

Research Article

A Novel Scheduling Algorithm for Common Rail Dual Automatic Guided Vehicles Particle Filtering Algorithm for Industrial Process Control

Yanghua Gao , Weidong Lou , and Hailiang Lu 

Information Center, China Tobacco Zhejiang Industrial Co., Ltd., Hangzhou 310008, China

Correspondence should be addressed to Yanghua Gao; yhgao@zju.edu.cn

Received 3 November 2020; Revised 20 November 2020; Accepted 27 November 2020; Published 6 January 2021

Academic Editor: Hussein Abulkasim

Copyright © 2021 Yanghua Gao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper explores deep into the collaborative scheduling of common rail dual automatic guided vehicles (AGVs). Firstly, a dual AGV scheduling model was constructed to minimize the overall time of material distribution. Then, a novel scheduling algorithm was developed to dynamically plan the orders based on time windows. To effectively minimize the distribution time, heuristic algorithms were adopted to initialize the distribution order of materials. On this basis, the collaboration between the two AGVs was innovatively designed based on dynamic planning and time windows, making up for the defects of traditional methods in AGV cooperation. This greatly shortens the running time of the entire system in material distribution. The computing results fully demonstrate the efficiency and rationality of our algorithm. Finally, our algorithm was proved to be superior to the benchmark method through experiments on actual industrial instances.

1. Introduction

With economic growth and industrial development, there is a rising demand for automation and intelligent transport. This gives birth to a series of automated logistics equipment, such as automatic identification system, automatic guided vehicle (AGV) system, and cargo tracking system. The AGV system boasts greater application potential than the other systems, owing to its advantages in automation and efficiency.

The operations in many factories are faced with a nondeterministic polynomial-time (NP) hard problem: two AGVs at the opposite ends of a rail need to deliver materials to tanks along the rail, and the material distribution should be planned reasonably to minimize the distribution time. This problem was defined by Erdoğan in 2013 as the twin robots scheduling problem (TRSP) [1].

The TRSP applies widely in the robotic industry, where gantry robots are employed by job-shops for packaging, assembly, palletizing, and depalletizing [2]. Similar scheduling problems have arisen in the context of automated parking garages [3] and automated libraries [4]. In addition, the TRSP can be extended to multidimensional problems,

namely, robots working on shelves, and multirobot problems like multirobot cooperation system [5, 6].

The dispatching of common rail dual AGVs is a typical TRSP in factories to quickly dispatch materials between tanks or automatically schedule express delivery in warehouses. Traditionally, this novel and specific problem is solved by simple heuristic algorithms, such as genetic algorithms (GA) and [7–10] particle swarm optimization (PSO) [11–13]. Due to the sheer scale of the problem, it is difficult for these algorithms to converge to the global optimal solution. Thus, it is urgent an efficient and feasible scheduling algorithm for common rail dual AGVs.

This paper establishes a model of the common rail dual AGVs and presents an efficient solution to the model. The remainder of this paper is organized as follows: Section 2 sets up a dual AGV scheduling model to minimize the running time of the AGV system; Section 3 presents a novel heuristic scheduling algorithm that dynamically plans material distribution tasks based on time windows; Section 4 compares our method with the benchmark method through experiments on a set of benchmark instances; Section 5 wraps up this work.

2. Model Construction

To model the common rail dual AGV system, the actual factory scheduling problem was abstracted and redefined as follows (Table 1).

The common rail dual AGV system has two AGVs, one rail, and two independent hangars. During the operation, the two AGVs automatically drive in opposite directions along the same rail. Each end of the rail leads to a hangar. The tanks are evenly arranged along one or both sides of the rail for material stacking. Multiple materials can be stacked in the same tank but can only be picked up from top to bottom.

Each AGV is preallocated by the system. The operation of each AGV includes three steps: picking, transporting, and placing along the rail. The AGV can only handle one material at a time. After completing all tasks, the AGV must return to its hangar to be reset for the next operation.

To avoid collisions, a safe distance must be maintained between the two AGVs. Suppose that the two AGVs operate at the same speed on the rail, which can be calculated at time t when passing by a tank. Then, the two AGVs take the same amount of time (on average), T , to pick up and put down the material (Figure 1).

After abstracting the actual problem, the objective is to design a dual AGV scheduling strategy that minimizes the times for the two AGVs to complete the assigned material tasks and return to their respective hangars.

2.1. Dual AGV Scheduling Model with Minimized Time

2.1.1. Motion States and Directions of AGVs. To clearly illustrate the coordinate change of AGVs, the rail was abstracted into a numerical axis with the leftmost end as the origin. It is assumed that the tanks are arranged at equal intervals. The length of each interval and the length of each tank are denoted as $l1$ and $l2$, respectively. Then, the total length of each slot, that is, each tank and its left interval, can be calculated by

$$L = l1 + l2. \quad (1)$$

The current loading states of the two AGVs can be described by the state variable z_{Ai} ($i = 1, 2$):

$$z_{Ai} = \begin{cases} 0, & \text{AGV } Ai \text{ is unloaded,} \\ 1, & \text{AGV } Ai \text{ is loaded,} \\ 2, & \text{AGV } Ai \text{ is picking up materials,} \\ 3, & \text{AGV } Ai \text{ is putting down materials.} \end{cases} \quad (2)$$

The current directions of the two AGVs can be represented by a 0–1 decision variable O_{Ai} ($i = 1, 2$):

$$O_{Ai} = \begin{cases} 0, & \text{AGV } Ai \text{ moves to the left,} \\ 1, & \text{AGV } Ai \text{ moves to the right.} \end{cases} \quad (3)$$

Taking the picking and placing processes as static state (through which the two AGVs avoid collision), the current motion states of the two AGVs can be illustrated by another 0–1 decision variable m_{Ai} ($i = 1, 2$):

TABLE 1: The symbol description.

