max
ErrorHuman	9	10	11
ErrorRobot	5	7,5	10

Tabella 9 - Parametri delle distribuzioni triangolari associate al tasso di errore

A.2.3 – Carico ergonomico

Anche il carico ergonomico associato all'esecuzione dei task è stato modellato esclusivamente come attributo delle risorse, assumendo valori differenti per operatori umani e robotici (Tabella 10).

(%)	Min	Mode	Max
LoadHuman	30	35	40
LoadRobot	10	12,5	15

Tabella 10 - Parametri delle distribuzioni triangolari associate al carico ergonomico

A.3 – Pesi dell'indice decisionale Q

In questa sezione sono riportati i valori dei pesi associati all'indice decisionale Q , utilizzato per l'assegnazione dei task alle risorse (Tabella 11). I parametri A , B e C rappresentano rispettivamente il peso attribuito al tempo di esecuzione, al tasso di errore ed al carico ergonomico.

I pesi sono stati definiti separatamente per operatori umani e robotici, al fine di riflettere differenti priorità decisionali nelle due tipologie di risorsa.

Come discusso nel corpo della tesi, tali pesi non modificano le prestazioni fisiche delle risorse, ma modellano le preferenze del sistema nel processo di assegnazione.

	A	B	G
Human	0,3	0,4	0,3
Robot	0,45	0,5	0,05

Tabella 11 - Pesi associati all'indice decisionale Q

Si è scelto di fissare un valore del parametro C pari a 0,2 (20%), introducendo una quantità di incertezza che renda meno rigida o deterministico il calcolo dell'indice decisionale Q la conseguente assegnazione dei task. In questo modo il modello è in grado di riflettere in maniera più realistica contesti operativi caratterizzati da informazioni imperfette e decisioni locali non perfettamente ottimizzate.

A.4 – Procedura di determinazione dei costi orari delle risorse

La presente sezione riporta la procedura adottata per la determinazione dei costi orari delle risorse umane (Tabella 12) e robotiche (Tabella 13) utilizzate nel modello di simulazione.

I costi non rappresentano un caso industriale specifico, ma sono stati definiti al fine di costruire uno scenario economico plausibile e coerente con l'obiettivo comparativo dell'analisi.

Human	
Costo annuo lordo (€)	42000
Ore settimanali (h/sett)	40
Settimane annue (sett/anno)	52
Ore annue (h/anno)	2080
c_H (€/h)	20,0

Tabella 12 - Determinazione del costo orario delle risorse umane

Robot	
Costo acquisto (€)	50000
Periodo ammortamento (anno)	5
Ammortamento annuo (€/anno)	10000
Manutenzione annua (€/anno)	5000
Consumo robot (kW)	1
Costo energia (€/kWh)	0,2
Costo consumo energetico (€/h)	0,2
Ore settimanali	40
Settimane annue	52
Ore annue	2080
c_R (€/h)	7,4

Tabella 13 - Determinazione del costo orario delle risorse robotiche

A.5 – Modello di simulazione completo

Si riporta nella presente sezione la rappresentazione completa del modello di simulazione sviluppato sul software *Arena* (Figura 47, Figura 48, Figura 49, Figura 50).

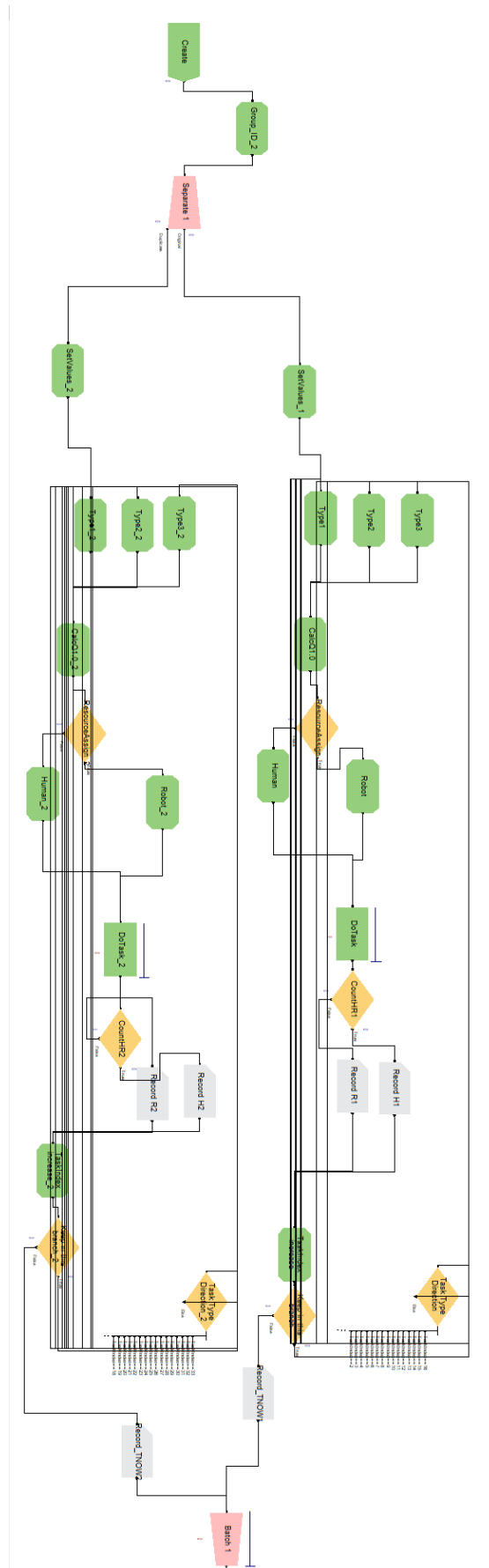


Figura 47 - Modello di simulazione implementato in Arena (parte 1) (fonte: Elaborazione propria)

