Research on PBS buffer scheduling strategy problems based on genetic algorithms

Yaodan Zhang [*], *Yuanbo* Shi, *Jianjun* Xu, and *Zhiye* Wei

School of Artificial Intelligence and Software, Liaoning Petrochemical University Fushun 130001, China

Abstract. In automotive manufacturing, production constraints between the painting and assembly workshops often disrupt the vehicle production sequence. To solve this, a Painted Body Store (PBS) buffer was introduced. This adjusts the sequence from painting to match the assembly workshop's needs.A model was created using rules to match vehicle and channel attributes, forming an initial exit sequence for painted vehicles. A genetic algorithm optimized vehicle selection from the PBS, adhering to constraints.Analysis showed this method improved efficiency by nearly 40% compared to traditional linear scheduling. This highlights its effectiveness in optimizing automotive manufacturing.

1 Introduction

The automotive sector is pivotal for economic growth, with China leading in production and sales, boasting a vehicle parc of 435 million by the end of 2023, indicating market potential. New energy vehicles (NEVs) are on the rise, reaching 20 million by 2021, supported by technology and policy.

Automotive manufacturing involves welding, painting, and assembly lines, each with unique demands. The Painted Body Store (PBS) bridges painting and assembly, aligning production sequences and offering flexibility for market shifts.

Genetic algorithms optimize production scheduling, enhancing control, efficiency, and cost-effectiveness. This improves market responsiveness and supports economic progress.

Researchers have made strides in optimizing scheduling across industries, using genetic algorithms and other techniques for enhanced efficiency.

These studies collectively advance production scheduling, offering robust solutions to logistical challenges.

2 Model

The PBS workflow includes two processes: body inbound and body outbound. Body inbound involves transferring bodies from the paint shop to the PBS buffer storage lanes using the vehicle receiving traverse. Body outbound involves retrieving bodies from the storage lanes with the vehicle delivery traverse.

[*] Corresponding author: zhangyaodan0519@163.com

The information entering the scheduling area includes vehicle type, powertrain, and drive information. Vehicle type consists of two categories: A and B, but it does not affect the objective function, so it is not recorded. Powertrain information includes fuel and hybrid types. Drive information includes two-wheel drive (2WD) and four-wheel drive (4WD).

A body's attribute is determined by its powertrain type—fuel or hybrid. When a body enters a lane, the lane's attribute updates to match the last vehicle's attribute in that lane if it is not empty.

Once lane attributes are set, their priorities must be established. This involves determining which attribute is currently most important. For example, if the powertrain information has the highest priority, it indicates that powertrain information is the most important, followed by drive information, with other attributes being less important.

The matching model based on body information is as follows:

$$\max f(i) = \begin{cases} \sum_{k=1}^{n} \left(s(k) \times 2^{n-k}\right), \text{Lane } i \text{ is not empty} \\ 2^{n-1} - 1, \text{Lane } i \text{ is empty} \end{cases} \tag{1}$$

where:

- i represents the lane index.
- f represents the matching degree between the body and lane i.
- n represents the number of rules, where $n > 2$.
- k represents the priority of the rule, where $k = 1, 2, \dots, n$. A higher k indicates lower priority.
- $s(k)$ represents the attribute consistency indicator.

$$s(k) = \begin{cases} 1, v_{i,k} = c_k \\ 0, v_{i,k} \neq c_k \end{cases} \tag{2}$$

where:

- y denotes the attribute of lane i with priority k.
- c denotes the attribute of the body with priority k.
- $s(k)$ evaluates whether the attribute of the body matches the attribute of the lane. If they match, it equals 1; otherwise, it equals 0.

The primary factors influencing the final assembly workshop are variations in powertrain type and drive mode. Therefore, different weights are assigned to these factors: 0.4 for powertrain information, 0.3 for drive mode, 0.2 for return lane usage frequency, and 0.1 for total scheduling time. The objective of adjusting the body sequence in the linear PBS buffer area with return lanes is to maximize the evaluation function of the outbound queue in the buffer area. The mathematical model is represented as:

$$value = (100 - x_1) \times 0.4 + (100 - x_2) \times 0.3 + (100 - x_3) \times 0.2 + (100 - x_4) \times 0.1 \tag{3}$$

where x_1 represents the score of the powertrain sorting, x_2 represents the drive sorting score, x_3 represents the return lane usage score, x_4 represents the scheduling time score, and 0.4, 0.3, 0.2, and 0.1 are the weight coefficients for each indicator. $value$ represents the total score.

