Optimizing Dynamic Cellular Manufacturing System: A Deep Reinforcement Learning Approach to Profit Maximization and Inventory Management

Accepted for publication in the International Journal of Systems Science: Operations and Logistics

https://doi.org/10.1080/23302674.2025.2582482

Optimizing Dynamic Cellular Manufacturing System: A Deep Reinforcement Learning Approach to Profit Maximization and Inventory Management

Mostafa Jafari[1], Peyman Akhavan[2], Amir Hossein Akbari*[3]

1. Department of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran
ORCID: https://orcid.org/0000-0002-2794-5915, Email: jafari@iust.ac.ir
2. Essex Business School, University of Essex, Elmer Approach, Southend-on-Sea, Postal code: SS1 1LW, UK
ORCID: https://orcid.org/0000-0001-6256-3288, Email: peyman.akhavan@essex.ac.uk
3. Department of Industrial Engineering, Iran University of Science & Technology, Tehran, Iran
ORCID: https://orcid.org/0000-0002-2242-3130, Email: akbari_amir@ind.iust.ac.ir

Abstract: This paper presents a novel dynamic cellular manufacturing system that incorporates order rejection, tardiness costs, and the costs of purchasing and holding raw materials. Orders arrive at different times, requiring real-time decisions on acceptance or rejection. Accepted orders necessitate raw materials, which must be procured at the optimal time. A mathematical model with two objectives—maximizing profit and minimizing the number of rejected orders—is proposed. To solve this problem, an iteration-based hierarchical solution method with three steps has been developed. First, machines are assigned to cells using a genetic algorithm. Next, a deep reinforcement learning (DRL) algorithm with a double network is employed to manage order acceptance, schedule operations for accepted orders, assign them to the most suitable machines, procure raw materials, and determine optimal safety stock levels. The inventory management within the DRL framework is further supported by an artificial neural network. Finally, a boxing match algorithm is introduced to optimize machine placement based on DRL outputs. A case study was conducted to evaluate the performance of the proposed method, and comparative results between real-world data and the algorithm's results demonstrate the effectiveness of the proposed approach.

Keywords: cellular manufacturing, order rejection, inventory, deep reinforcement learning, stone paper.

1. Introduction

Manufacturers today are compelled to adopt innovative strategies in response to fluctuating customer demands and intense competition (Zhang et al. 2023). These strategies enhance organizational competitiveness, improve customer service, and increase flexibility and agility, all of which are essential in today's global market (Danilovic and Ilic 2019). One effective approach to achieving flexibility and agility is the implementation of cellular manufacturing systems (CMS) (Salimpour, Pourvaziri, and Azab 2021). CMS improves flexibility, optimizes intra- and inter-cell transportation, improves resource utilization, and strengthens an organization's competitive capabilities (Motahari et al. 2023). However, CMS also face limitations that challenge their ability to consistently deliver high-quality customer service. A key challenge lies in inventory management. If raw materials do not arrive at the production system on time, product completion is delayed, leading to customer dissatisfaction. Therefore, purchasing the right quantities at optimal times is critical for maximizing profits (Cuartas and Aguilar 2023). Recent studies have considered raw material inventory in scheduling contexts (Qian, Chang, and Zhang 2024). For example, (Beck 2002) examined raw material consumption in a static environment, while (Terekhov et al. 2012) and (Dehaybe, Catanzaro, and Chevalier 2024) explored inventory management within the supply chain. However, these works do not address the dynamic challenges arising from fluctuating order arrivals and raw material availability. Given capacity limitations, manufacturing systems often cannot fulfill all customer demands, making order rejection necessary to reduce tardiness costs (Perea, Yepes-Borrero, and Menezes 2023). Strategically prioritizing high-profit orders, along with coordinating order acceptance, rejection, and scheduling, can significantly enhance both production efficiency and overall sales performance (Li et al. 2023).
This paper investigates a multi-objective dynamic cellular manufacturing system (DCMS) that integrates order rejection, tardiness costs, and the costs of purchasing and holding raw materials. Orders arrive at different times, necessitating decisions on acceptance or rejection. Accepted orders require the timely procurement of raw materials, with purchasing restricted to specific periods.

Dynamic manufacturing systems require real-time, large-scale data analysis, which often exceeds the capabilities of traditional shop floor management systems and human expertise (Elia, Margherita, and Passiante 2020). This necessitates the use of automated knowledge generation and self-adaptive control mechanisms (Liu, Piplani, and Toro 2022; Liu et al. 2022). Reinforcement learning (RL) is one such method for self-adaptive control that has recently gained increasing attention and has been applied to various problems (Yan et al. 2022). Deep reinforcement learning (DRL), an emerging approach, has been increasingly utilized in inventory management (Kaynov et al. 2024), scheduling (Du and Li 2024), order acceptance, and other areas (Zhou et al. 2024). However, DRL is highly sensitive to initial data, where changes in these inputs can significantly impact the results.

This paper presents several key contributions to the field of dynamic cellular manufacturing systems (DCMS), addressing the complexities of order rejection, tardiness costs, and inventory management. Our study highlights the challenges posed by variability in order arrival times, which can lead to order rejections and necessitate a strategic approach to raw material procurement.

To address these challenges, we propose a novel hierarchical algorithm named GDRLB, which integrates three innovative components:

1. A genetic algorithm for optimal machine-to-cell assignment.
2. A deep reinforcement learning (DRL) model with a double network for real-time decision-making in production processes.
3. A boxing match algorithm to improve machine positioning.

Within this framework, inventory management is handled using a heuristic method, with safety stock levels determined by an artificial neural network (ANN). Additionally, we introduce a mathematical model that balances short-term profit maximization with the long-term objective of minimizing rejected orders. This approach provides a comprehensive strategy to enhance customer satisfaction and support business growth. A case study conducted at a stone paper factory in Gilan City, Iran, demonstrates the effectiveness of the proposed method.

The remainder of this paper is structured as follows: Section 2 reviews the literature on cellular manufacturing systems (CMS), order rejection, inventory management, and deep reinforcement learning (DRL). Section 3 introduces the mathematical modelling approach. In Section 4, a hierarchical heuristic algorithm is proposed to solve the problem. Section 5 analyzes the results of a numerical example and case study. Section 6 presents the discussion. Finally, section 7 provides the conclusions.

2. Literature Review

CMS aims to optimize production processes by organizing workstations and equipment into efficient cellular arrangements. Some researchers have focused on CMS in dynamic environments and on scheduling. (Wu et al. 2021) minimized the long-run expected makespan of stochastic customer orders under a budget constraint. (Wang and Tang 2020) created a stochastic optimization model aimed at reducing the overall expected cost of the system. (Zhang et al. 2023) drew inspiration from a real-world dynamic cellular manufacturing system designed for processing printed circuit boards. (Motahari et al. 2023) introduced a multi-objective model aimed at scheduling part families and designing a group layout within CMS. Some researchers have focused on CMS in a dynamic environments and inventory management. (Lamba et al. 2020) modeled a dynamic cellular facility layout problem that minimizes net electric energy consumption along with material handling and rearrangement costs. (Maalouf et al. 2022) proposed a distributed approach for smart production management in a CMS, integrating planning, scheduling, and material handling allocation. (Deep 2020) developed a simulated annealing algorithm based on a genetic algorithm to solve the cell formation problem.

However, limited system capacity often forces organizations to reject some orders, which can help reduce tardiness and customer dissatisfaction. The OAS problem was presented by (Slotnick and Morton 2007). The OAS problem has been studied in various contexts involving dynamic order arrivals, unknown order information, and constraints such as limited capacity, due dates, and the deterioration effect. (Mao et al. 2024) considered the OAS problem with deteriorating jobs and delivery times. (An et al. 2023b) focused on the real-time OAS problem in a flexible job-shop environment with multi-level imperfect maintenance constraints. (An et al. 2023a) introduced a real-time OAS problem in a flexible job-shop environment by incorporating condition-based preventive maintenance. (D'Haen, Braekers, and Ramaekers 2023) proposed an integrated solution approach to solve the OAS problem with dynamic order arrivals. (Rahman, Janardhanan, and Nielsen 2019) focused on a real-time OAS problem in a flow shop production system and presented a hybrid genetic algorithm (GA) and particle swarm optimization (PSO) to solve it. (Sarvestani et al. 2019) addressed an OAS problem in a single-machine environment with multiple customers. (Ju and Woo 2023) integrated long-term planning and mid-term scheduling. (Fu et al. 2021) proposed a stochastic bi-objective two-stage open shop scheduling problem that models a vehicle maintenance process, where tasks are assigned to be completed by multiple third-party companies with professional equipment. (Li et al. 2022) proposed a bi-objective parallel-testing-site scheduling-location model. According to the above studies, OAS is an important feature for organizations, and organizations can improve system quality, customer satisfaction, and profit by analyzing it. (Wang, Zhang, and Yin 2022) considered the OAS problem with a profit maximization objective and proposed a hybrid algorithm named the adaptive simulated annealing genetic algorithm to solve it. (Leng et al. 2023) developed a dual DRL approach for the OAS problem with dynamically arriving orders.

One of the essential requirements for order processing, which reduces scheduling deviations, is the availability of raw materials at the right time. Some papers have focused on inventory management in scheduling problems. (Beck 2002) formulated the job shop scheduling problem by modelling the production, consumption, and storage of inventory within a constraint-directed framework. (Grigoriev, Holthuijsen, and Van De Klundert 2005) investigated the intricacies of single-machine scheduling aimed at minimizing makespan and the number of tardy jobs while considering raw material constraints. (Terekhov et al. 2012) examined the two-stage assembly scheduling problem, which incorporates component availability constraints within a supply chain consisting of two manufacturing plants and a merge-in-transit facility. (Rabiu and Ali 2024) proposed a smart approach to optimize inventory management. (Napoleone et al. 2023) studied the synchronization of material flows in mass-customized production systems. (Li et al. 2022) studied a bi-objective model for integrated inventory and transportation at both tactical and operational levels in a four-echelon supply chain. (Singh and Mishra 2024) used machine learning to manage inventory and minimize the total average cost effectively.

Providing an efficient solution method to achieve optimal results can greatly assist dynamic systems that face various constraints. DRL is an emerging method for solving large-scale Markov Decision Process (MDP) problems and has been initially applied in areas such as inventory management, scheduling, and order acceptance (Zhou et al. 2024). Some papers focusing on inventory management are as follows. (Cuartas and Aguilar 2023) proposed a hybrid algorithm combining reinforcement learning (RL) and inventory management techniques to determine the optimal timing and quantity of inventory. (Oroojlooyjadid et al. 2022) proposed a DRL algorithm to play the Beer Game. (Zhou et al. 2024) optimized the inventory system using a new multi-agent DRL algorithm. (Dehaybe, Catanzaro, and Chevalier 2024) proposed a DRL algorithm to solve the single-item lot-sizing problem. (Stranieri, Fadda, and Stella 2024) introduced a novel heuristic DRL to solve a two-echelon supply chain inventory management problem. Some papers focusing on scheduling are as follows. (Leng et al. 2021) used a DRL algorithm to solve an OAS problem. (Leng et al. 2023) presented a dual DRL algorithm to improve revenue in an OAS problem. (Liu, Piplani, and Toro 2022) proposed a new DRL approach with a hierarchical and distributed architecture to solve the DFJSP with constant job arrivals. (Luo 2020) used DRL to solve the DFJSP under new job insertions. (Lei et al. 2023) proposed a novel end-to-end hierarchical RL algorithm to solve the DFJSP where the processing information of newly arrived jobs is unknown. (Silva, Valladão, and Homem-de-Mello 2021) developed a data-driven approach for a dynamic optimization problem. (Hsieh, Lin, and Wang 2024) developed an algorithm based on dispatching rules to minimize tardiness in a flexible flow shop environment.

