Journal of Computer Science and Electrical Engineering
ISSN: 2663-1946
DOI: https://doi.org/10.61784/jcsee3048

PRODUCTION SCHEDULING OPTIMIZATION MODEL BASED ON DYNAMIC PROGRAMMING AND GENETIC ALGORITHM

XiangLong Huang*, ZhengTing Li, LiKe Zhong
School of Computer Science and Engineering, Guangdong Ocean University, Yangjiang 529500, Guangdong, China.
Corresponding Author: XiangLong Huang, Email: 13500020575@stu.gdou.edu.cn

Abstract: In view of the problem that enterprises can not improve their profits due to the difficulty in planning reasonable decision-making schemes in real life, this paper proposes a dynamic programming model of production decision-making based on genetic algorithm. Firstly, a joint target benefit model considering the detection cost of parts, the purchase price of parts, semi-finished products, the assembly cost of finished products, the loss of replacement, semi-finished products, the disassembly cost of finished products, the disassembly of semi-finished products into parts, the disassembly compensation of finished products into semi-finished products, and the market price is established. Then, the genetic algorithm is used to optimize the variables of whether to detect parts, whether to detect semi-finished products, and whether to disassemble finished products (0-1). Finally, taking an actual production plan as an example, the validity of the model is verified, and the decision-making plan when the profit is the largest is obtained. This paper proposes a genetic algorithm-based dynamic programming model for production decision-making, which comprehensively considers various cost and price factors to optimize decision variables, thereby improving production efficiency and profit margins, and is verified to be valid through an actual production plan.
Keywords: Production decision; Dynamic programming; Genetic algorithm

1 INTRODUCTION

The quality of production decision-making will directly affect the sales of products[1],and indirectly affect the economic benefits of enterprises. Only through scientific and reasonable decision-making, can enterprises maximize the efficiency of resources and improve their competitiveness and profitability. Therefore, enterprise decision makers need to have a huge information reserve, strong analysis and judgment ability, as well as keen market insight and extraordinary decision-making thinking, in order to make wise decisions and create more value for the enterprise. In order to better meet the needs of enterprises and achieve good economic and commercial benefits, it is necessary to solve the decision-making problems in the production process according to the actual situation[2].Through the establishment of some appropriate mathematical models, a more systematic and scientific method can be used to analyze and study the production products, formulate flexible and reasonable product processing strategies, and plan reasonable decision-making plans based on the analysis results, improve product sales, and bring higher profits to the enterprise[3]. In this paper, the dynamic programming model is optimized by genetic algorithm for the decision-making problem of enterprise production of electronic products. Then, the dynamic programming model is established and solved based on the lowest production cost and the highest economic benefit. Finally, the optimal enterprise production decision-making scheme is given[4]. In order to solve the problem that enterprises can not improve their income due to the difficulty of planning reasonable decision-making schemes in real life.

2 DYNAMIC PROGRAMMING MODEL OF PRODUCTION DECISION BASED ON MULTI-OBJECTIVE

2.1 Joint Benefit Objective Function

2.1.1 Definition of decision variables

Due to the increase in the number of spare parts, you can set 0-1 decision variables[5] x_i, i can be 1, 2,... In this paper, we consider the case of eight spare parts. w_i represents whether to detect semi-finished products, i= 1,2,3 represent three semi-finished products respectively, w_i = 0 means not to detect semi-finished products, and the detection of finished products still use y as the decision variable, z_i= 0 indicates no disassembly, i= 1,2,3 indicates whether the semi-finished products are disassembled, and i = 4 indicates whether the finished products are disassembled

$$x_i = \begin{cases} 0 & \text{Do not detect spare parts} \\ 1 & \text{detect spare parts} \end{cases} \quad (i = 1,2,3,\cdots,8)$$

$$w_i = \begin{cases} 0 & \text{Do not detect semi} - \text{finished products} \\ 1 & \text{detect semi} - \text{finished products} \end{cases} \quad (i = 1,2,3)$$

$$z_i = \begin{cases} 0 & \text{Not disassembled} \\ 1 & \text{disassembled} \end{cases} \quad (i = 1,2,3,4)$$

2.1.2 Establish the cost of testing

C_1 and C_2 are defined as the detection cost of spare parts 1 and 2, and C_i (i= 1,2,3,....,n) For the detection cost of the number of spare parts, the case of eight spare parts is also considered here. S_i (i = 1, 2, 3,...,n) is the detection cost of semi-finished products, C_f is the detection cost of finished products, and the detection cost is obtained as shown in Equation (1).

$$C = \sum_{i=1}^{8} x_i C_i + \sum_{j=1}^{3} w_j S_j + yC_f \tag{1}$$

Among them, C is the total inspection cost, which is composed of the inspection cost of spare parts, semi-finished products and finished products.

