A Decision Optimization Framework for Multi-Stage Manufacturing Processes: Integrating 0-1 Integer Programming and Genetic Algorithms

Yiran Ding *,#, Chenyu Zhang #, Kemeng Wang #

School of Artificial Intelligence, Hebei University of Technology, Tianjin, China, 300401

* Corresponding Author Email: IridiumL77@163.com

#These authors contributed equally.

Abstract. Optimizing multi-stage manufacturing is crucial for improving profitability, operational efficiency, and competitiveness in the market. The complexity of multi-stage manufacturing systems has been extensively studied, with numerous approaches exploring various optimization techniques. This study proposes a decision optimization framework that addresses critical challenges in inspection, disassembly, and resource allocation. The framework integrates a 0-1 integer programming model, which captures binary decisions and key operational parameters, with a genetic algorithm (GA) designed to deliver efficient solutions for large-scale manufacturing instances. This approach builds on established optimization methods and addresses contemporary challenges in production, including resource constraints and quality control. The 0-1 integer programming model formalizes the problem, while the GA aims to maximize expected profit by optimizing resource usage and process efficiency. Case studies demonstrate significant improvements, with profit increases of 10%-25% achieved through optimized strategies, such as selective inspection and strategic disassembly. This research provides practical guidance for manufacturers, balancing quality control, cost management, and resource optimization, and bridges the gap between theoretical modeling and real-world manufacturing applications, offering insights for smarter decision-making in the manufacturing sector.

Keywords: Multi-stage Manufacturing, Decision Optimization, 0-1 Integer Programming, Genetic Algorithm.

1. Introduction

Multi-stage manufacturing processes, involving sequential operations to assemble products, are vital to modern production systems. Optimizing these processes is essential for enhancing profitability, operational efficiency, and maintaining competitiveness [1,2]. However, the complexity of these systems presents significant challenges, particularly in balancing quality control, cost management, and meeting production deadlines [3,4]. Inefficiencies, such as suboptimal inspection or disassembly decisions, can lead to increased operational costs, waste, and diminished customer satisfaction.

To address these challenges, previous studies have employed various optimization techniques. Genetic algorithms (GAs) have been extensively used for solving complex scheduling and resource allocation problems in manufacturing contexts [5-10]. These algorithms are effective for large-scale optimization due to their ability to explore vast solution spaces efficiently. Concurrently, 0-1 integer programming models have been developed to handle discrete decisions, such as whether to inspect components or disassemble products [11,12]. These models provide a structured approach to optimizing binary choices in manufacturing.

Despite these advancements, existing methods face limitations when applied to large-scale, multi-stage systems. Traditional integer programming approaches encounter scalability issues as the number of decision variables grows, resulting in computationally intensive solutions [13]. Meanwhile, genetic algorithms, though efficient, may not adequately integrate quality-cost trade-offs, often prioritizing scheduling over quality considerations [14]. Furthermore, the integration of Industry 4.0 technologies, such as digital twins and big data analytics, has introduced new complexities that demand more sophisticated optimization frameworks [1,2,3,15].

To bridge these gaps, this study proposes a hybrid decision optimization framework that combines 0-1 integer programming with genetic algorithms. The integer programming model captures the binary nature of manufacturing decisions, while the genetic algorithm efficiently searches for near-optimal solutions in large-scale instances. This hybrid approach is inspired by prior research that successfully integrated these methodologies [11]. By leveraging the strengths of both techniques, the framework aims to maximize expected profit while ensuring a balance between quality control and operational efficiency.

The primary contributions of this research are twofold. First, it formulates a comprehensive optimization model addressing key decisions in multi-stage manufacturing, including inspection, disassembly, and resource allocation. Second, it demonstrates the framework's practical applicability through case studies, showing profit improvements of 10%-25% compared to traditional methods.

2. Mathematical Formulation:A 0-1 Integer Programming Model for Manufacturing Decisions

2.1. Data Sources

The data used in this study, including defect rates, purchase prices, inspection costs, assembly costs, market selling prices, and other operational parameters, are primarily derived from the problem description and datasets provided by the 2024 Higher Education Press Cup National Mathematical Modeling Competition B-Topic.

https://www.mcm.edu.cn/html_cn/node/a0c1fb5c31d43551f08cd8ad16870444.html

These datasets include specific parameter values for six different production scenarios (as shown in Table 1) and the assembly situation of two processes and eight components (as shown in Table 5).