3. Problem Description and Mathematical Model

There exists a dynamic cellular manufacturing system (DCMS) composed of different cells and machines. The number of cells is predetermined, and machines must be assigned to these cells. Orders arrive at different times, requiring a decision on whether to accept or reject them. Each accepted order consists of several operations that must be processed on different machines. Each order has an associated revenue, processing times on various machines, a due date, and a tardiness cost. Orders also require raw materials and cannot be processed until the raw materials are available. The system incurs purchasing and holding costs, which must be managed appropriately.

The general procedure of the problem is as follows:

- Machines are assigned to cells.
- Orders from customers are received at different times, each with its specific attributes.
- Suitable orders are accepted, the required raw materials are procured, and processing is carried out.
- The completed orders are delivered to customers.

The objectives of the problem are to maximize profit (calculated as revenue from accepted orders minus tardiness costs, intra- and inter-cell movement costs, and the costs of purchasing and holding raw materials) and to minimize the number of rejected orders.

The problem assumptions include the following:

- ➢ All parameters are known and deterministic.
- ➢ Machines are continuously available.
- ➢ The number of cells is predetermined.
- ➢ The time and cost of cell formation are considered to be zero.
- ➢ Each order consists of several operations that must be processed in a specific sequence.
- ➢ Processing of a component begins only when all its raw materials are available.
- ➢ There is no inventory at the beginning of the planning period.
- ➢ Purchasing incurs a fixed cost (A), and raw materials cannot be bought at any time (t_{allow}).

In Table 1, symbols, problem parameters, and decision variables are presented, and the mathematical model is presented.

Table 1. Indexes, parameters, and variables

notation	*description*	*notation*	*description*
index			
c	Index of cell c=1…C	j_p	Index of operation of order j=1…n_p
p	Index of order p=1…P	e	Index of raw material e=1…E
m	Index of machine m=1…M	t	Index of time t=1…T
parameter			
R_p	Revenue of order p	C_p	Tardiness cost of order p
C_{in}	Inter-cell cost	t_{jpm}	Time of operation j of order p in machine m
C_{out}	Intra-cell cost	d_p	Due date of part p
A_{jpm}	The ability of machine m for processing operations j of order p	Ord_e	The cost of buying raw material e
B_{epj}	Number of raw materials e required for operation j in order p	H_e	The cost of holding raw material e
Variable			

Table 1. Indexes, parameters, and variables

notation	*description*	*notation*	*description*
N_{mc}	Binary variable that is 1 if the machine m allocated to cell c and is zero otherwise	TF_P	Tardiness of order p
X_{jpmct}	Binary variable that is 1 if processed operations j of order p in machine m in cell c at period t and is zero otherwise	L_{et}	The level of the raw material e at the end of period t
COT_{jp}	completion time for operations j of order p	O_{et}	The amount of raw material e that is bought in period t ∈ t_{allow}
ST_{jp}	start time for operations j of order p	Y_p	If order p is accepted 1

The mathematical model is as follows.

Objective 1: The first objective function aims to maximize profit, which is calculated as the total revenue from accepted orders minus the costs associated with tardiness, intra- and inter-cell movements, and the purchase and holding of raw materials. If the sum of inter-cell movement is Mov_{in} and the sum of intra-cell movement is Mov_{out}, Objective Function 1 can be modelled as follows:

$$Max\ Z1 = \sum_p R_p Y_p - \sum_p C_p TF_p - C_{in} Mov_{in} - C_{out} Mov_{out} - \sum_{e,t} H_e L_{et} - \sum_{e,t} Ord_e O_{et} + \sum_{e,t} \mathrm{A} \times \min(1, O_{et})$$

Objective 2**:** The second objective function aims to minimize the number of rejected orders.

$$Min\ Z = \sum_p (1 - Y_p)$$

The objective function 2 is equivalent to maximizing the number of accepted orders.

Constraint:

$\sum_{j,p,m,c,t} X_{jpmct} \leq 1$	$\forall\ j,p$	(1)
$\sum_{c,t} X_{jpmct} \leq A_{jpm}$	$\forall\ j,p,m$	(2)
$\sum_c N_{mc} \leq 1$ and $\sum_{j,p} X_{jpmct} \leq N_{mc}$	$\forall\ m$ and $\forall\ m,c,t$	(3)
$ST_{jp} = \sum_{m,c,t} t.X_{jpmct}$	$\forall\ j,p$	(4)
$COT_{jp} \geq ST_{jp} + \sum_{m,c,t} X_{jpmct}\ t_{jpm}$	$\forall\ j,p$	(5)
$\sum_{j,p} \sum_{t'=1}^{t+t_{jpm}} X_{jpmct'} \leq 1$	$\forall\ m,c,t$	(6)
$TF_p \geq COT_{jp} - d_p$	$\forall\ j,p$	(7)
$L_{et} = O_{et} - \sum_{j,p,m,c} B_{epj}.\ X_{jpmct}$	$\forall\ t = 1, e$	(8)
$L_{et} = L_{e(t-1)} - \sum_{j,p,m,c} B_{epj}\ .X_{jpmct} + O_{et}$	$\forall\ t \geq 2, e$	(9)

Constraint (1) specifies the conditions under which orders are accepted by the model. Constraint (2) ensures that if an order is received, its operations can only be processed on the corresponding machines. Constraint (3) states that each machine can be assigned to only one cell at a time. Constraints (4) and

(5) represent the start and finish times of operations, respectively. Constraints (6) address overlapping operations on machines and within cells to prevent scheduling conflicts. Constraint (7) handles tardiness. by calculating any delays beyond due dates. Constraint (8) calculates the inventory level of raw material type (e) at the end of time period (t=1), which is equal to the number of available materials at the beginning of time (t) minus the amount used during the process. Constraint (9) illustrates the inventory level at the end of a period, as the previous period's inventory minus the amount consumed during the current period, plus any newly received materials.

4. **Methodology**

This section presents a hierarchical three-step solution method to address the problem.

1. Machine Assignment to Cells (Step 1): In the first step, machines are assigned to cells using a genetic algorithm (GA). This step generates an initial layout without prior information, meaning it may not be optimal.
2. Order Management and Scheduling with DRL (Step 2): In the second step, a deep reinforcement learning (DRL) algorithm is used to:
   - Determine order acceptance.
   - Schedule the accepted orders.
   - Assign the operations of these orders to suitable machines.
   - Purchase the necessary raw materials at optimal times.

   Since the DRL approach relies on decision-making in a dynamic environment, it is formulated as a Markov Decision Process (MDP). The MDP model defines the states, actions, rewards, and transitions necessary for optimizing order processing decisions. The DRL algorithm then learns an optimal policy based on this model to maximize profitability while minimizing order rejection.
3. Layout Optimization with BMA (Step 3): After order scheduling and raw material procurement, the locations of machines within the cells are adjusted using a boxing match algorithm (BMA) to achieve an optimal layout for the production system.
4. Iterative Refinement (Steps 2 and 3): An iterative procedure is applied in steps 2 and 3 to progressively enhance the solution. After obtaining an initial layout from Step 1, Steps 2 and 3 are repeated iteratively to refine both order processing decisions and machine layout. Since optimal order processing decisions influence machine layout and vice versa, this iterative adjustment continues until the difference between consecutive optimal solutions is below a predefined threshold, α.

Fig. 1 shows the solution method.

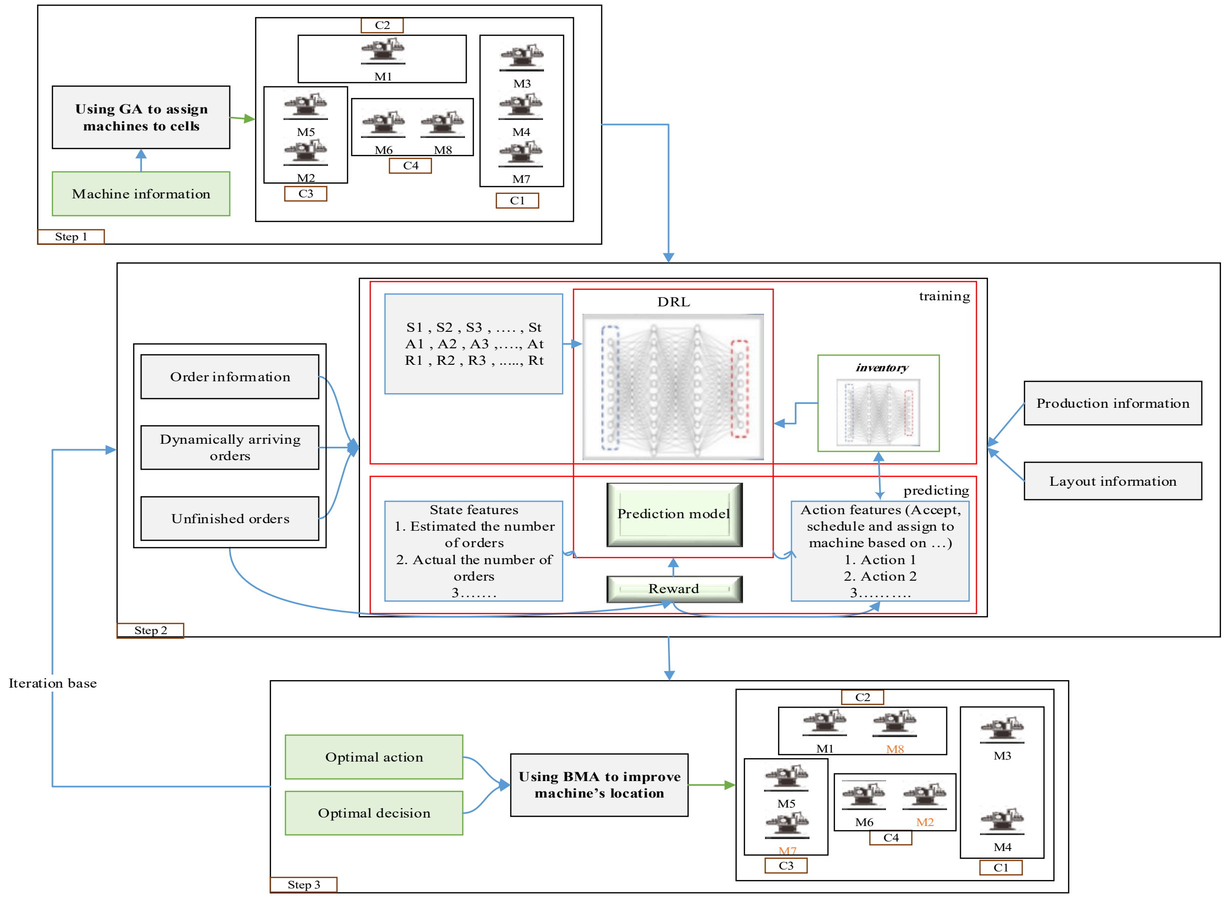

Fig. 1. The solution method

4.1. Assigning Machines to Cells by GA (Step 1 of Solution Method)

Since no historical data (such as order execution sequences, machine utilization levels, or task sequences) is available, achieving an optimal machine-to-cell assignment is highly challenging, if not impossible. Therefore, introducing heuristic methods to determine the best layout can significantly contribute to cost reduction and increased system efficiency. In this section, to minimize relocation and depreciation costs (by balancing the workload distribution across the system), we aim to enhance each cell's capability to perform various tasks, ensuring that all cells can execute all operations whenever feasible. This approach reduces intercellular movement costs, leading to overall system cost savings. It is important to note that no method can achieve an optimal layout without prior data. Consequently, in this study, the initial layout is proposed and then further refined in subsequent sections.