Symbol	Meaning
$l1$	The length of tank interval
$l2$	The length of each tank
L	The slot length
W_i	Material, where i is the material number
CT_i	Tank, where i is the tank number
M	The total number of tanks
z_{Ai}	The current loading states of each AGV
O_{Ai}	The current directions of each AGV
m_{Ai}	The motion states of each AGV
C_i	The current states of each tank
CT_k^F	Serial number of the target tank
T_{temp}	Material W_i is temporarily stored in the left neighbor of the target tank
T_{clr}	The time it takes to empty the target tank
$\varphi(A_i)$	Whether AGV Ai carries compatible materials
d_{k1k2}	The distance between the two AGVs
t	The time to pass through a tank
T	The time of picking up and putting down materials
T_{total}	The running time of the entire AGV system
D	The length of each AGV journey

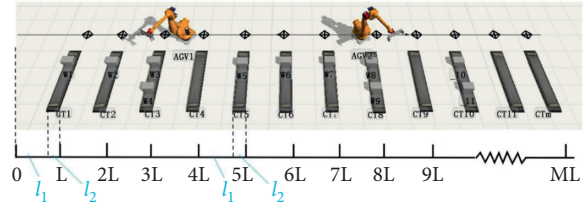


FIGURE 1: The abstract diagram of the dual AGV system.

$$m_{Ai} = \begin{cases} 0, & \text{AGV } Ai \text{ is at rest,} \\ 1, & \text{AGV } Ai \text{ is in motion.} \end{cases} \quad (4)$$

2.1.2. States of Tanks. If a material is placed in its target tank, the material is compatible with the tank. Otherwise, the material is not compatible with the tank. Let C_i be the current state of the tank. If the tank belongs to the state $C_i = -1$, all materials have been delivered to the target tank and the two AGVs have returned to the hangars:

$$C_i = \begin{cases} 0, & \text{the tank is empty,} \\ 1, & \text{there are incompatible materials in the tank,} \\ -1, & \text{the tank is full of compatible materials.} \end{cases} \quad (5)$$

2.1.3. Tank Constraints. The materials are handled by the last-in, first-out principle. That is, each material is transported to the current tank to the target tank CT_k^F , where k is the serial number of the target tank. The destination slot must be empty or only compatible before the material is placed:

$$CT_k^F = 0 \text{ or } -1. \quad (6)$$

If $CT_k^F = 1$, there are incompatible materials. The number of noncompatible materials is denoted as j . Then, it is necessary to judge the direction O_{Ai} of the AGV. If $O_{Ai} = 0$, the

AGV temporarily places the target material W_{inow} in the right neighbor of the target tank CT_{k+1}^F . If $O_{Ai} = 1$, the material W_{inow} is temporarily stored in the left neighbor of the target tank CT_{k-1}^F , which can be recorded as $T_{\text{temp}} = T$.

Next, the AGV drives to CT_k^F and transports j incompatible materials to the left neighbor CT_{k-1}^F or the right neighbor CT_{k+1}^F without temporary storage of W_{inow} until $CT_k^F = 0$ or -1 . The time it takes to empty the target tank is denoted as $T_{\text{clr}} = (2T + 2t)j$. Figure 2 illustrates the rules for material delivery.

2.1.4. Distribution Constraint. When being delivered to its destination tank, each material must be distributed by the specified AGV. Hence, an index $\varphi(A_i)$ was introduced to judge whether AGV A_i carries materials that are compatible with the target tank:

$$\varphi(A_i) = \begin{cases} 0, & \text{AGV } A_i \text{ carries compatible materials,} \\ 1, & \text{AGV } A_i \text{ carries incompatible materials,} \end{cases} \quad (7)$$

and, then, at any moment, make both $\varphi(A_i)$ values of the two AGVs zero:

$$\sum \varphi(A_i) = 0, \quad (8)$$

and this constraint ensures that each AGV is delivering the materials allocated to it at any time.

2.1.5. Safe Distance Constraint. The positions of AGV1 and AGV2 on the rail, denoted as X_{k1}^{A1} and X_{k2}^{A2} , can be, respectively, derived as

$$X_{k1}^{A1} = k1L, X_{k2}^{A2} = k2L, \quad (9)$$

where $k1$ and $k2$ are tank numbers of AGV1 and AGV2, respectively.

Then, the distance between A_1 and A_2 can be represented by d_{k1k2} . There is a phased conflict between the schedules of the two AGVs. To avoid directional conflict between A_1 and A_2 , at least one slot length L must be maintained between them as the safe distance [14]:

$$d_{k1k2} = X_{k1}^{A1} - X_{k2}^{A2} = bL, \quad (b \in Z^+, b \geq 2). \quad (10)$$

Thus, the relationship between the locations of the two AGVs can be described as

$$k2 = k1 + b, \quad (b \in Z^+, b \geq 2). \quad (11)$$

2.2. Dual AGV Scheduling Model with Minimal Time. Under the above constraints, a dual AGV scheduling model can be established to minimize the time of material distribution. The paths of the AGVs can be described by an undirected graph $G = (CT, E)$, where $CT = \{CT_1, CT_2, \dots, CT_m\}$ is the set of tanks, and $E = \{E_1, E_2, \dots, E_i\}$ is the set of intervals between tanks CT_i and CT_j . Then, the length of the journey of each AGV can be described as an integer multiple of slot length L :

$$E_{ij} = |i - j|L. \quad (12)$$

Suppose that an AGV changes direction for b times on the way to deliver materials. Let CT_i and CT_j be the start and end points of the AGV at the second change of direction B ($0 \leq BB \leq bb$), respectively. Then, the route of the AGV is denoted as E_{ij}^B . The travel length of the AGV can be expressed as

$$D = \sum_{b=0}^B E_{ij}^B. \quad (13)$$

Obviously, each AGV picks up the same amount of materials as those it delivers. Let n be the times of pick-ups (and also the times of placements) of the AGV, let t be the time to pass through a tank, and let T be the time of picking up and putting down materials. Then, the time for each AGV to complete its tasks and return to its hangar can be determined. The total time for AGV1 to complete its tasks and return to its hangar can be calculated by

$$T_{\text{total1}} = D_1 t + n_1 T, \quad (14)$$

where D_1 is the length of the journey of AGV1 and n_1 is the number of pick-ups or placements of AGV1.

