



Research article

Research on multi-load AGV path planning of weaving workshop based on time priority

Li-zhen Du^{1,*}, Shanfu Ke¹, Zhen Wang¹, Jing Tao², Lianqing Yu¹ and Hongjun Li¹

¹ School of Mechanical Engineering and Automation, Wuhan Textile University, 430200, Wuhan, China

² School of Mechanical and Electrical Engineering, Hubei Polytechnic University, 435003, Huangshi, China

* Correspondence: Email: dlzay@163.com; Tel: +15327347590.

Abstract: The multi-load AGV (Automatic Guided Vehicle) is a new kind of materials handling equipment used to load cloth automatically in an intelligent weaving workshop. It can transport multiple rolls of cloth and choose the correct, most effective path to improve the transportation efficiency without people engaged in. This paper creates a feasible path topology according to the layout of the workshop and the logistics environment, and uses the Warshall-Floyd algorithm to search for the optimal route between two arbitrary points. The aim of the path planning is to maximize the machine efficiency, which is constrained by environmental limits, load limits and work limits. This paper establishes the mathematical model of the path planning problem using the mixed genetic particle swarm optimization algorithm (GA-PSO) to solve the problem, and the particle iteration mechanism based on the time priority is proposed to make the evolution more directional and accelerate the convergence speed of the algorithm. The effectiveness and practicability of the model and methods are verified by simulation and benefit analysis.

Keywords: multi-load AGV; intelligent weaving workshop; time priority; path planning; GA-PSO algorithm

1. Introduction

Most of weaving workshops have adopted the mode of loading cloth manually, which lead to require a large amount of physical labor. Due to the poor quality of manual cloth loading, some cloth loading tasks would easily missed, resulting in cloth rolls with different lengths, thus affecting customer satisfaction. Therefore, there is an urgent need to replace people with machines in the cloth loading process and adopt an automatic method of cloth loading.

A multi-load AGV is mainly applied in the weaving workshop to automatically drop off the finished rolled cloth and transport it to the cloth inspection workshop, which can reduce the need for labor and reduce the operation costs [1]. Because of the unique layout and production mode, the path planning method of the multi-load AGV in a weaving workshop urgently needs to be discussed.

At present, with respect to the strategy of task allocation in the multi-load AGV's path planning problem, Li Guo Min [2] presented an improved harmony search algorithm to solve tasks assigning and sequencing of multiple AGVs. Dong-Hyun Lee [3] proposed a resource based task allocation algorithm for multi-robot systems.

In terms of shortest paths, Tian Jian Chen [4] presented a dynamic routing method for the shortest path planning and conflict prevention in the AGV system, which scheduled the shortest and conflict-free route by utilizing the Dijkstra algorithm and the Dynamic Time-Window Method. Adam [5] presented a modified particle swarm optimization method to search for a minimal time path for a solar-powered unmanned ground vehicle (UGV). Smolic-Rocak [6] proposed a routing method using time windows with a vectorial form. The use of time windows makes the algorithm prone to other scheduling and routing problems.

With respect to establishing the path planning model, KG Huo [7] regarded the minimum total operational costs as the planning target; the operational restrictions, the length of the time window and the load balance as constraint conditions; and the scheduling policy driven for events as the research method to establish a mixed-integer programming model for the scheduling of the multi-load AGV. Li Guo Min [8] established a mathematical model with a new objective function, which incorporates two indicators, that is, the standard deviation of the waiting time of computer numerical control material buffers and the total travel distance of automated guided vehicle. Maryam Mousavi [9] studied a multi-objective optimization method, including task scheduling, the minimum number of vehicles and the time costs. Fazlollahtabar [10] proposed an approach for finding an optimal path in a flexible jobshop manufacturing system considering two criteria of time and cost. H. Fazlollahtabar [11] considered time, costs and AGV capabilities as a triple standard and applied a mathematical programming method to achieve the optimal path selection of AGV transportation on the basis of the BOM.

With respect to the solution of the model, V. K. Chawla [12] proposed a combination of particle swarm optimization (PSO) for a global search and the Memetic Algorithm (MA) for a local search, termed the modified memetic particle swarm optimization algorithm (MMPSO), for the scheduling of multi-load AGVs in an FMS. Soovadeep Bakshi [13] presented an algorithm for the scheduling of an AGV that traverses desired locations on a manufacturing floor; the algorithm enables the AGV to find an optimal route. Hamed Fazlollahtabar [14] proposed a mathematical formulation that was first adapted to a minimum cost flow (MCF) model and then optimized using a modified network simplex

algorithm (NSA). Grunow [15] took advantage of the effectiveness of vehicles, proposed a scheduling algorithm based on priority to solve the problem for multi-load AGV scheduling, and established an MILP model to verify the effectiveness of the scheduling algorithm. Tong-Jin Park [16] proposed an algorithm to calculate the real path and obstacle's length and planned the path using the algorithm.