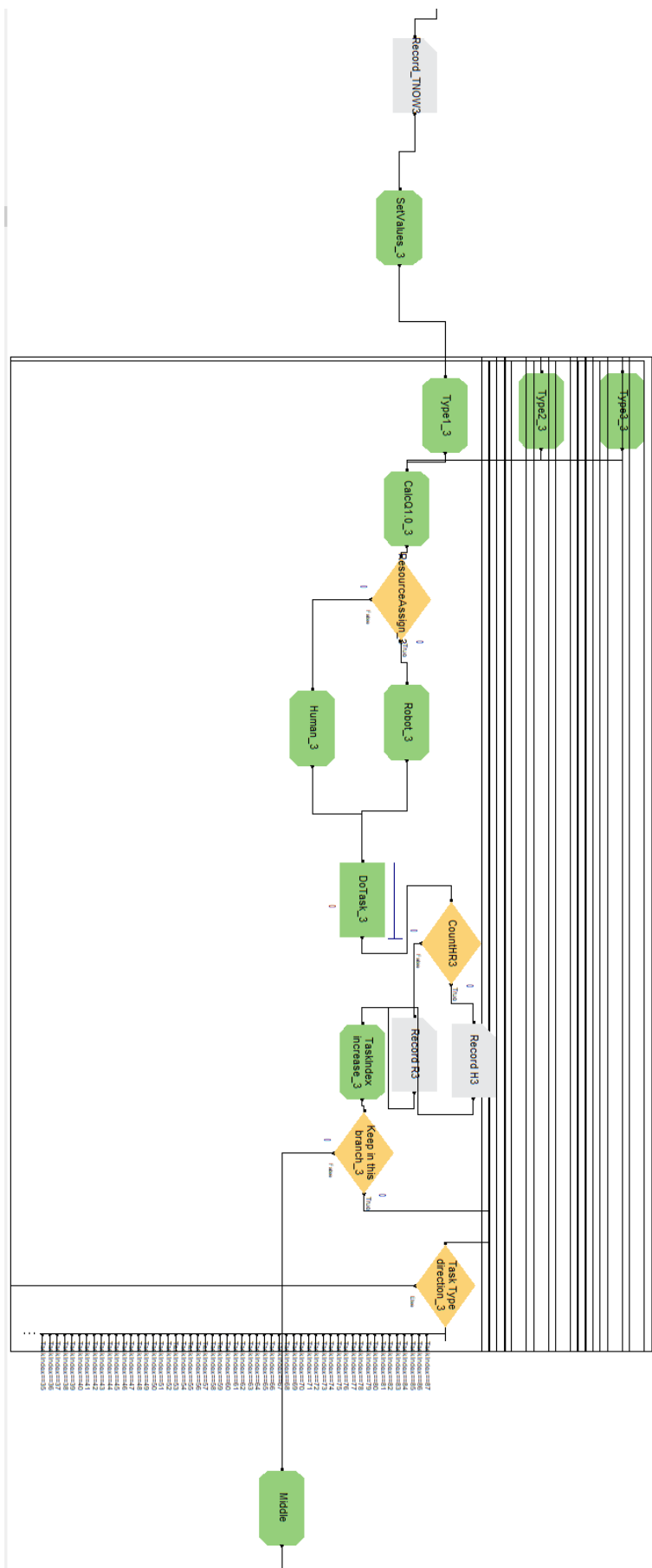


Figura 48 - Modello di simulazione implementato in Arena (parte 2) (fonte: Elaborazione propria)

Appendice B – Risultati completi delle simulazioni

La presente Appendice riporta in forma tabellare i valori completi degli indicatori di tempo, di produttività e di costo calcolati a partire dai risultati delle simulazioni.

Al fine di migliorare la leggibilità e facilitare il confronto tra le diverse configurazioni di risorse, gli indicatori sono stati raggruppati per tipologia e presentati in tabelle separate, ciascuna riferita a una specifica famiglia di grandezze.

Tutti gli indicatori sono riportati in funzione del numero di risorse umane (H) e robotiche (R) impiegate nella simulazione.

B.1 – Tempo di processo

In questa sezione è riportato il tempo medio di processo in funzione della specifica configurazione di risorse umane e robotiche coinvolte. I valori presentati in Tabella 14 sono stati ottenuti direttamente dagli output del software *Arena* al termine delle simulazioni.

Ogni simulazione è stata condotta introducendo nel sistema 35 entità di skateboard da produrre, secondo la distribuzione del tempo di inter-arrivo descritta nel capitolo relativo ai parametri di simulazione. Il tempo di processo riportato rappresenta la media del tempo totale di completamento calcolata sulle 100 repliche eseguite per ciascuna configurazione.

Humans	Robots	Avg Total Time (h)
1	1	2,71994
2	1	1,63906
3	1	1,60012
4	1	1,60650
5	1	1,61269
6	1	1,60345
7	1	1,60751
8	1	1,60723
9	1	1,60741
10	1	1,60771
1	2	2,70569
1	3	2,63727
1	4	2,63800
1	5	2,66118
1	6	2,66208
1	7	2,66097
1	8	2,66097
1	9	2,66097
1	10	2,74358
2	2	0,84324

Humans	Robots	Avg Total Time (h)
3	2	0,41431
4	2	0,40373
5	2	0,37622
6	2	0,38546
7	2	0,36846
8	2	0,38354
9	2	0,38635
10	2	0,34538
2	3	0,80668
2	4	0,85345
2	5	0,81387
2	6	0,81707
2	7	0,80779
2	8	0,80702
2	9	0,80847
2	10	0,86130
3	3	0,34805
4	3	0,21671
5	3	0,19056
6	3	0,19503
7	3	0,19345
8	3	0,18600
9	3	0,18360
10	3	0,18102
3	4	0,31982
3	5	0,33138
3	6	0,31702
3	7	0,32488
3	8	0,32070
3	9	0,32766
3	10	0,33413
4	4	0,19320
5	4	0,16727
6	4	0,15697
7	4	0,15611
8	4	0,15751
9	4	0,15332
10	4	0,15248
4	5	0,19083
4	6	0,19138
4	7	0,19669
4	8	0,18806
4	9	0,19513
4	10	0,19632
5	5	0,16014

Humans	Robots	Avg Total Time (h)
6	5	0,15100
7	5	0,14754
8	5	0,14715
9	5	0,14634
10	5	0,14604
15	5	0,14625
20	5	0,14623
25	5	0,14623
30	5	0,14623
35	5	0,14623
40	5	0,14623
45	5	0,14623
50	5	0,14623
5	6	0,16159
5	7	0,15866
5	8	0,15747
5	9	0,15771
5	10	0,15804
5	15	0,17353
5	20	0,15716
5	25	0,15716
5	30	0,15716
5	35	0,16333
5	40	0,15716
5	45	0,18072
5	50	0,15716
6	6	0,14914
7	6	0,14586
8	6	0,14392
9	6	0,14415
10	6	0,14383
6	7	0,15090
6	8	0,14876
6	9	0,14952
6	10	0,14880
7	7	0,14525
8	7	0,14361
9	7	0,14314
10	7	0,14303
7	8	0,14512
7	9	0,14522
7	10	0,14494
8	8	0,14348
9	8	0,14288
10	8	0,14266

Humans	Robots	Avg Total Time (h)
8	9	0,14282
8	10	0,14350
9	9	0,14287
10	9	0,14251
9	10	0,14269
10	10	0,14225
15	10	0,14233
20	10	0,14228
25	10	0,14233
30	10	0,14228
35	10	0,14233
40	10	0,14233
45	10	0,14233
50	10	0,14225
10	15	0,14255
10	20	0,14244
10	25	0,14255
10	30	0,14255
10	35	0,14255
10	40	0,14255
10	45	0,14255
10	50	0,14248
15	15	0,14216
20	15	0,14230
25	15	0,14230
30	15	0,14216
35	15	0,14230
40	15	0,14230
45	15	0,14230
50	15	0,14230
15	20	0,14230
15	25	0,14230
15	30	0,14228
15	35	0,14230
15	40	0,14230
15	45	0,14232
15	50	0,14230
20	20	0,14229
25	20	0,14227
30	20	0,14229
35	20	0,14229
40	20	0,14220
45	20	0,14229
50	20	0,14229
20	25	0,14234

Humans	Robots	Avg Total Time (h)
20	30	0,14229
20	35	0,14228
20	40	0,14229
20	45	0,14229
20	50	0,14229
25	25	0,14229
30	25	0,14228
35	25	0,14229
40	25	0,14229
45	25	0,14230
50	25	0,14229
25	30	0,14229
25	35	0,14229
25	40	0,14227
25	45	0,14229
25	50	0,14229
30	30	0,14222
35	30	0,14229
40	30	0,14232
45	30	0,14229
50	30	0,14229
30	35	0,14228
30	40	0,14229
30	45	0,14228
30	50	0,14229
35	35	0,14229
40	35	0,14227
45	35	0,14229
50	35	0,14229
35	40	0,14229
35	45	0,14229
35	50	0,14229
40	40	0,14234
45	40	0,14229
50	40	0,14229
40	45	0,14229
40	50	0,14229
45	45	0,14227
50	45	0,14229
45	50	0,14229
50	50	0,14226

Tabella 14 - Tempo di processo per diverse configurazioni

B.2 – Indicatori di produttività basati sul numero di skateboard prodotti per unità di tempo

In questa sezione sono riportati i valori degli indicatori di produttività definiti come numero medio di skateboard prodotti per unità di tempo, normalizzati rispetto al numero di risorse impiegate (Tabella 15). Gli indicatori distinguono tra produttività delle risorse umane, delle risorse robotiche e produttività media complessiva del sistema non distinta tra le due tipologie di risorse.

