EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

Optimization of Circular-Rail RGV Scheduling in a Tobacco Warehouse

Dingke Shi[1], Zhihui Ye[1], Nanzhe Ding[1], Jie Gao[1,*], Chao Cheng[1], Wenwen Lin[2]

[1] China Tobacco Zhejiang Industrial Co., Ltd., Hangzhou 310009, China
[2] Faculty of Mechanical Engineering & Intelligent manufacturing, Ningbo University, Ningbo 315211, Zhejiang, China

Abstract

Continuous operation in tobacco warehouses places high requirements on circular-rail rail guided vehicle (RGV) dispatching, because task concentration, track interference, and equipment abnormalities may quickly cause vehicle waiting and material-flow delay. To improve the response capability of this system, this study develops a hybrid scheduling method based on an improved genetic algorithm and simulated annealing. According to the operating characteristics of the circular rail, a scheduling model is formulated to minimize the overall task completion time while considering rail operation rules and composite task requirements. In the solution process, the simulated annealing temperature is used to adjust parent selection and mutation behavior, so that the algorithm can maintain search diversity in the early stage and improve convergence in the later stage. Case results show that the proposed method reduces the average maximum completion time from 150.2 s to 143.4 s and improves the best result from 145.7 s to 139.6 s compared with the conventional genetic algorithm. These results indicate that the proposed method can improve RGV response efficiency and provide decision support for stable warehouse logistics under disturbances such as task fluctuation and path conflict.

Keywords: rail guided vehicle scheduling, circular-rail, genetic algorithm, task optimization

Received on 11 June 2026, accepted on 09 July 2026, published on 15 July 2026

doi: 10.4108/eetsis.13463

1. Introduction

Intelligent manufacturing and flexible production have made production systems more responsive, but they have also increased their exposure to operational uncertainty. Task surges, equipment faults, logistics congestion, and temporary production interruptions can all interrupt the coordination between production and material handling. In this context, manufacturing resilience refers to the ability of a production system to preserve essential functions during disturbances and to return to a stable operating state after abnormal events occur [1-3]. Recent research further suggests that resilience is not limited to disturbance resistance. It also depends on whether material flow, information flow, and production flow can be reorganized through scheduling adjustment, resource reconfiguration, and logistics recovery [4]. For continuous-production enterprises, internal logistics is therefore not only a material-transfer function, but also a key factor affecting production rhythm and line continuity.

In tobacco manufacturing, the tobacco warehouse connects storage, material transfer, and production supply. Its operating state directly influences the stability of downstream production. As tobacco factories place higher requirements on production efficiency, delivery accuracy, and operational reliability, labor-intensive warehouse operations are increasingly unable to meet the needs of intelligent factories. Problems such as low handling efficiency, high error probability, and heavy labor demand become more prominent in continuous operation. To improve automation and controllability, tobacco enterprises have introduced automated logistics equipment into warehouse operations. Among these solutions, circular-rail rail guided vehicle (RGV) systems are suitable for tobacco-box handling because their routes are fixed, their movements are predictable, and

* Corresponding author. Email: Jie Gao (499766199@qq.com)

their transportation process can be controlled through scheduling.

In a cigarette factory, the warehouse RGV system is located between storage and production supply. Once several handling tasks arrive within a short time, the circular rail can easily become crowded because the vehicles move in the same direction and cannot bypass each other freely. If one RGV waits too long at a shared track segment, the following vehicles may also be delayed, and the delay may finally affect the material supply to the production line.

For this reason, this paper uses the maximum task completion time as the main scheduling index. This index records the finishing time of the last task in a task batch. When this value is reduced, the whole batch can be cleared earlier, and the waiting time on the circular rail is also likely to decrease. Although this index does not describe manufacturing resilience completely, it is directly related to the recovery of warehouse transportation after task concentration or temporary disturbance.

Previous work on RGV and AGV scheduling can be roughly divided into three lines. The first line focuses on vehicle-task assignment in relatively fixed production settings. For example, some RGV studies used greedy rules, heuristic search, or dynamic scheduling models to assign vehicles to machining or transfer tasks [5, 6]. These methods are easy to implement, but they usually simplify the interaction between vehicles on the track.