The total time for AGV2 to complete its tasks and return to its hangar can be calculated by

$$T_{\text{total2}} = D_2 t + n_2 T, \quad (15)$$

where D_2 is the length of the journey of AGV2 and n_2 is the number of pick-ups or placements of AGV2.

Then, the total running time of the entire AGV system equals the longer running time between the two AGVs:

$$T_{\text{total}} = \max\{T_{\text{total1}}, T_{\text{total2}}\}. \quad (16)$$

In this way, a dual AGV scheduling model can be established aiming to let the two AGVs complete their tasks and return to their respective hangar as early as possible, that is, to minimize the running time of the entire AGV system:

$$\begin{aligned} & \min T_{\text{total}} \\ & \left\{ \begin{array}{l} CT_k^F = 0 \text{ or } -1, \\ \sum \varphi(A_i) = 0, \\ T_{\text{temp}} = T, \\ T_{\text{clr}} = (2T + 2t)j, \\ d_{k1k2} = X_{k1}^{A1} - X_{k2}^{A2} = bL, \quad (b \in Z^+, b \geq 2), \\ k2 = k1 + b, \quad (b \in Z^+, b \geq 2), \\ E_{ij} = |i - j|L, \\ D = \sum_{b=0}^B E_{ij}^B, \\ T_{\text{total1}} = D_1 t + n_1 T, \quad T_{\text{total2}} = D_2 t + n_2 T, \\ T_{\text{total}} = \max\{T_{\text{total1}}, T_{\text{total2}} \} \end{array} \right. \quad (17) \end{aligned}$$

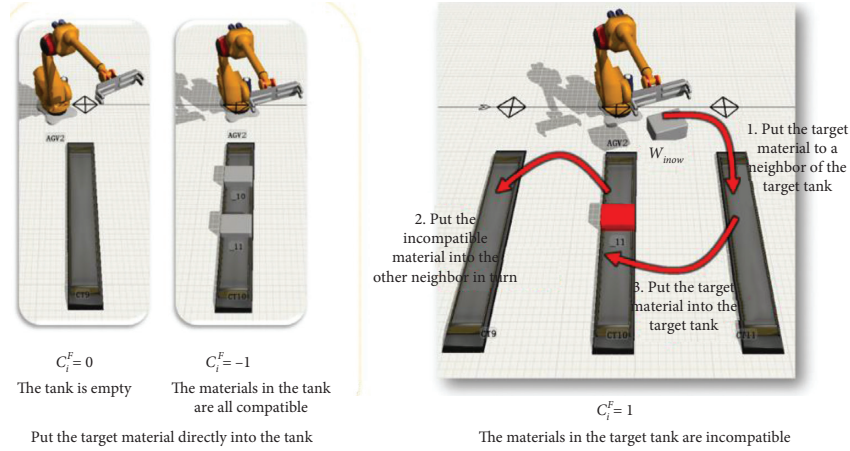


FIGURE 2: The rules for material delivery.

3. Scheduling Algorithm

3.1. Time Windows Model. Let r_i^j be the j -th occupied time window on the i -th tank, and let f_m^n be the n -th idle time window on the m -th tank. At each moment t , AGV1 and AGV2 occupy the time window where a tank is located. Then, the time $T_{arr}(f_{k0}^p, f_{kF}^q)$ to move from the current tank $k0$ of the material to the target tank kF can be calculated by

$$T_{arr}(f_{k0}^p, f_{kF}^q) = L(f_{k0}^p) + t_{k0kF} + t_{\Delta}, \quad (18)$$

where $L(f_{k0}^p)$ is the time to reach the starting tank $K0$ in a free time; t_{k0kF} is the time to move from the starting tank $K0$

to the end tank KF without considering safe distance; t_{Δ} is the additional time to move from $K0$ to KF incurred by the location and motion states of the other AGV [15].

3.2. Adding Safe Distance. Under the safe distance, the tank r_i^j where AGV is located and its left neighbor r_{i-1}^j and right neighbor r_{i+1}^j are occupied simultaneously; moreover, the other AGV cannot appear in this section with this AGV at the same time. Then, the current state v_i of the current slot can be depicted as [16]

$$v_i = \begin{cases} 0, & \text{the tank is occupied and the AGV is in the } i^{\text{th}} \text{ or } i \pm 1^{\text{th}} \text{ tank,} \\ 1, & \text{the tank is free and the AGV is not in the } i^{\text{th}} \text{ or } i \pm 1^{\text{th}} \text{ tank.} \end{cases} \quad (19)$$

For every time t , the state of each slot needs to be changed with the location of the AGV. Figure 3 explains the occupation state of the time window.

3.3. Heuristic Scheduling Algorithm for Dynamic Planning of Material Distribution Tasks Based on Time Windows (DPTW)

3.3.1. Workflow

Step 1. Find the shortest time delivery sequence for the independent delivery state of each AGV by the genetic algorithm (GA)

(1) Parameter initialization

Initialize the selection, crossover, and mutation probabilities according to the performance of the AGVs.

(2) Encoding

Set up a coding string of the length n_{A_i} and values between 1 and n_{A_i} , where n_{A_i} is the total number of materials assigned to A_i .

(3) Generating the initial solution

Randomly generate a coded string of length n_{A_i} ; that is, randomly arrange the order of material delivery, as the initial solution.

(4) Fitness calculation

Take the penultimate time required to complete the local materials in the current order ($1/T_{total}$) as the fitness function. The reciprocal value of the fitness is negatively correlated with the time cost and positively with the fitness of each individual.

(5) Population selection

Use the Tournament Selection Method (TSM) [17, 18] to remove some individuals from the population, and select one of the best fit individuals to enter the next population. Repeat this process until the new population equals the original population in size.

(6) Crossover and mutation

Implement the reverse mutation operator [19] and exchange the order of material distribution.