Humans	Robots	π_H (entità/h)	π_R (entità/h)	π_{tot} (entità/h)
1	1	12,8680	12,8680	6,4340
2	1	10,6768	21,3537	7,1179
3	1	7,2911	21,8733	5,4683
4	1	5,4466	21,7865	4,3573
5	1	4,3406	21,7029	3,6171
6	1	3,6380	21,8280	3,1183
7	1	3,1104	21,7728	2,7216
8	1	2,7221	21,7766	2,4196
9	1	2,4193	21,7741	2,1774
10	1	2,1770	21,7701	1,9791
1	2	12,9357	6,4679	4,3119
1	3	13,2713	4,4238	3,3178
1	4	13,2676	3,3169	2,6535
1	5	13,1521	2,6304	2,1920
1	6	13,1476	2,1913	1,8782
1	7	13,1531	1,8790	1,6441
1	8	13,1531	1,6441	1,4615
1	9	13,1531	1,4615	1,3153
1	10	12,7570	1,2757	1,1597
2	2	20,7532	20,7532	10,3766
3	2	28,1593	42,2390	16,8956
4	2	21,6727	43,3454	14,4485
5	2	18,6059	46,5149	13,2900
6	2	15,1334	45,4003	11,3501
7	2	13,5700	47,4949	10,5544
8	2	11,4070	45,6281	9,1256
9	2	10,0656	45,2952	8,2355
10	2	10,1339	50,6694	8,4449
2	3	21,6939	14,4626	8,6775
2	4	20,5050	10,2525	6,8350
2	5	21,5021	8,6009	6,1435
2	6	21,4180	7,1393	5,3545
2	7	21,6641	6,1898	4,8143

Humans	Robots	$\pi_H(\text{entità/h})$	$\pi_R(\text{entità/h})$	$\pi_{\text{tot}}(\text{entità/h})$
2	8	21,6846	5,4212	4,3369
2	9	21,6457	4,8102	3,9356
2	10	20,3181	4,0636	3,3863
3	3	33,5202	33,5202	16,7601
4	3	40,3767	53,8356	23,0724
5	3	36,7339	61,2232	22,9587
6	3	29,9093	59,8185	19,9395
7	3	25,8467	60,3090	18,0927
8	3	23,5210	62,7225	17,1061
9	3	21,1808	63,5423	15,8856
10	3	19,3346	64,4486	14,8727
3	4	36,4794	27,3596	15,6340
3	5	35,2059	21,1236	13,2022
3	6	36,8009	18,4004	12,2670
3	7	35,9110	15,3904	10,7733
3	8	36,3787	13,6420	9,9215
3	9	35,6063	11,8688	8,9016
3	10	34,9167	10,4750	8,0577
4	4	45,2901	45,2901	22,6451
5	4	41,8473	52,3091	23,2485
6	4	37,1617	55,7426	22,2970
7	4	32,0282	56,0493	20,3816
8	4	27,7765	55,5530	18,5177
9	4	25,3638	57,0686	17,5596
10	4	22,9533	57,3833	16,3952
4	5	45,8522	36,6818	20,3788
4	6	45,7216	30,4811	18,2886
4	7	44,4861	25,4207	16,1768
4	8	46,5273	23,2637	15,5091
4	9	44,8429	19,9302	13,7978
4	10	44,5702	17,8281	12,7343
5	5	43,7124	43,7124	21,8562
6	5	38,6310	46,3572	21,0714
7	5	33,8889	47,4444	19,7685
8	5	29,7321	47,5713	18,2967
9	5	26,5752	47,8354	17,0841
10	5	23,9652	47,9305	15,9768
15	5	15,9547	47,8641	11,9660
20	5	11,9671	47,8685	9,5737
25	5	9,5737	47,8685	7,9781
30	5	7,9781	47,8685	6,8384
35	5	6,8384	47,8685	5,9836
40	5	5,9836	47,8685	5,3187
45	5	5,3187	47,8685	4,7868
50	5	4,7868	47,8685	4,3517

Humans	Robots	$\pi_H(\text{entità/h})$	$\pi_R(\text{entità/h})$	$\pi_{\text{tot}}(\text{entità/h})$
5	6	43,3185	36,0987	19,6902
5	7	44,1189	31,5135	18,3829
5	8	44,4527	27,7829	17,0972
5	9	44,3859	24,6588	15,8521
5	10	44,2912	22,1456	14,7637
5	15	40,3395	13,4465	10,0849
5	20	44,5395	11,1349	8,9079
5	25	44,5395	8,9079	7,4233
5	30	44,5395	7,4233	6,3628
5	35	42,8586	6,1227	5,3573
5	40	44,5395	5,5674	4,9488
5	45	38,7329	4,3037	3,8733
5	50	44,5395	4,4540	4,0490
6	6	39,1126	39,1126	19,5563
7	6	34,2796	39,9928	18,4582
8	6	30,3993	40,5323	17,3710
9	6	26,9780	40,4670	16,1868
10	6	24,3335	40,5558	15,2084
6	7	38,6572	33,1347	17,8418
6	8	39,2140	29,4105	16,8060
6	9	39,0141	26,0094	15,6056
6	10	39,2019	23,5211	14,7007
7	7	34,4238	34,4238	17,2119
8	7	30,4653	34,8175	16,2482
9	7	27,1693	34,9319	15,2827
10	7	24,4711	34,9587	14,3947
7	8	34,4535	30,1468	16,0783
7	9	34,4313	26,7799	15,0637
7	10	34,4969	24,1478	14,2046
8	8	30,4925	30,4925	15,2462
9	8	27,2173	30,6195	14,4092
10	8	24,5340	30,6676	13,6300
8	9	30,6336	27,2299	14,4158
8	10	30,4887	24,3910	13,5505
9	9	27,2207	27,2207	13,6104
10	9	24,5598	27,2886	12,9262
9	10	27,2538	24,5284	12,9097
10	10	24,6040	24,6040	12,3020
15	10	16,3936	24,5905	9,8362
20	10	12,3001	24,6001	8,2000
25	10	9,8363	24,5909	7,0260
30	10	8,2000	24,6001	6,1500
35	10	7,0260	24,5909	5,4646
40	10	6,1477	24,5909	4,9182
45	10	5,4646	24,5909	4,4711

Humans	Robots	$\pi_H(\text{entità/h})$	$\pi_R(\text{entità/h})$	$\pi_{\text{tot}}(\text{entità/h})$
50	10	4,9209	24,6045	4,1008
10	15	24,5524	16,3682	9,8209
10	20	24,5720	12,2860	8,1907
10	25	24,5524	9,8209	7,0150
10	30	24,5524	8,1841	6,1381
10	35	24,5524	7,0150	5,4561
10	40	24,5524	6,1381	4,9105
10	45	24,5524	5,4561	4,4641
10	50	24,5653	4,9131	4,0942
15	15	16,4136	16,4136	8,2068
20	15	12,2982	16,3976	7,0275
25	15	9,8386	16,3977	6,1491
30	15	8,2068	16,4136	5,4712
35	15	7,0276	16,3977	4,9193
40	15	6,1491	16,3977	4,4721
45	15	5,4659	16,3977	4,0994
50	15	4,9193	16,3977	3,7841
15	20	16,3975	12,2982	7,0275
15	25	16,3975	9,8385	6,1491
15	30	16,3994	8,1997	5,4665
15	35	16,3975	7,0275	4,9193
15	40	16,3975	6,1491	4,4721
15	45	16,3951	5,4650	4,0988
15	50	16,3975	4,9193	3,7840
20	20	12,2985	12,2985	6,1492
25	20	9,8407	12,3009	5,4671
30	20	8,1990	12,2986	4,9194
35	20	7,0278	12,2986	4,4722
40	20	6,1534	12,3068	4,1023
45	20	5,4660	12,2986	3,7842
50	20	4,9194	12,2986	3,5139
20	25	12,2945	9,8356	5,4642
20	30	12,2985	8,1990	4,9194
20	35	12,2995	7,0283	4,4726
20	40	12,2985	6,1492	4,0995
20	45	12,2985	5,4660	3,7842
20	50	12,2985	4,9194	3,5139
25	25	9,8389	9,8389	4,9194
30	25	8,1997	9,8396	4,4726
35	25	7,0278	9,8389	4,0995
40	25	6,1493	9,8389	3,7842
45	25	5,4657	9,8383	3,5137
50	25	4,9194	9,8389	3,2796
25	30	9,8389	8,1990	4,4722
25	35	9,8389	7,0278	4,0995