The second line studies AGV scheduling together with routing decisions. Digital-twin-based methods have been used to monitor system uncertainty and coordinate physical and virtual operations [7], while hierarchical path-planning algorithms have been proposed to reduce travel distance and vehicle conflicts [8]. These studies are useful for multi-vehicle systems, but many of them assume more flexible path choices than those available in a closed circular rail.

More recent studies have added practical production constraints to the scheduling model. Battery capacity, unloading inspection, new task insertion, machine failure, processing-time variation, and vehicle state changes have been considered in different AGV/RGV scheduling problems [9-12]. Other studies treat scheduling as an integrated production-logistics problem and solve it by mixed-integer programming, cooperative genetic algorithms, ant colony optimization, reinforcement learning, or other metaheuristic methods [13-16]. These studies show that vehicle scheduling is affected not only by route length, but also by task sequence, vehicle state, path conflict, and production rhythm.

RGV scheduling has also been studied in manufacturing scenarios such as CNC equipment service, casting workshops, flexible transfer systems, and multi-objective scheduling problems [17-21]. In addition, landscape-based strategies have been used to improve the search quality of RGV dynamic scheduling problems [22]. However, most of these studies are not designed for tobacco warehouse logistics. In the circular-rail system considered in this paper, vehicles move in one direction, stations are densely arranged, and composite tobacco-box transportation tasks must follow fixed material-flow paths. These features make it necessary to build a model that records station-passing time and vehicle interference more explicitly.

Compared with general AGV/RGV scheduling problems, the circular-rail RGV system in a tobacco warehouse has more specific operating restrictions. The vehicles move along a closed track in one direction, backward movement is not permitted, and overtaking is highly limited by the track structure. In addition, loading and unloading stations are densely distributed along the rail, so the departure time of one vehicle may directly influence the passing time of following vehicles. Tobacco-box handling also involves composite transportation paths across different processing areas. These features make the problem different from ordinary route-selection or vehicle-assignment problems and require a scheduling model that explicitly records station-passing time and vehicle interference on the circular rail.

Metaheuristic algorithms are often used when the scheduling problem is too large or too constrained to be solved by simple dispatching rules. Genetic algorithms are suitable for this type of problem because they search through different task sequences and vehicle assignments by population evolution [23]. Simulated annealing is useful in another way: it can accept a worse solution with a certain probability, which gives the search a chance to move out of a local optimum [24]. For this reason, several studies have combined genetic search with local or stochastic search strategies to improve solution diversity and convergence behavior [25]. Multi-agent deep reinforcement learning has also been introduced into synchronized production and AGV scheduling, showing another possible direction for production-logistics coordination [26]. These studies suggest that combining different search mechanisms may be useful for RGV scheduling, especially when vehicle conflicts and task sequences are strongly coupled.

However, the existing methods still leave some gaps for the tobacco warehouse system studied in this paper. Many studies use transportation time, task completion time, or equipment utilization as the main performance indicators, but they seldom discuss how RGV scheduling affects logistics recovery and production continuity. In addition, some RGV models mainly describe pickup and unloading operations. For a circular rail, this is not enough, because the time at which a vehicle passes each intermediate station may also affect the movement of the following vehicles. In actual tobacco warehouse operation, tasks may arrive intensively, vehicles may wait on shared track segments, and the system may need to recover after equipment abnormalities or temporary production interruption. These features require a model that describes the station-passing process more explicitly.

Based on these considerations, this paper studies the scheduling problem of a circular-rail RGV system in a tobacco warehouse. The model is built according to the actual movement rule of the circular rail. It records the passing time of each RGV at each station and then calculates the task completion time by combining passing time and operation time. To solve the model, an improved genetic algorithm is combined with simulated annealing. The simulated annealing mechanism is used in parent selection and mutation adjustment so that the search can keep more diversity in the early stage and become more stable in the later stage. The purpose is to reduce the maximum task completion time and improve the response capability of the warehouse transportation system under concentrated tasks or temporary disturbances.

2. Problem Description of Circular-Rail RGV Scheduling in a Tobacco Warehouse

This paper mainly focuses on modeling and solving the scheduling problem of a circular-rail RGV system in a tobacco warehouse. As shown in Fig. 1, the layout of the circular-rail RGV system in the tobacco warehouse includes 16 loading stations {U|1, 3, 4, 5, ⋯, u} and 32 unloading stations {R|2, 7, 9, 11, ⋯, r}. The tobacco boxes in the automated high-bay warehouse are classified into full boxes and empty boxes, which follow different logistics paths.