All data used in this study are anonymized and do not involve proprietary information from specific companies. Future research could incorporate real-world production data from manufacturing enterprises to further validate the proposed framework.

2.2. The Structure of the 0-1 Programming Model

The 0-1 programming model is a specific type of integer programming used to solve optimization problems with binary choices (0 or 1). Each decision variable in this model represents "yes" or "no," or "select" or "not select." It is commonly applied in resource allocation, task scheduling, and production planning.

The 0-1 programming model includes four decisions: Parts 1 and 2 Detection Decision, Product Detection Decision, and Product Disassembly Decision.

Each 0-1 variable corresponds to a decision step.

$$x_i = \begin{cases} 0, not\ implementing\ the\ decision \\ 1, implementing\ the\ decision \end{cases} ,i=1,2,3,4 \tag{1}$$

The establishment of the objective function begins by introducing the assembly rate of parts. This is defined as the assembly rate of part *i-th*, which can be assembled into a finished product after going through decision *i-th* (1 or 2) and before decision 3. At the same time, this model introduces the assembly rate of the finished product, which is composed of two components. The non-conformance of either component will result in the finished product being non-conforming, the minimum conformance rate between the two components is adopted.

$$\gamma_i=1-x_i \cdot P_i\ , i=1,2 \tag{2}$$

where *i-th* takes the value of 1 or 2, representing the decision variable for Decision 1 or Decision 2, and P_i is the defect rate of part *i-th*.

$$\gamma_{min}=\min\{\gamma_i\}\ ,i=1,2 \tag{3}$$

where the finished product retention rate is introduced, indicating the retention rate of the finished product after Decision 3.

$$N=x_3 \cdot \gamma_{min} \cdot (1-p_3)+(1-x_3) \cdot \gamma_{min} \tag{4}$$

where p_3 is the defect rate of the finished product. If the company chooses to conduct finished product inspection in Decision 3, the final retention rate of the finished product is the product of the assembly rate and the non-defect rate; if the company chooses not to conduct finished product inspection in Decision 3, the final retention rate of the finished product is equal to the assembly rate, with no discarding or disassembly performed.

$$\max\ S=R+E_{reclaim}-E_{cost}-E_{basic\_detect}-E_{assemble}-E_{detect}-E_{exchange}-E_{dismantle} \tag{5}$$

where S is the objective function, representing the maximum expected profit per finished product. It denotes the expected revenue per finished product, referring to the total revenue generated by the company from selling qualified finished products.

$$R=\gamma_{min} \cdot (1-p_3) \cdot R_{price} \tag{6}$$

The variable γ_{min} represents the assembly rate of finished products, while R_{price} denotes the unit market price of the finished product. The final revenue should be calculated based on the rate of qualified finished products assembled.

$$E_{reclaim}=x_4 \cdot \sum_{i=1}^{2} C_i \cdot \left[\gamma_{min} \cdot P_3+\left(\gamma_i-\gamma_{min}\right)\right] \tag{7}$$

where $E_{reclaim}$ denotes the expected revenue from the recovery of a single defective product, x_4 is the decision variable for Decision 4, C_i represents the purchase price per component *i-th*, γ_i is the assembly rate of component *i-th*, $\gamma_{min} \cdot P_3$ indicates the ratio of defective finished products, and $\gamma_i-\gamma_{min}$ represents the ratio of excess unassembled components after assembling component *i-th*.

$$E_{cost}=\sum_{i=1}^{2} C_i \tag{8}$$

where E_{cost} denotes the expected total purchasing cost of components, referring to the fixed costs incurred by the company for purchasing components 1 and 2.

$$E_{basic\_detect}=\sum_{i=1}^{2} x_i \cdot C_{basic\_detect\ i} \tag{9}$$

where E_{basic_detect} denotes the expected total inspection cost of components, referring to the fixed costs incurred by the company if it chooses to inspect the components in Decision 1 and Decision 2. Where $C_{basic\_detect\ i}$ denotes the inspection cost of component *i*.