Humans	Robots	$\pi_H(\text{entità/h})$	$\pi_R(\text{entità/h})$	$\pi_{tot}(\text{entità/h})$
25	40	9,8407	6,1504	3,7849
25	45	9,8389	5,4660	3,5139
25	50	9,8389	4,9194	3,2796
30	30	8,2032	8,2032	4,1016
35	30	7,0278	8,1990	3,7842
40	30	6,1481	8,1975	3,5132
45	30	5,4660	8,1990	3,2796
50	30	4,9194	8,1990	3,0746
30	35	8,1998	7,0284	3,7845
30	40	8,1990	6,1493	3,5139
30	45	8,1997	5,4665	3,2799
30	50	8,1990	4,9194	3,0746
35	35	7,0278	7,0278	3,5139
40	35	6,1502	7,0288	3,2801
45	35	5,4660	7,0278	3,0746
50	35	4,9194	7,0278	2,8938
35	40	7,0278	6,1493	3,2796
35	45	7,0278	5,4660	3,0746
35	50	7,0278	4,9194	2,8938
40	40	6,1472	6,1472	3,0736
45	40	5,4660	6,1493	2,8938
50	40	4,9194	6,1493	2,7330
40	45	6,1493	5,4660	2,8938
40	50	6,1493	4,9194	2,7330
45	45	5,4671	5,4671	2,7335
50	45	4,9194	5,4660	2,5892
45	50	5,4660	4,9194	2,5892
50	50	4,9207	4,9207	2,4604

Tabella 15 - Indicatori di produttività basati sul numero medio di skateboard prodotti nel tempo

B.3 – Indicatori di produttività basati sul tempo medio per skateboard

La seguente Tabella 16 riporta i valori degli indicatori di produttività espressi in termini di tempo medio imputabile a una singola risorsa per la realizzazione di uno skateboard. Tali grandezze consentono di valutare il contributo temporale medio delle risorse, distinguendo tra operatori umani, robotici e risorse complessive del sistema. Anche in questo caso, è riportato un indicatore di produttività P_{tot} che rappresenta la produttività media del sistema non distinguendo per la tipologia di risorse impiegate.

Humans	Robots	P_H (h/entità)	P_R (h/entità)	P_{tot} (h/entità)
1	1	0,07771	0,07771	0,03886
2	1	0,02342	0,04683	0,01561
3	1	0,01524	0,04572	0,01143
4	1	0,01147	0,04590	0,00918
5	1	0,00922	0,04608	0,00768
6	1	0,00764	0,04581	0,00654
7	1	0,00656	0,04593	0,00574
8	1	0,00574	0,04592	0,00510
9	1	0,00510	0,04593	0,00459
10	1	0,00459	0,04593	0,00418
1	2	0,07731	0,03865	0,02577
1	3	0,07535	0,02512	0,01884
1	4	0,07537	0,01884	0,01507
1	5	0,07603	0,01521	0,01267
1	6	0,07606	0,01268	0,01087
1	7	0,07603	0,01086	0,00950
1	8	0,07603	0,00950	0,00845
1	9	0,07603	0,00845	0,00760
1	10	0,07839	0,00784	0,00713
2	2	0,01205	0,01205	0,00602
3	2	0,00395	0,00592	0,00237
4	2	0,00288	0,00577	0,00192
5	2	0,00215	0,00537	0,00154
6	2	0,00184	0,00551	0,00138
7	2	0,00150	0,00526	0,00117
8	2	0,00137	0,00548	0,00110
9	2	0,00123	0,00552	0,00100
10	2	0,00099	0,00493	0,00082
2	3	0,01152	0,00768	0,00461
2	4	0,01219	0,00610	0,00406
2	5	0,01163	0,00465	0,00332
2	6	0,01167	0,00389	0,00292
2	7	0,01154	0,00330	0,00256
2	8	0,01153	0,00288	0,00231
2	9	0,01155	0,00257	0,00210
2	10	0,01230	0,00246	0,00205
3	3	0,00331	0,00331	0,00166
4	3	0,00155	0,00206	0,00088
5	3	0,00109	0,00181	0,00068
6	3	0,00093	0,00186	0,00062
7	3	0,00079	0,00184	0,00055
8	3	0,00066	0,00177	0,00048
9	3	0,00058	0,00175	0,00044
10	3	0,00052	0,00172	0,00040

Humans	Robots	P_H (h/entità)	P_R (h/entità)	P_{tot} (h/entità)
3	4	0,00305	0,00228	0,00131
3	5	0,00316	0,00189	0,00118
3	6	0,00302	0,00151	0,00101
3	7	0,00309	0,00133	0,00093
3	8	0,00305	0,00115	0,00083
3	9	0,00312	0,00104	0,00078
3	10	0,00318	0,00095	0,00073
4	4	0,00138	0,00138	0,00069
5	4	0,00096	0,00119	0,00053
6	4	0,00075	0,00112	0,00045
7	4	0,00064	0,00112	0,00041
8	4	0,00056	0,00113	0,00038
9	4	0,00049	0,00110	0,00034
10	4	0,00044	0,00109	0,00031
4	5	0,00136	0,00109	0,00061
4	6	0,00137	0,00091	0,00055
4	7	0,00140	0,00080	0,00051
4	8	0,00134	0,00067	0,00045
4	9	0,00139	0,00062	0,00043
4	10	0,00140	0,00056	0,00040
5	5	0,00092	0,00092	0,00046
6	5	0,00072	0,00086	0,00039
7	5	0,00060	0,00084	0,00035
8	5	0,00053	0,00084	0,00032
9	5	0,00046	0,00084	0,00030
10	5	0,00042	0,00083	0,00028
15	5	0,00028	0,00084	0,00021
20	5	0,00021	0,00084	0,00017
25	5	0,00017	0,00084	0,00014
30	5	0,00014	0,00084	0,00012
35	5	0,00012	0,00084	0,00010
40	5	0,00010	0,00084	0,00009
45	5	0,00009	0,00084	0,00008
50	5	0,00008	0,00084	0,00008
5	6	0,00092	0,00077	0,00042
5	7	0,00091	0,00065	0,00038
5	8	0,00090	0,00056	0,00035
5	9	0,00090	0,00050	0,00032
5	10	0,00090	0,00045	0,00030
5	15	0,00099	0,00033	0,00025
5	20	0,00090	0,00022	0,00018
5	25	0,00090	0,00018	0,00015
5	30	0,00090	0,00015	0,00013
5	35	0,00093	0,00013	0,00012
5	40	0,00090	0,00011	0,00010

Humans	Robots	P_H (h/entità)	P_R (h/entità)	P_{tot} (h/entità)
5	45	0,00103	0,00011	0,00010
5	50	0,00090	0,00009	0,00008
6	6	0,00071	0,00071	0,00036
7	6	0,00060	0,00069	0,00032
8	6	0,00051	0,00069	0,00029
9	6	0,00046	0,00069	0,00027
10	6	0,00041	0,00068	0,00026
6	7	0,00072	0,00062	0,00033
6	8	0,00071	0,00053	0,00030
6	9	0,00071	0,00047	0,00028
6	10	0,00071	0,00043	0,00027
7	7	0,00059	0,00059	0,00030
8	7	0,00051	0,00059	0,00027
9	7	0,00045	0,00058	0,00026
10	7	0,00041	0,00058	0,00024
7	8	0,00059	0,00052	0,00028
7	9	0,00059	0,00046	0,00026
7	10	0,00059	0,00041	0,00024
8	8	0,00051	0,00051	0,00026
9	8	0,00045	0,00051	0,00024
10	8	0,00041	0,00051	0,00023
8	9	0,00051	0,00045	0,00024
8	10	0,00051	0,00041	0,00023
9	9	0,00045	0,00045	0,00023
10	9	0,00041	0,00045	0,00021
9	10	0,00045	0,00041	0,00021
10	10	0,00041	0,00041	0,00020
15	10	0,00027	0,00041	0,00016
20	10	0,00020	0,00041	0,00014
25	10	0,00016	0,00041	0,00012
30	10	0,00014	0,00041	0,00010
35	10	0,00012	0,00041	0,00009
40	10	0,00010	0,00041	0,00008
45	10	0,00009	0,00041	0,00007
50	10	0,00008	0,00041	0,00007
10	15	0,00041	0,00027	0,00016
10	20	0,00041	0,00020	0,00014
10	25	0,00041	0,00016	0,00012
10	30	0,00041	0,00014	0,00010
10	35	0,00041	0,00012	0,00009
10	40	0,00041	0,00010	0,00008
10	45	0,00041	0,00009	0,00007
10	50	0,00041	0,00008	0,00007
15	15	0,00027	0,00027	0,00014
20	15	0,00020	0,00027	0,00012

