INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
Online ISSN 1841-9844, ISSN-L 1841-9836, Volume: 20, Issue: 1, Month: February, Year: 2025
Article Number: 6874, https://doi.org/10.15837/ijccc.2025.1.6874

Intelligent Scheduling of Automotive Stamping Workshops Based on an Enhanced Genetic Algorithm

Zhiyuan Li, Desheng Wu, Cheng He, Xiaofeng Gan, Peichun Chen

Zhiyuan Li*
School of Intelligence Technology
Geely University of China, China
No.123, Chengjian Avenue, Chengdu City, China
*Corresponding author: lizhiyuan@guc.edu.cn

Desheng Wu
Zhejiang Geely New Energy Commercial Vehicle Group Co., LTD, China
desheng.wu@geely.com

Cheng He
Zhejiang Geely New Energy Commercial Vehicle Group Co., LTD, China
he.cheng@geely.com

XiaoFeng Gan
Zhejiang Geely New Energy Commercial Vehicle Group Co., LTD, China
284277279@qq.com

Peichun Chen
Research Servicesr
Coventry University, Coventry, UK
Priory Street Coventry CV1 5FB, UK
ad8230@coventry.ac.uk

Abstract

With the rapid development of the automotive manufacturing industry, enhancing production efficiency and flexibility in stamping workshops has become crucial for maintaining competitiveness. This paper presents an intelligent scheduling method based on an enhanced genetic algorithm to address the scheduling challenges in automotive stamping workshops. By incorporating the magic square selection method for initializing the population and an improved roulette wheel selection operation, this approach significantly enhances the search efficiency and solution quality of the genetic algorithm. During the algorithm's implementation, innovative crossover methods and decoding mechanisms were designed to better accommodate the production characteristics and constraint conditions of stamping workshops. The proposed method was validated using production data from an automotive factory. The scheduling results demonstrate that the enhanced genetic algorithm effectively addresses over 95% of the scheduling issues encountered in intelligent scheduling of automotive stamping workshops. The algorithm markedly improves workshop production efficiency, reduces production cycles, and enhances the stability and feasibility of production plans. This study provides a robust technical solution for intelligent production management in automotive stamping workshops, offering substantial theoretical significance and practical application value.

Keywords: Automotive Stamping Workshop; Intelligent Scheduling; Enhanced Genetic Algorithm; Optimization Scheduling.

1 Introduction

In automotive production management, the formulation of production plans is essential for the rational utilization of enterprise resources and the maximization of customer satisfaction. Effective production scheduling plays a key role in significantly reducing product wait times and in-process durations, as well as markedly improving equipment utilization. The optimization of production planning and scheduling has consistently been a research hotspot. With the continuous advancements in operations research and computer technology, production planning and scheduling have transitioned from manual operations to computer-aided automation. This transition has become a vital method for enterprises to conserve resources, reduce costs, improve productivity, and thereby enhance overall operational efficiency.

Workshop scheduling is a combinatorial problem and remains one of the critical challenges that many researchers seek to address. In many small automotive enterprises, the production scheduling of stamping workshops still largely depends on the expertise of experienced senior workers. However, when faced with highly complex issues, they may struggle to respond promptly, and their problem-solving approaches often have substantial limitations and subjectivity. Relying solely on personal work experience to determine production schedules cannot fully meet production needs[16].Therefore, there is a pressing need for an efficient and feasible production scheduling algorithm that can address the processing time and sequence of different workpieces under multiple constraints. Such an algorithm would ensure the optimization of relevant indicators for each workpiece, enabling the rational utilization and allocation of processing time and equipment for each batch of parts. This, in turn, would minimize production cycles and maximize production efficiency.

Currently, some scholars have employed hybrid genetic algorithms or effective genetic algorithms based on estimation of distribution to address multi-objective flexible job-shop scheduling problems. These algorithms demonstrate a degree of solving efficiency and accuracy, optimizing production time and equipment utilization. However, in practical production scenarios, these algorithms encounter significant limitations and are not ideal for scheduling purposes[22][18].

The enhanced genetic algorithm proposed in this paper can effectively solve intelligent scheduling problems in automotive stamping workshops and can be applied to address practical issues.