This section presents an innovative approach for the initial assignment of machines to cells. The primary goal of this method is to balance the operational capacity of each cell. If the capacity of each cell is determined by the capabilities of its machines, then this capacity can be calculated as follows.

$$AC_c = \sum_{j,p,m \in c} A_{jpm} \qquad (10)$$

If each cell's capacity to perform operations is fully utilized, the probability of inter-cell movement increases, while the probability of intra-cell movement decreases. Therefore, to assign machines to cells, we define the following problem.

$$Min\ Z = \sqrt{\frac{\sum_c (AC_c - Ave(AC_c))^2}{C}}$$

S. T

$$\sum_c N_{mc} \leq 1 \qquad \forall\, m \qquad (11)$$

The objective of the above problem is to enhance each cell's ability.
Considering that the above problem is nonlinear, a genetic algorithm is used to solve the problem.

Algorithm 1. The proposed algorithm of step 1

1. Input: iteration (iter)=300, number of papulation (npop)=100, number of mutation (nmut)=40, number of crossover (ncross)=60, AC_c,
2. Output: firstly layout
3. Randomly assign machine to cell and calculate the objective function for each solution
4. iter = 1
5. While iter < 300
5.1. Generate new solution based on one point crossover
5.2. Generate new solutions based on mutation
5.2.1. Assign machine $\boldsymbol{m}$ from the cell $\boldsymbol{c}$ with the highest AC_c into cell $\boldsymbol{c'}$ with the lowest AC_c
5.3. Combine the obtained solutions of the mutation, crossover, and previous iteration.
5.4. Calculate objective function of the all solutions.
5.5. Select the best solution (the number of the best solutions is equal to npop)
5.6. iter = iter + 1
6. end while
7. use the best solution as the initial layout

To evaluate the efficiency of the proposed GA, it is compared with the methods PSO, Classic GA, BMA, GWO, SA and ICA. Table 2 illustrates the improvement achieved by the proposed genetic algorithm compared to the other methods.

Table 2. The improvement of the genetic algorithm compared to other methods (percent)

	PSO	GA Classic*	BMA	GWO	SA	ICA
Objective	2.1	8.9	1.3	2.2	7.3	3.1
Time	0.3	-0.7	1.2	1.4	-0.9	2.1

* GA Classic: the GA with random crossover and mutation

4.2. DRL (Step 2 of Solution Method)

This section introduces a deep reinforcement learning (DRL) algorithm for order acceptance, scheduling the operations of accepted orders, and procuring the raw materials needed to process those orders. The DRL framework is designed to handle the dynamic nature of the production system, where orders arrive at different times and decisions must be made in real-time.
The problem is formulated as a Markov Decision Process (MDP), where the system state represents key variables such as machine availability, order attributes, and raw material inventory levels. The action space includes accepting or rejecting orders, scheduling operations, assigning machines, and determining the optimal raw material procurement strategy. The DRL algorithm is trained using an ε-greedy policy to balance exploration and exploitation, and a reward function is defined to maximize profit while minimizing order rejection.

4.2.1. State Representation

In production systems, reinforcement learning states may encompass variables like the number of machines, the remaining operations for each order, the processing time for each order, and the time left for an order's completion (Shahrabi, Adibi, and Mahootchi 2017; Shiue, Lee, and Su 2018). In this research, ten states have been defined to represent the system's current state. All the introduced states are within the range of [0,1], based on (Luo 2020). To define the states of the reinforcement learning algorithm, the following parameters are first introduced.

Table 3. General Parameters of All States

Parameter	Description
$COT_m(t)$	The completion time of the last operation assigned to machine M at rescheduling point t
$OC_p(t)$	The number of operations completed for order p at time t
$ONC_p(t)$	The number of operations that are incomplete for order p at time t
$U_m(t)$	the utilization rate of machine M at time t $U_m(t) = \sum_{p,c}^{P,C} \sum_{j=1}^{OC_p(t)} t_{jpm}.x_{jpmct} \Big/ COT_m(t)$
$CR_P(t)$	the completion rate of order p at t, $CR_P(t) = OC_p(t) \Big/ \sum_p OC_p(t)$
$H_e(t)$	the number of raw materials of type e remaining in the production system at rescheduled point t
$E_e(t)$	the number of raw materials of type e that consumed in period t
$NE_{pe}(t)$	the number of raw materials of type e needed to complete the order p

Using the above notations, the state features at each rescheduling point t can be outlined as follows.

4.2.1.1. Estimating the Number of Orders with Tardiness (*ETC(t)*)

This state represents the estimated number of orders that are expected to experience tardiness at time t. To calculate this state, we first define the following items.

Table 4. The parameters of the state 1

$SL_p(t)$	$d_p - \sum_{m,j\in remaining\ operation} t_{jpm} \Big/ M - t$
$C_p(t)$	The estimated completion time of order p at point t
NO_{loss}	Estimating the number of unprofitable orders
N_{total}	The number of uncompleted orders

The calculation method is given in Algorithm 2.

Algorithm 2. The procedure of calculating ***ETC(t)***

```
1.      Input: COT_m(t), OP_p(t), NOP_p(t), d_p
2.      Output: ETC(t)
3.      NO_loss = 0, N_total = 0
4.      for p = 1:P
4.1.        if NOP_p(t) > 0
4.2.          N_total = N_total + 1
4.3.          t̄_jpm = Σ_{j∈NOPp(t)} t_jpm / M
4.4.          C_p(t) = t̄_jpm + COT_m(t)
4.5.          SL_p = d_p - C_p(t)
4.6.          if SL_p < 0
4.6.1.            NO_loss = NO_loss + 1
4.6.2.          end if
5.      end for
6.      ETC(t) = NO_loss / N_total
```

4.2.1.2. The Actual Number of Orders with Tardiness (*ATC(t)*)

This state includes orders whose completion time will certainly exceed their due date. The calculation method of this state is similar to State 1.

4.2.1.3. Estimating the Number of Unprofitable Orders (*ETCN(t)*)

The calculation method of State 3 is similar to State 1. Line 4.6 and Line 6 are modified as follows:

Line 4.6: if $SL_p < 0$ && $R_p - |SL_p| * T_P < 0$

Line 6: $ETCN(t) = NO_{loss} / N_{total}$

4.2.1.4. The Actual Number of Unprofitable Orders (*ATCN(t)*)

This state includes orders whose cost of orders exceeds their revenue. The calculation method of this state is similar to State 3.

4.2.1.5. Estimating the Number of Orders Whose Tardiness Cost Exceeds a Fraction of Their Revenue (*ETCN'(t)*)

This state accounts for various costs, including tardiness, inter- and intra-cell movement costs, and raw material procurement and holding costs. A percentage (alpha) of the revenue from each order is deducted to calculate the total cost. This method follows a similar approach to State 3, adjusting the order revenues accordingly.

$\boldsymbol{R_p} = \boldsymbol{\alpha}.\boldsymbol{R_p} \quad \alpha € (0,1)$

4.2.1.6. The Actual Number of Orders Whose Tardiness Cost Exceeds a Fraction of Their Revenue (*ATCN'(t)*)

This state is similar to state 5 and considers the actual number of affected orders.

4.2.1.7. Estimating the Number of Orders with a Shortage of Raw Materials *ER(t)*

To define this state, first, NR_{loss} is considered as the number of orders whose raw materials are facing with shortage and N_{total} as the total number of orders in progress. The calculation method is provided in Algorithm 3.

Algorithm 3. The procedure of calculating ***ER(t)***

1. Input: $\boldsymbol{OP_p(t)}$, $\boldsymbol{NOP_p(t)}$, $\boldsymbol{B_{epj}}$, $\boldsymbol{L_{et}}$
2. Output: $\boldsymbol{ER(t)}$
3. $\boldsymbol{NR_{loss} = 0, N_{total} = 0}$
4. **for** p = 1:P

4.1. **if** $\boldsymbol{NOP_p(t) > 0}$

4.2. $\boldsymbol{N_{total} = N_{total} + 1}$

4.6. **if** $\sum_j \boldsymbol{B_{epj}} > \boldsymbol{L_{et}}$ for $\boldsymbol{j} \in \boldsymbol{NOPp(t)}$ ***and P*** $\in$ ***current order and accepted orders***

4.6.1. $\boldsymbol{NR_{loss}} = \boldsymbol{NR_{loss}} + 1$

4.6.2. **end if**

5. **end for**
6. $\boldsymbol{ER(t)} = \boldsymbol{NR_{loss}} / \boldsymbol{N_{total}}$

4.2.1.8. The Actual Number of Orders with a Shortage of Raw Materials *AR(t)*

This state is similar to the one above, but it specifically identifies orders that will face a shortage.

4.2.1.9. Order Consumption Rate in Each Period *OCR(t)*

Calculated by dividing the average amount of raw materials used in each period by the total quantity of raw materials purchased up to time t.

$$OCR_e(t) = \sum_{t'=0}^{t} E_e(t')/O_{et'} \qquad \forall e \qquad (12)$$

4.2.1.10. The Rate of Raw Materials in the System *RRM(t)*

Dividing the quantity of raw materials in the system by the raw materials needed for all orders, including those currently being processed and those accepted but still in the queue.

$$RRM_e(t) = L_{et}/\sum_p NE_{pe} \qquad \forall e \quad \forall p \in \mathit{accepted\ and\ uncomplete\ orders} \qquad (13)$$

4.2.2. Action Representation

The complexity of the problem, involving multiple continuous decision-making issues, makes it difficult to establish a single rule to address order acceptance and rejection, scheduling operations, assigning machines, and managing raw materials. As a result, various actions are necessary to achieve optimal outcomes. The research problem is divided into four sub-problems:

- Accepting appropriate orders.

- Scheduling the operations of accepted orders.
- Assigning the operations to suitable machines.
- Procuring the necessary raw materials.

The decision-maker must choose the best rule at each decision point, and the proposed algorithm introduces nine actions to guide optimal decision-making. Given that the processing time and order completion time must be estimated, and that the processing time of operations varies across different machines, for convenience, the value $\overline{t_{jp}}$ is the average processing time of operation j across all machines. This value is calculated as follows.

$$\overline{t_{jp}} = \frac{\sum_m t_{jpm}}{M} \qquad (14)$$

In all actions, three sub-problems must be solved:

1. Scheduling the remaining operations and the operations of newly accepted orders. The scheduling/rescheduling of operations is based on the scheduling of their respective orders. After scheduling, the operations are assigned to the first idle machine (if multiple operations can be assigned to a machine simultaneously, the operation with the earliest due date is selected).
2. Determining which orders to reject. After rejecting certain orders, the remaining orders must be scheduled (following the same approach as in sub-problem 1).
3. Procuring the required raw materials at the right time and in the correct amounts.

First, different rules are introduced for each sub-problem, and then a combination of these rules is considered as a set of possible actions.

4.2.2.1. Scheduling of Received Orders

Strategy 1: This strategy dynamically prioritizes orders by balancing profit potential and production time efficiency. The steps of Strategy 1 are as follows:

1. $TO_{jp} = \frac{\sum_{m=1}^{M} t_{jpm}}{M} \qquad \forall j,p \qquad (15)$
2. $TP_p = \sum_j TO_{jp} \qquad \forall p \qquad (16)$
3. $RT_p = \frac{R_p}{TP_p} \qquad \forall p \qquad (17)$
4. Arrange each order in descending order based RT_p.

