Scholar Publishing Group
International Journal of Big Data Intelligent Technology
https://doi.org/10.38007/IJBDIT.2023.040103
ISSN 2790-0932 Vol. 4, Issue 1: 17-25

Multi-AGV Task Scheduling Method for Intelligent Warehousing

Huiheng Suo[1,3,a], Qiang Hu[1,b], Jian Wu[1,c*], Xie Ma[2,d], Youxuan Cai[2,e], Shiai Bi[3,f], Jingwen Zhang[3,g], Xiushui Ma[3,h*]

[1]*Nanchang Hangkong University, Nanchang, China*

[2]*Ningbo University of Finance & Economics, Ningbo, China*

[3]*NingboTech University, Ningbo, China*

[a]*suohuiheng@163.com*, [b]*1586270278@qq.com*, [c]*flywujian@qq.com*, [d]*maxie@163.com*, [e]*1337367538@qq.com*, [f]*2246177205@qq.com*, [g]*1729629241@qq.com*, [h]*mxsh63@aliyun.com*

[*]*corresponding author*

Keywords: Intelligent Storage, AGV, Task Scheduling, PSO

Abstract: For the collaborative control of automatic guided vehicles (AGVs) in logistics and warehousing, task scheduling is critical for improving the efficiency of AGV execution. To address this issue, this paper first establishes a mathematical model for multi-task AGV task scheduling and proposes an efficiency evaluation index. Then, a particle swarm optimization (PSO) based order task scheduling algorithm is proposed. This algorithm assigns matching values to sort stations based on orders, and then sorting stations allocate tasks to AGVs. Finally, simulation results show that this algorithm effectively coordinates the operation of the entire intelligent warehouse, verifying the correctness of the algorithm and providing a solution for practical warehouse task scheduling problems.

1. Introduction

With the rapid development of the e-commerce industry and continuous advances in artificial intelligence technology, efficient and accurate logistics has become an indispensable factor for enterprises in today's highly competitive market. In order to meet market demands, modern intelligent warehousing systems have gradually transitioned from traditional "picking" modes to more efficient "goods-to-person" operation modes. Compared with traditional picking mode, an intelligent picking mode system not only has higher efficiency, but also can reduce the labor intensity of workers, thereby having prominent advantages in saving labor resources and other aspects [1]. In intelligent warehousing systems, the main replacement for labor is automated guided vehicles (AGVs), which can improve the overall operation efficiency of the warehouse and reduce the impact of human factors on logistics efficiency. Currently, multi-AGV-based intelligent warehousing systems have become an important research hotspot in warehousing and logistics [2]. By introducing multiple AGVs, an intelligent warehousing system can achieve more efficient logistics operations, thereby improving the processing efficiency and operational benefits of the entire warehouse.

2. Mathematical Models for Multi-AGV Task Scheduling

2.1. Scene Description

Assuming that there are M sorting stations Sta_i ($i \in [1, M]$) distributed in a square warehouse area $[0,s]\times[0,s]$ with a side length of s. Each sorting station Sta_i has a sorting task quantity of TG for a set of tasks represented by S_{TG}, so $S_{TG}=\{1,2,3,...,TG\}$. The number of shelves is represented by $\mathrm{shelf}Num$, and different types of products GS correspond to different shelf positions Pos_i, where $i \in [1, TG]$. Each sorting station Sta_i is assigned $N_{rob}=\{rob_1, rob_2, ..., rob_N\}$ AGVs to complete the picking of goods in the order.

The order tasks are initially unknown. Customers first make purchases of various products, and then generate corresponding orders, which are assigned to sorting stations with similar matching degrees. The sorting stations then allocate the picking tasks of various products in the order to AGVs for real-time processing. The objectives of multi-AGV task assignment include two aspects:

(1) Timely response and processing of received tasks;

(2) Reasonably allocate the resources of sorting stations and multi-AGVs to maximize task execution efficiency.