2 Literature Review

2.1 Initial Population Optimization

As a robust and versatile intelligent optimization algorithm, the genetic algorithm (GA) has been widely applied in the manufacturing industry, demonstrating outstanding performance particularly in production scheduling, process parameter optimization, and resource allocation [3][24]. Numerous scholars have conducted in-depth research on the scheduling problems in automotive stamping workshops using various solution algorithms. However, traditional genetic algorithms suffer from poor global and local search capabilities and are prone to premature convergence. Consequently, extensive research has been conducted to improve genetic algorithms. For instance, Wang et al. proposed an improved decimal-based group encoding scheme for the reverse job scheduling problem, which simultaneously optimizes processes and parameters[21]. Cheng et al. introduced a heuristic method and a machine chain optimization method based on short processing times and equipment balancing strategies to enhance the quality of the initial population for solving shop scheduling problems using genetic algorithms[2].

The studies primarily focus on population initialization, genetic operations, encoding, and the integration of other algorithms. However, it is evident that current methods for constructing the initial population of genetic algorithms are overly complex and fail to address several critical issues. These include the difficulty in determining the optimal initial population size, poor stability of initial solutions, challenges in ensuring the diversity of the initial population, and the negative impact on the genetic operation process[5][1].

2.2 Idealized objective function

The objective function in genetic algorithms for job shop scheduling problems is crucial as it not only evaluates the fitness of individuals and guides the search direction but also measures optimization effects and handles multi-objective optimization problems and constraints[23] . This enhances the robustness and diversity of the algorithm. For instance, Ponnambalam et al. introduced a genetic algorithm for bi-objective job shop scheduling problems, providing significant insights into multi-objective optimization in job shop scheduling[17]. Similarly, Yu et al. discussed the application of genetic algorithms for solving multi-objective flexible job shop scheduling problems, offering concrete solutions for multi-objective optimization[7]. Additionally, Wang proposed a hybrid optimization strategy that combines genetic algorithms with other optimization methods to address job shop scheduling problems, demonstrating the flexibility and adaptability of genetic algorithms[19].

The studies primarily focus on minimizing processing time under multi-objective functions. However, traditional multi-objective optimization methods often fail to obtain suitable solutions, leading to significant fluctuations in objective function values and poor optimization performance.

3 Enhanced Realization of Genetic Algorithms

3.1 Characteristics of Production in an Automotive Stamping Workshop

Automotive manufacturing generally includes four major workshops: stamping, welding, painting, and final assembly. During the production process, sheet metal sequentially goes through stamping, welding, painting, and finally reaches the final assembly stage, completing the entire process to become a finished vehicle. Among these stages, welding, painting, and final assembly can be regarded as flow-type production processes. Based on the final assembly plan, the required vehicle models are broken down according to the Bill of Materials (BOM), and the finished stamped parts are directly retrieved from the stamped parts inventory for the flow-type production. To address the challenge of denoising weak laser signals, this article employs the following research methodology[13].

The stamping workshop is significantly different from the other workshops. It typically contains several production lines, each composed of 5-6 stamping presses, depending on the number of operations required for the stamped parts. In the stamping workshop, various stamped parts for the car body are processed through operations such as uncoiling and shearing, drawing, shaping, trimming, punching, and flanging. These operations are performed by changing the dies, each operation following a fixed process route. The time required to complete a single operation on a stamped part is approximately 1-20 seconds[25]. However, when different operations are processed on the same stamping press, die changes are necessary, and the die change time typically ranges from 10-15 minutes. This die change time is a crucial factor to consider in the scheduling of the stamping workshop [11] . The production flow in the automotive stamping workshop is illustrated in Figure 1.

3.2 Workshop Scheduling Problem

As a relatively unique type of scheduling problem, the essence of the stamping workshop is to achieve reasonable resource allocation for multiple tasks under given multi-resource constraints, while maintaining the basic order of tasks. The stamping workshop falls under the category of job shop scheduling; however, due to the special nature of its production, it also exhibits many characteristics similar to flow shop scheduling. In the scheduling model of a stamping workshop, there are n different types of stamping parts that need to be arranged on a production line consisting of m stamping machines for stamping[20]. The number of operations and the processing time for each operation of the various stamping parts are known but vary, with the number of operations generally ranging from