3 Algorithms

1. Body Information Matching Model:

Attributes: Refers to the vehicle's powertrain type and drive mode, including hybrid and fuel types for powertrain, and two-wheel drive and four-wheel drive for drive mode.

Characteristics: Comprised of the two attributes of the vehicle, denoted as (Vp, Vd).

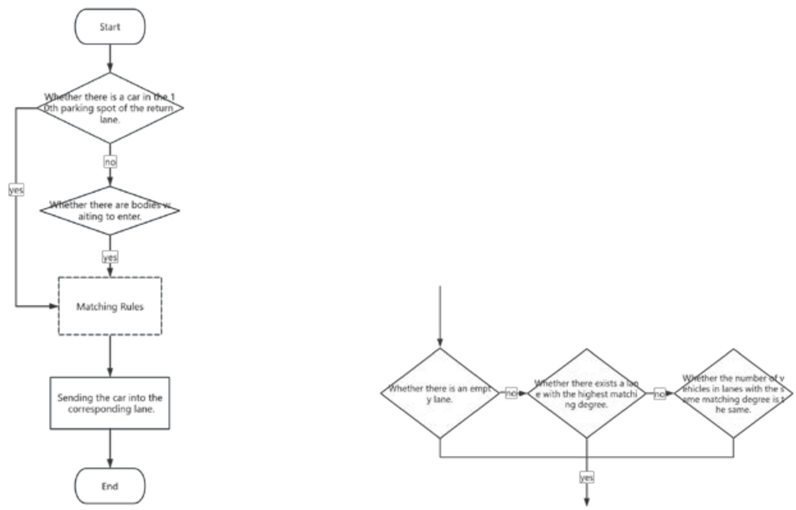
Consistency: Indicates whether the vehicle's attributes match those of the inbound lane.

Matching Degree: Refers to the degree of matching between the vehicle and a particular inbound lane.

2. Body Inbound Strategy:

Step 1: If an empty lane exists, the vehicle is sent directly to it. Step 2: Assign the characteristics of the last vehicle to enter the lane to the incoming vehicle. Step 3: Calculate the matching degree based on the current vehicle's characteristics and the lane's characteristics. Step 4: Select the lane with the highest matching degree for delivery. If degrees are the same, choose the lane with fewer vehicles. If equal, consider the lane closest to the vehicle receiving traverse. Step 5: Update the lane's attributes.

This process is illustrated in the body inbound process diagram.



(a) Inventory flow chart (b) Matching rules

Fig. 1. Vehicle entry flowchart.

In cases where a reverse loading operation is necessary in the PBS, shifting a body from the 1st parking spot of an inbound lane to the 10th parking spot of the return lane triggers changes to the entire inbound sequence. To address this, a genetic algorithm is employed to dynamically adjust the inbound sequence, aiming to meet constraints and optimize workshop production efficiency.

Table 1. The coding results for different data categories.

(a) Vehicle Power		(b) Drive Type		(c) Combined Coding Result			
Power Type	Code	Drive Type	Code	a_0	a_1	b_0	b_1
Fuel Vehicle	0	Two-Wheel Drive	a	A	B	C	D
Hybrid Vehicle	1	Four-Wheel Drive	b				

3.1 Encoding of data

Encode the vehicles during the body inbound process.