2.2. Task Allocation Pattern

Assuming there are multiple AGVs waiting in the waiting area of the warehouse and sorting stations located at their corresponding unloading points. The order tasks are assigned to sorting stations based on their current operating status. When assigning order tasks, they are allocated based on the matching value between the order and the sorting station.

Assuming that the product categories in the entire intelligent warehouse are represented by GS, and the product categories of a certain order can be represented by $OG=(p_1, p_2, p_3, ..., p_{GS})$, $p_i=0,1$. If the target product is in the task, it is 1, otherwise it is 0.

The total distance from the shelves where all the goods in an order are located to the corresponding sorting station is $H(o_g)$:

$$H(O_g)=\sum_{i=1}^{GS} H(O_i) \tag{1}$$

$$H(O)=|x-x_{sta}|+|y-y_{sta}| \tag{2}$$

Where, $H(O)$ represents the Manhattan distance from the shelf where the product is located to the sorting station.

The distance $Dis\left(o_g\right)$ between adjacent shelves corresponding to two adjacent products can be represented as:

$$Dis\left(O_g\right)=\sum_{i=1}^{GS} StaG\left(i\right)\times h\left(i,i+1\right) \tag{3}$$

$$h\left(i,i+1\right)=\left|x-x_{i+1}\right|+\left|y-y_{i+1}\right| \tag{4}$$

Where, $h(i,i+1)$ represents the Manhattan distance between the shelves corresponding to two products.

$StaG=\left(p_1,p_2,p_3,\ldots,p_{GS}\right)$ refers to the types of goods that the sorting station needs to process, The matching value between the order O_n and the sorting station Sta_m is:

$$OF_m=OG_n\times StaG_m^T \tag{5}$$

To ensure timely processing of orders in the system, a maximum order waiting time T_w is set, and the matching value of order O_n is defined as:

$$OP_{mn}=\frac{rand()\times\dfrac{OF_{mn}}{\left\|OG_n\right\|_1}+\left(1-rand()\right)\times\dfrac{t-T_{O_nre}}{T_w}}{H\left(O_g\right)+Dis\left(O_g\right)} \tag{6}$$

Where, $rand()$ is the weight coefficient, t represents the current time of the system, and T_{O_nre} represents the system time when the sorting station receives the order O_n.

When assigning order tasks to sort stations, the set of tasks TG is recalculated based on the product categories $S_{TG}=\{1,2,3,\ldots,TG\}$. The distance from the shelf a to the shelf b during movement is defined as Dis_{ab}, where, $\mathrm{a,b}\in[1,TG]$. The evaluation index is the average distance between AGV and the station or shelf, which is calculated as follows:

$$Fo=\frac{\sum_{k\in N} Dis_k}{N_{work}} \tag{7}$$

Where, Dis_k represents the total distance traveled by $AGVk$ during task execution in the scheduling, and N_{work} represents the number of AGVs still in the task after the scheduling ends. Obviously, the smaller the Fo indicator, the higher the overall efficiency of the system.

2.3. FIFO

The task scheduling FIFO strategy refers to the "first in, first out" policy, which means that tasks are processed in the order they are submitted based on their submission time. The earlier submitted tasks are processed first, and the later submitted tasks are processed later. The FIFO strategy is simple and easy to implement, and it ensures the fairness of task processing, avoiding situations where some tasks are left unprocessed for a long time. However, the FIFO strategy ignores the priority of tasks and cannot guarantee timely processing of high-priority tasks, so it may not be suitable for some scenarios.

2.4. Particle Swarm-Based Task Scheduling

2.4.1. PSO

The particle swarm algorithm originated from the study of bird hunting behavior. It simulates birds in a flock by designing a weightless particle with only two attributes: velocity and position. Velocity represents the speed of movement, and position represents the direction of movement [3]. Each particle individually searches for the optimal solution in the search space, and remembers it as its current individual extreme value. The individual extreme value is shared among all particles in the swarm, and the best individual extreme value is found as the current global optimum solution for the entire swarm. All particles in the swarm adjust their velocity and position based on their own current individual extreme value and the current global optimum value shared by the entire swarm.