When the automated high-bay warehouse outputs a full box, the box is transported by an RGV to Area 2 for blending. After the cut tobacco in the box is discharged, the box is conveyed by the transport system behind Area 2 to Area 3 for sorting. After sorting is completed, the box is transported by an RGV to Area 6 for storage. When the automated high-bay warehouse outputs an empty box, the box is transported by an RGV to Area 4 for cleaning. It is then conveyed by the transport system behind Area 4 to Area 5 for packing, and finally transported by an RGV to Area 6 for storage.

According to the material-flow diagram of tobacco boxes, the tasks performed by RGVs can be divided into four types: Task A, Area 1 → Area 2; Task B, Area 1 → Area 4; Task C, Area 3 → Area 6; and Task D, Area 5 → Area 6. Each task can only be executed by one RGV. Owing to the distribution of the areas, RGVs can also perform composite operations, including:

(1) from Area 1 to Area 2, then to Area 3, and finally to Area 6;

(2) from Area 1 to Area 2, then to Area 5, and finally to Area 6;

(3) from Area 1 to Area 4, then to Area 5, and finally to Area 6.

The main scheduling objective of the circular-rail RGV system in the tobacco warehouse is to complete cargo transportation within the shortest possible time.

To solve the RGV scheduling problem, the following constraints are considered:

(1) The circular-rail RGVs travel on the closed track in a fixed order, namely clockwise, and backward movement is not allowed.

(2) A single RGV can independently complete one task.

(3) At the beginning of operation, all RGVs are on standby at the origin point. This assumption corresponds to the common start-up state of a planned scheduling cycle and provides a unified initial condition for comparing different dispatching schemes.

In addition to the above constraints, the following assumptions are made:

(1) Unexpected emergencies during transportation are not considered.

(2) All prerequisites for transportation tasks are assumed to be ready, and an RGV can start the task immediately upon arrival.

(3) The time required for an RGV to pass through a curve is 2 s. To simplify distance calculation, the time spent passing through a curve is approximately converted into the travel time on a straight track over the same duration.

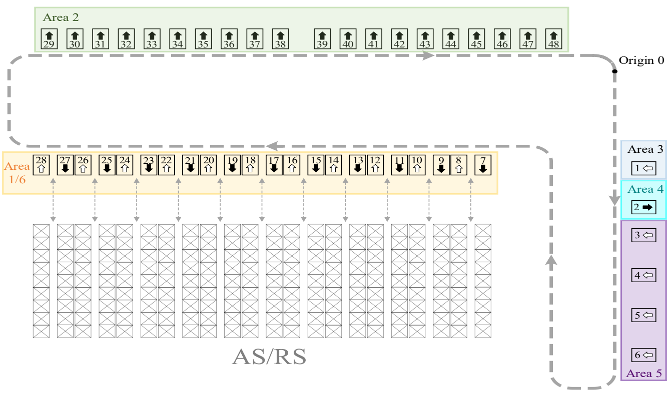

Figure 1. Layout of the circular-rail RGV system in the tobacco warehouse

3. RGV Scheduling Model

For clarity, the main notation used in the scheduling model is summarized below.

Table 1. Symbol Table

Symbol	Meaning
i	Index of RGV, $i = 1, 2, ..., I$
j	Index of transportation task, $j = 1, 2, ..., J$
q	Circuit or task-order index of an RGV
e	Index of station on the circular rail
I	Total number of RGVs
J	Total number of tasks
C_j	Completion time of task j
x_{ij}	Decision variable; $x_{ij} = 1$ if task j is assigned to RGV i, otherwise $x_{ij} = 0$
t	Arrival, departure, or station-passing time variable used in the model
A, B, C, D	Task types corresponding to different tobacco-box transportation paths
v	Travel speed of the RGV