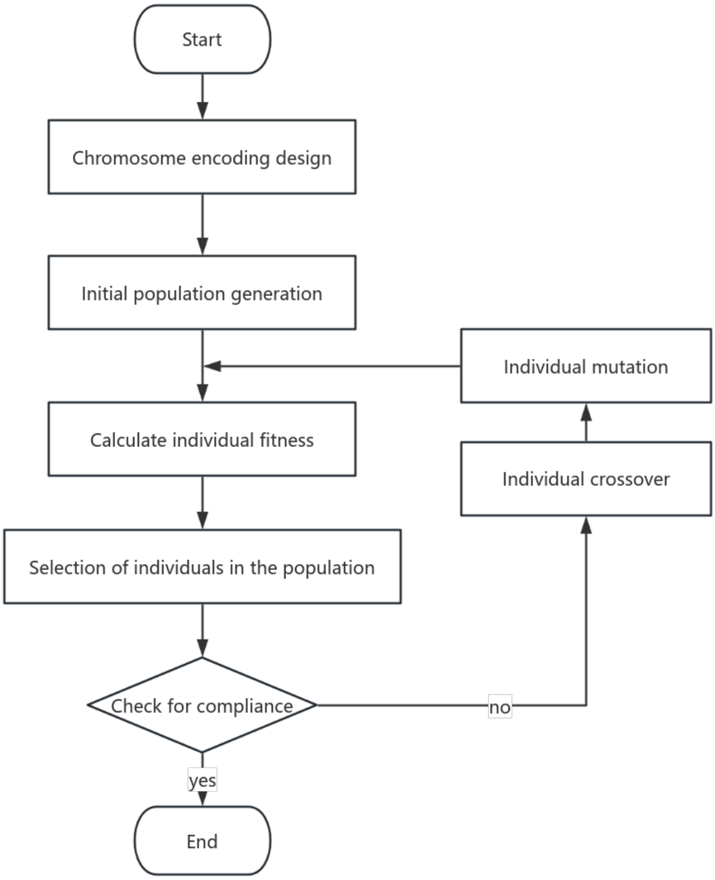


Fig. 2. Genetic algorithm flowchart.

When vehicle positions need rearrangement due to reverse loading, and the overall inbound sequence is affected, a genetic algorithm (GA) is considered to meet downstream workshop requirements. Vehicles are encoded with parking spots as genes of a chromosome. Multiple vehicles in one lane form one chromosome, representing a specific outbound plan, as shown in Table 2. The encoding method is based on vehicle location, where each gene represents the lane number.

Table 2. Scheduling example.

	10	9	8	7	6	5	4	3	2	1
ELane 6	B	A	A	B	A	B	A	A	A	A
ELane 5	C	C	D	C	D	C	C	C	C	C
RLane		D								
ELane 4	D	D	D	C	C	D	D	D		
ELane 3		B	B	A	A	B	B	B	B	B
ELane 2		B	B	B	B	A	A	A	A	A
ELane 1		C	D	C	D	C	C	D	C	C

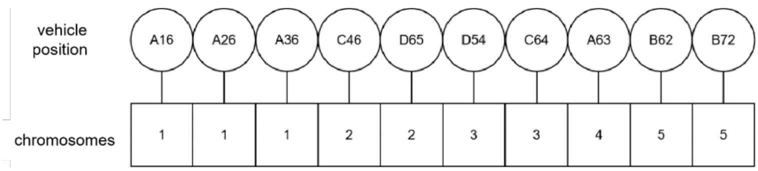


Fig. 3. Chromosome encoding schematic.

3.2 Generating Initial Population

Step 1: Form a set with vehicles in parking spot 1 and compare the next vehicle entering the buffer area with them. If there's a matching vehicle in the set, randomly select one to exit parking spot 1. Step 2: If no matching vehicle is found, randomly select one vehicle to exit parking spot 1. Step 3: Repeat steps 1 to 3 until all vehicles successfully exit. Each passing scheme on each lane is a chromosome, and each vehicle is a gene.

3.3 Designing fitness function

After generating the initial population, a fitness function is required to assess the fitness of individuals. This function, also known as the evaluation function, primarily relies on the objective function. Individuals with higher fitness in the population are more likely to survive, increasing the probability of passing their traits to the next generation.

3.4 Selection operator

The selection operator utilizes the fitness function to choose individuals from the population for reproduction, employing the fitness proportionate selection operator in this paper. This operator functions based on the principle that a uniformly distributed random number $r \in [0, F)$is utilized to minimize specified parameters, determining the selection of individuals.

$$i \leftarrow \underset{n\in[0,N)}{arg\,min}\left\{r < \sum_{i=0}^{n} f_i\right\} \tag{4}$$

where $N$is the number of individuals, and f_i is the fitness of the i-th individual.

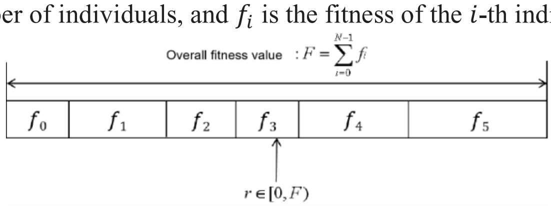


Fig. 4. Overall fitness.

3.5 Crossover operator

The crossover operation in genetic algorithms involves exchanging parts of genes between paired chromosomes randomly to generate new individuals. This paper utilizes a two-point crossover method, where two crossover points are randomly selected on the gene sequence, and gene segments between these points are exchanged. The operation is illustrated in Fig. 5.