Each particle has the following features:

1. Each particle has its own position and velocity attributes.
2. Each particle can know its own extreme value and the extreme value of other particles in the swarm.

During each iteration, the movement direction and velocity of the particle can be chosen from the following three possibilities:

1. Move in the same direction as the particle's original direction.
2. Move in the direction of its historical best position.
3. Move in the direction of the best position in its neighborhood.

This fusion of choices can be represented as follows:

$$v_{t+1} = c_1 v_t + c_2 \times \mathrm{rand}() \times \left(p_{i,t} - x_t\right) + c_3 \times rand() \times \left(p_{g,t} - x_t\right) \tag{8}$$

$$x_{i+1} = x_i + v_{i+1} \tag{9}$$

Where, $i = 1, 2, ..., N$, N is the total number of particles in the particle swarm. The first part represents the influence of velocity magnitude and direction. The second part is a vector pointing from the current point to the particles own best point, representing the part of the particle's action that comes from its own experience [4]. The third part is a vector pointing from the current point to the best point of the population, reflecting the collaborative cooperation and knowledge sharing among particles. v_t represents the velocity at time t, x_t represents the position at time t, $rand()$ represents a random coefficient between 0 and 1, $p_{i,t}$ represents the historical best position of the current particle at time t, $p_{g,t}$ represents the historical best position at time t. c_1, c_2, c_3 are trust coefficients, generally, $c_1 = 1$, $c_2 = c_3 = 2$, c_1 represents the trust level of the particle in its current motion state; c_2 represents the trust level in its own experience; c_3 represents the trust level for information sharing among particles in the swarm [5].

The update of particles is shown in Figure 1:

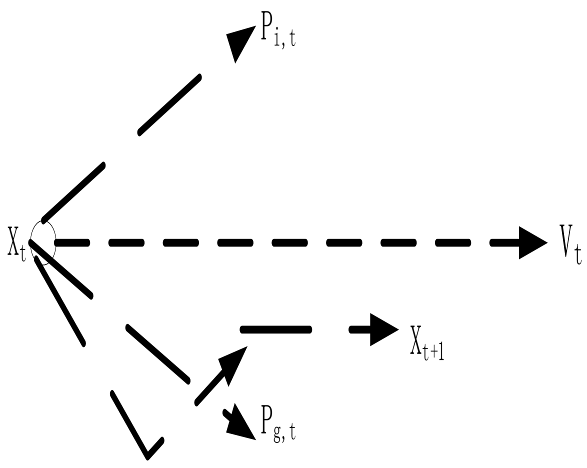

Figure 1. Particle update illustration

2.4.2. Particle Swarm Algorithm Flowchart

Step 1: Randomly initialize particles.
Step 2: Evaluate each particle and obtain the global best.
Step 3: Check if the result from Step 2 or Step 7 meets the termination condition. If yes, exit; otherwise, proceed to Step 4.
Step 4: Update the velocity and position of each particle.
Step 5: Evaluate the fitness value of each particle.
Step 6: Update the personal best position of each particle.
Step 7: Update the global best position of the swarm and go back to Step 3

3. Simulation Experiment Analyses

In order to demonstrate the correctness of the proposed model and the task scheduling effectiveness of the algorithm, simulation experiments were designed to compare the algorithm proposed in this paper with traditional $FIFO$ strategies in terms of task completion efficiency and AGV travel distance.

3.1. Simulation Parameter Settings

The main variable in the simulation is the number of AGVs, ranging from 20 to 200 units, increasing by 10 units at a time, with a total order quantity set at 1000.

3.2. Analysis of Simulation Results

Through the above configuration parameters, compared with the order processing method of the traditional strategy under the same conditions, the initial state of the simulation experiment is shown in Figure 2, and all AGVs in the figure are waiting for instructions from the host computer in the standby area.