The sequence obtained from this method is called Sequence_Method1 (SM1).

Strategy 2: This strategy focuses on meeting customer delivery deadlines. The steps of Strategy 2 are as follows:

1. Sort orders in ascending order based on their due date.

The sequence obtained from this method is called Sequence_Method2 (SM2).

Strategy 3: This strategy combines the strengths of both profit-time efficiency (SM1) and deadline sensitivity (SM2) through a weighted fusion approach. The steps of Strategy 3 are as follows:

1. Calculate SM1
2. Calculate SM2
3. Calculate FS, where $FS=W_1*SM_1 + W_2*SM_2 \qquad 0<W_1,W_2<1$
4. Sort (descending) orders based on the rank obtained in the ***FS***. If two orders have the same ranking, the order with higher revenue is prioritized.

4.2.2.2. Rejecting Orders

After determining the sequence of orders, operations, and their start and end times, the profitability of each order is calculated based on Objective 1. In this case, some orders yield a profit, while others incur a loss. The following strategies are used to reject orders:

Strategy 1: The order with the greatest loss is rejected, the operations are reassigned to the machines, and the profit of each remaining order is recalculated. This process continues until all unprofitable orders have been rejected. This iterative optimization process systematically eliminates unprofitable orders while dynamically re-evaluating system profitability.

Strategy 2: An order with a negative profit is randomly selected and removed. Then, the operations are reassigned to machines, and the profit of each remaining order is recalculated. This process continues until all unprofitable orders have been removed. This iterative optimization protocol enhances system profitability through targeted order rejection and dynamic resource reallocation.

Strategy 3: The difference between the due date and completion time (D) is calculated for each order. Orders with positive D (completed before the due date) are rescheduled as late as possible until D becomes negative. This allows some unprofitable orders to be completed earlier, reducing their tardiness costs. Then, the order with the highest loss is rejected, operations are reassigned to machines, and profits are recalculated. This process repeats until all unprofitable orders are rejected. This advanced strategy combines deadline management with profit maximization through a two-phase iterative process.

4.2.2.3. Buying and Holding of Raw Materials

Balancing procurement costs and inventory holding costs while preventing stockouts leads to significant improvements in production system performance and can substantially reduce overall inventory expenses. The main challenge in this section is choosing the right quantity and timing for purchasing raw materials. The following strategy determines the optimal amount of raw material to be purchased at the appropriate time:

First, the average amount of raw materials consumed in the past period (from 1 up to now) is calculated (***Ave_e***).

$$Ave_e = {\sum B_{epj}}/{t} \qquad \begin{array}{c} \text{t=1….t} \\ \text{p = total accepted orders} \\ \text{j=total operation} \end{array} \qquad (18)$$

The estimated number of raw materials that will be held in future periods (M_{et}) is as follows.

$$M_{e(t+n)} = M_{et} - (n+1)Ave_e \qquad \forall t, e \qquad (19)$$

According to the above equation, the expected amount of raw material held for k periods ($E(H_e^{\acute{t}})$) is calculated as follows.

$$E(H_e^{\acute{t}}) = M_{\mathrm{et}} - \sum_{i=0}^{t}(i+1)Ave_e \qquad \forall e, \acute{t} = t \dots t+k \qquad (20)$$

The purchase number of raw materials (O_{et}) is determined using Eq. 19 to minimize the combined costs of buying and holding over k periods. Raw materials cannot be bought in every period, and shortages are not allowed to avoid production stoppages. Algorithm 4 is employed to calculate the purchase quantity while adhering to these constraints. The future time frame for raw material purchasing is represented by the parameter Alpha, and decisions about buying or holding are made by considering the Alpha periods (E(I) = Alpha).

Algorithm 4. Procedure to buy raw materials

1. Input: $\boldsymbol{B_{epj}}$, $\boldsymbol{L_{et}}$, $\boldsymbol{M_{et}}$, $\boldsymbol{Ave_e}$, $\boldsymbol{H_e^{t'}}$, ***Alpha*** (for example Alpha=5)
2. Output: ***buying of raw materials***
3. analyze all policies t+1 up to t+5
4. select the best policy

for example, t+1, t+2, t+4, t+5 € $\boldsymbol{t_{allow}}$ and t+3 ∉ $\boldsymbol{t_{allow}}$

all policy for this sample is as follows

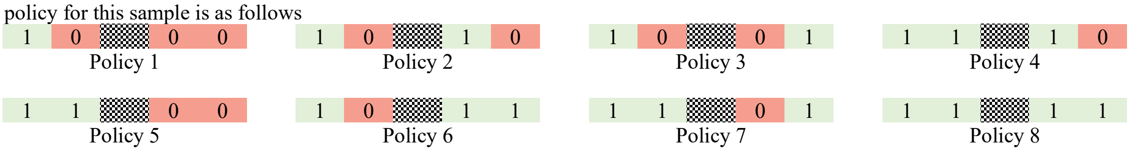

Fig. 2. The different policy of buying of raw materials

In Fig. 2, 1 means buying raw materials, and 0 means no buying. These policies are reviewed, and the best policy is selected. The purchase amount is equal to the total amount of consumption until the next purchase. (the average amount consumed in previous rounds).

Purchases cannot be made in every period, so precautionary raw materials, denoted as Ave_e, are kept in stock. This value is continuously updated based on past data, acting as a learning function. An artificial neural network (ANN) determines the amount of Ave_e, with input layers consisting of factors such as the number of accepted orders, total processing time, average due date, purchase restrictions for the next period, and current inventory levels. The output layer is Ave_e, and the ANN has 5 hidden layers. Initially, at t=1, Ave_e is set to zero, and from subsequent periods onward, the ANN updates its value. To optimize the solution process and reduce complexity, the ANN is run every five periods.

Actions:

According to the above definitions, the actions of the problem include all possible combinations. Therefore, the total number of actions is calculated as follows.

Scheduling strategy	***Rejection strategy***	***Buying strategy***	***The number of actions***
3	*3*	*1*	*3×3×1=9*

4.2.3. The Action Selection Policy

In this study, the ε-greedy approach is employed for action selection during the training phase. Once training is complete, the action with the highest Q-value is chosen for accepting and scheduling incoming orders, as well as for assigning operations to machines. Furthermore, the method outlined by (Luo 2020) is applied to prevent the algorithm from falling into local optimum.

4.2.4. Rewards Representation

The proposed problem includes two objective functions: maximizing profit and minimizing the number of rejected orders. Therefore, the reward rt for the pair of state and action (s_t, a_t) is defined by considering two characteristics: the current state s_t and the next state s_t+1.

In this case, if we define ***obj1(t)*** as the amount of profit obtained up to moment t, define ***obj2(t)*** as the amount of rejected orders up to moment ***t***, p(t) as the total number of orders received up to moment t, and w as the weight, the function for the first objective is; Then, the value of r(t) is calculated as follows.

The procedure to calculate r_t is given in Algorithm 5.

Algorithm 5: Procedure of rewards

if obj2(t+1) – obj2(t) = 0

$r_t = \frac{obj1(t+1)-obj1(t)}{obj1(t+1)}$

else

$r_t = w\frac{obj1(t+1)-obj1(t)}{obj1(t)} - (1-w)\frac{obj2(t+1)-obj2(t)}{P(t+1)}$

end

In the proposed algorithm, the amount of W is equal to 0.5.

4.2.5. Asynchronous Transition

The transition begins at the decision-making point at time step t and concludes when the operation is completed at time step t+1. However, time step t+1 does not always align with the subsequent decision-making point. The procedure is as follows:

1. Observe the state $\boldsymbol{s_t}$ when machines become idle, preprocess the state $\boldsymbol{Ø_t} = \boldsymbol{Ø}(\boldsymbol{s_t})$, and record $\boldsymbol{Ø_t}$
2. Take action $\boldsymbol{a_t}$, select an order to complete its operations
3. Until the operation is completed, calculate the reward $\boldsymbol{r_t}$ and record $\boldsymbol{Ø_t+1}$
4. Store the experience ($\boldsymbol{Ø_t}$, $\boldsymbol{a_t}$, $\boldsymbol{r_t}$, $\boldsymbol{Ø_{t+1}}$) in the replay memory

4.2.6. Network Structure

This study employs a deep neural network consisting of nine fully connected layers:

- One input layer,
- Seven hidden layers, and
- One output layer.

The number of nodes in the input layer corresponds to the state features (10 nodes), while the output layer contains as many nodes as there are available actions (9 nodes). Each hidden layer consists of 72 nodes.

The activation functions used are:

- "tansig" for the input layer,
- "tanh" for the hidden layers, and
- "softmax" for the output layer.

The network is trained over 10,000 epochs, with the following hyperparameters:

- Learning rate decaying from 10^{-2} to 10^{-3},
- Soft target update strategy (τ) of 0.01,
- Discount factor (γ) of 0.3,
- Exploration policy is ε-greedy with ε decreasing from 1 to 0.1,
- Batch Size is 64,
- Exploration decay rate (ε Decay) is 0.995,
- Target network update frequency is every 50 episodes, and
- Replay buffer size (N) is 1,000

4.2.7. Overall Framework of the Training Method

In the training process, the decision point t is defined as each time a new order arrives or an operation is completed. The training technique based on procedure follows the methodology proposed by (Luo 2020).

4.3. Improvement of Machine Placement in Cells by BMA (Step 3 of Solution Method)

This section introduces a mathematical model for optimizing machines placement in cells.

$X_{jj'pcc'}$	1 if operation j of order p is processed in cell c and operation j' of order p is processed cell c', otherwise 0
$X_{jj'pc}$	1 if the operation j and j' of order p is processed in cell c, otherwise 0

The mathematical model for improving machine placement in cells is as follows:

$$\min Z = \sum C^{out} X_{jj'pcc'} + \sum C^{in} X_{jj'pc}$$

S. T

Equation (1), (2), (3), (6), (7)

$$\sum_{c=1}^{C} N_{mc} \leq 1 \qquad \forall m \qquad (21)$$

The input data for the above model is based on a real case study covering one month. Decisions regarding order acceptance, scheduling, and raw material purchasing are made based on the output of the DRL algorithm.

Due to the high dimensionality of the problem, a meta-heuristic algorithm based on the boxing match algorithm (BMA) has been used to solve it. BMA was introduced by (Tanhaeean, Tavakkoli-Moghaddam, and Akbari 2022). This approach finds feasible solutions by partitioning the solution space into various sections and employing a semi-zigzag search within each section to generate new solutions. Before defining the algorithm, the following are defined.

$$AC_m^{out} = \frac{\sum_{j,p,c,c',t} X_{jpmct} X_{jj'pcc'}}{\sum_{j,p,c,c',t,m} X_{jpmct} X_{jj'pcc'}} \qquad \text{The number of movements from inside of the cell to outside of the cell} \qquad (22)$$

$$AC_m^{in} = \frac{\sum_{j,p,c,c',t} X_{jpmct} X_{jj'pc'c}}{\sum_{j,p,c,c',t,m} X_{jpmct} X_{jj'pc'c}} \qquad \text{the number of movements from outside of the cell to inside of the cell} \qquad (23)$$

$SC_m^{in} = \frac{\sum_{j,p,c,c',t} X_{jpmct} X_{jj'pc'c}}{\sum_{j,p,c,c',t,m} X_{jpmct} X_{jj'pc'c}}$	The standard deviation of machine m in relation to other machines for intra-cell movement.	(24)
$SC_m^{in} = \sqrt{\frac{\sum (AC_m^{in} - Ave\_AC_m^{in})^2}{M}}$	The standard deviation of machine m in relation to other machines for inter-cell movement.	(25)

The first line of the solution representation as call and the second line as machine. The proposed algorithm to assign machines to cells is given in Algorithm 6.