(7) Evolution

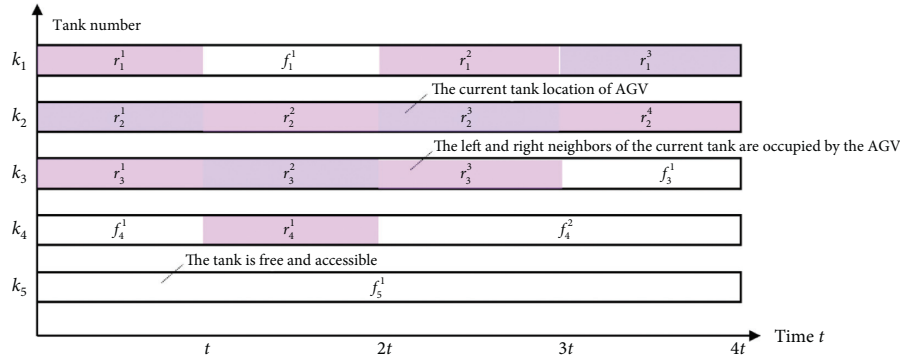


FIGURE 3: The occupation state of the time window.

Obtain the new population and repeat Step (4) until the 10,000-th iteration to get the best fit individuals.

(8) Decoding

Decode the genetic sequence of the order of local material distribution for each AGV. Store the results of AGV1 and AGV2 in one-dimensional (1D) arrays p_0^1 and p_0^2 , respectively, which reflect the minimum running time in the case of independent interaction.

Step 2. Order distribution based on the initial minimum running time

As shown in Figure 4, distribute the first task to AGV1 and AGV2 according to the initial scheduling plan p_0^1 , obtain the time window of the first task of each AGV, and judge whether there is an overlapping section RP(S) between the routes of the two AGVs.

If yes, let the AGV closer to the destination tank continue with its delivery, and halt the other AGV until the overlapping section is cleared.

If not, repeat Step 2 after the delivery to assign the second task as per the initial plan until an overlapping section arises. Once an overlapping section occurs, go to Step 3 for dynamic planning of the next task.

Step 3. Dynamic planning of the next task based on time windows

Assign the next task $RW_{i+1}(K0, KF, t_{i+1})$ to an AGV after it has placed the current task material and entered the unloaded state $z_{Ai} = 0$.

(1) Case 1: the two AGVs do not end the current task in a synchronized manner.

If A1 terminates the current task but A2 continues with the current task, dynamically plan the next task of A1 in combination with the time window of A2; conversely, dynamically plan the next task of A2 in combination with the time window of A1. Firstly, search for the nearest tanks on the left and right of the current slot of A1, and denote them as

optional task points RW1 and RW2, following the greedy strategy. Take the number of the next materials in the original plan p_0^1 as alternative task point RW3.

Next, obtain the time window of the alternative tank separately, and plan the route and time to reach the starting and target tanks.

Finally, compare A2 with the time windows of the current target tank, measure the set of overlapping sections RP1(SS), RP2(SS), and RP3(SS), and take the RW corresponding to $\min\{RP1(SS), RP2(SS), RP3(SS)\}$ as the next task.

(2) Case 2: the two AGVs end the current task in a synchronized manner.

If A1 and A2 terminate the current task at the same time and if the distribution of the next task to one of them affects that of the other, then A1 and A2 search for the nearest tanks on the left and right of their current slots and denote them as optional task points (RW_1^{A1}, RW_2^{A1}) and (RW_1^{A2}, RW_2^{A2}) following the greedy strategy and also take the number of the next materials in the original plan p_0^1 as alternative task points RW_3^{A1} and RW_3^{A2} .

Next, obtain the time windows of the three alternative task points of A1 and A2, respectively, and plan the route and time for each AGV to reach the starting and target tanks.

Finally, establish the set of overlapping sections, and take the least overlapping combinations as the next task of the two AGVs.

Step 4. Determine the motion state for the next task

(1) Case 1: the two AGVs do not end the current task in a synchronized manner.

Let an AGV receive the next task as a collaborator, compare the time windows for transporting the current materials, and choose to stop or stay away for cooperation with the AGV still engaged in material delivery at the end of the overlapping section, according to its location k_i and whether k_{i-2}/k_{i+2} is busy at the next moment:

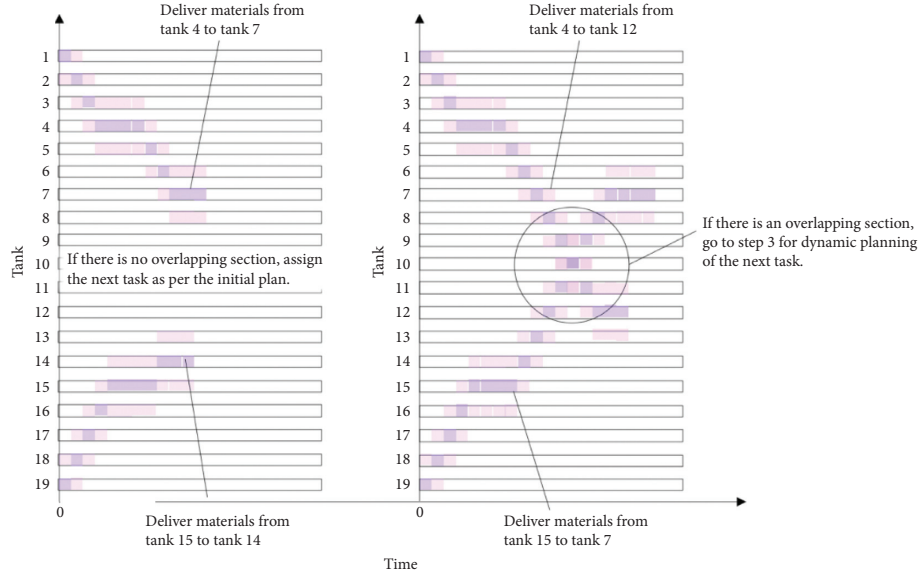


FIGURE 4: The handling of overlapping section.

$$\begin{cases} v_{ki} = 1, v_{ki-2} \parallel v_{ki+2} = 1, & \text{move left/right,} \\ v_{ki} = 1, v_{ki+2} = 0, & \text{move to the/end tank,} \\ v_{ki} = 0, v_{ki-2} \parallel v_{ki+2} = 1, & \text{Stop.} \end{cases} \quad (20)$$

(2) Case 2: the two AGVs end the current task in a synchronized manner.