Humans	Robots	P_H (h/entità)	P_R (h/entità)	P_{tot} (h/entità)
25	15	0,00016	0,00027	0,00010
30	15	0,00014	0,00027	0,00009
35	15	0,00012	0,00027	0,00008
40	15	0,00010	0,00027	0,00007
45	15	0,00009	0,00027	0,00007
50	15	0,00008	0,00027	0,00006
15	20	0,00027	0,00020	0,00012
15	25	0,00027	0,00016	0,00010
15	30	0,00027	0,00014	0,00009
15	35	0,00027	0,00012	0,00008
15	40	0,00027	0,00010	0,00007
15	45	0,00027	0,00009	0,00007
15	50	0,00027	0,00008	0,00006
20	20	0,00020	0,00020	0,00010
25	20	0,00016	0,00020	0,00009
30	20	0,00014	0,00020	0,00008
35	20	0,00012	0,00020	0,00007
40	20	0,00010	0,00020	0,00007
45	20	0,00009	0,00020	0,00006
50	20	0,00008	0,00020	0,00006
20	25	0,00020	0,00016	0,00009
20	30	0,00020	0,00014	0,00008
20	35	0,00020	0,00012	0,00007
20	40	0,00020	0,00010	0,00007
20	45	0,00020	0,00009	0,00006
20	50	0,00020	0,00008	0,00006
25	25	0,00016	0,00016	0,00008
30	25	0,00014	0,00016	0,00007
35	25	0,00012	0,00016	0,00007
40	25	0,00010	0,00016	0,00006
45	25	0,00009	0,00016	0,00006
50	25	0,00008	0,00016	0,00005
25	30	0,00016	0,00014	0,00007
25	35	0,00016	0,00012	0,00007
25	40	0,00016	0,00010	0,00006
25	45	0,00016	0,00009	0,00006
25	50	0,00016	0,00008	0,00005
30	30	0,00014	0,00014	0,00007
35	30	0,00012	0,00014	0,00006
40	30	0,00010	0,00014	0,00006
45	30	0,00009	0,00014	0,00005
50	30	0,00008	0,00014	0,00005
30	35	0,00014	0,00012	0,00006
30	40	0,00014	0,00010	0,00006
30	45	0,00014	0,00009	0,00005

Humans	Robots	P_H (h/entità)	P_R (h/entità)	P_{tot} (h/entità)
30	50	0,00014	0,00008	0,00005
35	35	0,00012	0,00012	0,00006
40	35	0,00010	0,00012	0,00005
45	35	0,00009	0,00012	0,00005
50	35	0,00008	0,00012	0,00005
35	40	0,00012	0,00010	0,00005
35	45	0,00012	0,00009	0,00005
35	50	0,00012	0,00008	0,00005
40	40	0,00010	0,00010	0,00005
45	40	0,00009	0,00010	0,00005
50	40	0,00008	0,00010	0,00005
40	45	0,00010	0,00009	0,00005
40	50	0,00010	0,00008	0,00005
45	45	0,00009	0,00009	0,00005
50	45	0,00008	0,00009	0,00004
45	50	0,00009	0,00008	0,00004
50	50	0,00008	0,00008	0,00004

Tabella 16 - Indicatori di produttività basati sul tempo medio necessario alla produzione di uno skateboard

B.4 – Costi unitari per skateboard attribuibili a una singola risorsa

In questa sezione sono riportati i valori del costo medio unitario per skateboard attribuibile a una singola risorsa umana o robotica, escludendo i periodi di inattività delle risorse. Si riporta anche il relativo costo di produzione totale di uno skateboard C_{work} , anche in questo caso non considerando i costi relativi alle risorse inattive nel sistema (Tabella 17).

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
1	1	1,5542	0,5751	2,1293
2	1	0,4683	0,3465	1,2832
3	1	0,3048	0,3383	1,2527
4	1	0,2295	0,3397	1,2577
5	1	0,1843	0,3410	1,2625
6	1	0,1527	0,3390	1,2553
7	1	0,1312	0,3399	1,2584
8	1	0,1148	0,3398	1,2582
9	1	0,1021	0,3399	1,2584
10	1	0,0919	0,3399	1,2586
1	2	1,5461	0,2860	2,1182
1	3	1,5070	0,1859	2,0646
1	4	1,5074	0,1394	2,0652
1	5	1,5207	0,1125	2,0833
1	6	1,5212	0,0938	2,0840

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
1	7	1,5206	0,0804	2,0832
1	8	1,5206	0,0703	2,0832
1	9	1,5206	0,0625	2,0832
1	10	1,5678	0,0580	2,1478
2	2	0,2409	0,0891	0,6601
3	2	0,0789	0,0438	0,3243
4	2	0,0577	0,0427	0,3161
5	2	0,0430	0,0398	0,2945
6	2	0,0367	0,0407	0,3018
7	2	0,0301	0,0390	0,2885
8	2	0,0274	0,0405	0,3003
9	2	0,0245	0,0408	0,3025
10	2	0,0197	0,0365	0,2704
2	3	0,2305	0,0569	0,6315
2	4	0,2438	0,0451	0,6681
2	5	0,2325	0,0344	0,6371
2	6	0,2334	0,0288	0,6396
2	7	0,2308	0,0244	0,6324
2	8	0,2306	0,0213	0,6318
2	9	0,2310	0,0190	0,6329
2	10	0,2461	0,0182	0,6743
3	3	0,0663	0,0245	0,2725
4	3	0,0310	0,0153	0,1697
5	3	0,0218	0,0134	0,1492
6	3	0,0186	0,0137	0,1527
7	3	0,0158	0,0136	0,1514
8	3	0,0133	0,0131	0,1456
9	3	0,0117	0,0129	0,1437
10	3	0,0103	0,0128	0,1417
3	4	0,0609	0,0169	0,2504
3	5	0,0631	0,0140	0,2594
3	6	0,0604	0,0112	0,2482
3	7	0,0619	0,0098	0,2543
3	8	0,0611	0,0085	0,2511
3	9	0,0624	0,0077	0,2565
3	10	0,0636	0,0071	0,2616
4	4	0,0276	0,0102	0,1512
5	4	0,0191	0,0088	0,1310
6	4	0,0149	0,0083	0,1229
7	4	0,0127	0,0083	0,1222
8	4	0,0113	0,0083	0,1233
9	4	0,0097	0,0081	0,1200
10	4	0,0087	0,0081	0,1194
4	5	0,0273	0,0081	0,1494
4	6	0,0273	0,0067	0,1498