2.1.3 Determine the unit price cost

The purchase unit price is 2 yuan, 8 yuan, 12 yuan. Considering the unit price cost of the purchase, the purchase unit price of the 8 parts is summed to determine the unit price cost :

$$D = \sum_{i=1}^{8} m_i \tag{2}$$

Where m_i (i = 1, 2, 3....., 8) respectively represent the purchase unit price of parts 1-8, and D is the unit price cost.

2.1.4 Calculating assembly costs

The cost of assembly of semi-finished products and finished products can be expressed as Ca_i, Ca_1, Ca_2 and Ca_3 represent the assembly cost of semi-finished products, Ca_4 represents the assembly cost of finished products, and the assembly cost can be listed as Equation (3).

$$C_a = \sum_{i=1}^{4} C_{ai} \tag{3}$$

Where Ca is the total assembly cost, which is composed of the assembly cost of semi-finished products and the assembly cost of finished products.

2.1.5 Calculation of switching costs

The replacement loss needs to take into account the defective rate of the finished product, and the defective rate of the finished product is affected by the defective rate of the semi-finished product, and the defective rate of the semi-finished product is affected by the defective rate of the parts. The defective rate adjustment function is introduced to solve the new semi-finished product defective rate and finished product defective rate [6].

The adjustment function of the defective rate of spare parts is introduced :

$$Q_j = \sum_{i=1}^{8} (1 + P_i)^{(1-x_i)} \cdot Q_{sj} \qquad (j = 1,2,3) \tag{4}$$

where P_i (i = 1, 2,......,8)The defective rate of 8 spare parts is represented respectively. Q_{s1}, Q_{s2} and Q_{s3} represent the defect rate of the original semi-finished products respectively, and Q_j represents the defect rate of the new semi-finished products.

The adjustment function of the defective rate of spare parts is introduced :

$$P_f = \sum_{j=1}^{3} (1 + Q_j)^{(1-w_j)} \cdot P_s \tag{5}$$

P_s represents the defective rate of the original finished product, and P_f represents the defective rate of the new finished product.

$$L = (1 - y) \cdot C_L \cdot P_f \tag{6}$$

Using the new finished product defect rate, combined with the exchange loss C_L, the total cost of the exchange can be obtained.

2.1.6 Calculation of disassembly cost and disassembly compensation

C_{di} is defined as the disassembly cost. C_{d1}, C_{d2} and C_{d3} represent the disassembly cost of semi-finished products. C_{d4} represents the disassembly cost of finished products. The disassembly cost can be expressed as the sum of the costs of semi-finished products or finished products to be disassembled, which can be expressed as Formula (7).

$$C_x = \sum_{i=1}^{4} z_i \cdot C_{di} \tag{7}$$

The dismantling of some unqualified products (finished or semi-finished products) may reduce the qualified probability of unqualified products. Here, the qualified rate is still defined as a decrease of 50 %. m_9 (22 yuan), m_{10} (22 yuan) and m_{11} (20 yuan) respectively represent the unit price of semi-finished products, which is obtained by summing the unit price of the assembled spare parts.

$$B_1 = \sum_{i=1}^{3} \sum_{j=1}^{3} z_j \cdot [(0.5 - P_{3j+i-3}) \cdot m_{3j+i-3}] \quad (3j + i - 3 < 9) \tag{8}$$

B_1 represents the disassembly compensation of semi-finished products into spare parts, and 3j + i-3 represents the number of spare parts respectively, which needs to be less than 9, because there are only 8 maximum spare parts.0.5-P_{3j+i-3} represents the qualified rate of spare parts, and the product with the unit price m_{3j+i-3} represents the return cost of qualified products , and the product with the decision variable z represents the disassembly compensation[7].

$$B_2 = \sum_{j=1}^{3} z_4 \cdot [(0.5 - Q_j) \cdot m_{j+8}] \tag{9}$$

B_2 represents the disassembly compensation of finished products into semi-finished products, and m_{j+8} represents the unit price of semi-finished products m_9, m_{10} and m_{11} respectively. The 0.5-Q_j represents the qualified rate of the semi-finished product, and the unit price m_{j+8} represents the return cost of the qualified product, and it is integrated with the decision variable z to represent the disassembly compensation.