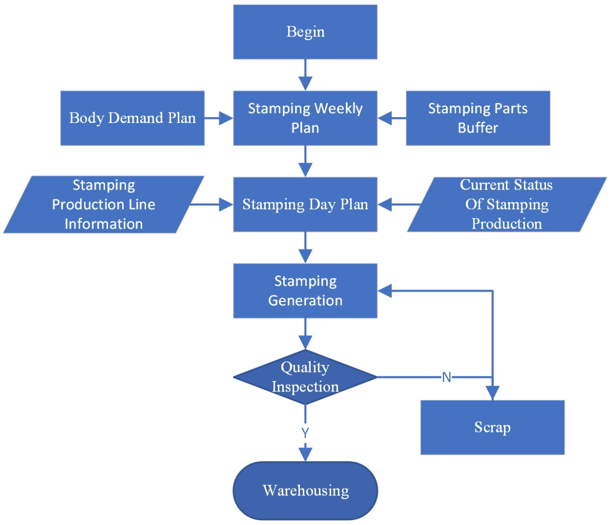

Figure 1: Production Flowchart of an Automotive Stamping Workshop

1 to 5. Each operation does not necessarily have to be processed on a specific stamping machine but must be processed on a machine that meets the processing conditions[9][14]. The setup time (die change time) for the start of processing each batch of stamping parts is known. Therefore, the scheduling model needs to determine the processing sequence of each batch of stamping parts, the sequence of stamping machines, and the start time of each operation. Short processing times and high equipment utilization rates result in lower production costs[6]. Thus, the objective function is set to minimize the processing time.

$$min : H = t \tag{1}$$

Where H is the objective function and t is the total time to complete all processing tasks, measured in seconds. The value of t can be expressed as:

$$t = max \{T (M_1) , T (M_2) , T(M_m)\} \tag{2}$$

Formula 2 indicates that the completion time of the job is equal to the latest completion time among all the machines, where T(Mm) represents the final completion time of machine m.

Let $M = \{M_1, M_2, \ldots, M_m\}$ represent the set of stamping machines, $J = \{J_1, J_2, \ldots, J_n\}$ represent the set of stamping tasks, $N = \{N_1, N_2, \ldots, N_n\}$ represent the batch quantities of the stamping parts, $O_i = \{O_{i1}, O_{i2}, \ldots, O_{ir_i}\}$ represent the set of operations for stamping part $J_i, p_i = p_{i1}, p_{i2}, \ldots, p_{ir_i}$ represent the unit processing time for each operation of stamping part i. The batch processing time for stamping part J_i can be expressed as formula 3.

$$T (O_{ik}) = p_{ik} \times N_i \tag{3}$$

Let m_{ik} represent the stamping machine for operation O_{ik} ,where i=1,. . . ,n, k=1,. . . ,r_i , $m_{ik} \subset M$, S_{ik} represent the batch processing time for operation O_{ik}, where i=1,. . . ,n and k=1,. . . ,r_i clearly $S_i1 = S_i$, i=1,. . . ,n. $TS_{(ij,i(j+1))}$ represent the transportation time from the j-th operation to the (j+1)-th operation for stamping part i, i=1,. . . ,n,k=1,. . . ,r_i as shown in formula 4.

$$TS_{ij,i(j+1)} = ts \times |m_{i(j+1)} - m_{ij}| \tag{4}$$

Let $WT_{ik}^{M_j}$ represent the waiting time for operation O_{ik} on stamping machine M_m before stating, which is the idle time of the equipment, where i=1,. . . ,n, k=1,. . . ,r_i. Let $ST_{ik,jl}^{M_s}$ represent the die change time from operation O_{ik} to operation O_{jt} on stamping machine M_s where i=0,. . . ,n,k=0,. . . ,r_i, j=1,. . . ,n,l=1,. . . ,r_j,s=1,. . . ,m.

Let $TS_{ij,i(j+1)}^{M_m}$ represent the transmission time for the operation processed on the stamping machine M_m and $T(O_{ik})^{M_m}$ represent the processing time for the operation on the stamping mache M_m. Then the completion time of the stamping machine $T(M_m)$ can be expressed as shown in formula 5.

$$T\left(M_m\right) = \sum WT_{ik}^{M_m} + \sum TS_{ij,i(j+1)}^{M_m} + \sum ST_{ik,jl}^{M_m} + \sum T(O_{ik})^{M_m} \tag{5}$$

Combining the above equations, the objective function can be expressed as formula 6.

$$min : H = \max\{\sum WT_{ik}^{M_m} + \sum TS_{ij,i(j+1)}^{M_m} + \sum ST_{ik,jl}^{M_m} + \sum T(O_{ik})^{M_m}\} \tag{6}$$