Algorithm 6. The proposed algorithm to assign machines to cells

1. Input: Iteration = 10, number of populations = 20, arrived orders = 1000, number of sections = 4 (the number of solutions in sections 1, 2, 3, and 4 are 8, 6, 4, and 2, respectively), initial energy = 1000, final energy= 1.
2. Output: reassign machine
3. Assign the machine to the cell based on section 4.1
4. Calculate objective function of problem.
5. iteration = 1
6. While iteration < 10
6.1. Generate new solution based one semi-zigzag search
The number of new solutions generated for this section is 8 × 4 = 32
The number of new solutions generated for this section is 6 × 3 = 18
The number of new solutions generated for this section is 4 × 2 = 8
The number of new solutions generated for this section is 2 × 1 = 2
* Method 1
6.2. Calculate Eq. 21, 22, 23, 24
6.3. Find machine m with $max_c(AC_m^{out} + AC_m^{in})$ and $min_c(SC_m^{out} + SC_m^{in})$, and cell c with the more movement for machine m
6.4. Assign machine m to cell c
6.5. If the cost of the new layout < the cost of the current layout
6.5.1. Go to line 6.2
6.6. end if

* Method 2
6.7. random change location m and m'

6.8. Combine new solutions in each section
6.9. Accept orders, schedule the operation of accepted orders and assign them to the machine based on the DRL result, and calculate objective function
6.10. Select the best solution (in each section is equal to the number of solutions in each section)
6.11. iteration = iteration + 1
6.12. decrease energy
end while

To evaluate the efficiency of the proposed BMA, it is compared with Methods PSO, GA Classic, the proposed GA in section 4.1, GWO, and SA. Table 5 illustrates the improvement achieved by the proposed genetic algorithm compared to other methods.

Table 5. The improvement of the BMA compared to other methods (percent)

	PSO	GA Classic*	The proposed GA in section 4.1	GWO	SA	ICA
Objective	5.2	11.3	1.9	4.1	12.07	8.3
Time	0.2	-2.4	-0.1	2.5	-2.9	1.9

* GA Classic: the GA with random crossover and mutation

5. Case Study

This study examines the innovative production process of stone paper, a sustainable alternative to traditional paper. Stone paper is made from stone powder (calcium carbonate) combined with polyethylene, eliminating the need for wood pulp. Production of stone paper in Gilan, Iran, began in 2014. The factory operates within a streamlined three-level supply chain: suppliers provide the essential stone and polyethylene, the production facility processes these raw materials, and a diverse customer base purchases various paper products.

Within the factory, three production halls play vital roles, with one hall exclusively dedicated to crafting stone paper packets. This particular hall features three specialized workstations (cells), equipped with nine machines and staffed by eight skilled operators.

Cell 1 (C1): Raw materials—comprising 80% stone powder and 20% polyethylene—are meticulously mixed to create a rich paper pulp. This cell is equipped with two machines responsible for precise blending of the raw materials.

Cell 2 (C2): The prepared pulp is transported to this cell, where it is processed into various stone paper products, including packets, labels, books, and carton packaging. This cell contains four machines, ensuring efficient and high-quality production of different products.

Cell 3 (C3): This stage focuses on the final finishing processes, where prints and designs are applied to enhance the visual appeal of the products. It is equipped with three machines that handle various printing and surface treatment operations.

To maintain operational efficiency, every order must be processed within a maximum of five steps (It is possible to return from the cell 3 to cell 2 -subsurface printing is performed for some products-), with operations tailored to the specifics of each request. This ensures a flexible and adaptable production process that meets customer demands efficiently.

For the purpose of this study, we analyzed production data from the past month, providing valuable insights into the dynamic processes and collaborative efforts that drive the success of stone paper manufacturing. This analysis highlights the importance of effective order scheduling, machine allocation, and resource management, ultimately contributing to a more sustainable future.

Several different problem instances based on the case study have been examined to evaluate the model's performance. Table 6 shows the different sizes of problem instances.

Table 6. The size of the problem instance

	Max Cell	orders	Machine	Operation	Raw material
Small	4	1000	10	5	2
Medium	8	5000	20	10	4
Large	12	10000	40	15	6

In the case study, some data (such as due date, revenue, processing time of operations, order arrival time, order arrival rate, etc.) are precise.

the performance of the proposed methods has been compared with three DRL algorithms from the literature ((Wu et al. 2024), (Liu, Tseng, and Weng 2024), (Gan et al. 2024)). Algorithms used for comparison are customized according to the proposed problem, and steps 1 and 3 of the proposed methods were incorporated to them. A total of 30 examples were produced in small, medium and large sizes. Table 7 shows the results of the proposed algorithm:

Table 7. The result of the proposed algorithm compared with three DRL in literature in problem instance of proposed model

Size	Instance	Proposed		(Wu et al. 2024)		(Liu, Tseng, and Weng 2024)		(Gan et al. 2024)	
		Obj1	%Obj2	Obj1	%Obj2	Obj1	%Obj2	Obj1	%Obj2
Small	1	**5242**	**91.42**	5226	88.22	5163	90.1	5199	90
	2	5715	90.5	5671	89.09	5743	88.69	5673	89.77
	3	6066	92.79	5985	91.03	6030	92.59	6026	92.64
	4	5676	91.27	5641	88.02	5628	90.8	5603	90.26
	5	5063	90.17	5047	91.69	4999	89.36	4972	90.92
	6	5774	90.45	5677	91.35	5556	90.06	5731	89.26
	7	6046	87.11	6017	86.37	5920	86.89	6012	87.33
	8	**5799**	**90.43**	5796	87.07	5669	87.77	5703	89.67
	9	4878	90.44	4927	87.06	4888	89.58	4835	87.39
	10	5622	92.66	5547	90.41	5602	90.27	5477	89.19
Time (min)		26.14		19.24		17.14		18.01	
Medium	1	26413	90.04	26225	89.92	25602	89.19	26389	90.69
	2	28696	89.5	28044	90.78	27907	89.87	28622	90.09
	3	32050	89.7	31931	89.33	32049	88.88	31908	88.72
	4	30341	90.19	29669	86.73	29910	88.68	30285	87.22
	5	24595	89.86	23801	89.49	24567	88.79	24549	88.92

	6	28853	91.47	29040	88.45	28755	86.56	28979	85.6
	7	29219	90.1	29584	87.77	29552	88.14	28895	87.42
	8	**28478**	**94.16**	28470	90.12	27979	91.87	27821	91.93
	9	**24925**	**89.06**	24247	86.38	24878	87.4	24568	87.11
	10	27548	93.97	27504	91.51	27023	93.1	27321	93.01
Time (min)		49.24		36.57		29.73		27.1	
Large	1	49842	87.73	49793	88.09	49777	88.7	50261	87.47
	2	60670	90.58	59887	90.52	60221	90.47	61101	89.41
	3	**70091**	**92.28**	69878	89.54	68378	91.39	68584	91.35
	4	60046	90.93	57502	87.06	58919	87.96	59094	88.9
	5	51422	89.29	50795	86.95	51376	86.54	50790	88.3
	6	58909	91.86	57938	91.28	57152	93.53	58374	93.46
	7	56554	88.05	56261	89.01	56254	87.75	56506	87.41
	8	54920	88.13	55646	87.23	55316	88.83	55415	87.36
	9	52716	87.1	52548	88.45	51533	87.9	52403	87.78
	10	**57496**	**89.75**	55910	88.61	55465	89.45	56366	89.7
Time (min)		84.17		71.38		52.61		51.95	
Number of best		23	20	4	5	1	3	2	2

In Table 7, the proposed method outperformed other approaches in terms of both objective functions for 6 problems instances. Specifically, the number of superior solutions obtained by the proposed method was 23 for the first objective function (profit maximization) and 20 for the second objective function (minimizing rejected orders). These results demonstrate the method's effectiveness in solving complex production scheduling and machine allocation problems.

Fig. 3 highlights the impact of the proposed BMA on optimizing machine locations. By improving the layout of the machines, the total profit increased by approximately 9%, underscoring the importance of optimal machine placement. Additionally, the number of accepted orders rose by 2.5%.

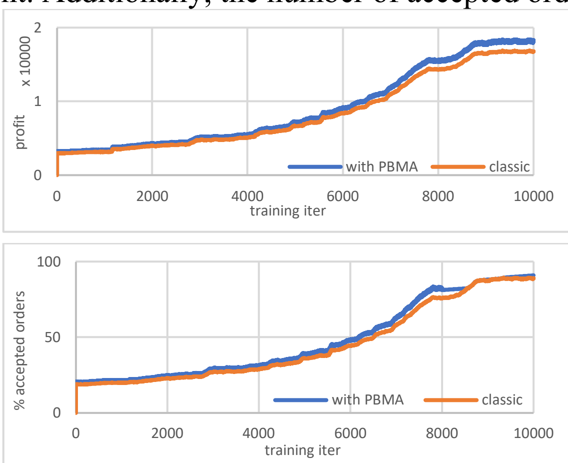

Fig. 3. The result of BMA in the production system.

Given the hierarchical nature of steps 2 and 3 in the proposed solution approach, the iterative process continues until an efficient solution is obtained. Table 8 presents the average number of iterations required for small, medium, and large problem instances, along with the improvement in the final solution relative to the first iteration.

Table 8. The details of the improvements achieved in different iterations (α = 0.02)

	Small	Medium	Large
Number of iterations	13	19	31
% Improvement	14.23	17.11	15.19

Based on the results presented in Table 8, the hierarchical approach improves the initial solutions by an average of 15.51%, demonstrating the effectiveness of the proposed solution method.

6. Discussion

This paper addresses a DCMS problem that simultaneously considers account order rejection, inventory management, and tardiness penalties, with the dual objectives of maximizing profit and minimizing order rejection.

6.1. The impact of the Second Objective: Focusing on the second objective encourages the system to accept more orders and attract a broader customer base. Although, this may temporarily reduce immediate profits due to increased tardiness costs, it functions as a long-term investment in the system's competitiveness. By attracting and retaining customers the production system strengthens its market position and gains strategic advantages. Fig. 4 demonstrates the significance of **W**, the weight of profit maximization, showing that setting **W** to 1 result in a 25% rejection rate, while setting it to 0 reduces rejections to 0%. Decision-makers can adjust W based on their organization's specific needs and conditions.

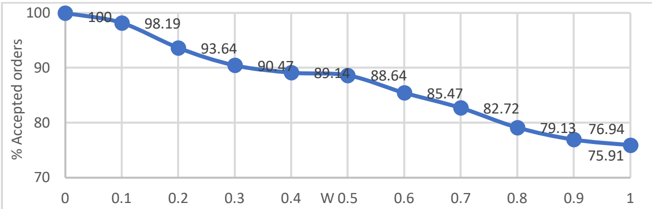

Fig. 4. The result of different w

6.2. The impact of Inventory: Purchasing raw materials at the optimal time and in the right quantity reduces costs and enhances the system's competitive advantage. Fig. 5 illustrates the sensitivity of the problem concerning the parameter E(I) for raw materials 1 and 2. When E(I) is low, fewer periods are evaluated, leading to more frequent purchases and lower inventory levels. This increase in purchase orders is due to a lack of long-term foresight, requiring adjustments in preparation for future periods. On the other hand, when E(I) is high, more periods are reviewed, and the uncertainty regarding order arrivals increases. As a result, determining the exact quantity of raw materials needed becomes a greater challenge, leading to an increase in the number of orders.