2.2 Establishment of Objective Function

Considering the profit and cost, combined with the market price M = 200 yuan, the objective function for solving the optimal value can be listed[8], that is, $\mathrm{Max}\omega = \mathrm{M} - \mathrm{C} - \mathrm{D} - \mathrm{Cx} - \mathrm{Ca}$,, $\mathrm{C} + \mathrm{D} + \mathrm{Cx} + \mathrm{Ca}$ represents the total cost, so that it is as small as possible. The objective function can also be converted into the maximum value of ω, and the constructed objective function is shown in Equation (10).

$$Max\ \omega = M - D - C - C_a - C_x + (B_1 + B_2) - L$$

$$st.\begin{cases} x_i, w_i, z_i, y = \begin{cases} 0 \\ 1 \end{cases} \quad (i = 1,2,3...) \\ Q_j = \sum_{i=1}^{8} (1+P_i)^{(1-x_i)} \cdot Q_{sj} \qquad (j = 1,2,3) \\ P_f = \sum_{j=1}^{3} (1+Q_j)^{(1-w_j)} \cdot P_s \\ D = \sum_{i=1}^{8} m_i \\ C = \sum_{i=1}^{8} x_i C_i + \sum_{j=1}^{3} w_j S_j + yC_f \\ C_a = \sum_{i=1}^{4} C_{ai} \\ C_x = \sum_{i=1}^{4} z_i \cdot C_{di} \\ B_1 = \sum_{i=1}^{3}\sum_{j=1}^{3} z_j \cdot [(0.5 - P_{3j+i-3}) \cdot m_{3j+i-3}] \quad (3j+i-3<9) \\ B_2 = \sum_{j=1}^{3} z_4 \cdot [(0.5 - Q_j) \cdot m_{j+8}] \\ L = (1-y) \cdot C_L \cdot P_f \end{cases} \tag{10}$$

ω is the income of the enterprise, which is required to make the income as large as possible.

3 ANALYSIS PROCESS OF DYNAMIC PROGRAMMING MODEL BASED ON GENETIC ALGORITHM

3.1 The Working Principle of Genetic Algorithm

The basic principle of genetic algorithm is derived from the basic concepts of Darwin 's natural selection and genetics[9]. In the genetic algorithm, each instance of the solution is regarded as an ' individual ', and the entire solution space forms a ' population '. Each individual is represented by a string of ' genes ' that encode the specific parameters of the solution[10]. The genetic algorithm continuously improves the quality of the population and approaches the optimal solution through the iterative process. The core steps include selection, crossover and mutation.

Consider a simplified genetic algorithm model, whose fitness function $f(x)$ is used to evaluate the performance of each individual, where x is a vector that encodes individual features. The objective of the algorithm is to maximize the fitness function[11]. An iteration of genetic algorithm can be expressed as the following steps :

3.1.1 Selection

The probability that an individual is selected for reproduction is proportional to its fitness. If we set the probability that p_i the first i-individual is selected.

$$p_i = \frac{f(x_i)}{\sum_{j=1}^{N} f(x_j)} \tag{11}$$

In formula (11), $f(x_i)$ is the fitness of the first i individual, and N is the total number of individuals in the population.

3.1.2 Crossing

The selected individuals generate new offspring through crossover operation. If the crossing point is k and two individuals are considered x_i and x_j, the descendant x_{new} can be expressed as :

$$x_{new} = \left(x_{i1}, x_{i2}, \ldots, x_{ik}, x_{j(k+1)}, \ldots, x_{jn}\right) \tag{12}$$

3.1.3 Mutation

Some genes of the newborn individuals are modified with a small probability μ to introduce variation and increase the diversity of the population. For gene x_{nk}, the mutation operation can be expressed as

$$x_{nk}^{'} = x_{nk} + \delta, \quad \text{with probability} \mu \tag{13}$$

In formula (13), δ is a small random variable and μ is the mutation rate.
By repeating these steps, the genetic algorithm can gradually improve the quality of the solution after multiple generations of iterations, close to the optimal solution[12]. The fitness function of each generation is usually improved, indicating the effectiveness of the algorithm in solving specific problems.