After determining the objective function, the following constraints need to be satisfied:

(1) Operation constraints. The processing time of the subsequent operation in a batch of stamping parts cannot be earlier than the completion time of the first operation of the previous step, as shown in formula 7.

$$S_{ik} \geq S_{i(k-1)},\ i = 1, \ldots, n; k = 2, \ldots, r_i \tag{7}$$

(2) Stamping machine constraints-1: Each stamping machine can only process one operation at a time, as shown in formula 8 and 9.

$$S_{ik} \geq E_{jl,ik}^{M_{ik}} \left(S_{jl} + N_j \times p_{jl} + ST_{jl,ik}^{M_{ik}}\right), i, j = 1, \ldots, n, k \neq l, k = 1, \ldots, r_i, l = 1, \ldots, r \tag{8}$$

$$E_{jl,ik}^{M_{ik}} = \begin{cases} 0, & \text{if Operation } O_{ik} \text{ is the first operation processed on machine } M_s \\ 1 & \end{cases}$$
$$i = 0, \ldots, n,\ k = 0, \ldots, r,\ j = 1, \ldots, n,\ l = 1, \ldots, r_j,\ s = 1, \ldots, m \tag{9}$$

(3) Stamping machine constraints-2: The equipment's rated stamping force must meet or exceed the stamping force required by the operation.

3.3 Initialization of the Population Based on Magic Square

To ensure the diversity of the genetic algorithm population and improve the quality of the initial solutions, the current scheduling problem in the stamping workshop employs methods such as swap encoding, random encoding, and quantum bit encoding. However, these encoding methods encounter the issue of overly complex genetic operations, which hinders the acquisition of the optimal solution[12].

Magic square is a mathematical problem that originated in China. Currently, it has a wide range of applications in artificial intelligence, experimental design, game theory, position analysis, and has gradually become an important direction in combinatorial mathematics research.

An N-order magic square possesses several special properties:

(1) The order of a magic square refers to the number of rows or columns in the square.

(2) The sum of the numbers in each row, column, or diagonal of the magic square is the same. This sum is called the magic constant. For an N-order magic square, the magic constant is given by the value $N\frac{1+N^2}{2}$.

When the number of operations in an automobile stamping workshop is N, the population can be constructed using $L = \lceil \sqrt{J_n} \rceil$, where J_n is the total number of operations for n workpieces. The constructed magic square can be rotated clockwise by 90 degrees each time to obtain new transformed magic squares.

For instance, suppose there are 12 operations in an automobile stamping workshop. A 4-order magic square can cover all operations. After creating a 4-order magic square, it can be rotated clockwise by 90 degrees successively, as shown in Figure 2.

By applying elementary row operations to the resulting magic square, a new transformed magic square can be obtained. The new magic square after the elementary row operations is shown in Figure 3.

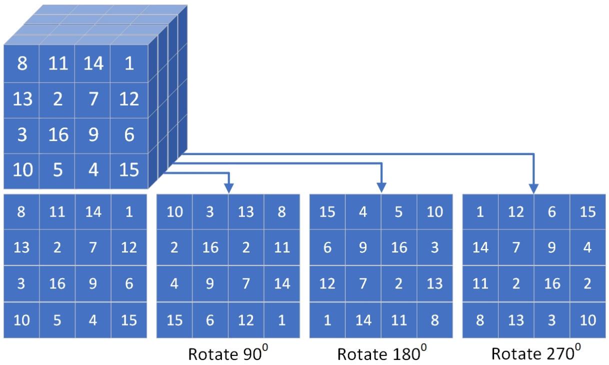

Figure 2: Production Flowchart of an Automotive Stamping Workshop

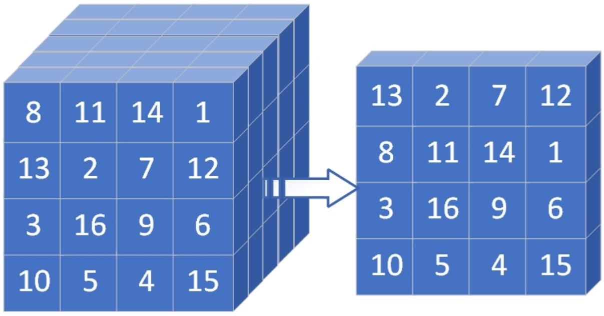

Figure 3: Production Flowchart of an Automotive Stamping Workshop

By flattening the data of each magic square obtained through row operations, and removing each value greater than the initial procedure's data, a new initial population will be generated, as shown in Figure 4.