Fig. 5 indicates that the lowest amount of raw material purchases occurred at E(I) = 5, which also resulted in the lowest inventory costs (both buying and holding). In Fig. 5, B-e1 and B-e2 represent the number of purchases for raw materials types 1 and 2, while AH-e1 and AH-e2 represent the average inventory held throughout the entire period for raw materials types 1 and 2.

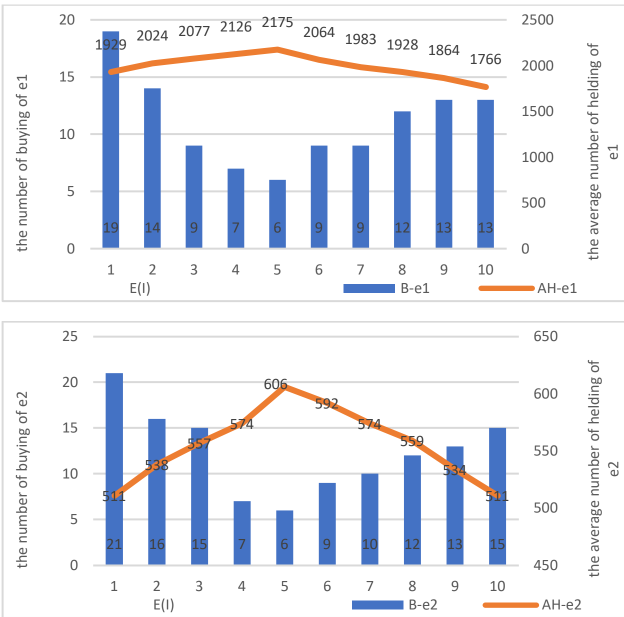

Fig. 5. The result of the different E(I)

Fig. 6 shows the cost of inventory in the case study. When E(I) = 5, the cost of inventory is at minimum.

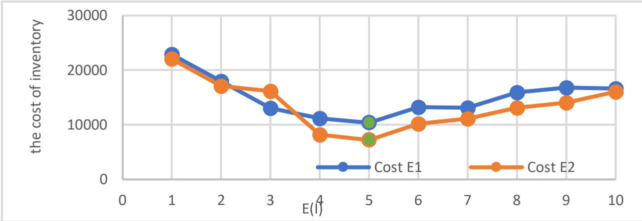

Fig. 6. The cost of different E(I)

In Fig. 6, Cost E1 represents the cost of raw material type 1, while Cost E2 represents the cost of raw material type 2. The cost of raw material type 1 decreased by approximately 68% in the best case and 55% in the worst case. This highlights that inventory management is a critical decision in production systems, and careful analysis and selection of inventory policies can lead to significant cost reductions. Fig. 7 compares the difference between the inventory required in the real case and the result of the proposed algorithm over a three-month period. The neural network proposed to improve inventory decisions helps to gradually reduce the gap between the actual case and the proposed method. The difference between the real case and the proposed algorithm for raw materials types 1 and 2 is 5.2% and 4.8%, respectively.

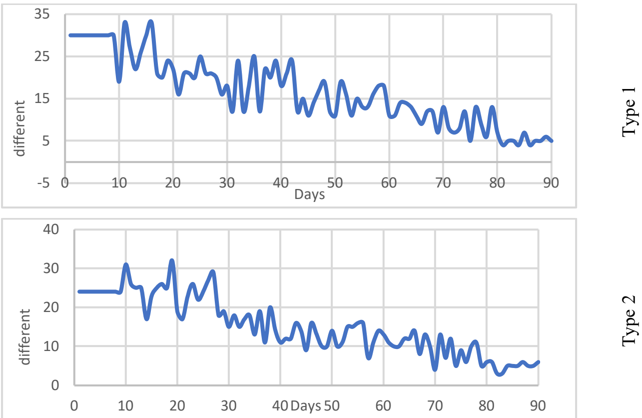

Fig. 7. The difference between the actual required raw material and the result of the proposed algorithm

6.3. Impact of DRL on Order Management

To achieve an optimal scheduling policy, it is essential to schedule the best set of accepted orders efficiently, assign them to the most suitable machines, and ensure raw materials are supplied at the right time and in the right quantity. The DRL algorithm evaluates multiple strategies to make these decisions, aiming to maximize overall system performance. This section focuses on the impact of each individual strategy used in the problem-solving process. It is worth noting that the overall impact of these strategies, in comparison with other solution methods, has been analyzed in Sections 5, 6.1, and 6.2. Below, only the effect of removing each individual strategy on the problem is examined.

Strategies Implemented for Scheduling Received Orders: There are three strategies for this decision. The integration of these three strategies provides the DRL agent with a structured environment, enhancing its ability to learn effective scheduling policies. Comparative analysis in Table 9 shows that removing any of the strategies results in suboptimal performance, validating their collective importance. In real-world applications, such strategies can be adapted to align with business goals like profitability, responsiveness, or operational stability. Table 9 shows the average of obj1 for all samples.

Table 9. Comparative analysis for strategies implemented for scheduling received orders

	Deleted strategy						
	Strategy 1	Strategy 2	Strategy 3	Strategy 1, 2	Strategy 2, 3	Strategy 1, 3	Strategy 1, 2, 3 (random scheduling)

Decreasing of total profit	5.1	4.9	5.4	15.7	18.1	17.8	31.2

Strategies Used for Order Acceptance and Rejection: There are three strategies for this decision. The combination of these strategies helps the DRL agent make more informed, balanced decisions about which orders to accept, thereby increasing the system's profitability and reliability. These strategies reflect practical decision-making processes commonly used by experienced production managers. Comparative analysis in table 10 shows that removing any of the strategies results in suboptimal performance, validating their collective importance.

Table 10. Comparative analysis for strategies used for order acceptance and rejection

	Deleted strategy						
	Strategy 1	Strategy 2	Strategy 3	Strategy 1, 2	Strategy 2, 3	Strategy 1, 3	Strategy 1, 2, 3 (random rejection)
Decreasing of total profit	6.1	5.08	7.09	18.09	17.78	16.32	34.6

Strategies Used for Assigning Accepted Orders to Machines: In this strategy, orders are assigned to the earliest available machine. Given the absence of machine failures or deterioration, this approach can lead to the best decision for improving system efficiency and ensuring on-time delivery to customers.

Strategies Used for Raw Material Procurement: There is one primary strategy for this decision. The raw material procurement strategy integrated into the DRL framework directly affects production continuity, cost efficiency, and responsiveness. Removing or misconfiguring it negatively impacts performance, highlighting the need for intelligent material planning in automated manufacturing systems. The impact of this strategy is analyzed in Section 6.2.

6.4. Theoretical implications

This research contributes to the body of knowledge in dynamic cellular manufacturing systems (DCMS), particularly in addressing the complexities of order rejection, tardiness costs, and inventory management within a dynamic environment. Several key theoretical implications arise from the findings of this study:

- Integration of Multi-Objective Optimization: The study introduces a multi-objective optimization framework that considers both profit maximization and minimizing order rejection. This dual objective framework expands the traditional scope of DCMS, where previous studies often focused solely on optimizing production efficiency or profit. By integrating order rejection and raw material procurement into the model, this research broadens the understanding of how trade-offs between competing objectives can be managed in complex manufacturing environments. Therefore, the current research develops upon the studies of (An et al. 2023b), (An et al. 2023a) and (D'Haen, Braekers, and Ramaekers 2023). The aim of the aforementioned studies has been to maximize profit or minimize costs, which are short-term objectives and do not ensure the long-term survival of production systems. Given that maintaining competitive advantage and system sustainability in the long run is a crucial strategy, this research seeks to achieve a competitive edge by incorporating a second objective—reducing order rejections and increasing customer satisfaction. Accepting a customer's order with low profit contributes to customer loyalty to the organization.
- Application of Deep Reinforcement Learning (DRL) in DCMS: The use of deep reinforcement learning (DRL) to handle real-time decision-making for order acceptance, scheduling, and inventory management marks a significant theoretical contribution. While DRL has been applied in areas like scheduling and inventory management individually, its application in a

multi-faceted manufacturing system like DCMS demonstrates its potential to handle dynamic, large-scale, real-time problems. This work adds to the growing literature on the use of DRL in industrial applications and presents an innovative use case where DRL is combined with other heuristic algorithms for system optimization. Therefore, the current research develops upon the studies of (Oroojlooyjadid et al. 2022), (Zhou et al. 2024), (Dehaybe, Catanzaro, and Chevalier 2024), (Stranieri, Fadda, and Stella 2024), (Leng et al. 2021), (Leng et al. 2023), (Luo 2020) and (Lei et al. 2023). This development is done by integrating DRL with heuristic algorithms such as genetic algorithms and the boxing match algorithm. Unlike prior works that focus on isolated aspects of scheduling or inventory control, this study demonstrates how these techniques can be synergistically applied to optimize multiple facets of a production system simultaneously. This holistic approach enhances both short-term operational efficiency and long-term system adaptability, making it a valuable contribution to the field.

- Hierarchical Solution Approach: The study presents a hierarchical algorithmic approach integrating a genetic algorithm, DRL, and a boxing match algorithm. This multi-layered approach contributes to the theoretical development of hybrid solution methodologies for complex, high-dimensional optimization problems. The combination of these algorithms addresses the limitations of using a single method, providing a more comprehensive approach to solve multi-stage problems like machine assignment, order scheduling, and raw material procurement. This approach can inspire future research into hybrid algorithms for solving similarly complex manufacturing problems. This research builds upon the work of (Lei et al. 2023) and (Liu, Piplani, and Toro 2022) by developing a reinforcement learning approach. This building is done by integrating reinforcement learning with heuristic optimization techniques, this study extends the methodologies presented in prior works to address multi-stage decision-making in dynamic manufacturing environments. Unlike previous studies that primarily focused on static/dynamic single-stage optimization, this research demonstrates how reinforcement learning can dynamically adapt to real-time changes in order arrivals, machine availability, and inventory levels. This advancement paves the way for future studies to explore more adaptive and intelligent decision-support systems in complex industrial settings.
- Artificial Neural Networks in Inventory Management**:** The integration of artificial neural networks (ANN) to determine safety stock levels introduces a new perspective on inventory management in manufacturing systems. While ANN has been used in forecasting and demand planning, its application in determining precautionary raw materials within a dynamic system adds to the theoretical understanding of how machine learning techniques can enhance inventory control in real-time. This novel application underscores the potential of ANN for reducing the complexity and computational burden of real-time decision-making in DCMS. The proposed problem enhances the research of (Grigoriev, Holthuijsen, and Van De Klundert 2005), (Terekhov et al. 2012), (Cuartas and Aguilar 2023) and (Zhou et al. 2024) by offering an effective solution method for inventory management. This enhancement is done by leveraging ANN for safety stock determination, this study extends prior research by incorporating real-time data processing and adaptive learning to optimize inventory levels dynamically. Unlike traditional inventory models that rely on static safety stock calculations, the proposed approach enables manufacturers to adjust stock levels in response to fluctuating order arrivals and processing times. This advancement enhances decision-making efficiency and reduces unnecessary inventory costs, making it a valuable contribution to intelligent inventory management in dynamic production environments.