The implementation of a multi-load AGV in a weaving workshop can be regarded as task assignment and path planning problems in the transportation network. The above literature has substantial reference value for the task assignment strategy, the establishment of a mathematical model and the solution for multi-load AGV path planning, but it is not fully applicable to a weaving workshop. At present, the scheduling problem of weaving workshop is focused on production scheduling, there is a lack of research results on the logistics scheduling of weaving workshop. On the basis of the above documents, this paper studies the problem of multi-load AGV path planning in a weaving workshop; considers the speed differences of AGVs on main streets, roadways and corners to search for the path with the shortest time; presents a method for multi-load AGV path planning based on the shortest time; and proposes the GA-PSO algorithm with a better search performance to solve the path planning model. The result of this research on multi-load AGV is practical and can be applied to the weaving workshops and other industries.

2. Problem description

The layout of the weaving workshop and the horizontal transportation path are shown in Figure 1. The loom is arranged in a laneway pattern; the blue line in the Figure indicates the path, and blue circle indicates the task point needed for load cloth. The multi-load AGV (automated guided vehicle) starts from the cloth inspection workshop (arriving at point P00 can be regarded as entering the weaving workshop) and travels along the feasible path to the path node corresponding to the set weaving machine. After loading the cloth automatically, the AGV transfers the cloth roller to the inspection workshop (to point P00 point that can be considered as entering the inspection workshop). After unloading the cloth roller, the AGV returns to the weaving workshop to carry the finished cloth and handles all the loaded cloth.

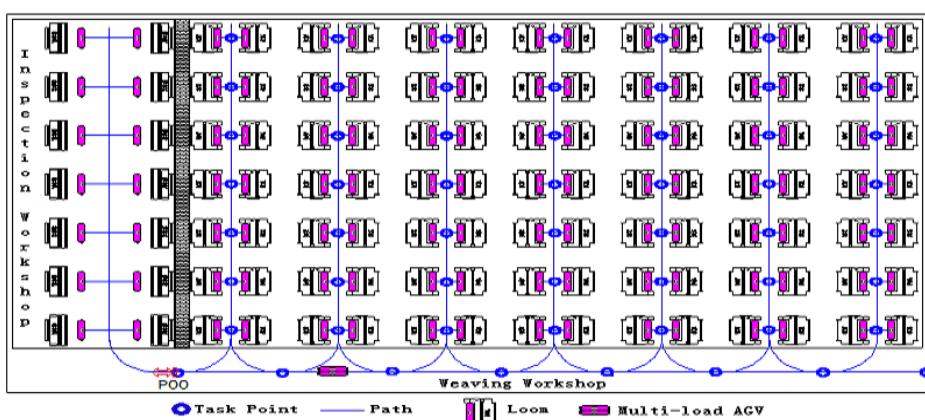


Figure 1. Layout of the weaving workshop.

The path planning method consists of completing all tasks, carrying a full load in each single task (the capacity of each AGV is 4), minimizing the transport costs, and maximizing the production efficiency, which is the goal of path planning. The AGV serves each task point only once in each task batch, and the AGV has a load limitation. Therefore, the problem can be considered as a TSP problem with limited load.

According to the actual layout of the workshop, the following constraints are mainly considered in constructing the transportation network:

- (1) Path constraint: the AGV has only one shortest path between the task points in the transportation process;
- (2) Task constraint: the AGV shall finish all tasks;
- (3) Capacity limitation constraint: The number of cloth rolls on the AGV must not exceed the upper limit (4 rolls).

The key questions are how to schedule the AGV reasonably according to tasks, and which scheduling rules can get the highest benefits. According to the practical situation, this paper considers scheduling the AGV according to the minimum loom waiting and AGV working times, and the model for multi-load AGV path planning is based on the lowest time.

3. Mathematics model generation

3.1. Topology diagram of the workshop

To facilitate the description of the problem, in this study, the cloth loading point is 0, the loom is $1, 2, \dots, 98$, the task point number is $1, 2, \dots, n$, and the set of task points for cloth loading is $R = \{1, \dots, i, \dots, j, \dots, n\}$. Figure 2 shows the topological diagram of the weaving workshop.

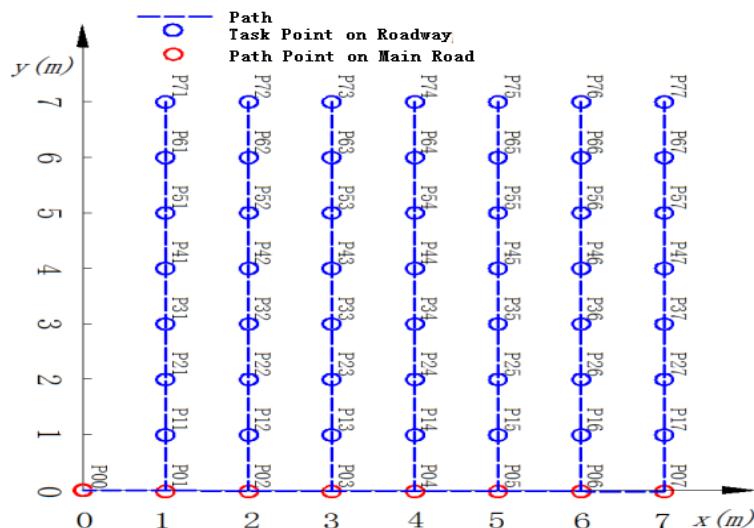


Figure 2. Topological diagram of the weaving workshop.