When initializing the population in a genetic algorithm, setting the population size too large can result in excessive computation, leading to decreased algorithm efficiency. When using a magic square to initialize the population size, the generated population individuals are functions of the completed magic square. It can be approximately considered that the number of population individuals is related to the number of operations by the relationship described in formula 10.

$$\begin{cases} \frac{L(L-1)}{2}, & L < 20 \\ 200, & L \geq 20 \end{cases} \tag{10}$$

It can be verified that when the total number of operations is close to L^2, the initial population constructed using magic squares exhibits diversity.

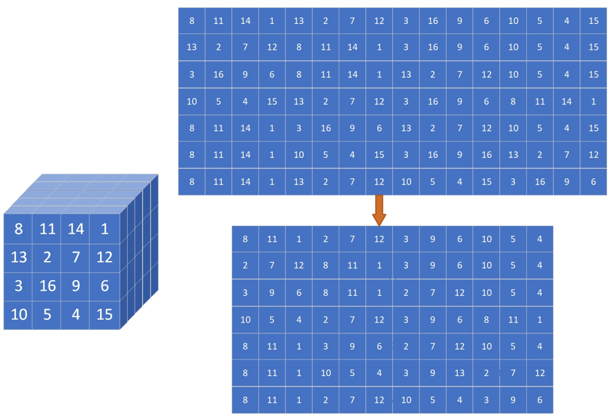

Figure 4: Production Flowchart of an Automotive Stamping Workshop

3.4 Selection Operation

The most common method for the selection operation is the roulette wheel method. However, although this method ensures that individuals with low fitness also have a chance to be selected, it can result in high-fitness individuals being eliminated, thereby potentially preventing the optimal individuals from being passed on to the next generation[8].

A suitable selection strategy has a significant impact on the convergence speed of the algorithm. To accelerate the convergence speed of the algorithm, improve search efficiency, and maintain individual diversity to avoid local convergence, the roulette wheel method can be optimized. By preserving the individual with the highest fitness value from the previous generation and directly passing it on to the next generation, the specific approach is as follows.

The individuals in the next generation population need to be generated based on the fitness of the individuals in the previous generation population using the roulette wheel selection method. The probability of selecting individual i is shown in formula 11.

$$P\left(V_k\right) = f(v_k)\sum_{i=1}^{N} f(v_i) \tag{11}$$

Where $f\left(v_k\right)$ is the fitness value of individual k . Individuals with higher fitness values have a greater chance of being selected. However, both high and low fitness individuals have the possibility of being passed on to the next generation, maintaining population diversity. Meanwhile, the individual with the highest fitness in each generation is directly retained for the next generation and does not participate in crossover or mutation operations, thereby maximizing the quality of the population.

3.5 Crossover Operation

The crossover operation allows the genes of excellent individuals in the population to be propagated and disseminated, thereby producing superior individuals within the population and driving the evolutionary process of the entire group[15]. For the scheduling problem in a stamping workshop, to better inherit the superior gene segments of the parent generation, we can use the method of exchanging gene segments after the crossover points of two parent chromosomes to produce offspring individuals. The specific steps are as follows:

(1) Determine the two parent chromosomes to be crossed based on the crossover probability of the parent generation and use a random function to generate the crossover positions (Pos) for the two parent chromosomes.

(2) Exchange the gene segments after the Pos positions of the two parent chromosomes and mark the duplicated gene operations in the two parent chromosomes after the exchange.

(3) Exchange the duplicated genes of the two parents, thereby generating offspring individuals.

Figure 5 shows the two offspring chromosomes generated after performing crossover on two randomly selected parent chromosomes.

3.6 Mutation Operation

The basic content of mutation operation is to make changes to certain gene values of individuals in the population. Mutation operation is random and can ensure the diversity of individuals in the population, but it cannot guarantee the generation of better individuals after mutation. In fact, many times it may even destroy the excellence of the previous generation individuals[10].