- Long-Term and Short-Term Decision-Making: The study's focus on the trade-off between long-term and short-term objectives, particularly through the γ parameter in the optimization model, adds to the theoretical debate on balancing profit maximization with customer satisfaction and system longevity. By demonstrating that a balance between the two can improve competitive advantage, the research contributes to multi-period optimization theory and provides insights into strategic decision-making in dynamic manufacturing contexts.
- Expanded Scope of DCMS Research**:** Finally, this research expands the theoretical scope of DCMS by incorporating the often-overlooked aspects of raw material procurement and order rejection in a dynamic environment. These factors are critical to the success of real-world manufacturing systems but have received limited attention in existing DCMS research. The study highlights the importance of integrating supply chain considerations into manufacturing models, pushing the boundaries of traditional DCMS theory. Therefore, the current research develops upon the studies of (An et al. 2023b), (An et al. 2023a), (D'Haen, Braekers, and Ramaekers 2023), (Rahman, Janardhanan, and Nielsen 2019), (Sarvestani et al. 2019), (Ju and Woo 2023) and (Negri 2023). This development is done by addressing raw material procurement and order rejection within a dynamic framework. This research builds upon previous studies by offering a more comprehensive representation of real-world manufacturing constraints. Unlike prior models that primarily focus on machine allocation and scheduling, the proposed approach integrates supply chain factors, enhancing the adaptability of DCMS in uncertain environments. This contribution ensures that manufacturing systems can optimize both production efficiency and supply chain resilience, leading to improved long-term sustainability and competitiveness.

In summary, the study's theoretical contributions lie in its multi-objective optimization framework, the innovative application of DRL and hybrid algorithms, the use of ANN for inventory management, and its emphasis on balancing short-term and long-term objectives. These contributions deepen the theoretical understanding of dynamic manufacturing systems and open new avenues for research in this field.

6.5. Practical implications

The proposed model offers significant practical benefits for manufacturing managers, production planners, and decision-makers operating in dynamic cellular manufacturing systems (DCMS). Industries with fluctuating demand, strict order deadlines, and complex inventory constraints—such as paper production, automotive manufacturing, electronics assembly, and customized manufacturing—would benefit the most. By optimizing order acceptance, machine assignments, and raw material procurement, the model improves operational efficiency, reduces costs, and enhances overall system responsiveness. Unlike existing models in the literature, which often focus on isolated aspects such as scheduling or inventory control, our approach integrates multiple decision-making layers into a single optimization framework. This includes a deep reinforcement learning (DRL)-based scheduling mechanism and the boxing match algorithm (BMA) for machine layout optimization. This holistic approach ensures real-time adaptability to dynamic production conditions, making it more robust and practical than conventional heuristic or static optimization methods.

Our results indicate that managers should strategically balance short-term profitability and long-term competitiveness by fine-tuning order acceptance policies and optimizing machine placement. The study findings demonstrate that:

- Optimized machine placement increases total profit by approximately 9%.
- Smart raw material procurement reduces inventory costs by 15.4% and lowers raw material expenses by up to 68% in the best case.

- Adjusting the weight of profit maximization can reduce order rejection rates by up to 25%, making the production system more competitive.

From an implementation perspective, the model can be integrated into existing enterprise resource planning (ERP) systems with moderate adjustments. However, challenges such as data availability, workforce training, and potential resistance to AI-driven decision-making must be addressed. With proper data collection and phased implementation, these obstacles can be mitigated, making this approach a valuable, scalable, and practical solution for modern manufacturing operations.

In total, the proposed model and solution methods improve operational efficiency, decision-making, and competitiveness in manufacturing environments through several key innovations:

- Enhanced Order Acceptance with Deep Reinforcement Learning (DRL): Real-time, data-driven order acceptance prioritizes high-profit orders, reduces tardiness costs, and improves responsiveness to customer demands.
- Optimized Scheduling and Machine Utilization: Genetic algorithms and the boxing match algorithm improve machine assignments, reduce idle times, and enhance production efficiency, particularly in resource-constrained environments.
- Improved Inventory Management with Artificial Neural Networks (ANN): Predictive safety stock management minimizes stockouts and excessive inventory costs while ensuring smooth production.
- Balancing Short-Term and Long-Term Goals: The γ parameter allows managers to adjust strategies, either maximizing immediate profits or attracting more customers for long-term gains.
- Hybrid Algorithm Effectiveness: The integration of genetic algorithms, DRL, and the boxing match algorithm efficiently addresses complex, real-time decision-making challenges in dynamic production environments.
- Competitive Advantage: The model helps manufacturers accept more orders, optimize resources, and adapt to market changes, leading to greater profitability and stronger market positioning.

7. Conclusion

This paper presents a dynamic cellular manufacturing system (DCMS) that incorporates order rejection, tardiness costs, and the costs associated with purchasing and holding raw materials. Orders arrive at different times, and decisions regarding acceptance or rejection are made based on multiple factors. Each received order includes details such as revenue, processing time, due date, and tardiness cost. Accepted orders require raw materials, which must be purchased at appropriate times, as purchasing is not allowed arbitrarily.

In the inventory management aspect, the quantity of raw materials purchased is determined using a heuristic method, while the amount of precautionary raw materials is based on an artificial neural network (ANN). A mathematical model with two objectives—profit maximization and minimization of rejected orders—has been proposed. Given the NP-hard nature of the problem, a hierarchical heuristic algorithm has been developed to handle larger problem instances.

Initially, machines are assigned to cells using a genetic algorithm. Subsequently, a deep reinforcement learning (DRL) algorithm is applied to accept orders, schedule operations, assign them to machines, and determine the timing and quantity of raw material purchases. Additionally, a boxing match algorithm (BMA) is introduced to enhance the location of machines based on the output of the deep reinforcement learning model.

A case study based on a stone paper factory demonstrates the performance of the proposed method. Results show that the BMA effectively improves machine locations, resulting in approximately 9% higher total profit and a 2.5% increase in accepted orders.
The impact of tardiness costs is also significant, causing a 25% reduction in accepted orders. Comparing real-case data with the algorithm's results reveals that the number of raw material purchases increased on average by 1.5 times, while the average inventory holding decreased by 15.4%. Moreover, the proposed ANN for raw material procurement proved effective, with discrepancies between real and predicted raw material quantities of 5.2% and 4.8% for types 1 and 2, respectively.

Limitations and Future Research

This study, while offering valuable insights into the optimization of dynamic cellular manufacturing systems using deep reinforcement learning, is subject to several important limitations that influence its practical applicability.
One key limitation lies in the assumption of full machine availability, which does not reflect the operational realities of most manufacturing environments. In practice, machines are subject to wear and tear, deterioration, and unexpected breakdowns, all of which can disrupt production schedules and reduce overall system efficiency. Another limitation involves the risks associated with raw material procurement. In real-world supply chains, the timely arrival of raw materials is not guaranteed due to various uncertainties such as transportation delays, supplier issues, or geopolitical factors. Failure to receive materials on time can lead to suboptimal scheduling and production delays. Another limitation concerns the relocation of machines. In most industrial settings, relocating machines is costly and time-consuming, and managers are typically reluctant to undertake such changes unless absolutely necessary. This inflexibility may hinder the discovery of globally optimal solutions.
To overcome these limitations and improve the model's practical relevance, future research could explore the following directions:

- Incorporating machine deterioration and failure dynamics into the scheduling framework to better reflect real-world production challenges.
- Modelling raw material procurement risks, including stochastic lead times and delivery uncertainties, to develop more resilient and adaptive scheduling strategies.
- Differentiating between movable and fixed machines within the system, allowing for a more nuanced approach to cellular layout design and enabling a realistic evaluation of layout flexibility's impact on performance.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

References

An, Youjun, Xiaohui Chen, Kaizhou Gao, Lin Zhang, Yinghe Li, and Ziye Zhao. 2023a. 'A hybrid multi-objective evolutionary algorithm for solving an adaptive flexible job-shop rescheduling problem with real-time order acceptance and condition-based preventive maintenance', *Expert Systems with Applications*, 212: 118711. https://doi.org/10.1016/j.eswa.2022.118711

An, Y., Chen, X., Gao, K., Zhang, L., Li, Y., & Zhao, Z. 2023b. 'Integrated optimization of real-time order acceptance and flexible job-shop rescheduling with multi-level imperfect maintenance constraints', *Swarm and Evolutionary Computation*, 77: 101243. https://doi.org/10.1016/j.swevo.2023.101243

Beck, J Christopher. 2002. 'Heuristics for scheduling with inventory: dynamic focus via constraint criticality', *Journal of Scheduling*, 5: 43-69. https://doi.org/10.1002/jos.91

Cuartas, Carlos, and Jose Aguilar. 2023. 'Hybrid algorithm based on reinforcement learning for smart inventory management', *Journal of Intelligent Manufacturing*, 34: 123-49. https://doi.org/10.1007/s10845-022-01982-5

D'Haen, Ruben, Kris Braekers, and Katrien Ramaekers. 2023. 'Integrated scheduling of order picking operations under dynamic order arrivals', *International Journal of Production Research*, 61: 3205-26. https://doi.org/10.1080/00207543.2022.2078747

Danilovic, Milos, and Oliver Ilic. 2019. 'A novel hybrid algorithm for manufacturing cell formation problem', *Expert Systems with Applications*, 135: 327-50. https://doi.org/10.1016/j.eswa.2019.06.019

Deep, Kamal. 2020. 'Machine cell formation for dynamic part population considering part operation trade-off and worker assignment using simulated annealing-based genetic algorithm', *European Journal of Industrial Engineering*, 14: 189-216. https://doi.org/10.1504/EJIE.2020.105720

Dehaybe, Henri, Daniele Catanzaro, and Philippe Chevalier. 2024. 'Deep Reinforcement Learning for inventory optimization with non-stationary uncertain demand', *European Journal of Operational Research*, 314: 433-45. https://doi.org/10.1016/j.ejor.2023.10.007

Du, Yu, and Jun-qing Li. 2024. 'A deep reinforcement learning based algorithm for a distributed precast concrete production scheduling', *International Journal of Production Economics*, 268: 109102. https://doi.org/10.1016/j.ijpe.2023.109102

Elia, Gianluca, Alessandro Margherita, and Giuseppina Passiante. 2020. 'Management engineering: A new perspective on the integration of engineering and management knowledge', *IEEE Transactions on Engineering Management*, 68: 881-93. https://doi.org/10.1109/TEM.2020.2992911

Fu, Yaping, Haobin Li, Min Huang, and Hui Xiao. 2021. 'Bi-objective modeling and optimization for stochastic two-stage open shop scheduling problems in the sharing economy', *IEEE Transactions on Engineering Management*. https://doi.org/10.1109/TEM.2021.3095954

Gan, Xuemei, Ying Zuo, Guanci Yang, Ansi Zhang, and Fei Tao. 2024. 'Dynamic scheduling for dual-objective job shop with machine breakdown by reinforcement learning', *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, 238: 3-17. https://doi.org/10.1177/09544054231167086

Grigoriev, Alexander, Martijn Holthuijsen, and Joris Van De Klundert. 2005. 'Basic scheduling problems with raw material constraints', *Naval Research Logistics (NRL)*, 52: 527-35. https://doi.org/10.1002/nav.20095

Hsieh, Feng Yu, Yang Kuei Lin, and Yi-Chi Wang. 2024. 'Developing algorithm and dispatching rules for scheduling a realistic flexible flow shop with setups to minimize total weighted tardiness', *The International Journal of Advanced Manufacturing Technology*: 1-21. https://doi.org/10.1007/s00170-024-14509-4

Ju, SuHeon, and Jong Hun Woo. 2023. 'Integration of long-term planning and mid-term scheduling of shipbuilding', *Production Planning & Control*, 34: 524-42. https://doi.org/10.1080/09537287.2021.1940344