$$E_{assemble}=C_{assemble} \cdot \gamma_{min} \tag{10}$$

where $E_{assemble}$ denotes the expected cost of finished product assembly, and $C_{assemble}$ represents the assembly cost of the finished product.

$$E_{detect}=x_3 \cdot \gamma_{min} \cdot C_{detect} \tag{11}$$

where E_{detect} denotes the expected cost of finished product inspection, and x_3 represents the decision variable for Decision 3.

$$E_{exchange}=(1-x_3) \cdot \gamma_{min} \cdot P_3 \cdot C_{exchange} \tag{12}$$

where $E_{exchange}$ denotes the expected cost of replacing defective finished products, and $C_{exchange}$ represents the loss incurred from replacing the defective finished products.

$$E_{dismantle}=x_4 \cdot \gamma_{min} \cdot P_3 \cdot C_{dismantle} \tag{13}$$

where $E_{dismantle}$ denotes the expected cost of disassembling defective finished products, and $C_{dismantle}$ represents the disassembly cost of the defective finished products.

$$S \geq 0 \tag{14}$$

where S represents the profit, ensuring that the model guarantees a positive profit for the enterprise.

2.3. Optimal Decision-Making Model for Enterprises Based on Genetic Algorithm

This model is primarily designed to solve the problem of how an enterprise can make optimal decisions during the production process, especially in a multi-stage production process. The model diagram is shown in Figure 1.

This model decompose the production process into *m+1* stages, with each stage considering only the inspection and disassembly decisions for the products at that stage. In the first stage, there is no disassembly decision, only the inspection decision for the components. For the subsequent stages (semi-finished products and finished products), based on the defect rate of the products, decisions are made regarding whether to inspect and disassemble, in order to optimize the costs and revenues of the entire production process.

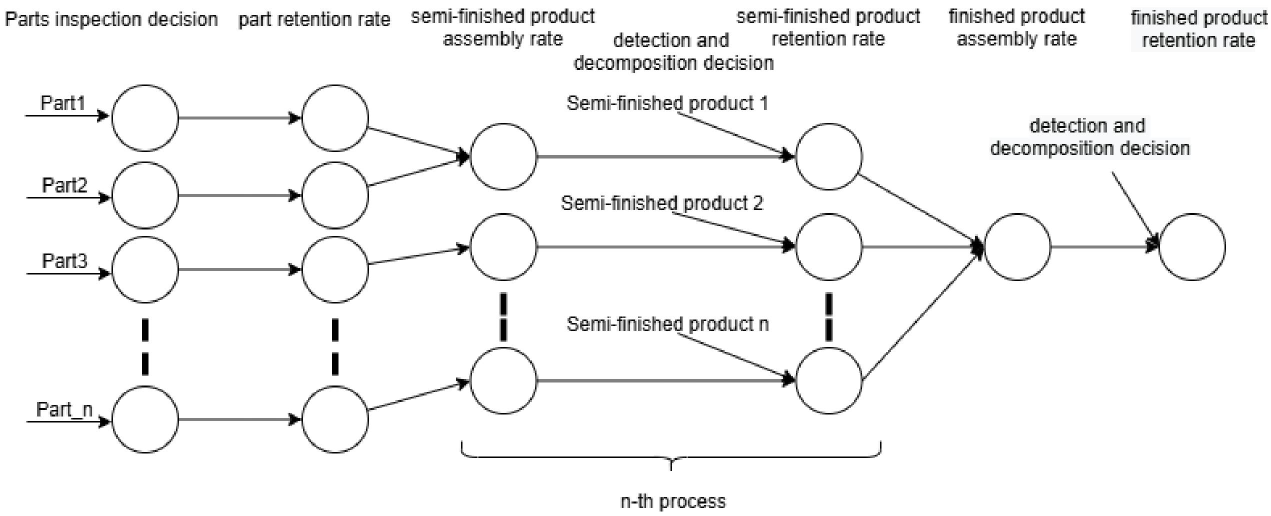

Figure 1. model diagram

This model introduce the assembly rate $\gamma_{i,j+1}$ and retention rate $N_{i,j}$:

Assembly Rate: This refers to the assembly rate of completed product *i-th* at the *(j+1)-th* stage.

$$\gamma_{i,j+1} = \min\{N_{i,j}\},\ i=p,p+1,\ldots,q \tag{15}$$

where the $N_{i,j}$ is Retention Rate, and p and q are the lower and upper bounds of the component part numbers for product *i-th*.