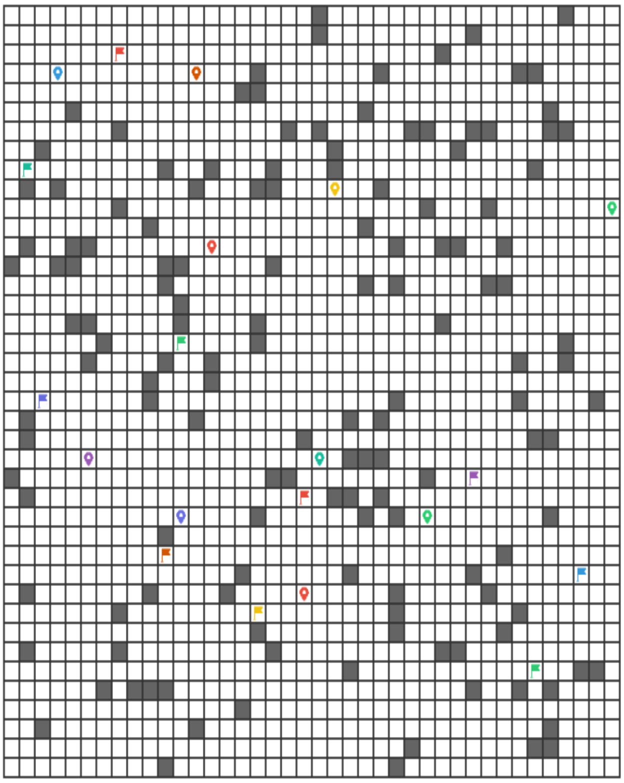

Figure 2. Initial state of simulation experiment

The simulation experiment process is shown in Figure 3. The AGV assigned to the order starts to perform the handling task. The results of the simulation experiment are shown in Figure 4. All AGVs assigned to the order transport the goods to the target point.

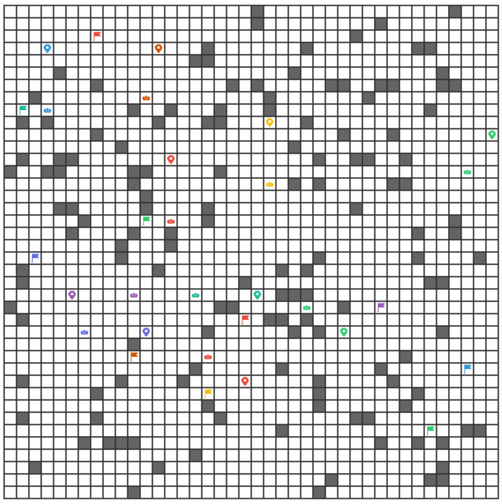

Figure 3. Simulation experiment process diagram

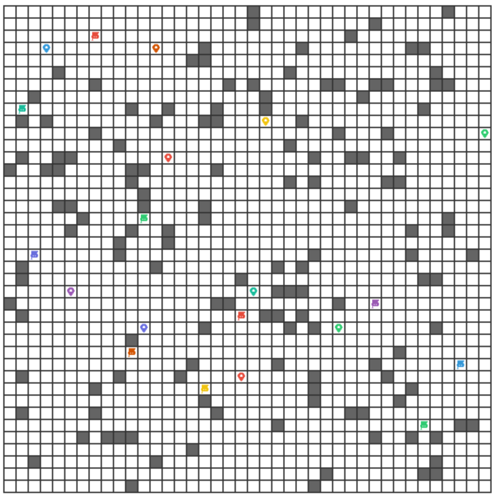

Figure 4. Simulation experiment results

The running time results of the order scheduling strategy system are shown in Figure 5.