- (1) P00 represents the entrance and exit of the inspection workshop.
- (2) The blue nodes (P1, P21,...and P77) represent the task nodes (decision points). Each node in the path corresponds to the loading task of two looms, such as the AGV running to P22, which can execute the loading tasks of looms 16 and 23.
- (3) The red nodes (P01, P02..., and P07) represent the turning nodes of the AGV when it enters the roadway.

3.2. Model description

- (1) The time of the occurrence of the loading tasks are known, and all are different.
- (2) Starting from P00, the AGV will arrive at each task node that needs loaded cloth, and each task point will be visited only once. After loading 4 rolls of cloth, the AGV will return to P00 to unload, and then start the next task.

In actual production, if the AGV arrives at the cloth loading point later than the cloth loading time, it will cause the loom to weave fabric beyond the original length. Therefore, how to execute the task with a lower loom waiting time and a lower AGV working time to improve the machine utilization rate is the key problem.

3.3. Parameters instructions

Table 1. Parameters instructions.

Parameters	Descriptions
T_i	The moment that task i occurs
F_i	Cloth loading time for the AGV at task i
D_i	Loom waiting time at task i
W_i	The time when the AGV arrives at task i
v_r	Average velocity of the AGV in the roadway
v_m	Average speed of the AGV in the main roads
v_c	Average speed of the AGV in curves
d_{ij}	The shortest distance for the AGV from task i to task j
d_i	Distance from P00 to task i
t_i	Time from P00 to task i
t_{ij}	The minimum time that the AGV requires to travel from task i to task j

3.4. Calculate t_{ij}

The shortest path problem model is one of the most basic and core models in the optimal design of networks. The shortest path problem that satisfies all kinds of path weight constraints can be summed up as an NP-hard problem of multi-objective combinatorial optimization. In this paper, since the velocities of the AGVs in the main road, roadways and curves are different, the route with shortest distance between two points is not necessarily the route with the shortest time consumption. In order to make the AGV execute the task as quickly as possible, we need to find the shortest time

consumption path between any two task points in the network.

In order to ensure the normal operation of AGVs, it is necessary to set the turning radius. The formula for the minimum outer circle turning radius is as follow:

$$R = \sqrt{(l+d)^2 + (r+b)^2} \quad (1)$$

The AGV wheelbase is l , the front suspension is d , the inner ring radius is r , and the width of the AGV is b .

There are many paths between task points in an intelligent weaving workshop. In this paper, we use the Warshall-Floyd [17] algorithm to find the shortest path between two task points.

The basic idea of the Warshall-Floyd algorithm is that for any vertex $v_k \in V$, the shortest path from vertex v_i to vertex v_j is passed or not passed by vertex v_k . Compare the value of the vertex d_{ij} to the vertex $d_{ik} + d_{kj}$. If $d_{ij} > d_{ik} + d_{kj}$, then $d_{ij} = d_{ik} + d_{kj}$, keeping d_{ij} as the shortest distance between the vertex v_i and the vertex v_j for the current search. Repeat this process, and finally, when all vertices v_k are searched, the shortest distance from the vertex v_i to the vertex v_j is obtained.

We construct an undirected weighted graph $G = (V, E)$, where V is a set of vertices that consist of the intersecting points between the curves, the main road and the roadways in the weaving workshop and all the decision points of the layout in the roadway, and E is the side formed by the adjacent vertices that marks the length of each side. According to the following formula, the time that the AGV needs in order to travel on this side is calculated as the weight value of each side of graph G .

$$t_{ij} = \frac{S_c}{v_c} + \frac{S_r}{v_r} + \frac{S_m}{v_m} \quad (2)$$

where S_c , S_r , and S_m , respectively, represent the curve distance, the roadway distance and the main road distance of the sides in Figure G.

The Warshall-Floyd algorithm is used to calculate the shortest time for the AGV to travel between each two task points t_{ij} , and the matrix is expressed as follows:

$$\begin{pmatrix} t_{0,0} & t_{0,1} & t_{0,2} & L & t_{0,49} \\ t_{1,0} & t_{1,1} & t_{1,2} & L & t_{1,49} \\ t_{2,0} & t_{2,1} & t_{2,2} & L & t_{2,49} \\ M & M & M & O & M \\ t_{49,0} & t_{49,1} & t_{49,2} & L & t_{49,49} \end{pmatrix}$$

t_{ij} represents the shortest time between two path points, where a subscript with 0 represents the time between P00 and the weaving task point.