Retention Rate: This refers to the retention rate of completed product *i-th* at the *j-th* stage.

$$N_{i,j} = X_{detect\ i,j} \cdot \gamma_{i,j} \cdot \left(1-P_{i,j}\right) + (1-X_{detect\ i,j}) \cdot \gamma_{i,j} \tag{16}$$

where $X_{detect\ i,j}$ is the inspection decision variable, $\gamma_{i,j}$ is the assembly rate, and $P_{i,j}$ is the defect rate of the product.

In this model, the following objective function is constructed to maximize the expected profit of a single finished product. The objective function can be represented as:

$$\max \quad S = R - C_{1,m+1} \tag{17}$$

where *R* is the expected revenue from the final product.

$$R = R_{price} \cdot N_{1,m+1} \tag{18}$$

where R_{price} is the fixed market price per unit, and $N_{1,m+1}$ is the retention rate of the final product.

$$C_{1,m+1}=E_{cost\ i,j}+E_{assemble\ i,j}+E_{detect\ i,j}+E_{exchange\ i,j}+E_{dismantle\ i,j}-E_{reclaim\ i,j} \tag{19}$$

where $E_{cost\ i,j}$ is the expected total cost of the final product, and $C_{1,m+1}$ is the expected total purchase cost.

$$E_{cost\ i,j+1}=\sum_{i=p}^{q} C_{i,j} \cdot N_{1,j} \tag{20}$$

where $C_{i,j}$ is the product cost.

$$E_{assemble\ i,j}=C_{assemble\ i,j} \cdot \gamma_{i,j} \tag{21}$$

where $E_{assemble\ i,j}$ is the expected assembly cost, and $C_{assemble}$ is the assembly cost of the finished product.

$$E_{detect\ i,j}=x_{detect\ i,j} \cdot \gamma_{\ i,j} \cdot C_{detect\ i,j} \tag{22}$$

where $E_{detect\ i,j}$ is the expected inspection cost, $x_{detect\ i,j}$ is the decision variable for inspection and $C_{detect\ i,j}$ is the inspection cost.

$$E_{exchange\ i,j}=(1-x_{detect\ 1.m+1}) \cdot \gamma_{1,m+1} \cdot P_{1,m+1} \cdot C_{exchange} \tag{23}$$

where $E_{exchange\ i,j}$ is the expected replacement cost for defective finished products, and $C_{exchange}$ is the replacement cost for defective products. Replacements only occur for the final finished product.

$$E_{dismantle\ i,j}=x_{dismantle\ i,j} \cdot \gamma_{i,j} \cdot P_{i,j} \cdot C_{dismantle\ i,j} \tag{24}$$

where $x_{dismantle\ i,j}$ is the disassembly decision variable and $C_{dismantle\ i,j}$ is the disassembly cost for defective finished products.

$E_{reclaim\ i,j}$is the expected revenue from recovering a single defective product.

$$E_{reclaim\ i,j+1}=x_{detect\ i,j+1} \cdot x_{dismantle\ i,j+1} \cdot \sum_{i=p}^{q} C_{i,j} \cdot \left[\gamma_{i,j+1} \cdot P_{i,j+1}+\left(\gamma_{i,j}-\gamma_{i,j+1}\right)\right] \tag{25}$$

where $\gamma_{i,j+1} \cdot P_{i,j+1}$ is the defective product ratio, and $\gamma_{i,j}-\gamma_{i,j+1}$ is the ratio of unused parts after assembly of component *i*.

This model use a genetic algorithm to solve the above optimization problem. The genetic algorithm simulates the "survival of the fittest" mechanism in biological evolution, making it suitable for solving complex multivariable optimization problems, especially nonlinear and discrete objective functions, with minimal requirements for the objective function. Therefore, it is ideal for solving the optimal decision-making model.

Step 1: Initialize Population: Randomly generate multiple production decision solutions. Each solution is encoded as a chromosome, with each gene representing a decision variable (e.g., whether to inspect components, semi-finished products, etc.). The initial population size is preset.