Since there is an overlapping section between the two AGVs, choose the collaborator based on their distances from the starting tank to the target tank. Make the AGV farther away from its target tank as the collaborator, and choose to stop or stay away. The determination of the motion state for the next task is illustrated in Figure 5.

Step 5. Repeating Steps 3 and 4 until the end of material distribution

Repeat Steps 3 and 4 until all material tasks are completed, return to the hangar, count the transport time of each task, and obtain the shortest time $\min T_{\text{total}}$.

(2) Running time optimization

To minimize the overall time cost of placing the target materials, the two AGVs can cooperate with each other to empty their target tank under the following conditions.

One AGV can issue an emptying request to the other AGV if their distance $d_{k_1 k_2}$ satisfies

$$d_{k_1 k_2} \geq \left(\frac{2T}{t} + 2 \right) L. \quad (21)$$

Suppose that AGV1 is the one issuing the request. Let k_2 and k_{A_2} be the current position and starting tank of AGV2, respectively; let k_{A_1} be the target tank of AGV1 ($k_2 < k_{A_1} < k_{A_2}$). If AGV2 is unloaded currently and moves in

the same direction (right) as AGV1, then AGV2 will accept the request under the following conditions:

$$\begin{cases} O_{A_1} = O_{A_2} = 1, \\ z_{A_2} = 0, \\ k_2 < k_{A_1} < k_{A_2}. \end{cases} \quad (22)$$

Suppose that AGV2 is the one issuing the request. Let k_1 and k_{A_1} be the current position and starting tank of AGV1, respectively; let k_{A_2} be the target tank of AGV2 ($k_1 < k_{A_2} < k_{A_1}$). If AGV1 is unloaded currently and moves in the same direction (right) as AGV2, then AGV1 will accept the request under the following conditions:

$$\begin{cases} O_{A_1} = O_{A_2} = 1, \\ z_{A_1} = 0, \\ k_1 < k_{A_2} < k_{A_1}. \end{cases} \quad (23)$$

Figure 6 explains the process of dual AGVs collaborating with each other.

3.3.2. *Algorithm Flow.* Figure 7 presents the workflow of the proposed algorithm.

4. Computational Results in Industrial Implementation

To verify its performance, the proposed algorithm was applied to minimize the overall time of the common rail dual AGVs in handling 16 actual orders from a factory. For comparison, the classical GA was also tested to test whether our algorithm can outshine the common baseline search algorithm. Tables 2 to 17 provide the details of the 16 orders.

As shown in Table 18, our algorithm greatly shortened the overall distribution time of the GA, an evidence to its rationality and high performance. Take order 7 as an

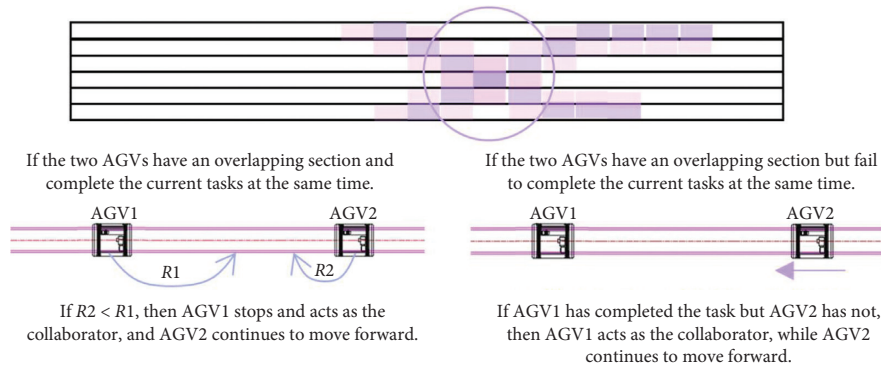
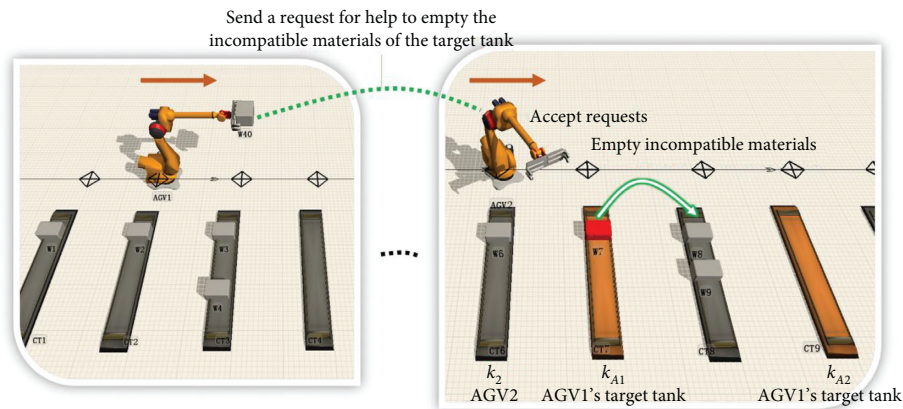


FIGURE 5: The determination of the motion state for the next task.



AGV2 can help empty the incompatible materials of AGV1, because the target tank of AGV1 is empty and before the target tank of AGV2.

FIGURE 6: The collaboration between the two AGVs.