3.5. Mathematical model

3.5.1. Task sequencing based on time priority

Each cloth loading task is generated based on the degree of urgency, which is usually determined by one or more characteristic parameters of the task. In the cloth loading task, the time of the task's occurrence, T_i , and the distance between the task point and P00, d_i , can reflect the task urgency, and so these 2 parameters are the bases for determining the priority of the task. To facilitate the comparison of the size of the emergency coefficient, we unify the units of the two characteristic parameters according to the relation between time and distance and convert d_i to t_i . Thus, the task urgency coefficient is as follow:

$$P_i = \omega_1 T_i + \omega_2 t_i \quad (3)$$

where ω_1 and ω_2 are the weights of the two parameters.

3.5.2. Mathematical modeling

The set of task points is $R = \{0, 1, 2, \dots, i, j, \dots, n\}$, and the 0-1 variable x_{ij} indicates the existence of the path between the task point i and the task point j . The definition of the variable is as follow:

$$x_{ij} = \begin{cases} 1 & \text{AGV from } i \text{ to } j \\ 0 & \text{or} \end{cases}$$

The objective function of path planning is to calculate:

$$\min f = \sum_{i=1}^R D_i + \sum_{i=1}^R \sum_{j=1}^R t_{ij} x_{ij} \quad (4)$$

subject to the following:

$$W_j = (D_i + T_i + F_i + t_{ij}) * x_{ij} \quad (5)$$

$$D_j = \max\{W_j - T_j, 0\} \quad (6)$$

$$\sum_{i=0}^n x_{ij} = 1 \quad j \in R \quad (7)$$

$$\sum_{j=0}^n x_{ij} = 1 \quad i \in R \quad (8)$$

where Eq 5 represents the time for the AGV to reach the task point j ; Eq 6 represents the waiting time of the loom at task point j , which is non-negative; and Eqs 7 and 8 indicate that the AGV serves each task point only once.

4. Proposed algorithm

4.1. Algorithm process

In the traditional GA-PSO algorithm, in order to improve the ability to get the optimal solutions by increasing the population diversity, we can randomly select the location of the cross-mutation of the particle. However, with the increase in population diversity, the convergence speed of the algorithm decelerates. Many scholars have conducted numerous studies [18–26] on how to improve the convergence speed and optimization ability of the algorithm. Moreover, there are some practical problems regarding the path planning of the multi-load AGV in an intelligent weaving workshop. For example, tasks with a higher urgency coefficient will be prioritized with higher probabilities. In a bid to solve this problem, this paper proposed the time-priority GA-PSO algorithm on the premise of preserving the mechanism of the cross-mutation, Introducing the task orderings based on time priority to guide the evolution of particles to a certain extent makes the evolution have a stronger directivity and improves the algorithm's convergence speed.

Step 1: Individual coding based on task ordering

Individual particle coding adopts the method of integer coding based on task ordering. Each particle represents the order in which the cloth loading vehicle performs the loading task. For example, the individual code is [8 7 6 4 3 9 5 1 2 12 11 10], indicating that the cloth loading vehicle executes the task successively with the priority of 8, 7, 6....

Step 2: Calculate the fitness value

The particle is a feasible solution for the planning scheme. The fitness value of the particle is expressed as the objective function's value. Since the cloth loading vehicle must return to the starting and ending point after loading 4 rolls, the actual walking plan of the loading vehicle needs to add the starting and ending points. If the individual code is [8 7 6 4 3 9 5 1 2 12 11 10], the actual walking route of the loading vehicle is [0 8 7 6 4 0 3 9 5 1 0 2 12 11 10 0]. The definition of the variable is as follow:

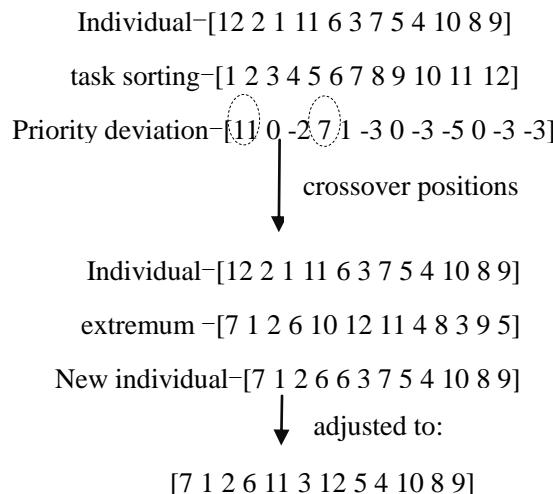
$$fitness(i) = \sum_{i=1}^R D_i + \sum_{i=1}^R \sum_{j=1}^R t_{ij} x_{ij} \quad (9)$$

Step 3: Crossover operation

The crossover method adopts the integer crossover method, and the individual is updated by intersecting it with the extremum of the individual and the extremum of the group. The position of the crossover is selected according to the degree of deviation between the individual and the task sorting, and the two positions in the individual with the largest deviation from the task sorting are selected as the cross positions. New individuals are generated after the intersection and, if there is a

duplicate position, replace the repeated task points with the task points not included in the individual.