Step 2: Fitness Evaluation: Evaluate the quality of each decision solution based on the objective function and calculate the expected profit. The solution with a higher fitness value is considered better. The fitness function is:

$$S=R-C_{1,m+1} \tag{26}$$

Step 3: Selection of Elite Individuals: The roulette wheel selection method is used to choose chromosomes for the next generation based on their fitness values. Individuals with higher fitness have a greater chance of being selected, improving the overall quality of the population.

Step 4: Multi-Point Crossover: Crossover is performed by exchanging genes between chromosomes to explore new solution spaces and avoid local optima.

Step 5: Mutation: With a 10% probability, gene mutation is applied to increase population diversity and further explore the solution space.

3. Results

3.1. The establishment of simulation model

Based on the parameters in Table 1, this model formulates four decisions: the detection decisions for parts 1 and 2, the detection decision for finished products, and the disassembly decision for finished products.

Table 1. Results of Maximum Profit Calculation Under Six Different Production Scenar

Situation	Component 1			Component 2			Finished Product	Defective Finished Product				
	Defective Rate	Purchase Unit Price	Detection Cost	Defective Rate	Purchase Unit Price	Detection Cost	Defective Rate	Assembly Cost	Detection Cost	Market Selling Price	Replacement Loss	Disassembly Cost
1	10%	4	2	10%	18	3	10%	6	3	56	6	5
2	20%	4	2	20%	18	3	20%	6	3	56	6	5
3	10%	4	2	10%	18	3	10%	6	3	56	30	5
4	20%	4	1	20%	18	1	20%	6	2	56	30	5
5	10%	4	8	20%	18	1	10%	6	2	56	10	5
6	5%	4	2	5%	18	3	5%	6	3	56	10	40

In this module, this paper first outlines the four decisions that enterprises must optimize at each stage of operations to maximize profits while ensuring product quality.

Decision 1 & Decision 2: Part detection decision, which determines whether to inspect the purchased spare parts (spare part 1 and spare part 2). If inspection is chosen, any non-conforming spare parts will be discarded; if no inspection is conducted, the non-conforming spare parts will proceed directly to the assembly stage.

Decision 3: Finished product detection decision, which determines whether to inspect the assembled finished products. If inspection is chosen, only the conforming finished products will enter the market; if no inspection is conducted, all finished products will proceed directly to the market.

Decision 4: Finished product disassembly decision, which determines whether to disassemble the non-conforming finished products identified during inspection. If disassembly is chosen, the spare parts will be recovered, and the steps for spare part detection and finished product detection will be repeated; if disassembly is not chosen, the non-conforming finished products will be discarded.

In this subsection, this model has derived the optimal inspection and disassembly schemes under six different production scenarios, along with their corresponding maximum expected profits.

This model have derived the optimal detection and disassembly strategies under six different production scenarios, along with their corresponding maximum expected profits, as shown in Table.2.

Table 2. Results of Maximum Profit Calculation under Six Different Production Scenarios.

Scenario	Decision 1	Decision 2	Decision 3	Decision 4	Maximum Expected Profit per Unit
1	0	0	0	1	23.5
2	0	0	0	1	19
3	0	0	0	1	21.1
4	0	0	1	1	18.2
5	0	0	0	1	23.1
6	0	0	0	0	24.7

Through literature and data analysis, the acceptable range of profit loss for the enterprise is determined to be between 0% and 10%. The results indicate that, in a global operational context, adopting the suboptimal decision-making scheme, which includes inspection of finished products, can achieve a higher expected profit per unit. The specific results for Scenario 1 are presented in Table 3.

An analysis of Table 2 leads to the conclusion that the profit loss ratio falls outside the acceptable range. Therefore, it is determined that the optimal decision-making scheme will not be changed, and this model will continue to opt for the option of not inspecting the finished products.

Table 3. Suboptimal Schemes for Decision-Making Involving Inspection of Finished Products in Scenario 1.

Strategy	Decision 1	Decision 2	Decision 3	Decision 4	Expected Profit per Unit
1	0	0	1	0	19.4
2	0	0	1	1	21.1
3	0	1	1	0	12.66
4	0	1	1	1	14.19
5	1	0	1	0	15.06
6	1	0	1	1	16.59
7	1	1	1	0	10.26
8	1	1	1	1	11.79
9	0	0	0	1	23.5

Through similar calculations, it was found that for Scenarios 2, 3, and 5, this model modifies the decision-making scheme, while for Scenario 6, it retains the decision-making scheme unchanged. The final results are presented in Table 4.