Based on the above problem definition and constraints, the objective can be described as follows: under a given task volume, all transportation tasks should be completed in the shortest possible time, namely,

$$\min f(x) = max\left(T_j^{end}\right), j \in \{1,2,\cdots,J\} \qquad (1)$$

where T_j^{end} denotes the completion time of task j, and J is the total number of tasks. Since the completion time of a task is determined by the arrival time of the RGV after completing the last transportation segment of that task, when task j is assigned to the i-th RGV and executed as the q-th task of this RGV, there is:

$$T_j^{end} = T_{iq}^e \tag{2}$$

where T_{iq}^e denotes the time at which the *i*-th RGV completes its q-th task and arrives at the target station e. Therefore, the maximum completion time of all tasks can be obtained by taking the maximum value of T_{iq}^e in the task sequences of all RGVs:

$$f(x) = \max_{i \in I, q \in Q_i} T_{iq}^e \tag{3}$$

It is calculated as follows:

$$T_{iq}^e = t_e + n_0 \times t_{load} + T_{i(q-1)}^{47} + (q-1) \times t_o \ \ (i = 1) \tag{4}$$

$$T_{i1}^1 = t_1 + n \times t_{load} + t_{wait} \times (i-1) \ \ (i \neq 1) \tag{5}$$

$$T_{iq}^1 = MAX(T_{i(q-1)}^{47}, T_{(i-1)q}^{47}) + t_o + \frac{p_1}{v} + n \times t_{load} (i \neq 1, q \neq 1) \tag{6}$$

$$T_{iq}^e = MAX(T_{iq}^{e-1}, T_{(i-1)q}^e) + \frac{p_e - p_{e-1}}{v} + n \times t_{load} (i \neq 1, e \neq 1) \tag{7}$$

$$i \in [1, I], e \in [1, E], q \in [1, Q], T_{i0}^e = 0$$

$$n_0 \in [0,1,2,3,4,5] \tag{8}$$

$$n = \begin{cases} 1 & x_{ij} \times y_{qj} \times (r_j, u_j) = e \\ 0 & x_{ij} \times y_{qj} \times (r_j, u_j) \neq e \end{cases} \tag{9}$$

where x_{ij} is a decision variable indicating that task j is transported by the i-th RGV; y_{qj} indicates that task j is executed in the q-th circuit; T_{iq}^e is the time at which the i-th RGV leaves station e in the q-th circuit; t_e is the ideal travel time of the RGV from the origin point to station e; n_0 is the number of task stations that the first RGV has already passed within one circuit; t_{load} the time required for RGV loading/unloading; t_o is the time required for the RGV to return from station 47 to origin point; n is used to determine whether the RGV needs to execute a task at the current station, where $n = 1$ indicates that execution is required and $n = 0$ indicates that it is not required; t_{wait} is the departure waiting time of the RGV; p_e is the actual position of station e relative to the origin; and v is the travel speed of the RGV.

According to the operating mode of RGVs on the circular rail, the arrival conditions of RGVs can be divided into four cases.

First, when the first RGV travels to any station, its arrival time is calculated using Eq. (4). It consists of the basic travel time, cumulative loading/unloading time, return time from the previous circuit, and the additional return time caused by multiple circuits.

Second, when an RGV other than the first RGV departs for station 1, its arrival time is calculated using Eq. (5). Since subsequent RGVs must wait until the preceding RGV has departed before moving, the arrival time consists of the basic travel time, task execution time, and departure waiting time.

Third, when an RGV other than the first RGV returns to origin point in the q-th circuit, its return time is calculated using Eq. (6). It consists of the larger value between the time at which the RGV reaches station 47 in the previous circuit and the time at which the preceding RGV passes station 47, plus the time required to return from station 47 to the origin point, the basic travel time, and the task execution time.

Fourth, when an RGV other than the first RGV travels to any station other than station 1, its arrival time is calculated using Eq. (7). It consists of the larger value between the time at which the RGV passes the preceding station and the time at which the preceding RGV passes the current station, plus the ideal travel time from the preceding station to the target station and the task execution time.