3.2 Dynamic Programming Model Solving Process

The solving process of the dynamic programming model is shown in Figure 1. The specific steps are as follows :
Step 1 : Construct a multi-objective normalization function. By setting the parameters of the objective function, the total profit and the defective rate penalty are combined into a multi-objective function. Normalized by the following formula : fitness value = total profit-defect rate penalty coefficient × defect rate
Step 2 : Define the optimization variables and algorithm parameters. A total of 16 binary decision variables including parts detection strategy (8 variables), semi-finished product detection strategy (3 variables), finished product detection strategy (1 variable), disassembly strategy (3 variables) and finished product disassembly strategy (1 variable) are taken as the optimization objectives. Set genetic algorithm parameters : population size, number of iterations, crossover probability, mutation probability. Variable upper and lower limits, etc.
Step 3 : Constraint condition verification and strategy generation. 1.Substandard product rate constraint : If the spare parts are not detected, the semi-finished product rate is cumulatively punished according to the formula : semi-finished product rate * = (1 + spare parts defective rate). If the semi-finished product is not detected, the finished product defective rate is cumulatively punished according to the formula : finished product defective rate * = (1 + semi-finished product defective rate). If the finished product is not detected, the total cost needs to be superimposed on the exchange loss × finished product defect rate. Disassembly compensation constraint : If you choose to disassemble semi-finished products or finished products, the total cost needs to be deducted from the compensation value. 3.Validation of policy effectiveness : Traverse all policy combinations and select the optimal policy that satisfies the constraints.
Step 4 : Joint optimization and result output : Combining genetic algorithm with dynamic programming to optimize the total profit maximization strategy. The maximum total profit strategy and the corresponding detection and dismantling scheme are output.

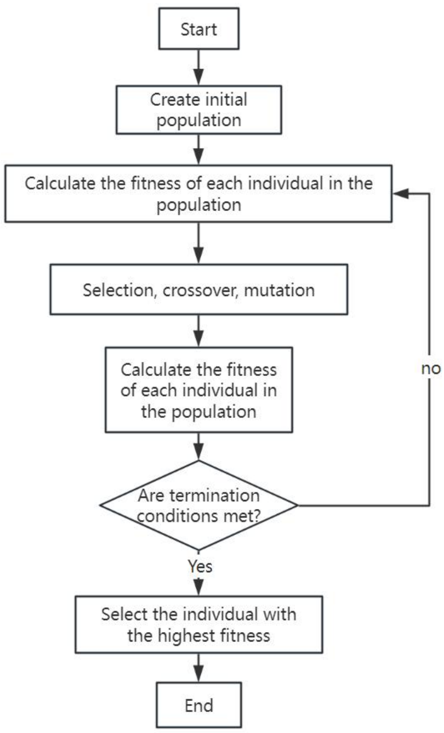

Figure 1 Dynamic Programming Model Solving process

4 AN INNOVATIVE SIMULATION - BASED SOLUTION APPROACH

The results of the improved fusion model : By initializing the population function, establishing the fitness function, evaluating the fitness of each individual, calculating the total profit and the defective rate, taking the total profit as the fitness value, and then performing a series of operations such as crossover and mutation of the genetic algorithm, 65536 different decision-making schemes are obtained after running the python software. Considering the large amount of sample data, there is no specific decision-making scheme here, only the total profit comparison under different strategies is shown, as shown in Figure 2.