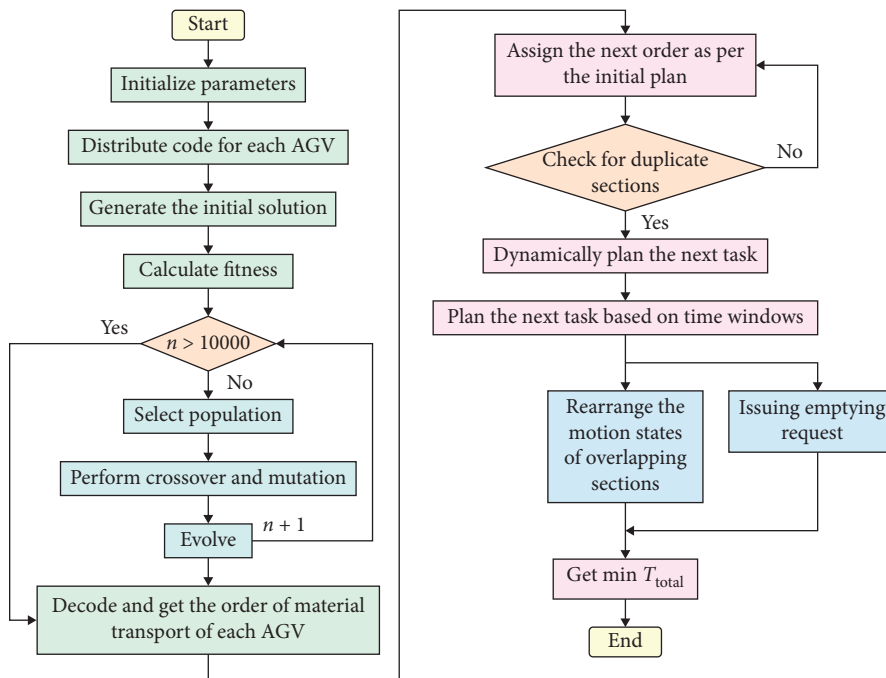


FIGURE 7: The workflow of the proposed algorithm.

TABLE 2: Instance 1 (number of tanks: 20).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	20
2	1	2	3
3	2	4	11
4	1	7	8
5	1	9	7
6	1	10	9
7	2	11	12
8	1	14	7
9	2	16	5
10	2	17	8
11	1	18	1
12	2	19	12
13	1	20	13

TABLE 3: Instance 2 (number of tanks: 20).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	10
2	2	3	9
3	1	4	4
4	2	5	1
5	1	6	8
6	2	7	2
7	2	10	18
8	1	11	16
9	1	12	5
10	1	13	17
11	2	15	15
12	2	16	14
13	1	17	8
14	2	18	18
15	1	19	8
16	2	20	8

TABLE 4: Instance 3 (number of tanks: 20).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	1	8
2	2	2	14
3	1	3	6
4	1	3	18
5	1	5	15
6	2	7	1
7	2	9	11
8	2	9	19
9	2	10	2
10	2	10	4
11	1	10	5
12	2	12	20
13	1	13	10
14	2	18	12
15	1	19	7

TABLE 5: Instance 4 (number of tanks: 20).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	13
2	2	3	6
3	2	5	13
4	1	7	2
5	1	9	4
6	2	10	5
7	1	11	12
8	2	11	4
9	2	11	4
10	1	12	16
11	2	12	5
12	1	16	11
13	1	16	17
14	1	17	17
15	2	17	16
16	1	18	10
17	1	19	19

TABLE 6: Instance 5 (number of tanks: 20).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	1	18
2	2	2	2
3	1	2	14
4	1	3	17
5	2	4	20
6	2	7	8
7	2	8	13
8	2	8	16
9	1	8	9
10	1	9	12
11	2	10	9
12	2	10	5
13	2	11	12
14	1	13	11
15	1	13	1
16	1	13	12
17	2	15	20
18	1	18	12
19	1	20	7

TABLE 7: Instance 6 (number of tanks: 25).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	1	2
2	2	2	11
3	2	3	10
4	2	7	10
5	1	9	5
6	1	10	16
7	1	11	20
8	2	12	14
9	1	13	17
10	2	14	21

TABLE 7: Continued.

Material serial number	Allocated AGV	Current tank	Target tank
11	1	15	24
12	2	16	14
13	2	17	16
14	2	18	21
15	1	19	16
16	2	20	6
17	2	22	19
18	2	23	6
19	1	25	13

TABLE 9: Continued.

Material serial number	Allocated AGV	Current tank	Target tank
16	1	19	1
17	2	20	7
18	2	21	15
19	2	22	23
20	1	23	23
21	1	24	16
22	1	25	25

TABLE 8: Instance 7 (number of tanks: 25).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	1	18
2	2	3	4
3	1	4	3
4	1	4	10
5	1	7	6
6	2	7	10
7	2	7	16
8	1	8	13
9	2	9	15
10	2	10	2
11	1	11	10
12	2	12	2
13	1	12	17
14	1	13	17
15	2	14	15
16	1	15	14
17	2	18	18
18	2	18	3
19	1	21	24
20	2	22	20

TABLE 10: Instance 9 (number of tanks: 25).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	3	19
2	1	5	4
3	2	6	23
4	2	7	5
5	1	9	7
6	1	10	20
7	1	11	2
8	2	14	22
9	1	14	9
10	2	15	4
11	1	15	18
12	2	17	7
13	2	20	20
14	2	20	7
15	1	21	21
16	2	22	25
17	1	22	16
18	1	24	5

TABLE 9: Instance 8 (number of tanks: 25).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	3
2	1	2	3
3	2	3	2
4	1	4	7
5	1	5	19
6	2	7	17
7	2	8	12
8	2	9	4
9	1	10	3
10	1	12	20
11	2	13	24
12	1	15	4
13	2	16	4
14	2	17	2
15	1	18	11

example to illustrate the rationality and effectiveness of our algorithm. Figure 8 explains the next routes of the two AGVs in the event of a conflict [20].

By our algorithm, the final results of AGV operations and order of material distribution were obtained. Then, the distribution order was randomly changed to obtain new solutions to AGV running time. After the changes, the running time of the AGVs and the relevant solutions were observed to see if our algorithm could lead to the best solution.

Figure 9 presents the randomly changed order and the new running time of the 30 orders in Tables 2 to 17. It can be seen that all the new solutions were greater than the original running time, indicating that the scheduling plan of our algorithm is the best possible solution.

The next step is to verify whether any solution in the 16 solutions is the optimal solution. Taking order 7 as the object, the order of material distributions was changed randomly twice, three times, and four times, respectively. The running time of each change is shown in Figure 10.