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
4	7	0,0281	0,0059	0,1540
4	8	0,0269	0,0050	0,1472
4	9	0,0279	0,0046	0,1528
4	10	0,0280	0,0042	0,1537
5	5	0,0183	0,0068	0,1254
6	5	0,0144	0,0064	0,1182
7	5	0,0120	0,0062	0,1155
8	5	0,0105	0,0062	0,1152
9	5	0,0093	0,0062	0,1146
10	5	0,0083	0,0062	0,1143
15	5	0,0056	0,0062	0,1145
20	5	0,0042	0,0062	0,1145
25	5	0,0033	0,0062	0,1145
30	5	0,0028	0,0062	0,1145
35	5	0,0024	0,0062	0,1145
40	5	0,0021	0,0062	0,1145
45	5	0,0019	0,0062	0,1145
50	5	0,0017	0,0062	0,1145
5	6	0,0185	0,0057	0,1265
5	7	0,0181	0,0048	0,1242
5	8	0,0180	0,0042	0,1233
5	9	0,0180	0,0037	0,1235
5	10	0,0181	0,0033	0,1237
5	15	0,0198	0,0024	0,1358
5	20	0,0180	0,0017	0,1230
5	25	0,0180	0,0013	0,1230
5	30	0,0180	0,0011	0,1230
5	35	0,0187	0,0010	0,1279
5	40	0,0180	0,0008	0,1230
5	45	0,0207	0,0008	0,1415
5	50	0,0180	0,0007	0,1230
6	6	0,0142	0,0053	0,1168
7	6	0,0119	0,0051	0,1142
8	6	0,0103	0,0051	0,1127
9	6	0,0092	0,0051	0,1128
10	6	0,0082	0,0051	0,1126
6	7	0,0144	0,0046	0,1181
6	8	0,0142	0,0039	0,1165
6	9	0,0142	0,0035	0,1171
6	10	0,0142	0,0031	0,1165
7	7	0,0119	0,0044	0,1137
8	7	0,0103	0,0043	0,1124
9	7	0,0091	0,0043	0,1121
10	7	0,0082	0,0043	0,1120
7	8	0,0118	0,0038	0,1136

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
7	9	0,0119	0,0034	0,1137
7	10	0,0118	0,0031	0,1135
8	8	0,0102	0,0038	0,1123
9	8	0,0091	0,0038	0,1119
10	8	0,0082	0,0038	0,1117
8	9	0,0102	0,0034	0,1118
8	10	0,0102	0,0030	0,1123
9	9	0,0091	0,0034	0,1118
10	9	0,0081	0,0033	0,1116
9	10	0,0091	0,0030	0,1117
10	10	0,0081	0,0030	0,1114
15	10	0,0054	0,0030	0,1114
20	10	0,0041	0,0030	0,1114
25	10	0,0033	0,0030	0,1114
30	10	0,0027	0,0030	0,1114
35	10	0,0023	0,0030	0,1114
40	10	0,0020	0,0030	0,1114
45	10	0,0018	0,0030	0,1114
50	10	0,0016	0,0030	0,1114
10	15	0,0081	0,0020	0,1116
10	20	0,0081	0,0015	0,1115
10	25	0,0081	0,0012	0,1116
10	30	0,0081	0,0010	0,1116
10	35	0,0081	0,0009	0,1116
10	40	0,0081	0,0008	0,1116
10	45	0,0081	0,0007	0,1116
10	50	0,0081	0,0006	0,1115
15	15	0,0054	0,0020	0,1113
20	15	0,0041	0,0020	0,1114
25	15	0,0033	0,0020	0,1114
30	15	0,0027	0,0020	0,1113
35	15	0,0023	0,0020	0,1114
40	15	0,0020	0,0020	0,1114
45	15	0,0018	0,0020	0,1114
50	15	0,0016	0,0020	0,1114
15	20	0,0054	0,0015	0,1114
15	25	0,0054	0,0012	0,1114
15	30	0,0054	0,0010	0,1114
15	35	0,0054	0,0009	0,1114
15	40	0,0054	0,0008	0,1114
15	45	0,0054	0,0007	0,1114
15	50	0,0054	0,0006	0,1114
20	20	0,0041	0,0015	0,1114
25	20	0,0033	0,0015	0,1114
30	20	0,0027	0,0015	0,1114

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
35	20	0,0023	0,0015	0,1114
40	20	0,0020	0,0015	0,1113
45	20	0,0018	0,0015	0,1114
50	20	0,0016	0,0015	0,1114
20	25	0,0041	0,0012	0,1114
20	30	0,0041	0,0010	0,1114
20	35	0,0041	0,0009	0,1114
20	40	0,0041	0,0008	0,1114
20	45	0,0041	0,0007	0,1114
20	50	0,0041	0,0006	0,1114
25	25	0,0033	0,0012	0,1114
30	25	0,0027	0,0012	0,1114
35	25	0,0023	0,0012	0,1114
40	25	0,0020	0,0012	0,1114
45	25	0,0018	0,0012	0,1114
50	25	0,0016	0,0012	0,1114
25	30	0,0033	0,0010	0,1114
25	35	0,0033	0,0009	0,1114
25	40	0,0033	0,0008	0,1114
25	45	0,0033	0,0007	0,1114
25	50	0,0033	0,0006	0,1114
30	30	0,0027	0,0010	0,1113
35	30	0,0023	0,0010	0,1114
40	30	0,0020	0,0010	0,1114
45	30	0,0018	0,0010	0,1114
50	30	0,0016	0,0010	0,1114
30	35	0,0027	0,0009	0,1114
30	40	0,0027	0,0008	0,1114
30	45	0,0027	0,0007	0,1114
30	50	0,0027	0,0006	0,1114
35	35	0,0023	0,0009	0,1114
40	35	0,0020	0,0009	0,1114
45	35	0,0018	0,0009	0,1114
50	35	0,0016	0,0009	0,1114
35	40	0,0023	0,0008	0,1114
35	45	0,0023	0,0007	0,1114
35	50	0,0023	0,0006	0,1114
40	40	0,0020	0,0008	0,1114
45	40	0,0018	0,0008	0,1114
50	40	0,0016	0,0008	0,1114
40	45	0,0020	0,0007	0,1114
40	50	0,0020	0,0006	0,1114
45	45	0,0018	0,0007	0,1114
50	45	0,0016	0,0007	0,1114
45	50	0,0018	0,0006	0,1114

Humans	Robots	γ_H (€/entità)	γ_R (€/entità)	C_{work} (€/entità)
50	50	0,0016	0,0006	0,1114

Tabella 17 - Indicatori di costo medio per skateboard escludendo i periodi di inattività delle risorse

B.5 – Tempo di processo

La Tabella 18 riporta i valori dell'indicatore di costo reale complessivo per skateboard, nonché il costo sostenuto dall'azienda per la disponibilità delle risorse durante l'intero processo produttivo.

Humans	Robots	C (€/entità)
1	1	2,1293
2	1	2,2198
3	1	3,0814
4	1	4,0117
5	1	4,9487
6	1	5,8366
7	1	6,7699
8	1	7,6871
9	1	8,6065
10	1	9,5268
1	2	2,6902
1	3	3,1798
1	4	3,7384
1	5	4,3339
1	6	4,8982
1	7	5,4588
1	8	6,0214
1	9	6,5840
1	10	7,3685
2	2	1,3203
3	2	0,8854
4	2	1,0935
5	2	1,2340
6	2	1,4846
7	2	1,6296
8	2	1,9155
9	2	2,1503
10	2	2,1196
2	3	1,4336
2	4	1,6971
2	5	1,7905
2	6	1,9703
2	7	2,1187

Humans	Robots	C (€/entità)
2	8	2,2873
2	9	2,4624
2	10	2,8054
3	3	0,8174
4	3	0,6328
5	3	0,6653
6	3	0,7924
7	3	0,8965
8	3	0,9683
9	3	1,0607
10	3	1,1492
3	4	0,8187
3	5	0,9184
3	6	0,9456
3	7	1,0378
3	8	1,0922
3	9	1,1852
3	10	1,2792
4	4	0,6050
5	4	0,6194
6	4	0,6709
7	4	0,7565
8	4	0,8532
9	4	0,9182
10	4	1,0003
4	5	0,6379
4	6	0,6802
4	7	0,7407
4	8	0,7479
4	9	0,8173
4	10	0,8638
5	5	0,6268
6	5	0,6773
7	5	0,7461
8	5	0,8282
9	5	0,9073
10	5	0,9889
15	5	1,4082
20	5	1,8258
25	5	2,2436
30	5	2,6615
35	5	3,0793
40	5	3,4971
45	5	3,9149
50	5	4,3327