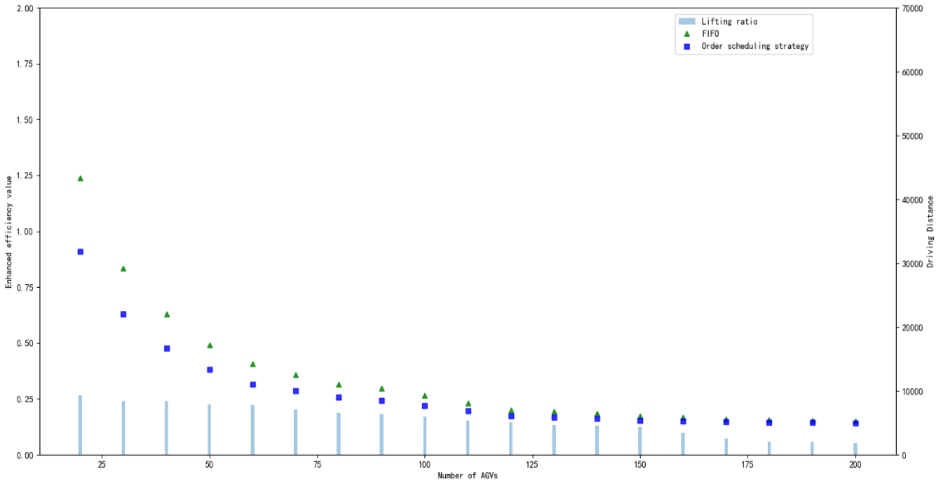

Figure 5. System Running Time of Order Scheduling Strategy

Observing Figure 5, it can be seen that under the same total task volume, when the number of AGVs is less than 100, the runtime of the order scheduling strategy using the particle swarm algorithm is significantly reduced compared to the traditional FIFO strategy. However, when the number of AGVs exceeds 150, the improvement of the algorithm's performance relative to the traditional FIFO strategy is not significant. Therefore, when the number of AGVs is small, using this scheduling algorithm can improve efficiency and reduce the number of handling operations.

Under the same experimental conditions, the comparison of robot running distance for the order scheduling strategy is shown in Figure6.

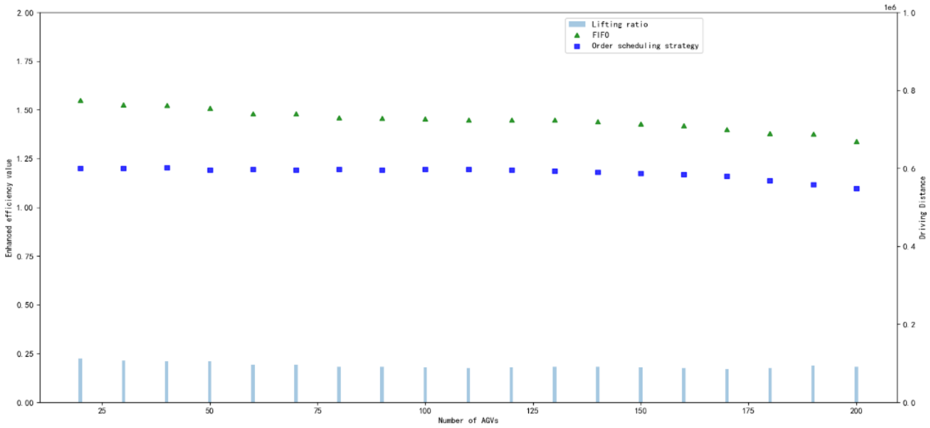

Figure 6. Order scheduling strategy and AGV travel distance

Figure 6 shows that the use of the particle swarm algorithm reduces the travel distance of AGVs between shelves, but the improvement in time efficiency is not as significant. This is because as the number of AGVs increases, the occurrence of path conflicts also increases, resulting in increased time spent on handling conflicts and overall time consumption. However, the overall order efficiency is still improved.