Kaynov, Illya, Marijn van Knippenberg, Vlado Menkovski, Albert van Breemen, and Willem van Jaarsveld. 2024. 'Deep reinforcement learning for one-warehouse multi-retailer inventory management', *International Journal of Production Economics*, 267: 109088. https://doi.org/10.1016/j.ijpe.2023.109088

Lamba, Kuldeep, Ravi Kumar, Shraddha Mishra, and Shubhangini Rajput. 2020. 'Sustainable dynamic cellular facility layout: A solution approach using simulated annealing-based meta-heuristic', *Annals of Operations Research*, 290: 5-26. https://doi.org/10.1007/s10479-019-03340-w

Lei, Kun, Peng Guo, Yi Wang, Jian Zhang, Xiangyin Meng, and Linmao Qian. 2023. 'Large-Scale Dynamic Scheduling for Flexible Job-Shop With Random Arrivals of New Jobs by Hierarchical Reinforcement Learning', *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2023.3272661

Leng, Jiewu, Jiwei Guo, Hu Zhang, Kailin Xu, Yan Qiao, Pai Zheng, and Weiming Shen. 2023. 'Dual deep reinforcement learning agents-based integrated order acceptance and scheduling of mass individualized prototyping', *Journal of Cleaner Production*, 427: 139249. https://doi.org/10.1016/j.jclepro.2023.139249

Leng, Jiewu, Guolei Ruan, Yuan Song, Qiang Liu, Yingbin Fu, Kai Ding, and Xin Chen. 2021. 'A loosely-coupled deep reinforcement learning approach for order acceptance decision of mass-individualized printed circuit board manufacturing in industry 4.0', *Journal of cleaner production*, 280: 124405. https://doi.org/10.1016/j.jclepro.2020.124405

Li, Yantong, Jean-François Côté, Leandro C Coelho, Chuang Zhang, and Shuai Zhang. 2023. 'Order assignment and scheduling under processing and distribution time uncertainty', *European Journal of Operational Research*, 305: 148-63. https://doi.org/10.1016/j.ejor.2022.05.033

Li, Yantong, Xin Wen, Tsan-Ming Choi, and Sai-Ho Chung. 2022. 'Optimal establishments of massive testing programs to combat COVID-19: A perspective of parallel-machine scheduling-location (ScheLoc) problem', *IEEE Transactions on Engineering Management*. https://doi.org/10.1109/TEM.2022.3199039

Liu, Chien-Liang, Chun-Jan Tseng, and Po-Hao Weng. 2024. 'Dynamic Job-Shop Scheduling via Graph Attention Networks and Deep Reinforcement Learning', *IEEE Transactions on Industrial Informatics*. https://doi.org/10.1109/TII.2024.3371489

Liu, Qihao, Xinyu Li, Liang Gao, and Jiaxin Fan. 2022. 'A multi-MILP model collaborative optimization method for integrated process planning and scheduling problem', *IEEE Transactions on Engineering Management*, 71: 4574-86. https://doi.org/10.1109/TEM.2022.3208431

Liu, Renke, Rajesh Piplani, and Carlos Toro. 2022. 'Deep reinforcement learning for dynamic scheduling of a flexible job shop', *International Journal of Production Research*, 60: 4049-69. https://doi.org/10.1080/00207543.2022.2058432

Luo, Shu. 2020. 'Dynamic scheduling for flexible job shop with new job insertions by deep reinforcement learning', *Applied Soft Computing*, 91: 106208. https://doi.org/10.1016/j.asoc.2020.106208

Maalouf, Elie, Joanna Daaboul, Julien Le Duigou, and Bassam Hussein. 2022. 'Production management for mass customization and smart cellular manufacturing system: NSGAII and SMPSO for factory-level planning', *The International Journal of Advanced Manufacturing Technology*, 120: 6833-54. https://doi.org/10.1007/s00170-022-09188-y

Mao, Rong-Rong, Dan-Yang Lv, Na Ren, and Ji-Bo Wang. 2024. 'Supply chain scheduling with deteriorating jobs and delivery times', *Journal of Applied Mathematics and Computing*, 70: 2285-312. https://doi.org/10.1007/s12190-024-02052-0

Motahari, Rasool, Zeinolabedin Alavifar, Abdullah Zareh Andaryan, Maxwell Chipulu, and Morteza Saberi. 2023. 'A multi-objective linear programming model for scheduling part families and designing a group layout in cellular manufacturing systems', *Computers & Operations Research*, 151: 106090. https://doi.org/10.1016/j.cor.2022.106090

Napoleone, Alessia, Emilio Moretti, Marco Macchi, and Marco Melacini. 2023. 'Synchronisation of material flows in mass-customised production systems: a literature-based classification framework and industrial application', *Production Planning & Control*: 1-19. https://doi.org/10.1080/09537287.2023.2214529

Negri, Elisa. 2023. "Effective methods for integrated process planning and scheduling: by Xinyu Li and Liang Gao, Beijing, Springer-Science Press, 2020, 462 pp., 155, 99$(Hardcover), ISBN: 978-3-662-55303-9." In.: Taylor & Francis. https://doi.org/10.1080/09537287.2021.1918889

Oroojlooyjadid, Afshin, MohammadReza Nazari, Lawrence V Snyder, and Martin Takáč. 2022. 'A deep q-network for the beer game: Deep reinforcement learning for inventory optimization', *Manufacturing & Service Operations Management*, 24: 285-304. https://doi.org/10.1287/msom.2020.0939

Perea, Federico, Juan C Yepes-Borrero, and Mozart BC Menezes. 2023. 'Acceptance Ordering Scheduling Problem: The impact of an order-portfolio on a make-to-order firm's profitability', *International Journal of Production Economics*, 264: 108977. https://doi.org/10.1016/j.ijpe.2023.108977

Qian, Jin, Guisong Chang, and Xue Zhang. 2024. 'Single-machine common due-window assignment and scheduling with position-dependent weights, delivery time, learning effect and resource allocations', *Journal of Applied Mathematics and Computing*, 70: 1965-94. https://doi.org/10.1007/s12190-024-02023-5

Rabiu, Sani, and Majid Khan Majahar Ali. 2024. 'Optimizing inventory dynamics: a smart approach for non-instantaneous deteriorating items with linear time function dependent variable demands and holding costs, shortages with backlogging', *Journal of Applied Mathematics and Computing*: 1-25. https://doi.org/10.1007/s12190-024-02089-1

Rahman, Humyun Fuad, Mukund Nilakantan Janardhanan, and Izabela Ewa Nielsen. 2019. 'Real-time order acceptance and scheduling problems in a flow shop environment using hybrid GA-PSO algorithm', *IEEE Access*, 7: 112742-55. https://doi.org/10.1109/ACCESS.2019.2935375

Salimpour, Saeideh, Hani Pourvaziri, and Ahmed Azab. 2021. 'Semi-robust layout design for cellular manufacturing in a dynamic environment', *Computers & Operations Research*, 133: 105367. https://doi.org/10.1016/j.cor.2021.105367

Sarvestani, Hossein Khorrami, Alborz Zadeh, Majid Seyfi, and Morteza Rasti-Barzoki. 2019. 'Integrated order acceptance and supply chain scheduling problem with supplier selection and due date assignment', *Applied Soft Computing*, 75: 72-83. https://doi.org/10.1016/j.asoc.2018.10.045

Shahrabi, Jamal, Mohammad Amin Adibi, and Masoud Mahootchi. 2017. 'A reinforcement learning approach to parameter estimation in dynamic job shop scheduling', *Computers & industrial engineering*, 110: 75-82. https://doi.org/10.1016/j.cie.2017.05.026

Shiue, Yeou-Ren, Ken-Chuan Lee, and Chao-Ton Su. 2018. 'Real-time scheduling for a smart factory using a reinforcement learning approach', *Computers & industrial engineering*, 125: 604-14. https://doi.org/10.1016/j.cie.2018.03.039

Silva, Thuener, Davi Valladão, and Tito Homem-de-Mello. 2021. 'A data-driven approach for a class of stochastic dynamic optimization problems', *Computational Optimization and Applications*, 80: 687-729. https://doi.org/10.1007/s10589-021-00320-4

Singh, Ranu, and Vinod Kumar Mishra. 2024. 'Machine learning based fuzzy inventory model for imperfect deteriorating products with demand forecast and partial backlogging under green investment technology', *Journal of the Operational Research Society*, 75: 1223-38. https://doi.org/10.1080/01605682.2023.2239868

Slotnick, Susan A, and Thomas E Morton. 2007. 'Order acceptance with weighted tardiness', *Computers & Operations Research*, 34: 3029-42. https://doi.org/10.1016/j.cor.2005.11.012

Stranieri, Francesco, Edoardo Fadda, and Fabio Stella. 2024. 'Combining deep reinforcement learning and multi-stage stochastic programming to address the supply chain inventory management problem', *International Journal of Production Economics*, 268: 109099. https://doi.org/10.1016/j.ijpe.2023.109099

Tanhaeean, M, R Tavakkoli-Moghaddam, and AH Akbari. 2022. 'Boxing match algorithm: A new meta-heuristic algorithm', *Soft Computing*, 26: 13277-99. https://doi.org/10.1007/s00500-022-07518-6

Terekhov, Daria, Mustafa K Doğru, Ulaş Özen, and J Christopher Beck. 2012. 'Solving two-machine assembly scheduling problems with inventory constraints', *Computers & Industrial Engineering*, 63: 120-34. https://doi.org/10.1016/j.cie.2012.02.006

Wang, Lili, Zhe Zhang, and Yong Yin. 2022. 'Order acceptance and scheduling problem with outsourcing in seru production system considering lot-spitting', *European Journal of Industrial Engineering*, 16: 91-116. https://doi.org/10.1504/EJIE.2022.119371

Wang, Ye, and Jiafu Tang. 2020. 'Optimized skill configuration for the seru production system under an uncertain demand', *Annals of Operations Research*: 1-21. https://doi.org/10.1007/s10479-020-03805-3

Wu, Lang, Yaping Zhao, Yuanyue Feng, Ben Niu, and Xiaoyun Xu. 2021. 'Minimizing makespan of stochastic customer orders in cellular manufacturing systems with parallel machines', *Computers & Operations Research*, 125: 105101. https://doi.org/10.1016/j.cor.2020.105101

Wu, Xinquan, Xuefeng Yan, Donghai Guan, and Mingqiang Wei. 2024. 'A deep reinforcement learning model for dynamic job-shop scheduling problem with uncertain processing time', *Engineering Applications of Artificial Intelligence*, 131: 107790. https://doi.org/10.1016/j.engappai.2023.107790

Yan, Yimo, Andy HF Chow, Chin Pang Ho, Yong-Hong Kuo, Qihao Wu, and Chengshuo Ying. 2022. 'Reinforcement learning for logistics and supply chain management: Methodologies, state of the art, and future opportunities', *Transportation Research Part E: Logistics and Transportation Review*, 162: 102712. https://doi.org/10.1016/j.tre.2022.102712

Zhang, Biao, Chao Lu, Lei-lei Meng, Yu-yan Han, Hong-yan Sang, and Xu-chu Jiang. 2023. 'Reconfigurable distributed flowshop group scheduling with a nested variable neighborhood descent algorithm', *Expert Systems with Applications*, 217: 119548. https://doi.org/10.1016/j.eswa.2023.119548

Zhou, Yifan, Kai Guo, Cheng Yu, and Zhisheng Zhang. 2024. 'Optimization of multi-echelon spare parts inventory systems using multi-agent deep reinforcement learning', *Applied Mathematical Modelling*, 125: 827-44. https://doi.org/10.1016/j.apm.2023.10.039