Humans	Robots	C (€/entità)
5	6	0,6667
5	7	0,6881
5	8	0,7163
5	9	0,7507
5	10	0,7857
5	15	1,0461
5	20	1,1136
5	25	1,2798
5	30	1,4459
5	35	1,6753
5	40	1,7782
5	45	2,2358
5	50	2,1105
6	6	0,7005
7	6	0,7685
8	6	0,8405
9	6	0,9242
10	6	1,0044
6	7	0,7407
6	8	0,7616
6	9	0,7971
6	10	0,8248
7	7	0,7960
8	7	0,8690
9	7	0,9480
10	7	1,0290
7	8	0,8260
7	9	0,8572
7	10	0,8862
8	8	0,8986
9	8	0,9765
10	8	1,0565
8	9	0,9246
8	10	0,9594
9	9	1,0066
10	9	1,0855
9	10	1,0355
10	10	1,1136
15	10	1,5209
20	10	1,9268
25	10	2,3342
30	10	2,7398
35	10	3,1475
40	10	3,5542
45	10	3,9608

Humans	Robots	C (€/entità)
50	10	4,3651
10	15	1,2667
10	20	1,4162
10	25	1,5681
10	30	1,7188
10	35	1,8695
10	40	2,0202
10	45	2,1709
10	50	2,3204
15	15	1,6693
20	15	2,0775
25	15	2,4841
30	15	2,8878
35	15	3,2972
40	15	3,7038
45	15	4,1103
50	15	4,5169
15	20	1,8214
15	25	1,9718
15	30	2,1220
15	35	2,2727
15	40	2,4231
15	45	2,5739
15	50	2,7240
20	20	2,2279
25	20	2,6340
30	20	3,0410
35	20	3,4476
40	20	3,8515
45	20	4,2607
50	20	4,6672
20	25	2,3791
20	30	2,5288
20	35	2,6790
20	40	2,8296
20	45	2,9800
20	50	3,1305
25	25	2,7849
30	25	3,1912
35	25	3,5980
40	25	4,0045
45	25	4,4113
50	25	4,8176
25	30	2,9353
25	35	3,0857

Humans	Robots	C (€/entità)
25	40	3,2355
25	45	3,3866
25	50	3,5370
30	30	3,3402
35	30	3,7484
40	30	4,1557
45	30	4,5615
50	30	4,9681
30	35	3,4920
30	40	3,6427
30	45	3,7928
30	50	3,9435
35	35	3,8988
40	35	4,3047
45	35	4,7119
50	35	5,1185
35	40	4,0493
35	45	4,1997
35	50	4,3501
40	40	4,4573
45	40	4,8624
50	40	5,2689
40	45	4,6062
40	50	4,7567
45	45	5,0118
50	45	5,4193
45	50	5,1632
50	50	5,5683

Tabella 18 - Indicatore di costo reale complessivo per skateboard

Appendice C – Stima della funzione di complessità di interazione

La presente Appendice riporta i risultati numerici ottenuti per la stima della funzione di complessità di interazione tra risorse umane e robotiche.

Come discusso nel capitolo di analisi dei risultati, la procedura di stima è stata articolata in più fasi: il calcolo dei valori osservati della funzione di complessità di interazione $f_c(H,R)$ mediante l'inversione della funzione di tempo di processo, la stima a fini comparativi dei modelli alternativi mediante regressione ai minimi quadrati ordinari e selezione finale della funzione migliore, basandosi su risultati numerici ed interpretabilità economica.

In questa Appendice sono riportati i risultati numerici relativi all'analisi effettuata.

C.1 – Calcolo dei valori osservati della funzione di complessità

Per ciascuna configurazione simulata (H,R) , il valore della funzione di complessità di interazione è stato ottenuto invertendo la relazione generale del tempo di processo, assumendo $\beta=1$. I risultati ottenuti sono riportati in Tabella 19.

Humans	Robots	$f_c(H,R)$
1	1	0,004372
2	1	-0,07772
3	1	0,209871
4	1	0,525426
5	1	0,843236
6	1	1,142818
7	1	1,459172
8	1	1,76962
9	1	2,080849
10	1	2,392395
1	2	0,474879
1	3	0,901318
1	4	1,365707
1	5	1,854436
1	6	2,32351
1	7	2,790025
1	8	3,257929
1	9	3,725834
1	10	4,354984
2	2	-0,37724
3	2	-0,61388
4	2	-0,54565

Humans	Robots	$f_c(H,R)$
5	2	-0,50384
6	2	-0,4171
7	2	-0,37154
8	2	-0,27164
9	2	-0,19156
10	2	-0,2105
2	3	-0,2624
2	4	-0,06957
2	5	0,0304
2	6	0,17812
2	7	0,306777
2	8	0,44745
2	9	0,592211
2	10	0,847702
3	3	-0,61443
4	3	-0,71801
5	3	-0,71518
6	3	-0,67077
7	3	-0,63603
8	3	-0,61406
9	3	-0,58352
10	3	-0,55437
3	4	-0,58948
3	5	-0,51636
3	6	-0,48157
3	7	-0,4116
3	8	-0,36277
3	9	-0,29133
3	10	-0,21858
4	4	-0,71464
5	4	-0,72057
6	4	-0,70742
7	4	-0,67883
8	4	-0,64549
9	4	-0,62525
10	4	-0,59781
4	5	-0,68458
4	6	-0,65003
4	7	-0,60572
4	8	-0,58995
4	9	-0,54024
4	10	-0,5029
5	5	-0,70434
6	5	-0,692
7	5	-0,67052

Humans	Robots	$f_c(H,R)$
8	5	-0,64293
9	5	-0,6166
10	5	-0,58911
15	5	-0,4471
20	5	-0,30573
25	5	-0,1643
30	5	-0,02288
35	5	0,118548
40	5	0,259973
45	5	0,401399
50	5	0,542824
5	6	-0,67323
5	7	-0,65126
5	8	-0,62619
5	9	-0,5979
5	10	-0,56925
5	15	-0,37448
5	20	-0,29529
5	25	-0,15711
5	30	-0,01894
5	35	0,16314
5	40	0,257421
5	45	0,604818
5	50	0,533777
6	6	-0,66956
7	6	-0,64862
8	6	-0,62546
9	6	-0,59698
10	6	-0,57004
6	7	-0,63914
6	8	-0,6181
6	9	-0,58986
6	10	-0,56566
7	7	-0,62456
8	7	-0,60102
9	7	-0,57465
10	7	-0,54731
7	8	-0,59936
7	9	-0,57357
7	10	-0,54889
8	8	-0,57615
9	8	-0,55027
10	8	-0,52338
8	9	-0,55299
8	10	-0,52563

Humans	Robots	$f_c(H,R)$
9	9	-0,52521
10	9	-0,49882
9	10	-0,50069
10	10	-0,47471
15	10	-0,33677
20	10	-0,19943
25	10	-0,06148
30	10	0,075762
35	10	0,213816
40	10	0,351465
45	10	0,489115
50	10	0,625861
10	15	-0,34827
10	20	-0,22356
10	25	-0,09761
10	30	0,027721
10	35	0,153053
10	40	0,278385
10	45	0,403717
10	50	0,528246
15	15	-0,21259
20	15	-0,0742
25	15	0,063409
30	15	0,19986
35	15	0,338645
40	15	0,476262
45	15	0,61388
50	15	0,751498
15	20	-0,08671
15	25	0,038395
15	30	0,163374
15	35	0,288611
15	40	0,413719
15	45	0,539061
15	50	0,663935
20	20	0,050878
25	20	0,188259
30	20	0,326099
35	20	0,463713
40	20	0,600256
45	20	0,738942
50	20	0,876556
20	25	0,176365
20	30	0,301087
20	35	0,426072