TABLE 11: Instance 10 (number of tanks: 25).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	3	14
2	2	5	14
3	1	8	10
4	1	10	18
5	1	11	14
6	2	12	7
7	2	12	19
8	2	14	24
9	2	15	5
10	2	16	4
11	1	16	24
12	2	17	14
13	1	17	18
14	1	19	11
15	1	19	14
16	1	19	2
17	2	19	12
18	1	20	17
19	1	20	7
20	1	21	16
21	2	21	3
22	2	22	17
23	1	24	25

TABLE 13: Instance 12 (number of tanks: 30).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	5
2	2	2	2
3	2	3	12
4	1	7	28
5	1	8	13
6	1	9	21
7	2	9	26
8	2	10	16
9	1	11	20
10	1	11	9
11	1	14	1
12	2	16	1
13	2	18	6
14	2	24	25
15	2	24	13
16	2	25	20
17	1	26	3
18	1	27	15
19	2	27	21
20	2	29	11
21	1	30	28
22	2	30	25

TABLE 12: Instance 11 (number of tanks: 30).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	1
2	1	2	27
3	1	2	1
4	2	5	13
5	1	6	28
6	2	6	30
7	1	8	10
8	2	9	29
9	1	10	19
10	1	11	9
11	2	11	7
12	1	13	12
13	1	13	25
14	2	13	24
15	1	15	26
16	2	16	7
17	2	17	23
18	2	17	27
19	2	18	10
20	2	18	4
21	2	24	14
22	2	27	5
23	1	28	23
24	1	29	12

TABLE 14: Instance 13 (number of tanks: 30).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	1	14
2	2	2	6
3	2	3	25
4	1	4	28
5	2	5	15
6	1	6	24
7	1	7	3
8	2	7	17
9	2	8	9
10	1	12	11
11	1	12	7
12	1	12	5
13	1	14	20
14	1	15	15
15	1	15	1
16	2	16	15
17	2	17	9
18	2	18	28
19	2	18	12
20	2	19	6
21	1	19	7
22	1	21	26
23	2	24	7
24	2	25	14
25	2	26	3
26	2	26	3
27	1	27	29

TABLE 15: Instance 14 (number of tanks: 30).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	1	27
2	1	1	8
3	1	1	28
4	1	3	11
5	2	3	20
6	1	4	18
7	1	4	20
8	2	6	9
9	2	8	16
10	2	8	27
11	1	9	25
12	1	10	22
13	2	10	13
14	2	11	18
15	1	11	10
16	1	11	1
17	1	13	16
18	2	13	20
19	1	14	9
20	2	14	1
21	1	16	22
22	1	16	20
23	1	16	10
24	2	17	14
25	1	19	28
26	1	19	6
27	2	19	8
28	1	20	23
29	2	24	11
30	2	24	25
31	2	26	30
32	2	27	19
33	2	29	27

TABLE 16: Instance 15 (number of tanks: 30).

Material serial number	Allocated AGV	Current tank	Target tank
1	2	3	22
2	1	4	29
3	2	4	1
4	1	4	29
5	2	6	21
6	2	7	16
7	1	7	5
8	1	8	5
9	1	8	25
10	2	11	1
11	1	11	1
12	2	12	24
13	1	13	30
14	2	14	4
15	1	14	19
16	2	15	24
17	1	16	9
18	1	16	2
19	1	17	24

TABLE 16: Continued.

Material serial number	Allocated AGV	Current tank	Target tank
20	1	20	23
21	1	20	12
22	1	22	21
23	2	22	8
24	2	23	25
25	2	26	9
26	1	26	5
27	2	29	16

TABLE 17: Instance 16 (number of tanks: 40).

Material serial number	Allocated AGV	Current tank	Target tank
1	1	4	15
2	2	4	18
3	1	7	19
4	1	7	18
5	2	11	28
6	2	11	27
7	2	12	2
8	2	12	32
9	2	12	24
10	2	15	26
11	1	15	24
12	2	19	26
13	1	21	28
14	2	22	28
15	1	23	4
16	1	23	25
17	2	24	16
18	2	24	28
19	1	25	29
20	2	25	15
21	1	26	25
22	2	26	32
23	2	26	36
24	1	28	16
25	1	28	16
26	1	29	16
27	1	30	28
28	1	30	37
29	1	31	23
30	2	32	15
31	2	36	8
32	1	37	13
33	1	38	20
34	2	38	28
35	1	39	20
36	2	39	22

It can be seen that, after the changes in material distribution order, the final new solutions were greater than the benchmark value of 785. Hence, the proposed algorithm can always obtain the optimal solution.

TABLE 18: The computing results (unit: s).

Instance	DPTW	GA
Instance 1	550	755
Instance 2	620	805
Instance 3	595	780
Instance 4	650	855
Instance 5	920	1365
Instance 6	665	915
Instance 7	785	1190
Instance 8	995	1375
Instance 9	910	1120
Instance 10	975	1235
Instance 11	1355	1860
Instance 12	1075	1665
Instance 13	1485	1910
Instance 14	1875	2350
Instance 15	1595	2130
Instance 16	2295	3345



FIGURE 8: The illustration of conflict avoidance in order 7.

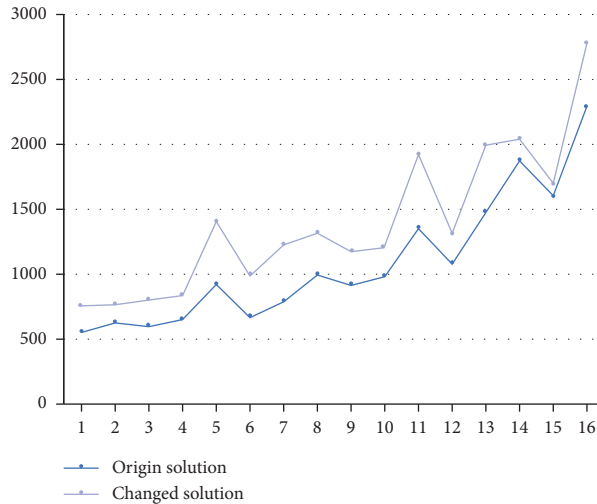


FIGURE 9: The running time after random changes of material distribution order.