The circular-rail RGV scheduling model is subject to the following constraints:

$$\sum_{i=1}^{I} x_{ij} = 1, j \in [1, J] \tag{10}$$

$$\sum_{q=1}^{Q} y_{qj} = 1, j \in [1, J] \tag{11}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{J} (x_{ij} \times y_{qj} \times G_j) \neq A + B, q \in [1, Q], i \in [1, I] \tag{12}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{J} (x_{ij} \times y_{qj} \times G_j) \neq C + D, q \in [1, Q], i \in [1, I] \tag{13}$$

$$\sum_{i=1}^{I} \sum_{j=1}^{J} (x_{ij} \times y_{qj} \times G_j) \neq B + C, q \in [1, Q], i \in [1, I] \tag{14}$$

Eq. (10) ensures that each task is assigned to only one RGV, thereby avoiding repeated execution or resource waste. Eq. (11) ensures that each task is completed within only one circuit, so a transportation operation will not be split across different circuits. Eqs. (12)-(14) restrict the task sequence according to the feasible composite-task types. In engineering terms, these constraints correspond to the actual tobacco-box flow among Area 1, Area 2, Area 3, Area 4, Area 5, and Area 6. By identifying the task types executed by an RGV in each circuit, the model allows only the predefined composite paths and prevents infeasible combinations that do not match the warehouse material-flow process.

4. Simulated Annealing Genetic Algorithm

Algorithm 1. Procedure of the proposed SAGA

Step 1. Initialize the population, algorithm parameters, initial temperature, and cooling coefficient.
Step 2. Calculate the fitness value of each individual according to the maximum task completion time.
Step 3. Select parent individuals through tournament selection and elitist preservation.
Step 4. Accept selected inferior individuals according to the simulated annealing acceptance probability under the current temperature.
Step 5. Apply two-point crossover to generate offspring and repair infeasible offspring so that task-assignment constraints are satisfied.
Step 6. Adjust the mutation rate according to the current temperature and perform mutation to maintain population diversity.
Step 7. Update the population and decrease the temperature according to the cooling coefficient.
Step 8. Repeat Steps 2-7 until the maximum number of iterations is reached, and output the best scheduling scheme.

4.1. Encoding Method

The proposed simulated annealing genetic algorithm (SAGA) adopts a binary encoding scheme. The variable x_{ij} is defined as a two-dimensional 0–1 array, where $x_{ij} = 1$ if the j-th task is transported by the i-th RGV, and $x_{ij} = 0$ otherwise. Table 2 presents an example of an individual x_{ij}, in which each column represents an independent gene.

The individual is generated randomly. Specifically, tasks are randomly assigned to RGVs. When the first task arrives, one of the 10 RGVs is randomly selected to execute the task. Then, the RGV assignment for the next task is performed. This process is repeated until all tasks have been assigned to RGVs, indicating that a single individual has been completely generated.

Table 2. Example of individual x_{ij}

x_{ij}	1	2	3	……	J-1	J
1	0	1	0	……	1	0
2	0	0	0	……	0	0
3	0	0	0	……	0	0
4	1	0	0	……	0	1
5	0	0	0	……	0	0
6	0	0	1	……	0	0
7	0	0	0	……	0	0
8	0	0	0	……	0	0
9	0	0	0	……	0	0
10	0	0	0	……	0	0

4.2. Parent Selection

The parent-selection process combines tournament selection, simulated-annealing acceptance, and elitist preservation. Tournament selection first compares a fixed number of individuals and selects the one with the best fitness as a candidate parent. Then, the simulated annealing criterion allows some inferior individuals to be accepted with a probability related to the current temperature. When the temperature is high, inferior individuals have a higher chance of being accepted, which helps maintain population diversity. When the temperature decreases, the acceptance probability becomes lower, and the search gradually focuses on better individuals.

Elitist preservation is used at the same time to retain high-quality individuals in each generation. This design reduces the risk of losing good genes during crossover and mutation while still allowing the algorithm to escape local optima in the early search stage.

4.3 Fitness Function

The fitness value is a key indicator for evaluating the quality of the current scheduling scheme. The fitness function is defined as follows:

$$f = min\, f(x) \tag{15}$$

$$f(x) = max(T_j^{end}), (j \in (1 \sim J)) \tag{16}$$

The fitness value is defined by the last task completed in the current schedule. A schedule with a smaller maximum completion time is regarded as better. During evaluation, an individual is first decoded into the task sequences assigned to different RGVs. The departure and station-passing times are then calculated according to the movement rules of the circular rail. Finally, the completion time of each task is obtained, and the largest completion time is taken as the fitness value.