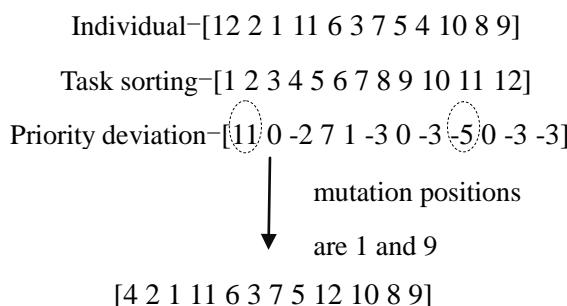
Experiments show that under this crossover rule, the convergence effect of the first few iterations is obvious, but it is also easier to fall into the local optimum. To avoid the algorithm from prematurely converging, if the position with the highest priority deviation is less than the median of the priority, we randomly select the intersection position. The operation method is as follow:



To ensure that the individual after crossing can keep the strategy of the outstanding individual, the particle is updated only when the new particle's fitness value is better than the old one.

Step 4: Mutation operation

It uses the method of exchanging two positions within an individual. Select the two positions where one has a non-negative value with the maximum priority deviation and the other has a non-positive value with the maximum priority deviation as mutation positions and exchange the two mutation positions. The operation method is as follow:



Similarly, in order to ensure that a mutated individual retains the strategies of an excellent individual, the particle is updated only when the fitness of the new particle is better than the old particle. The algorithm flow chart is as follow:

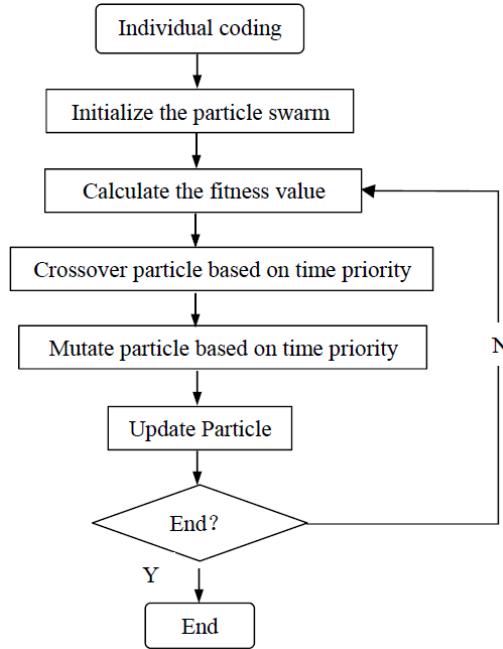


Figure 3. Flowchart of the GA-PSO based on time priority.

5. Experimental results

Taking 12 loading data in a certain period of time as an example, in the algorithm, we set the size of the particle swarm to 300 and the number of iterations to 200.

- (1) Construct the adjacency matrix between the nodes according to the layout of the workshop. The turning radius of the AGV is 2.3 m. Considering the different speeds of the AGV in the main roads, the roadway and the curves, assign the weights to each side accordingly. The Floyd algorithm is used to find t_{ij} , which is the minimum time required for the AGV to travel between nodes. The calculation results are shown in Figure 4.

```

>> floyd
***** The minimum time required for the AGV to travel between nodes *****
c =

```

	0	7.7000	12.8000	17.9000	23.0000	28.1000	33.2000	38.3000	10.9000	16.0000	21.1000
7.7000	0	5.1000	10.2000	15.3000	20.4000	25.5000	30.6000	12.1000	17.2000	22.3000	
12.8000	5.1000	0	5.1000	10.2000	15.3000	20.4000	25.5000	17.2000	22.3000	27.4000	
17.9000	10.2000	5.1000	0	5.1000	10.2000	15.3000	20.4000	22.3000	27.4000	32.5000	
23.0000	15.3000	10.2000	5.1000	0	5.1000	10.2000	15.3000	27.4000	32.5000	37.6000	
28.1000	20.4000	15.3000	10.2000	5.1000	0	5.1000	10.2000	32.5000	37.6000	42.7000	
33.2000	25.5000	20.4000	15.3000	10.2000	5.1000	0	5.1000	37.6000	42.7000	47.8000	
38.3000	30.6000	25.5000	20.4000	15.3000	10.2000	5.1000	0	42.7000	47.8000	52.9000	
10.9000	12.1000	17.2000	22.3000	27.4000	32.5000	37.6000	42.7000	0	5.1000	10.2000	
16.0000	17.2000	22.3000	27.4000	32.5000	37.6000	42.7000	47.8000	5.1000	0	5.1000	
21.1000	22.3000	27.4000	32.5000	37.6000	42.7000	47.8000	52.9000	10.2000	5.1000	0	

Figure 4. Part of the results for solving t_{ij} using the Floyd algorithm.