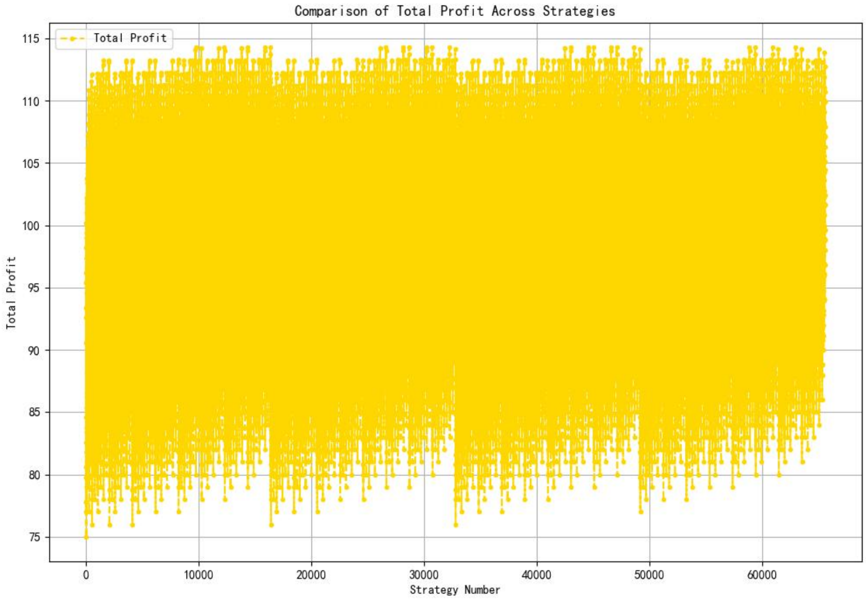

Figure 2 Total Profit Comparison under Different Strategies

Through the information in Figure 2, it can be seen that under more than 60,000 strategies, the total profit of the decision-making scheme shows a relatively regular change, and its total profit is roughly distributed between 83-112 yuan.

The highest total profit of 114.29 yuan is obtained by screening the output results in descending order. The decision-making scheme is to detect spare parts 1, 2, 4, not to detect spare parts 3, 5, 6, 7, 8, not to detect semi-finished products 1-3, detect finished products, disassemble unqualified semi-finished products 1-3, disassemble unqualified finished products. The minimum total profit is 75 yuan. The decision-making scheme is to detect parts 1-8, semi-finished products 1-3, finished products, unqualified semi-finished products 1-3, and unqualified finished products.

Table 1 Top 3 Decision Schemes of Total Profit

Policy number	decision plan	gross profit	Finished product rate
12257	testing spare parts 1,2,4, not testing spare parts 3,5,6,7,8, not testing semi-finished products 1-3, testing finished products, dismantling unqualified semi-finished products 1-3, dismantling unqualified finished products	114.2927	0.1565
14305	testing spare parts 1,2,5, not testing spare parts 3,4,6,7,8, not testing semi-finished products 1-3, testing finished products, dismantling unqualified semi-finished products 1-3, dismantling unqualified finished products	114.2927	0.1565
15841	testing spare parts 1,2,7, not testing spare parts 3,4,5,6,8, not testing semi-finished products 1-3, testing finished products, dismantling unqualified semi-finished products 1-3, dismantling unqualified finished products	114.2927	0.1565

Table 1 shows the decision-making schemes in the top 3 of the total profit. Through the data in Table 1, the decision-making scheme in the production process can be clearly provided for the enterprise. How to make decisions can maximize the benefits, and the finished product defect rate is taken as the corresponding index of the decision-making scheme.

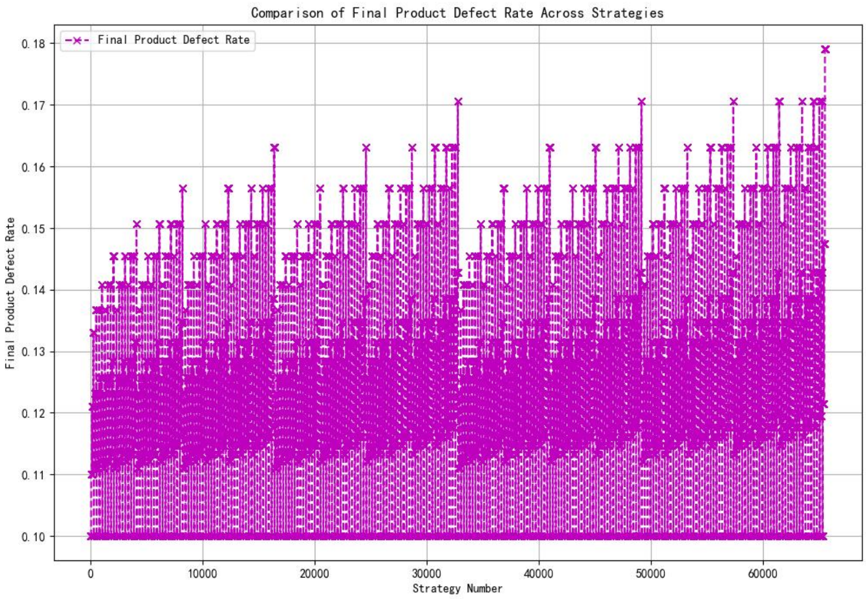

Figure 3 Comparison of Finished Product Defect Rate under Different Strategies

As depicted in Figure 3, the graph illustrates the comparison of the final product defect rate across various strategies. This visualization clearly demonstrates the fluctuation and distribution of defect rates under different strategies. It can be observed that the defect rate shows a certain degree of randomness and variability, without a clear - cut upward or downward trend. This further confirms that the defective rate of finished products under different strategies can indeed be used as the corresponding index result. After providing the specific decision - making scheme and its basis, analyzing this graph allows us to gain a more intuitive and comprehensive understanding of how different strategies impact the defect rate of finished products.