4.4. Crossover

Fig. 2 illustrates the two-point crossover operation. In this operation, two parent individuals are copied to form two offspring, and two crossover positions are randomly selected on the chromosome. The gene segment between the two positions is exchanged between the offspring. Because this exchange may produce infeasible task assignments, the offspring are checked and repaired before they enter the next population.

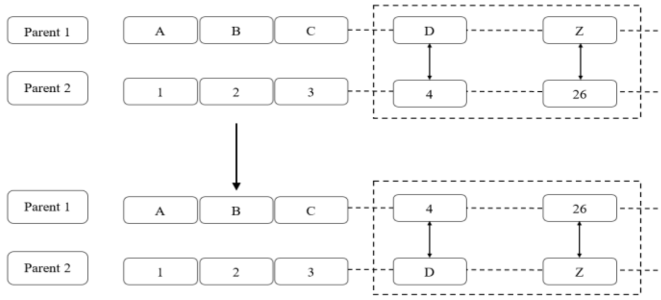

Figure 2. Schematic diagram of two-point crossover

4.5. Mutation

The mutation rate should not remain unchanged during the whole search process. If it is kept too high, the population can retain diversity, but the search may fluctuate for a long time. If it is too low, the algorithm may converge quickly but lose the chance to explore other task assignments. In this study, the mutation rate is adjusted according to the simulated annealing temperature. At the beginning of the search, the temperature is high, so a larger mutation rate is used to explore more scheduling schemes. As the temperature decreases, the mutation rate is gradually reduced, and the search becomes more focused on the better individuals already found.

5. Case Analysis

This paper conducts a case analysis of a circular-rail RGV system in a tobacco warehouse. The total length of the adopted rail is 172.13 m, including 158.00 m of straight rail and 14.13 m of curved rail. The station width is 1.20 m. The number of RGVs is $I = 10$, the number of task stations is $E = 48$, and the number of tasks is 20. The task types and loading/unloading stations are randomly generated. The genetic algorithm and the SAGA are each tested five times for comparison. Table 3 shows the relevant parameters of this experiment.

Table 3. Experimental Parameters

	Population Size	Number of Iterations	Mutation Rate	Crossover Rate	Temperature	Temperature Coefficient
Genetic Algorithm	100	100	0.1	0.8		
SAGA	100	100	0.1	0.8	1000	0.95

Note: The temperature controls the probability of accepting inferior solutions in the simulated annealing process, and the temperature coefficient determines the cooling speed during iterations.

The experimental results are shown in Table 4. The average maximum completion time obtained by the genetic algorithm is 150.2 s, with a minimum value of 145.7 s. In contrast, the average maximum completion time obtained by the SAGA is 143.4 s, with a minimum value of 139.6 s. Compared with the traditional genetic algorithm, SAGA reduces the average maximum completion time by 6.8 s, corresponding to a 4.53% reduction, and shortens the best value by 6.1 s, corresponding to a 4.19% reduction. Across the five runs, the SAGA results are also concentrated within a relatively narrow range, which indicates that the proposed method can improve both average search performance and best-case solution quality under the same parameter settings. Fig. 3 shows the fitness convergence curve and temperature convergence curve of the SAGA.

Table 4. Comparison of Algorithm Optimization Results

Experiment Number	1	2	3	4	5	Average Value/s	Best Value/s
Genetic Algorithm	153.4	150.6	146.2	154.9	145.7	150.2	145.7
SAGA	144.6	139.6	142.9	144.9	144.9	143.4	139.6

Note: Compared with GA, SAGA reduces the average maximum completion time by 4.53% and the best value by 4.19%.

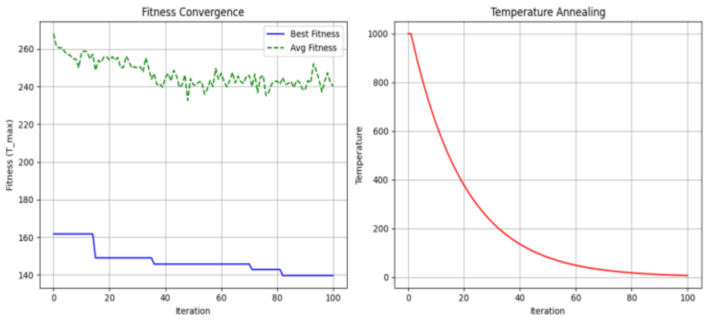

Figure 3. Fitness convergence curve and temperature convergence curve of the SAGA

Fig. 3 shows that the temperature drops as the iteration proceeds. At the beginning of the search, the temperature is still high, so the algorithm can accept a small number of worse individuals instead of only keeping the current best ones. This helps the population test more task-assignment combinations. When the temperature becomes lower, such acceptance is gradually restricted, and the fitness value changes more slowly. The curve therefore reflects the transition of SAGA from a relatively broad search to a more focused search around better scheduling schemes.