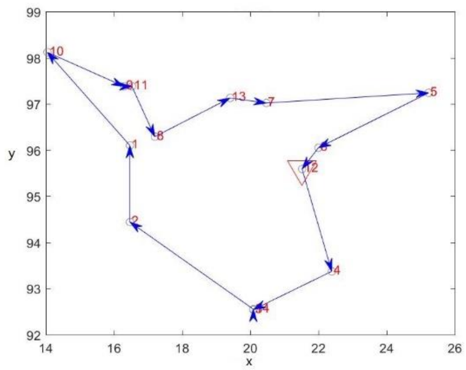

Figure 7. Trajectory using traditional strategy

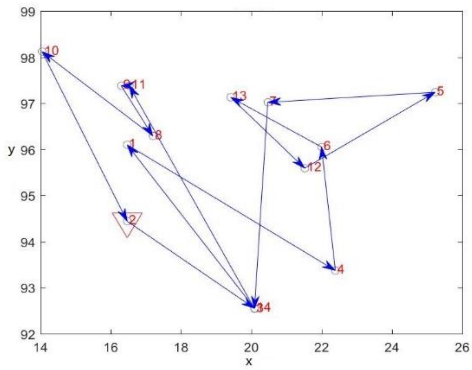

Figure 8. Trajectory using PSO

Figure 7 shows the random route map before optimizing task allocation for a single AGV, Figure 8 shows the optimized route map using the improved discrete particle swarm algorithm.

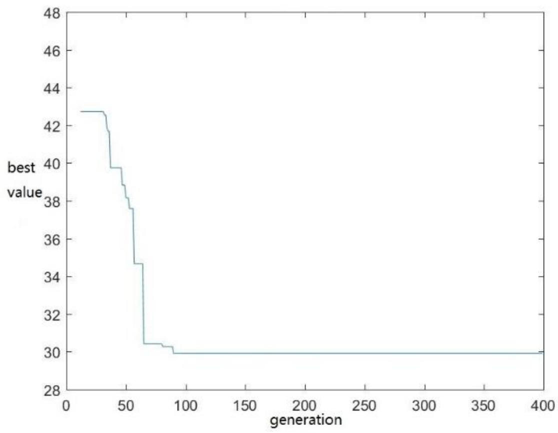

Figure 9. Algorithm convergence process

Figure 9 shows the convergence speed of the improved discrete particle swarm algorithm.

4. Conclusion

This paper takes multi-AGV task allocation in intelligent warehousing environment as the research goal, and discusses how to cooperate to complete warehousing tasks. Firstly, a mathematical model is established for multi-AGV task scheduling, and scene description, task allocation mode and efficiency evaluation index are proposed. Then, particle swarm optimization is introduced. The flow chart is generated by the established mathematical model and particle swarm optimization algorithm and the simulation experiment is designed. The experimental results show that the algorithm has improved execution time and AGV running distance.

Funding

This paper is supported by Projects of major scientific and technological research of Ningbo City(2020Z065, 2021Z059,2022Z056,2023Z050(the second batch)), Major instrument special projects of the ministry of science and technology of China(2018YFF01013200), Projects of major scientific and technological research of Beilun District, Ningbo City(2021BLG002, 2022G009), Projects of engineering research center of Ningbo City (Yinzhou District Development and Reform Bureau [2022] 23), Projects of scientific and technological research of colleges student's of China(202213001008).

Data Availability

Data sharing is not applicable to this article as no new data were created or analysed in this study.

Conflict of Interest

The author states that this article has no conflict of interest.

References

[1] *Ren Fang. The ' goods to people ' picking scheme and its innovative development. Logistics Technology and Applications, 2017, 22(10): 80-84.*

[2] *Kong Qingqing. The development outline and trend of intelligent AGV in warehousing and logistics industry. Fortune Today, 2018(24).*

[3] *Chen T, Zhu Y, Teng J. Beetle swarm optimisation for solving investment portfolio problems. The Journal of Engineering, 2018, 2018(16): 1600-1605.*

[4] *Nie S K, Wang Y J, Xiao S, et al. An adaptive chaos particle swarm optimization for tuning parameters of PID controller. Optimal Control Applications and Methods, 2017, 38(6): 1091-1102.*

[5] *Li Dan. Research on a hybrid active power filters .Hefei University of Technology, 2010.*