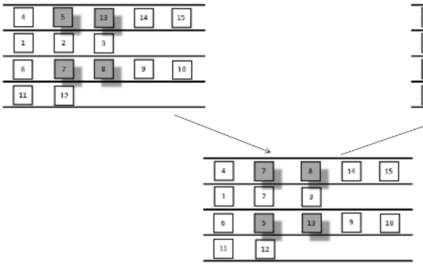


Fig. 5. Crossover operator.

3.6 Mutation operator

In genetic algorithms, individual mutation operation involves replacing the gene value at a specific gene locus in the individual chromosome encoding string with one of the remaining allele genes at that locus to create a new individual. The mutation operator can enhance the local search capability of the genetic algorithm. This paper adopts single-point mutation, which is relatively simple. Single-point mutation randomly selects a gene for mutation in the chromosome's gene encoding. The mutation operation is illustrated in Fig. 6.

Single-point mutation is relatively simple; it randomly selects one gene to change in the gene encoding of a random individual.

3.7 Algorithm termination

The algorithm terminates under the following conditions:

a) The maximum number of iterations is reached.

b) The optimal individual remains unchanged for consecutive generations (this paper takes 10 generations).

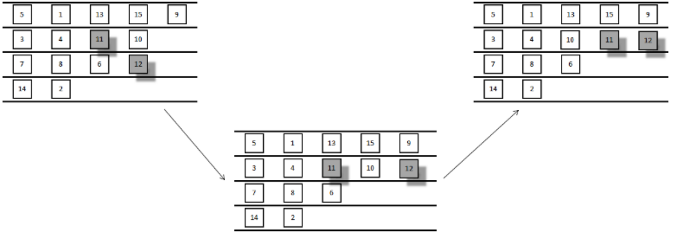
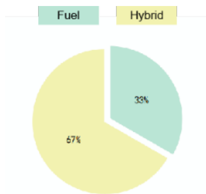


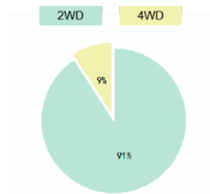
Fig. 6. Single-point mutation.

4 Experimental results

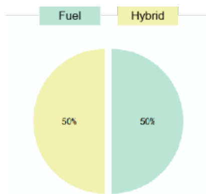
The vehicle information used in the experiment is divided into two types. The first type of data consists of normal data after desensitization of production data, while the other type is adjusted data used to test the adaptability of the model and algorithms. The data information is as follows:



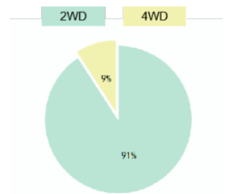
(a) Proportion of Power Types 1



(b) Proportion of drive types 1



(a) Proportion of Power Types 2



(b) Proportion of drive types 2

Fig. 7. Vehicle information statistics graph.

In actual production, some vehicles may need re-spraying, so the model must account for this. Although the vehicle information provided here is balanced, real production may involve varying entry sequences, vehicle types, or even 300 identical vehicles, significantly reducing model efficiency. Thus, preprocessing the data and integrating other algorithms like simulated annealing may be necessary.

```
100%|██████████| 1000/1000 [06:28<00:00,  2.73it/s]
36.679
[4 2 4 3 3 6 1 4 3 4 5 5 3 2 1 3 2 5 4 3 6 2 4 5 2 3 1 1 3 1 4 4 1 5 1 3 3
 5 3 3 4 3 2 1 4 4 6 3 2 3 3 3 2 4 1 5 2 2 5 2 3 4 1 6 2 3 1 1 4 5 6 3 5 2
 6 5 4 3 1 3 6 1 5 2 6 3 3 2 6 3 4 6 4 5 2 5 2 4 5 3 5 3 4 2 5 4 5 2 3 4 5
 6 2 4 4 2 4 6 1 6 3 6 2 2 6 3 2 5 3 3 6 6 3 4 6 4 4 5 2 3 2 2 2 6 6 5 5 3
 5 1 4 2 4 2 4 5 2 5 4 1 3 3 1 1 3 2 2 2 4 5 6 4 3 1 4 2 3 1 3 4 1 1 2 6 1
 5 3 5 3 6 6 5 2 4 4 1 3 6 3 3 4 4 4 3 5 1 2 3 3 5 4 2 2 3 6 3 2 6 4 5 4 5
 3 2 6 4 4 5 5 3 5 6 5 6 6 3 2 1 1 2 1 6 4 6 5 3 4 1 3 2 4 1 3 6 5 6 2 5 1
 3 6 4 5 5 6 2 5 2 2 1 1 5 4 6 3 3 3 4 3 6 6 3 5 6 1 2 4 6 2 2 3 1 3 1 6 5
 2 3 6 4 4 3 5 1 3 5 4 3 4 4 6 6 5 4 5 3 1 3]

Process finished with exit code 0
```

Fig. 8. The optimal solution under fully constrained conditions.