6. Conclusions

This study investigates the scheduling optimization of a circular-rail RGV system used in a tobacco warehouse. A hybrid solution approach is developed by integrating an improved genetic algorithm with simulated annealing. According to the operating features of the circular-rail system, a scheduling model is formulated to minimize the overall task completion time. The model takes into account both rail-operation constraints and the execution requirements of composite transportation tasks. In the proposed algorithm, the temperature-control mechanism of simulated annealing is embedded into the genetic search process to adjust parent selection and mutation operations. This design helps the algorithm avoid premature convergence, strengthen its global search ability, and improve its adaptability to scheduling changes.

In the case study, SAGA obtains a shorter average completion time and a better best result than the conventional genetic algorithm. This means that, under the same parameter setting, the proposed algorithm can find a more compact task arrangement for the studied circular-rail RGV system. The improvement is useful in scenarios where handling tasks are released intensively and vehicles have to wait on shared rail segments. In such cases, reducing the completion time of a task batch can also reduce the pressure on downstream material supply.

The proposed method is mainly intended for circular-rail RGV scheduling in tobacco warehouses. It can be used when the system has fixed one-way movement, dense loading and unloading stations, and composite tobacco-box transportation tasks. In future work, dynamic task arrivals, equipment failures, and recovery after production interruption should be considered in the model. These extensions will help test whether the algorithm remains stable when the warehouse operates under more disturbed conditions.

References

[1] Alexopoulos K, Anagiannis I, Nikolakis N, Chryssolouris G. A quantitative approach to resilience in manufacturing systems. International Journal of Production Research, 2022, 60(24): 7178-7193. DOI: 10.1080/00207543.2021.2018519.

[2] Lee J K, Siahpour S, Jia X, Brown P. Introduction to resilient manufacturing systems. Manufacturing Letters, 2022, 32: 24-27. DOI: 10.1016/j.mfglet.2022.02.002.

[3] Leng J, Xie J, Li R, Zhou X, Gu X, Liu Q, et al. Resilient manufacturing: A review of disruptions, assessment, and pathways. Journal of Manufacturing Systems, 2025, 79: 563-583. DOI: 10.1016/j.jmsy.2025.02.006.

[4] Agote-Garrido A, Martín-Gómez A M, Lama-Ruiz J R. Resilience as a driver of industrial manufacturing systems. Mechatronics and Automation Technology: Proceedings of the 3rd International Conference, 2024: 372-378. DOI: 10.3233/ATDE241264.

[5] Fang W J, Lai W P, Geng K L, et al. Research on RGV dynamic scheduling strategy. Digital Communication World, 2020, (11): 223-224.

[6] Cai J C, Cao M. RGV dynamic scheduling model. Software, 2020, 41(05): 113-116.

[7] Lou P, Zhong Y, Hu J, Fan C, Chen X. Digital-Twin-Driven AGV Scheduling and Routing in Automated Container Terminals. Mathematics, 2023, 11(12): 2678. DOI: 10.3390/math11122678.

[8] Yuan Z, Yang Z, Lv L, Shi Y. A Bi-Level Path Planning Algorithm for Multi-AGV Routing Problem. Electronics, 2020, 9(9): 1351. DOI: 10.3390/electronics9091351.

[9] Singh N, Dang Q V, Akcay A, Adan I, Martagan T. A matheuristic for AGV scheduling with battery constraints. European Journal of Operational Research, 2022, 298(3): 855-873. DOI: 10.1016/j.ejor.2021.08.008.

[10] Boccia M, Masone A, Sterle C, Murino T. The parallel AGV scheduling problem with battery constraints: A new formulation and a matheuristic approach. European Journal of Operational Research, 2023, 307(2): 590-603. DOI: 10.1016/j.ejor.2022.10.023.