Table 4. Maximum Profit Solutions under Six Different Production Scenarios after Modifying the Decision-Making Scheme.

Scenario	Decision 1	Decision 2	Decision 3	Decision 4	Expected Maximum Profit per Unit
1	0	0	0	1	23.5
2	0	0	1	1	17.2
3	0	0	1	1	21.1
4	0	0	1	1	18.2
5	0	0	1	1	22.1
6	0	0	0	0	24.7

3.2. Results and Evaluation of the Genetic Algorithm-Based Decision-Making Model

In this section, this paper applies the proposed optimal decision-making model based on genetic algorithm to address the problem of two processes and eight components. The specific situation is illustrated in Figure 2, and the corresponding values are presented in Table 5.

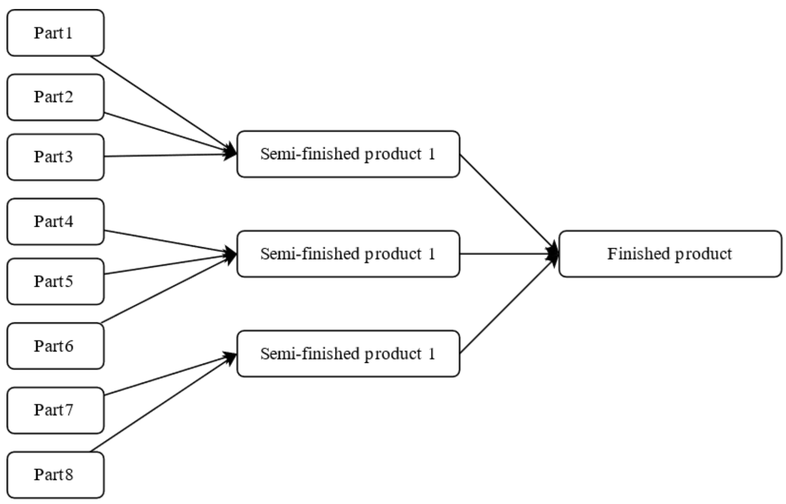

Figure 2. The Assembly Situation of Two Processes and Eight Components.

This model employs a genetic algorithm to address the enterprise decision-making problem, aiming to maximize expected profit by optimizing production decisions. Through the genetic algorithm, it simulates the 'survival of the fittest' evolutionary process, progressively refining decision points in the production process, such as whether to inspect or disassemble components. The resulting optimal decision plan is presented in Table 6.

The optimization model's results indicate that all components and semi-finished products need to be inspected. Semi-finished product 1 is not disassembled, while semi-finished products 2 and 3 are disassembled, based on a balance of quality control costs. Since the final products have already undergone strict quality control in earlier stages, no further inspection or disassembly is required. These decisions help minimize the impact of defective products while controlling production costs. This approach allows the company to ensure product quality while minimizing waste and additional expenses in the production process.

Table 5. Situations Encountered by the Company in Production

Component	Defective Rate	Purchase Price	Inspection Cost	Semi-finished Product	Defective Rate	Assembly Cost	Inspection Cost	Disassembly Cost
1	10%	2	1	1	10%	8	4	6
2	10%	8	1	2	10%	8	4	6
3	10%	12	2	3	10%	8	4	6
4	10%	2	1					
5	10%	8	1	Finished Product	10%	8	6	10
6	10%	12	2					
7	10%	8	1		Market Selling Price		Exchange Loss	
8	10%	12	2	Finished Product	200		40	

Table 6. Optimal Decision for Eight Parts in Dual Process

Decision1	Decision2	Decision3	Decision4	Decision5	Expected Profit
1	1	0	0	0	50
1	1	1	-1	-1	
1	1	1	-1	-1	
1	-1	-1	-1	-1	
1	-1	-1	-1	-1	
1	-1	-1	-1	-1	
1	-1	-1	-1	-1	
1	-1	-1	-1	-1	

4. Conclusions

This study addressed optimized decision-making in multi-stage manufacturing. A hybrid framework combining 0-1 integer programming and a genetic algorithm (GA) was proposed for inspection, disassembly, and resource allocation. The 0-1 model captures binary decisions and key parameters (defect/assembly rates, costs), while the GA efficiently identifies near-optimal solutions for large-scale problems.