Inspired by the use of transgenic technology in biotechnology, where good gene fragments with desired traits are introduced into cells to induce mutation and generate breakthrough changes. After conducting a large number of simulation experiments on the stamping workshop scheduling model, the experimental results indicate that when several closely related process codes are arranged in sequence, the individual often exhibits better fitness. At this point, these closely related process codes can be defined as excellent gene segments. Numerous excellent gene segments constitute an excellent gene pool. During each mutation operation, the process is checked against the excellent gene pool to avoid

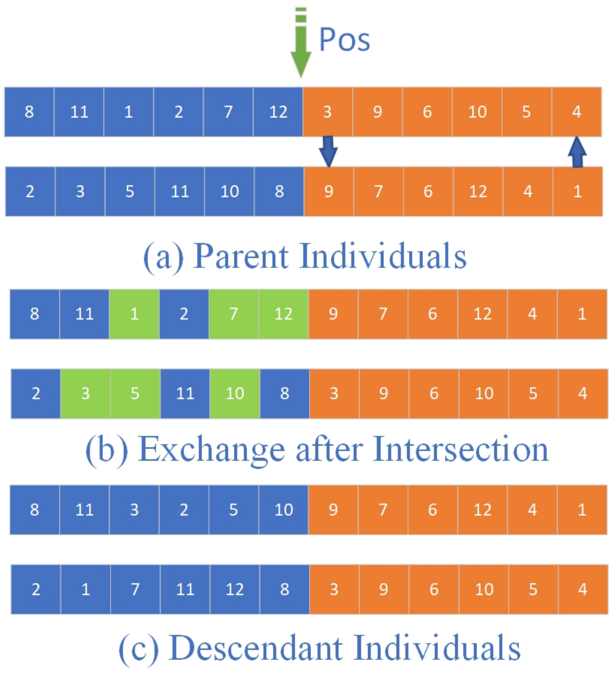

Figure 5: Production Flowchart of an Automotive Stamping Workshop

mutation in the excellent gene segments, thus ensuring that mutation only occurs in non-excellent gene segments.

3.7 Decode Operation

To describe the machining process of a specific operation for a workpiece, it is necessary to use the workpiece number, operation number, machine number, and processing time. This is encapsulated in the processing description matrix $D = (X, X_P, X_M, X_T)$, where D is individual elements respectively represent the sequence of workpiece codes, the corresponding sequence of operations for the workpiece codes, the sequence of machines, and the sequence of processing times[4].

For a chromosome X, the decoding steps are as follows:

(1) Slice the chromosome, converting the interchange encoding into operation encoding through slicing.

(2) Following the workpiece processing order of chromosome X', generate the corresponding set of processing operations, set of processing machine sequences, and set of processing time sequences for chromosome X' in reference to the workpiece operation collection.

(3) Define the matrix of start times for processing workpieces on machines, the current time array on machines, and the matrix of end times for processing workpieces on machines. If there are leftover operations on a machine, the current time array on that machine should be set to the leftover times from unfinished operations of the previous period.

(4) If a workpiece is being processed for the first time or if the current time of the machine scheduled to process the workpiece is greater than the completion time of the previous operation for that workpiece, then begin processing the current workpiece. Otherwise, the machine must wait until the completion of the previous operation before starting the current operation.

4 Experiment and Results

To evaluate the effectiveness of the improved genetic algorithm proposed in this paper for scheduling in an automotive stamping workshop, experimental tests were conducted using specific scheduling data from an automobile factory over the course of one day. The scheduling task list includes five stamping parts. Table 1 outlines the quantity produced of each stamping part, the operations in-

Table 1: Production task and related process parameters

Stamping parts	J1	J2		J3			J4				J5				
Batch quantity	700	500		600			500				500				
Operation	1	1	2	1	2	3	1	2	3	4	1	2	3	4	5
Processing time(s)	10	12	12	12	12	12	15	15	15	15	13	13	13	13	13
Mode change time(m)	30	30	30	30	30	30	30	30	30	30	30	30	30	30	30
Press force(tons)	600	600	400	600	400	400	600	400	400	400	600	400	400	400	400

Table 2: Table 2 Device Press Force

Device	M1	M2	M3	M4	M5
Press force(tons)	630	400	400	400	630

cluded for each type of stamping part, the production time for each operation, and the die change time required before producing each operation.

The production line equipment for processing this production task and their corresponding stamping forces are shown in the table 2. When selecting equipment for the operation, it is necessary to choose based on the magnitude of the stamping force. The equipment's rated stamping force must meet or exceed the stamping force required by the operation.

Using the improved genetic algorithm mentioned in this article to simulate and solve the scheduling data, we can obtain the Evolution Process shown in Figure 6, As can be seen from Figure 6, the population constructed by the method mentioned in this paper exhibits good diversity, and the convergence of the algorithm is also relatively ideal.