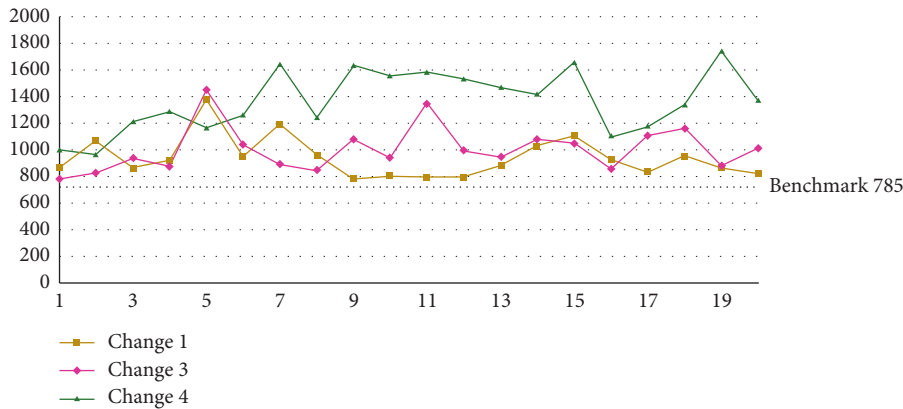


FIGURE 10: The running time after three types of changes in material distribution order.

5. Conclusions

This paper models and solves the problem of common rail dual AGVs scheduling in a novel way. The proposed solving algorithm for the problem consists of five steps, each of which is described in detail, and provides the scheduling plans considering the various situations of AGV motions. The performance of our algorithm was fully verified through experiments. The results show that our algorithm can solve the problem more efficiently and is more effective than the GA in all cases. Even if the solution structure was partly changed by random, our algorithm could still find the optimal solution very quickly. The future research will extend proposed method to multi-AGV cooperative scheduling or develop even more efficient online dynamic programming algorithms for this problem.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] G. Erdoğan, M. Battarra, and G. Laporte, "Scheduling twin robots on a line," *Naval Research Logistics (NRL)*, vol. 61, no. 2, pp. 119–130, 2014.
- [2] P. G. Ranky, "Collaborative, synchronous robots serving machines and cells," *Industrial Robot: An International Journal*, vol. 30, no. 3, 2003.
- [3] A. Mathijssen and A. J. Pretorius, "Verified design of an automated parking garage," in *Proceedings of the International Workshop on Parallel and Distributed Methods in Verification*, pp. 165–180, Bonn, Germany, August 2006.
- [4] K. E. Dimitri, F. D. Gallo, J. E. Kulakowski, R. J. Means, J. L. Thrall, and D. J. Winarski, "Automated data storage library dual picker interference avoidance," U.S. Patent 6,038,490, 2000.
- [5] W. Burgard, M. Moors, C. Stachniss, and F. E. Schneider, "Coordinated multi-robot exploration," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 376–386, 2005.

- [6] T. Arai, E. Pagello, and L. E. Parker, "Advances in multi-robot systems," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 655–661, 2002.
- [7] M. Gemeinder and M. Gerke, "GA-based path planning for mobile robot systems employing an active search algorithm," *Applied Soft Computing*, vol. 3, no. 2, pp. 149–158, 2003.
- [8] H. Özdemir, R. Sever, and Ö. Polat, "GA-based optimization of SURF algorithm and realization based on Vivado-HLS," *Traitement du Signal*, vol. 36, no. 5, pp. 377–382, 2019.
- [9] Y. Lu, X. Pei, C. Zhang, H. Luo, B. Liu, and Z. Ma, "Design of multimodal transport path optimization model and dual pheromone hybrid algorithm," *Journal Européen des Systèmes Automatisés*, vol. 52, no. 5, pp. 477–484, 2019.
- [10] A. P. Gopi, V. Lakshman Narayana, and N. Ashok Kumar, "Dynamic load balancing for client server assignment in distributed system using genetical algorithm," *Ingénierie des Systèmes d'Information*, vol. 23, no. 6, pp. 87–98, 2018.
- [11] W. Zhang, X. L. Zhang, and Y. Li, "Path planning for intelligent robots based on improved particle swarm optimization algorithm," *Journal of Computer Applications*, vol. 34, no. 2, pp. 510–513, 2014.
- [12] D. Wu, "Multi-objective Decision-making of new retailing terminals based on particle swarm optimization and genetic algorithm," *Journal Européen des Systèmes Automatisés*, vol. 52, no. 6, pp. 607–615, 2019.
- [13] Z. Li, Y. Zhou, and R. Bao, "An image classification method based on optimized fuzzy bag-of-words model," *Traitement du Signal*, vol. 36, no. 3, pp. 239–244, 2019.
- [14] K. L. Peng, "Safe following distance warning system and method for a vehicle," U.S. Patent Application 12/057,725, 2009.
- [15] N. Smolic-Rocak, S. Bogdan, Z. Kovacic, and T. Petrovic, "Time windows based dynamic routing in multi-AGV systems," *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 1, pp. 151–155, 2009.
- [16] D. I. Ferguson and D. Dolgov, "Safely navigating on roads through maintaining safe distance from other vehicles," U.S. Patent 8,725,342, 2014.
- [17] O. Andrzej and K. Stanislaw, "A new constraint tournament selection method for multicriteria optimization using genetic algorithm," *Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No. 00TH8512)*, vol. 1, pp. 501–508, 2000.
- [18] M. Vard, M. Neyestan, and A. Ghorbanian, "Supplier selection and order allocation problem modeling with the aim of comparing incremental discounts versus wholesale discounts by using GA and NSGA algorithms," *Journal Européen des Systèmes Automatisés*, vol. 52, no. 1, pp. 23–34, 2019.
- [19] T. P. Hong and H. S. Wang, "A dynamic mutation genetic algorithm," *IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems (Cat. No. 96CH35929)*, vol. 3, pp. 2000–2005, 1996.
- [20] R. H. Möhring, E. Köhler, E. Gawrilow, and B. Stenzel, "Conflict-free real-time AGV routing," in *Proceedings of the Operations Research Proceedings*, pp. 18–24, Tilburg, Netherlands, July 2005.