(2) By calculating the path node and root corresponding to the loading cloth loom and calculating T_i according to the loading time, the emergency coefficient of each task can be calculated. The calculation results are shown in Table 1.

Table 2. The original data.

Loading time	Loom ID	$T_i(S)$	node	P_i	number
10/24 0:01	62	60	20	99.6	1
10/24 0:12	88	720	4	743	2
10/24 0:13	49	780	28	817.9	3
10/24 0:14	48	840	27	882.8	4
10/24 0:16	23	960	37	988.8	5
10/24 0:21	1	1260	43	1314.3	6
10/24 0:40	74	2400	12	2431.3	7
10/24 0:43	35	2580	28	2617.9	8
10/24 1:00	72	3600	10	3621.1	9
10/24 1:19	89	4740	5	4768.1	10
10/24 1:24	26	5040	40	5084.1	11
10/24 1:32	69	5520	13	5556.4	12

(3) Use the algorithmic logic proposed in this paper to calculate the order of the AGV to perform the loading task. The calculation result is shown in Figure 5. After comparing the GA-PSO algorithm for randomly selecting the crossover mutation position, the convergence effect is shown in Figure 6. The fitness value of the optimal solution after 10 runs of the algorithm is shown in Table 2, the overall operational result of the algorithm is shown in Table 3.

```
***** Best Individual *****
tourGbest =
1 4 3 2 5 6 11 8 10 9 7 12
***** Path of AGV *****
Best_Route =
0 20 27 28 4 0 37 43 40 28 0 5 10 12 13 0
*****: The optimal solution of the objective function *****
value =
795.6000
```

Figure 5. Results of path planning.

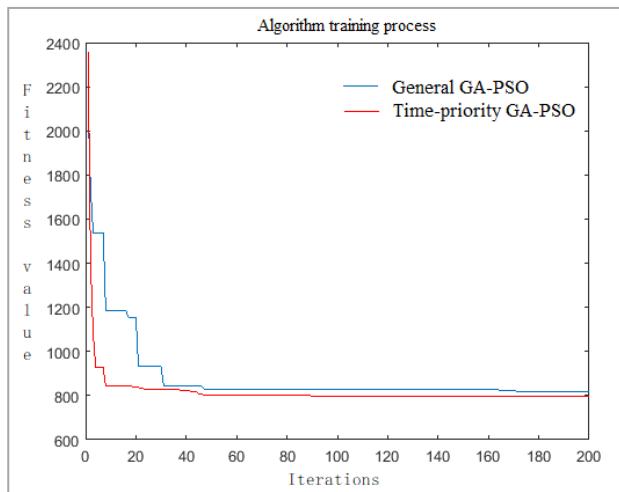


Figure 6. Comparison of the algorithms' convergence effects.

Table 3. The fitness value of the algorithm after 10 runs.

Algorithm	1	2	3	4	5	6	7	8	9	10
General GA-PSO	801.4	795.6	833.1	811.5	795.6	809.8	795.6	815.4	798.5	845.6
Time-priority GA-PSO	795.6	795.6	795.6	810.8	795.6	795.6	795.6	795.6	795.6	795.6

Table 4. The overall operational results.

Algorithm	Run times	Optimal	Average
General GA-PSO	10	795.6	810.2
Time-priority GA-PSO	10	795.6	797.1

6. Conclusion

The planning of the multi-load AGV is an NP-hard problem. This paper proposes a task-sorting strategy based on the time priority with consideration of the timing of the task in the weaving workshop. In addition, it applies this strategy to guide the evolutionary direction of the particle in the GA-PSO algorithm to obtain a GA-PSO algorithm with faster convergence and a better search ability. Figure 7 and 8 are comparison of the algorithm results for 12 tasks and 40 tasks, Figure 9 and 10 are comparison of the iterations versus the values of PR for 12 tasks and 40 tasks, The value of PR is the number that the algorithm optimization results under the current iteration divided by the referenced optimal solution, the smaller the value of PR, the algorithm convergence and optimization ability better. The simulation results show that the GA-PSO algorithm proposed in this paper shows better convergence speed and optimization stability when dealing with a small number of tasks. When the number of tasks is large, it can find not only the approximate optimal solution with a faster convergence speed but also the GA-PSO algorithm with a better ability than random selection for cross-variation positioning. The proposed algorithm can be applied to the multi-load AGV path planning problem in a weaving workshop.