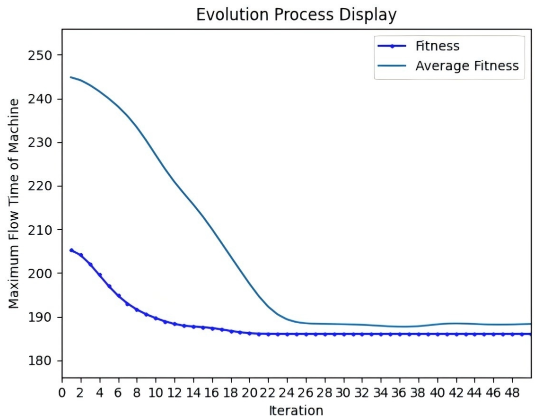

Figure 6: Production Flowchart of an Automotive Stamping Workshop

It can be determined that the algorithm's scheduling result for this production task is 26,700 seconds, and the scheduling outcome is quite satisfactory from the Figure 7.

The results of the scheduling by the algorithm in this paper indicate that both the equipment utilization and the balance among different pieces of equipment are relatively high as can be seen from Table 3.

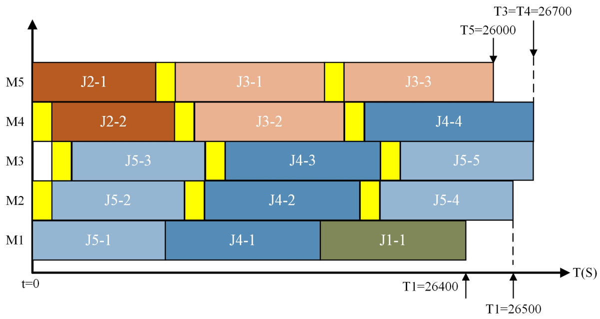

Figure 7: Production Flowchart of an Automotive Stamping Workshop

Table 3: Scheduling Result Data Analysise

	M1	M2	M3	M4	M5
Completion Time(s)	26400	26500	26700	26700	26000
Idle Time(s)	0	600	800	600	200
Utilization Rate	100%	98.35%	97%	97.75%	99.23%
Mean Deviation(s)	-60	40	240	240	-460

5 Conclusion

To address the intelligent scheduling issues in automotive stamping workshops, this paper improves the traditional genetic algorithm by introducing a magic square-based encoding design, which reduces the spatial complexity of the population's chromosomes, simplifies the encoding process of the genetic algorithm, and enhances its problem-solving capabilities. Additionally, modifications are made to the commonly used roulette wheel selection operation, allowing individuals with higher fitness to be retained in the next generation, ensuring the quality of the population.

Although the enhanced genetic algorithm proposed in this paper improves production efficiency and resource utilization, the inherent randomness of genetic algorithms means that the algorithm cannot always guarantee reaching the global optimum. Consequently, the scheduling plans derived from the algorithm require further optimization and enhancement in practical application. Future research will continue to focus on the scheduling issues in stamping workshops, integrating the strengths of other algorithms, and conducting dynamic scheduling research within workshops to better address scheduling challenges.

Funding

This paper is sourced from 2024 10th International Conference on Computers Communications and Control

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

References

[1] Bektas T. (2004). The multiple traveling salesman problem: an overview of formulations and solution procedures, *Omega*, 34(3), 209–219, 2004.

[2] Cheng R.W.; Mitsuo Gen.; Yasuhiro Tsujimura. (1999). A Tutorial Survey of Job-shop Scheduling Problems Using Genetic Algorithms Part II: Hybird Genetic Search Strategies, *Computers and Industrial Engineering*, 36(2), 343–364, 1999.

[3] Das, S.; Suganthan, P.N. (2011). Differential evolution: A survey of the state-of-the-art, *IEEE Transactions on Evolutionary Computation*, 15(1), 4–31, 2011.

[4] D.N. Zhou.; V Cherkassky.; T.R. Baldwin. (1990). Scaling Neural Network for Job-shop Scheduling, *International Joint Conference on Neural Networks*, USA: IEEE, 889-894, 1990.

[5] Gao K.Z.; Gao Z.G.; Zhang L. (2019). A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems, *IEEE/CAA Journal of Automatica Sinica*, 6(4), 904–916, 2019.