By establishing a production scheduling optimization model based on dynamic programming and genetic algorithm, remarkable results have been achieved : in 65536 decision-making schemes, the highest total profit reaches 114.29 yuan, and the corresponding decision-making scheme is to detect spare parts 1,2,4, do not detect spare parts 3,5,6,7,8, do not detect semi-finished products 1-3, detect finished products, disassemble unqualified semi-finished products 1-3, disassemble unqualified finished products ; the minimum total profit is 75 yuan, and the corresponding decision-making scheme is to detect all spare parts and semi-finished products, detect finished products, but not disassemble any non-conforming products.

5 CONCLUSION

This paper focuses on solving the problem that the profit cannot be improved due to the difficulty in formulating a reasonable decision-making plan in the actual operation of the enterprise. In view of the limitations of traditional dynamic programming methods in dealing with large-scale and complex problems, this paper introduces a dynamic programming model based on genetic algorithm. Genetic algorithm can effectively avoid local optimum and improve search efficiency. By combining genetic algorithm with dynamic programming, the advantages of both can be fully utilized to solve the problems that traditional dynamic programming is difficult to deal with. The actual production plan is taken as an example to verify the application. The results show that the model is effective and the corresponding decision-making scheme is successfully found when the profit reaches the maximum value.The total profit is roughly distributed between 83-112 yuan. Future work can consider using more intelligent optimization algorithms, such as particle swarm optimization algorithm, simulated annealing algorithm or ant colony algorithm, to further improve the efficiency of the model and the quality of the solution, and provide better production decision-making solutions for enterprises.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCE

[1] Chauhan M, Negi R, Ravi, et al. Empowering Small-Scale Vegetable Farmers with Drone-based Decision Support Systems for Sustainable Production. International Journal of Environment and Climate Change, 2025, 15(1): 203-216. DOI: 10.9734/ijecc/2025/v15i14685.

[2] Song H, Li X, Xin L, et al. Forging new pathways: How farmland abandonment affects decision-making of non-grain production - Insight from China's mountainous areas. Journal of environmental management, 2024, 373123753.

[3] Zhang L, Zhou J. Joint Pricing-Production Decisions for a Capital-Constrained Supplier in a Marketplace Platform. Journal of Theoretical and Applied Electronic Commerce Research, 2024, 19(4): 3547-3570.
[4] Liu T. The Research on Production Decision-Making Based on Cost-Benefit Analysis Models. Industrial Engineering and Innovation Management, 2024, 7(4): 28-36.
[5] Jin Z, Zhao Y, Sun Y, et al. Force feedback controller of a parallel haptic device via online adaptive dynamic programming. Mechatronics, 2025, 106103293-103293.
[6] Yang X, Wu C, Huang Z, et al. A two-stage dynamic planning for rural hybrid renewable energy systems under coupled carbon-green certificate trading. Energy, 2025, 316134336-134336.
[7] Wang K, Song H, Guo Z, et al. Automated multi-dimensional dynamic planning algorithm for solving energy management problems in fuel cell electric vehicles. Energy, 2025, 316134408-134408.
[8] Chen H, Lin Z, Chen Z, et al. Adaptive DWA algorithm with decision tree classifier for dynamic planning in USV navigation. Ocean Engineering, 2025, 321120328-120328.
[9] Qazi M I, Muneer A, Sonia R, et al. RDF Query Path Optimization Using Hybrid Genetic Algorithms: Semantic Web vs. Data-Intensive Cloud Computing. International Journal of Cloud Applications and Computing (IJCAC), 2022, 12(1): 1-16.
[10] Tian-Wei L, Jiang-Bo B, Nicholas F, et al. Multi-objective optimisation designs for thin-walled deployable composite hinges using surrogate models and Genetic Algorithms. Composite Structures, 2022, 280.
[11] Evandro M N, Bruno B G H, Sergio S G H, et al. Variable selection for estimating individual tree height using genetic algorithm and random forest. Forest Ecology and Management, 2022, 504.
[12] Genghan Y, Yang T, Peng W, et al. Optimizing 3D printing of chicken meat by response surface methodology and genetic algorithm: Feasibility study of 3D printed chicken product. LWT, 2022, 154.