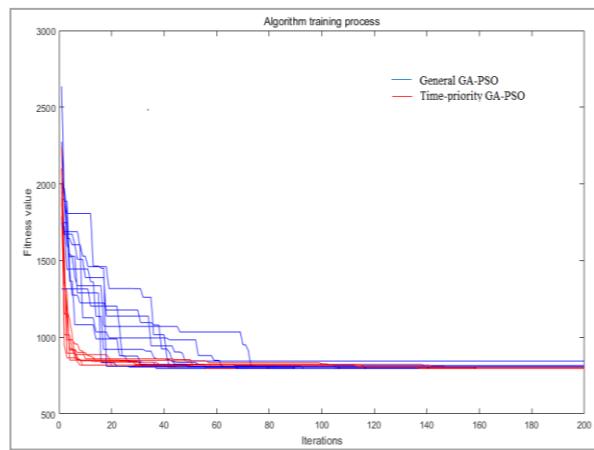


Figure 7. Comparison of the algorithm for 12 tasks.

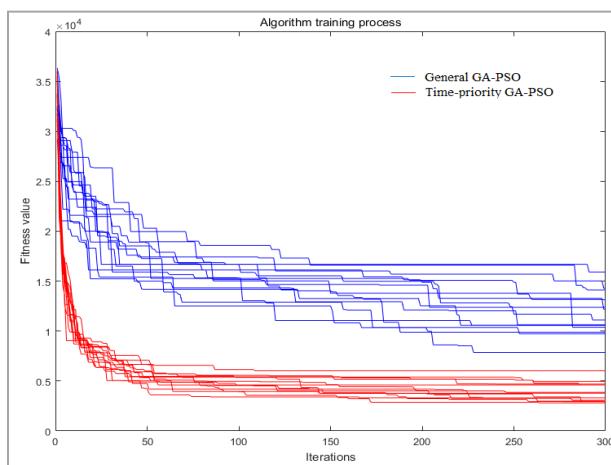


Figure 8. Comparison of the algorithm for 40 tasks.

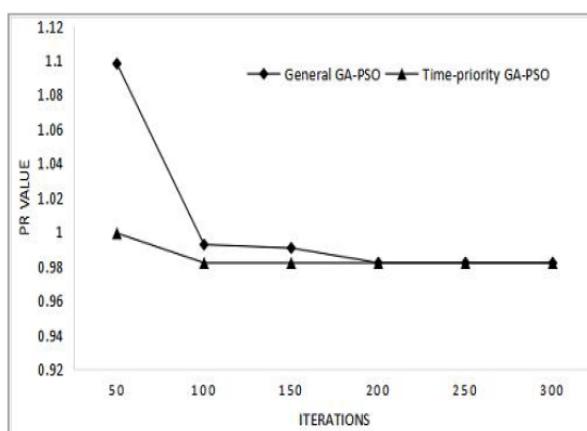


Figure 9. Comparison of the iterations versus the values of PR for 12 tasks.

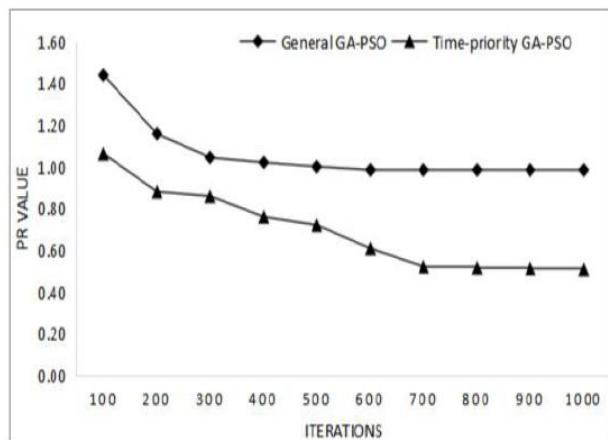


Figure 10. Comparison of the iterations versus the values of PR for 40 tasks.

In addition, compared with the actual production situation, applying the planning method in this study can reduce the cloth failure rate of the workshop from 15% to 2.5%.

The correctness and feasibility of the multi-load AGV path planning method proposed in this paper is apparent. It ensures that the scheduling is carried out in an orderly manner while improving the utilization rate of the machines. This has important practical significance to reduce the costs of loading cloth, to improve the efficiency of loading cloth and to improve production efficiency.

Acknowledgments

This study was funded by Hubei provincial key laboratory of digital textile equipment, project number DTL2017010.

Conflict of interest

The authors declare no conflict of interest in this paper.

References

1. H. Fazlollahtabar and M. Saidi-Mehrabad, Models for AGVs' scheduling and routing, *Autonomous Guided Vehicles*, **20** (2015), 1–15.
2. G. M. Li, X. Y. Li, L. Gao, et al., Tasks assigning and sequencing of multiple AGVs based on an improved harmony search algorithm, *J. Amb. Inter. Hum. Comp.*, 2018. Available from: DOI, 10.1007/s12652-018-1137-0.
3. D. H. Lee, Resource-based task allocation for multi-robot systems, *Robot Auton. Syst.*, **103** (2018), 151–161.
4. T. J. Chen, Y. Sun, W. Dai, et al., On the shortest and conflict-free path planning of multi-AGV system based on dijkstra algorithm and the dynamic time-window method, *Adv. Mater. Res.*, **645** (2013), 267–271.