[6] G Buxey. (1989). Production Scheduling: Practice and Theory, *European Journal of Operational Research*, 39, 17–311, 1989.

[7] H.B. Yu.; W Liang. (2001). Neural Network and Genetic Algorithm based Hybrid Approach to Expanded Job-shop Scheduling, *Computers & Industrial Engineering*, 39(3-4), 337-356, 2001.

[8] H Jeong.; Jinwoo Park.; R C Leachman. (1999). A Batch Splitting Method for A Job Shop Scheduling Problem in an MRP Environment, *International Journal of Production Research*, 37(15), 3583–3598, 1999.

[9] I Sabuncuolu.; B Gurgun. (1996). A Neural Network Model for Scheduling Problems, *European Journal of Operational Research*, 93(2), 288-299, 1996.

[10] J B Lasserre. (1992). An Integrated Model for Job-shop Planning and Scheduling, *Management Science*, 38(8), 1201-1211, 1992.

[11] Jiang T H.; Deng G L. (2018). Optimizing the low-carbon flexible job shop scheduling problem considering energy consumption, *IEEE Access*, 6, 46346-46355, 2018.

[12] JK Lenstra.; AHG Kan.; P Brucker. (1977). Complexity of Machine Scheduling Problems, *Studies in Integer Programming*, 343-361, 1977.

[13] J M van Laarhoven.; HL Aarts. (1992). Jan Karel Lenstra. Job Shop Scheduling by Simulated Annealing, *Operations Research*, 40(1), 113-125, 1992.

[14] Li K.; Deb K.; Zhang Q. (2019). An evolutionary many-objective optimization algorithm based on dominance and decomposition, *IEEE Transactions on Evolutionary Computation*, 23(4), 605-617, 2019.

[15] Li K.; Zhang Q. (2019). An efficient and robust evolutionary algorithm for real-parameter optimization, *IEEE Transactions on Evolutionary Computation*, 13(3), 451-466, 2019.

[16] PHU-ANG A. (2021). An improve artificial immune algorithm for solving the travelling salesman problem, *Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunication Engineering*, 261-264, 2021.

[17] S G Ponnambalam.; N Jawahar.; P Aravindan. (1999). A simulated Annealing Algorithm for Job Shop Scheduling, *Production Planning & Control*, 10(8), 767-777, 1999.

[18] TANG K.; Liu S.; YANG P. (2021). Few-shots parallel algorithm portfolio construction via coevolution, *IEEE Transactions on Evolutionary Computation*, 25(3), 595-607, 2021.

[19] Wang Y.; Cai Z.; Zhang Q (2019). Differential evolution with dynamic stochastic selection for constrained optimization, *IEEE Transactions on Evolutionary Computation*, 14(4), 678-696, 2019.

[20] Wang Y H.; FU L Q.; SU Y Q. (2018). Genetic algorithm in flexible workshop scheduling based on multi-objective optimization, *Journal of Interdisciplinary Mathematics*, 21(5), 1249-1254, 2018.

[21] Wang Y.; Yang O.; Wang S N. (2019). A solution to single-macgine inverse job-Shop scheduling problem, *Journal Citation Reports*, 18(2), 335-342, 2019.

[22] WU C.; FU X, PEI. (2021). A novel sparrow search algorithm for the traveling salesman problem, *IEEE Access*, 9, 153456, 2021.

[23] YUAN F.; SUN H.; Kang L. (2022). Joint optimization of train scheduling and dynamic passenger flow control strategy with headway-dependent demand, *Transportmetrica*, 10(1), 627-651, 2022.

[24] Zhang J.; Ding G.; Zou Y. (2019). Review of job shop scheduling research and its new perspectives under Industry 4.0, *Journal of Intelligent Manufacturing*, 30(4), 1809-1830, 2019.

[25] Zhang R.; Chiong R. (2016). Solving the energy-efficient job shop scheduling problem: a multi-objective genetic algorithm with enhanced local search for minimizing the total weighted tardiness and total energy consumption, *Journal of Cleaner Production,*, 112, 3361-3375, 2016.

Cite this paper as:

Zhiyuan Li.; Desheng Wu.; Cheng He.; Xiaofeng Gan.; Peichun Chen. (2025). Intelligent Scheduling of Automotive Stamping Workshops Based on an Enhanced Genetic Algorithm, *International Journal of Computers Communications & Control*, 20(1), 6874, 2025.
https://doi.org/10.15837/ijccc.2025.1.6874