Humans	Robots	$f_c(H,R)$
20	40	0,551296
20	45	0,6764
20	50	0,801505
25	25	0,313589
30	25	0,451091
35	25	0,588817
40	25	0,726431
45	25	0,864143
50	25	1,001659
25	30	0,438693
25	35	0,563796
25	40	0,688583
25	45	0,814004
25	50	0,939108
30	30	0,575505
35	30	0,713921
40	30	0,851878
45	30	0,989149
50	30	1,126763
30	35	0,701256
30	40	0,826514
30	45	0,951467
30	50	1,076722
35	35	0,839025
40	35	0,976336
45	35	1,114253
50	35	1,251867
35	40	0,964128
35	45	1,089232
35	50	1,214336
40	40	1,102439
45	40	1,239357
50	40	1,376971
40	45	1,226846
40	50	1,35195
45	45	1,364017
50	45	1,502074
45	50	1,489564
50	50	1,626491

Tabella 19 - $f_c(H,R)$

C.2 – Regressione lineare senza termine di interazione

I valori osservati sono stati inizialmente utilizzati per stimare un modello lineare privo del termine di interazione, della forma:

$$f_c(H, R) = c + X_0H + X_1R$$

La Tabella 20 riporta i risultati ottenuti dalla regressione ai minimi quadrati ordinari.

Intercetta	0,704737
X₀	-0,01102
X₁	-0,00762
R²	0,1595
R²_{adj}	0,1507

Tabella 20 - Regressione lineare senza termine di interazione

C.3 – Regressione lineare con termine di interazione

Infine, è stata stimata la funzione lineare comprensiva del termine di interazione tra risorse umane e robotiche, della forma:

$$f_c(H, R) = c + X_0H + X_1R + X_2HR$$

I risultati sono riportati in Tabella 21.

Intercetta	0,894081
X₀	-0,02561
X₁	-0,0222
X₂	0,000743
R²	0,218083
R²_{adj}	0,205865

Tabella 21 - Regressione lineare con termine di interazione

Bibliografia

- A. Dahiya, A. M. (2023, March). *A survey of multi-agent Human–Robot Interaction systems*. Tratto da <https://www.sciencedirect.com/science/article/abs/pii/S092188902200224X>
- A. Hentout, M. A. (2019, June). *Human–robot interaction in industrial collaborative robotics: a literature review of the decade 2008–2017*. Tratto da <https://www.scopus.com/pages/publications/85068623215>
- Amedeo Cappelli, E. G. (2003, November). *L’interazione Uomo-Robot*. Tratto da https://www.researchgate.net/profile/Emiliano-Giovannetti/publication/265031857_L'INTERAZIONE_UOMO-ROBOT_HUMAN-ROBOT_INTERACTION/links/0912f50c85bd5de3a3000000/LINTERAZIONE-UOMO-ROBOT-HUMAN-ROBOT-INTERACTION.pdf
- Barravecchia, F. (2025). *Human-Robot Interaction Dynamics: Modeling Multi-Human Multi-Robot Collaboration*. Turin.
- Cocchi, A. (2020). *L'utilizzo dei Cobot nelle industrie Automotive*. Tratto da Universal-robots: <https://www.universal-robots.com/it/blog/lutilizzo-dei-cobot-nelle-industrie-automotive/>
- D. Gross, C. H. (1998). *Fundamentals of Queuing Theory* (3 ed.). Wiley.
- F. Barravecchia, M. B. (2023, August). *Redefining Human–Robot Symbiosis: a bio-inspired approach*. Tratto da <https://iris.polito.it/retrieve/048bb90f-01d8-4145-84b5-5e4e10775aab/IJAMT%20v.128%2c%20n.5-6%2c%202023%2c%20pp.2043-2058%20Redefining%20Human%20Robot%20Symbiosis%20%28FB%20MB%20LM%20FF%29.pdf>
- FUTURMOTIVE. (2022). *Più vicini, più sicuri: come l’industria automotive può ritararsi passando da delocalizzazione a un modello industriale a filiera*. Tratto da [futurmotive.com: https://www.futurmotive.com/it/Piu-vicini-piu-sicuri-come-l-industria-automotive-puo-ritararsi-passando-da/n6](https://www.futurmotive.com/it/Piu-vicini-piu-sicuri-come-l-industria-automotive-puo-ritararsi-passando-da/n6)
- G. L. Curry, R. M. (2011). *Manufacturing Systems Modeling and Analysis* (2 ed.). Springer.
- Global, K. (2026). *Lavorazione meccanica: Fresatura*. Tratto da KUKA Global: <https://www.kuka.com/it-it/applicazioni/lavorazione-meccanica/fresatura-robot>
- I. Maurtua, A. I. (2017, July). *Human–robot collaboration in industrial applications: Safety, interaction and trust*. Tratto da <https://journals.sagepub.com/doi/full/10.1177/1729881417716010>
- J. Banks, J. S. (2010). *Discrete-Event System Simulation*. Pearson.

- Kelton, W. D. (2015). *Simulation with Arena* (6 ed.). McGrawHill.
- Krüger, J. (2009, October). *Cooperation of human and machines in assembly lines*. Tratto da <https://www.sciencedirect.com/science/article/abs/pii/S0007850609001760>
- L. Wang, R. G. (2019). *Symbiotic human-robot collaborative assembly*. Tratto da <https://www.sciencedirect.com/science/article/abs/pii/S0007850619301593>
- Lasi H., F. P. (2014). *Industry 4.0*. Tratto da <https://link.springer.com/article/10.1007/s12599-014-0334-4>
- Law, M. (2024). *Simulation Modeling and Analysis* (6 ed.). McGrawHill.
- R. Jahanmahin, S. M. (2022, December). *Human-robot interactions in manufacturing: A survey of human behavior modeling*. Tratto da <https://www.sciencedirect.com/science/article/abs/pii/S0736584522000916>
- S. El Zaatari, M. M. (2019, June). *Cobot programming for collaborative industrial tasks: An overview*. Tratto da <https://www.sciencedirect.com/science/article/abs/pii/S092188901830602X?via%3Dihub>
- S. Robla-Gomez, V. M.-S.-O. (2017, December 22). *Working Together: A Review on Safe Human-Robot Collaboration in Industrial Enviroments*. Tratto da [xplorestaging.ieee: https://xplorestaging.ieee.org/ielx7/6287639/7859429/08107677.pdf?arnumber=8107677&utm_source=scopus&getft_integrator=scopus](https://xplorestaging.ieee.org/ielx7/6287639/7859429/08107677.pdf?arnumber=8107677&utm_source=scopus&getft_integrator=scopus)
- Schlechtriem, M. (2022). *Robots Can Help Optimize Material Handling Processes in the Supply Chain*. Tratto da Robotics247: https://www.robotics247.com/article/robots_can_help_optimize_material_handling_processes_supply_chain
- SICK. (2026). *Soluzioni per robot di KUKA*. Tratto da sick.com: <https://www.sick.com/ch/it/soluzioni-per-robot-di-kuka/s/kuka>
- Star-ai. (2021). *Human-Cobot Collaboration for Robust Quality Inspections*. Tratto da Star-ai: <https://star-ai.eu/human-robot-collaboration>
- Universal-robots. (2026). *Mobile cobots handle round-the-clock production*. Tratto da Universal-robots: <https://www.universal-robots.com/case-stories/evco-plastics/>
- Villani V., P. F. (2018, November). *Sciencedirect*. Tratto da https://www.sciencedirect.com/science/article/abs/pii/S0957415818300321?utm_source=chatgpt.com
- Y. Cohen, S. S. (2021, January). *Deploying cobots in collaborative systems: major considerations and productivity analysis*. Tratto da <https://www.scopus.com/pages/publications/85099412169>

Yang, U. O. (2023, June 17). *Human–Robot Collaborations in Smart Manufacturing Environments: Review and Outlook*. Tratto da mdpi.com:
<https://www.mdpi.com/1424-8220/23/12/5663>