5. A. Kaplan, N. Kingry, P. Uhing, et al., Time-optimal path planning with power schedules for a solar-powered ground robot, *IEEE T. Autom. Sci. Eng.*, **14** (2017), 1235–1244.
6. N. Smolic-Rocak, S. Bogdan, Z. Kovacic, et al., Time windows based dynamic routing in multi-AGV systems, *IEEE T. Autom. Sci. Eng.*, **7** (2010), 151–155.
7. K. G. Huo, Y. Q. Zhang, Z. H. Hu, et al., Research on scheduling problem of multi-load AGV at automated container terminal, *J. Dalian Univ. Techno.*, **6** (2016), 243–251.
8. G. M. Li, B. Zeng, W. Liao, et al., A new AGV scheduling algorithm based on harmony search for material transfer in a real-world manufacturing system, *Adv. Mech. Eng.*, **10** (2018), 1–13.
9. M. Mousavi, H. J. Yap, S. N. Musa, et al., Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization, *PLoS One*, **12** (2017), 1–24.
10. F. Hamed and M. Nezam, An optimal path in a bi-criteria AGV-based flexible job shop manufacturing system having uncertain parameters, *Int. J. Ind. Syst. Eng.*, **13** (2010), 27–55.
11. H. Fazlollahtabara and M. Saidi-Mehrabad, Optima path in an intelligent AGV based manufacturing system, *Transp. Lett.*, **7** (2015), 219–228.
12. V. K. Chawla, A. K. Chanda, S. Angra, et al., Multi-load AGVs scheduling by application of modified memetic particle swarm optimization algorithm, *J. Braz. Socy. Mech. Sci.*, 2018. Available from: [10.1007/s40430-018-1357-4](https://doi.org/10.1007/s40430-018-1357-4).
13. S. Bakshi, Z. Y. Yan, D. M. Chen, et al., A fast algorithm on minimum-time scheduling of an autonomous ground vehicle using a traveling salesman framework, *J. Dyn. Syst. Meas. Control*, 2018. Available from: DOI, [10.1115/1.4040665](https://doi.org/10.1115/1.4040665).
14. H. Fazlollahtabar and S. Hassanli, Hybrid cost and time path planning for multiple autonomous guided vehicles, *Appl. Intell.*, **48** (2018), 482–498.
15. M. Grunow, H. Gunther, M. Lehmann, et al., Dispatching multi-load AGVs in highly automated seaport container terminals, *OR Spectrum*, **26** (2014), 211–235.
16. T. J. Park, J. W. Ahn, C. S. Han, et al., A path generation algorithm of an automatic guided vehicle using sensor scanning method, *J. Mech. Sci. Technol.*, **16** (2002), 137–146.
17. H. Y. Wang, Q. Huang, C. T. Li, et al., Graph theory algorithm and MATLAB simulation, BeiJing (2009), Beihang University Press.
18. A. Ali and M Tawhid, A hybrid particle swarm optimization and genetic algorithm with population partitioning for large scale optimization problems, *Ain. Shams. Eng. J.*, **8** (2017), 191–206.
19. X. Y. Li and L. Gao, An effective hybrid genetic algorithm and tabu search for flexible job shop scheduling problem, *Int. J. Prod. Econ.*, **174** (2016), 93–110.
20. P. B. C. Miranda and R. B. C. Prudêncio, Generation of particle swarm optimization algorithms: An experimental study using grammar guided genetic, *Appl. Soft. Comput.*, **60** (2017), 281–296.
21. S. Hamed and K. Govindan, A hybrid particle swarm optimization and genetic algorithm for closed loop supply chain network design in large scale networks, *Appl. Math. Model.*, **39** (2015), 3990–4012.
22. K. Deb and N. Padhye, Enhancing performance of particle swarm optimization through an algorithmic link with genetic algorithms, *Comput. Optim. Appl.*, **57** (2014), 761–794.

23. X. Y. Li, L. Gao, Q. K. Pan, et al., An effective hybrid genetic algorithm and variable neighborhood search for integrated process planning and scheduling in a packaging machine workshop, *IEEE T. Syst. Man. Cy. S.*, 2018.
24. A. M. Bertram, Q. Zhang, S. C. Kong, et al., A novel particle swarm and genetic algorithm hybrid method for diesel engine performance optimization, *Int. J. Engine. Res.*, **17** (2016), 732–747.
25. X. Y. Li, C. Lu, L. Gao, et al., An effective multi-objective algorithm for energy efficient scheduling in a real-life welding shop, *IEEE T. Ind. Inform.*, **14** (2018), 5400–5409.
26. Y. N. Yan, L. Z. Liu, B. Y. Luo, et al., Arrangement of garment production line by particle swarm algorithm, *J. Text. Res.*, **39** (2018), 120–124.



AIMS Press

©2019 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0>)