Using the established model and genetic algorithm adjustment strategy, data testing is performed with the following parameters: population size $N = 50$, number of iterations $T = 1000$, and mutation rate$= 0.1$. The obtained solution is as follows:

5 Conclusion

This paper establishes a vehicle entry model using a genetic algorithm to optimize PBS objectives. Genetic algorithms are robust and capable of finding global optimal solutions, making them suitable for complex optimization problems. However, they have slow convergence and poor local search ability. Combining genetic algorithms with other heuristic algorithms, such as simulated annealing or particle swarm optimization, can enhance speed and accuracy. Additionally, increasing the use of return lanes can reduce algorithm calls and improve overall model efficiency.

References

1. Ministry of Public Security, "National Motor Vehicle Ownership Reaches 435 Million, with 523 Million Drivers and New Energy Vehicle Ownership Exceeding 20 Million," Jan. 2024. [Online]. Available: https://www.gov.cn/lianbo/bumen/202401/content_6925362.htm
2. K. Ding, Y. Xu, H. Guan, S. Li, and B. Liwei, "Medium and Long-Term Ship Scheduling Optimization Based on Greedy-Genetic Algorithm," *Journal of Wuhan University of Technology (Transportation Science & Engineering Edition)*, vol. 47, no. 4, pp. 770–774, 2023.
3. M. Yavuz, "Iterated Beam Search for the Combined Car Sequencing and Level Scheduling Problem," *International Journal of Production Research*, vol. 51, no. 12, pp. 3698–3718, Jun. 2013, doi: 10.1080/00207543.2013.765068.
4. H. Zohali, B. Naderi, M. Mohammadi, and V. Roshanaei, "Reformulation, Linearization, and a Hybrid Iterated Local Search Algorithm for Economic Lot-Sizing and Sequencing in Hybrid Flow Shop Problems," *Computers & Operations Research*, vol. 104, pp. 127–138, Apr. 2019, doi: 10.1016/j.cor.2018.12.008.
5. Z. Shen, Q. Tang, T. Huang, T. Xiong, H. Y. K. Hu, and Y. Li, "Dynamic Production Scheduling Modeling and Multi-objective Optimization for Automobile Mixed-Model Production," in *Intelligent Computing and Internet of Things*, vol. 924, K. Li, M. Fei, D. Du, Z. Yang, and D. Yang, Eds., in *Communications in Computer and Information Science*, vol. 924. , Singapore: Springer Singapore, 2018, pp. 25–33. doi: 10.1007/978-981-13-2384-3_3.
6. A. Jalilvand-Nejad and P. Fattahi, "A Mathematical Model and Genetic Algorithm to Cyclic Flexible Job Shop Scheduling Problem," *Journal of Intelligent Manufacturing*, vol. 26, no. 6, pp. 1085–1098, Dec. 2015, doi: 10.1007/s10845-013-0841-z.
7. N. Boysen and M. Zenker, "A Decomposition Approach for the Car Resequencing Problem with Selectivity Banks," *Computers & Operations Research*, vol. 40, no. 1, pp. 98–108, Jan. 2013, doi: 10.1016/j.cor.2012.05.012.
8. T. Hu, "Research and Application of PBS Buffer Body Routing and Scheduling Strategy," *Master's Thesis*, Hefei University of Technology, 2017.
9. Z. Shenyu, "Production Scheduling and Dispatching Problem of Automobile Mixed Production Line," *Master's Thesis*, Chongqing University, 2021.
10. Y. Li, "Research on Multi-level Flow Shop Scheduling and Dispatching Problem for Automobile Mixed Flow Production," *Master's Thesis*, Chongqing University, 2022.
11. Z. Tianzhipeng, "Research and Application of Stamping Workshop Scheduling and Assembly Line Sorting Method for Automobile Mixed Flow Production," *Doctoral Dissertation*, Huazhong University of Science and Technology, 2017.
12. Z. Chenzhengmao, "Research on Multi-Workshop Correlated Sorting Based on Sorting Buffer," *Master's Thesis*, Huazhong University of Science and Technology, 2010.
13. G. Huang, "Planning and Execution Process Management of Mixed-flow Assembly Production," *Doctoral Dissertation*, Huazhong University of Science and Technology, 2009.