[11] Zou W, Zou J, Sang H, Meng L, Pan Q. An effective population-based iterated greedy algorithm for solving the multi-AGV scheduling problem with unloading safety detection. Information Sciences, 2024, 657: 119949. DOI: 10.1016/j.ins.2023.119949.

[12] Ren F, Liu H. Dynamic scheduling for flexible job shop based on MachineRank algorithm and reinforcement learning. Scientific Reports, 2024, 14(1): 29741. DOI: 10.1038/s41598-024-79593-8.

[13] Li W, Li H, Wang Y, Han Y. Optimizing flexible job shop scheduling with automated guided vehicles using a multi-strategy-driven genetic algorithm. Egyptian Informatics Journal, 2024, 25: 100437. DOI: 10.1016/j.eij.2023.100437.

[14] Amirteimoori A, Babaee Tirkolaee E, Simic V, Weber G W. A parallel heuristic for hybrid job shop scheduling problem considering conflict-free AGV routing. Swarm and Evolutionary Computation, 2023, 79: 101312. DOI: 10.1016/j.swevo.2023.101312.

[15] Han X, Cheng W, Meng L, Zhang B, Gao K, Zhang C, Duan P. A dual population collaborative genetic algorithm for solving flexible job shop scheduling problem with AGV. Swarm and Evolutionary Computation, 2024, 86: 101538. DOI: 10.1016/j.swevo.2024.101538.

[16] Xu Y, Liu W, Yuan H. Multi-AGV scheduling and path planning based on an improved ant colony algorithm. Vehicles, 2025, 7(3): 102. DOI: 10.3390/vehicles7030102.

[17] Li B F, Guo L, Yang T F, et al. Dynamic scheduling strategy of intelligent RGV based on simulation optimization model. Manufacturing Automation, 2022, 44(10): 120-123.

[18] Wang J, Wu J, Fang Y F, et al. Design and application of RGV flexible dynamic scheduling system in casting workshop. Mechanical & Electrical Engineering Technology, 2023, 52(09): 62-65+104.

[19] Ding F Y, Zhang Y B, Qiu H B, et al. RGV dynamic programming scheduling model combined with greedy strategy. Electronic Technology & Software Engineering, 2021, (21): 121-124.

[20] Wang J L, Zhu X C, Tian H L, et al. Virtual simulation technology of intelligent RGV dynamic scheduling based on genetic algorithm. Research and Exploration in Laboratory, 2021, 40(10): 109-111+143.

[21] Wei W, Yang Z H, Xie H R, et al. Multi-objective programming model for intelligent RGV dynamic scheduling. Computer Applications and Software, 2020, 37(04): 178-185.

[22] Li W, Tian F, Li K. Fitness Distance Correlation Strategy for Solving the RGV Dynamic Scheduling Problem. International Journal of Cognitive Informatics and Natural Intelligence, 2020, 14(3): 20-40. DOI: 10.4018/IJCINI.2020070102.

[23] Türkyılmaz A, Şenvar Ö, Ünal İ, Bulkan S. A hybrid genetic algorithm based on a two-level hypervolume contribution measure selection strategy for bi-objective flexible job shop problem. Computers & Operations Research, 2022, 141: 105694. DOI: 10.1016/j.cor.2021.105694.

[24] Lim K C W, Wong L P, Chin J F. Simulated-annealing-based hyper-heuristic for flexible job-shop scheduling. Engineering Optimization, 2023, 55(10): 1635-1651. DOI: 10.1080/0305215X.2022.2106477.

[25] Escamilla-Serna N J, Seck-Tuoh-Mora J C, Medina-Marin J, Barragan-Vite I, Corona-Armenta J R. A hybrid search using genetic algorithms and random-restart hill-climbing for flexible job shop scheduling instances with high flexibility. Applied Sciences, 2022, 12(16): 8050. DOI: 10.3390/app12168050.

[26] Popper J, Yfantis V, Ruskowski M. Simultaneous production and AGV scheduling using multi-agent deep reinforcement learning. Procedia CIRP, 2021, 104: 1523-1528. DOI: 10.1016/j.procir.2021.11.257.