Case studies demonstrated the framework's effectiveness. Optimized strategies, such as selective inspection and disassembly, yielded 10%-25% profit improvements. The GA consistently balanced quality control costs and revenue loss from defects. This framework offers a practical tool for manufacturers to enhance quality, reduce waste, and maximize profitability, bridging theoretical modeling with real-world application.

While promising, future research could:

Incorporate Uncertainty: Use stochastic or robust optimization for parameters like defect rates, which vary in reality (machine breakdowns, material quality).

Multi-Objective Optimization: Extend the framework to include environmental and social objectives alongside economic ones, aligning with sustainability trends.

Integrate Machine Learning: Predict parameters like defect rates using real-time data to improve adaptiveness.

Explore Other Algorithms: Investigate different metaheuristics or hybrid approaches for potentially better solutions in complex systems.

These enhancements would further improve decision-making in modern manufacturing.

References

[1] Rossit D G, Nesmachnow S, Rossit D A. A Multiobjective Evolutionary Algorithm based on Decomposition for a flow shop scheduling problem in the context of Industry 4.0[J]. 2022.

[2] Aheleroff S, Zhong R Y, Xu X. A digital twin reference for mass personalization in industry 4.0[J]. Procedia Cirp, 2020, 93: 228-233.

[3] Liu W, Kong C, Niu Q, et al. A method of NC machine tools intelligent monitoring system in smart factories[J]. Robotics and computer-integrated manufacturing, 2020, 61: 101842.

[4] Hidri L, Tlija M. Multi-stage hybrid flow shop scheduling problem with lag, unloading, and transportation times[J]. PeerJ Computer Science, 2024, 10: e2168.

[5] Triska Y, Agostino Í R S, Penna P M, et al. Integrated production and maintenance planning method with simulation-based optimization[J]. IFAC-PapersOnLine, 2021, 54(1): 349-354.

[6] Zhang X, Liao Z, Ma L, et al. Hierarchical multistrategy genetic algorithm for integrated process planning and scheduling[J]. Journal of Intelligent Manufacturing, 2022, 33(1): 223-246.

[7] Zhang B, Pan Q, Meng L, et al. An automatic multi-objective evolutionary algorithm for the hybrid flowshop scheduling problem with consistent sublots[J]. Knowledge-Based Systems, 2022, 238: 107819.

[8] Yuan S, Li T, Wang B. A discrete differential evolution algorithm for flow shop group scheduling problem with sequence-dependent setup and transportation times[J]. Journal of Intelligent Manufacturing, 2021, 32: 427-439.

[9] Wang K M, Wang K J, Chen C C. Capacitated production planning by parallel genetic algorithm for a multi-echelon and multi-site TFT-LCD panel manufacturing supply chain[J]. Applied Soft Computing, 2022, 127: 109371.

[10] Mahmudy W, Sarwani M, Rahmi A, et al. Optimization of multi-stage distribution process using improved genetic algorithm[J]. Int J Intell Eng Syst, 2021, 14(2): 211-219.

[11] Mansouri S A, Ahmarinejad A, Sheidaei F, et al. A multi-stage joint planning and operation model for energy hubs considering integrated demand response programs[J]. International Journal of Electrical Power & Energy Systems, 2022, 140: 108103.

[12] Rajak S, Vimal K E K, Arumugam S, et al. Multi-objective mixed-integer linear optimization model for sustainable closed-loop supply chain network: A case study on remanufacturing steering column[J]. Environment, development and sustainability, 2022, 24(5): 6481-6507.

[13] Xu N, Hou X Y, Jia N. Optimization of multi-stage production scheduling of automated production[J]. Int. J. Simul. Model, 2022, 21(1): 160-171.

[14] Khakifirooz M, Chien C F, Chen Y J. Bayesian inference for mining semiconductor manufacturing big data for yield enhancement and smart production to empower industry 4.0[J]. Applied Soft Computing, 2018, 68: 990-999.

[15] Panichella A. An improved Pareto front modeling algorithm for large-scale many-objective optimization[C]//Proceedings of the genetic and evolutionary computation conference. 2022: 565-573.