ISSN: 2664-9640

Scheduling Research of Unmanned Warehouse Handling Robot Based on Improved A* Algorithm

Luyuan Wang

College of energy and power, Shenyang Institute of Engineering, Shenyang Liaoning Province, 110136, China

Abstract

The AGV system is an automatic handling system that integrates a variety of high technologies and is mainly used for loading and unloading of goods in warehouses, and can be quickly connected to the storage system, which greatly improves the automation of storage and transportation. In the course of daily work, the robot is mainly responsible for transporting goods between picking platforms and storage spaces. Inevitably, however, due to problems with the physical facility parameters and the scheduling of the AGVs, the robots often collide or deadlock, which reduces transport efficiency and causes damage to the equipment. Therefore, in this paper, the design and scheduling of AGV systems are studied in depth.

Keywords

A* algorithm; ant colony algorithm; temporal conflict constraint; topological-raster map.

1. Introduction

With the rise of e-commerce, unmanned warehouses have gradually become the development direction and goal of automated warehouse logistics systems. The earliest ones include Amazon's Kiva robots, and later AGV handling robots, SHUTTLE shelf shuttles, DELTA sorting robots and other various, highly automated robots are tailored for the unmanned warehouse. The scheduling problem of handling robots in unmanned warehouses is one of the core issues. Task balancing area division To better balance the load of picking stations and to prevent local congestion of handling robots, the warehouse map is dynamically partitioned according to the number of picking stations and goods in stock[1]. That is, a default picking station is assigned to each pallet on the storage position in the warehouse. In this paper, an optimization model is developed so that the total amount of goods corresponding to the pallets at each picking station is as even as possible, while the sum of the distances from all pallets to their default picking stations is minimized. Further, the workload of each picking station in a certain period of time is more rationally balanced according to the inventory distribution of the required goods in a certain period of time, combined with the AGV scheduling algorithm[2].

2. Assumptions and notations

2.1. Assumptions

We use the following assumptions.

(1) AGV power is sufficient, i.e., the impact of power on the trolley is not considered.

(2) The warehouse inventory can meet the order demand at all times.

(3) AGVs can sufficiently bear the weight of pallets and goods.

(4) Each AGV can carry only one shelf at a time.

(5) All AGVs are of the same quality and move at a uniform speed of one unit per second.

(6) AGVs can only walk through each node once when performing the same set of tasks.

ISSN: 2664-9640

3. Model construction and solving

3.1. Modeling the scheduling time of unmanned warehouse robots

An unmanned warehouse robot scheduling model is developed here.

Objective function.

$$\text{Min } Z = \max_{k\in K}\left[\sum_{i\in W}\sum_{j\in W, j\neq i} X^k_{ij} T^k_{ij} + \sum_{i\in N} Z^k_i S_i + \sum_{i\in N} T^k_{o_k i} + \sum_{i\in N} A_i\right] \tag{1}$$

Binding Conditions.

$$\sum_{j\in N} X^k_{o_{kj}} = 1 \quad \forall k\in K \tag{2}$$

$$\sum_{i\in N} X^k_{io_k} = 1 \quad \forall k\in K \tag{3}$$

$$\sum_{k\in K}\sum_{j\in W, j\neq i} X^k_{ij} = 1 \forall i\in N \tag{4}$$

$$\sum_{k\in K}\sum_{i\in W, i\neq j} X^k_{ij} = 1 \forall j\in N \tag{5}$$

$$\sum^k_{ih} - \sum_{j\in W, j\neq h} X^k_{hj} = 0 \forall k\in K, h\in W \tag{6}$$

$$Z^k_i = \sum_{j\in W, j\neq h} X^k_{ij} \forall i\in N, k\in K \tag{7}$$

$$T^k_{ij} = \frac{|X_i - X_j| + |Y_i - Y_j|}{V_k} \quad \forall k\in K, i\in N, j\in N \tag{8}$$

$$T^k_{o_k i} = \frac{|M_k - X_i| + |N_k - Y_i|}{V_k} \forall k\in K, i\in N \tag{9}$$

$$T^k_{o_k i} = \frac{|M_k - X_i| + |N_k - Y_i|}{V_k} \forall k\in K, i\in N \tag{10}$$

$$S_i = S^g_i + S^s_i + S^r_i \quad \forall i\in N \tag{11}$$

$$F_k = \sum_{i\in W}\sum_{j\in W, j\neq i} X^k_{ij} T^k_{ij} + \sum_{i\in N} Z^k_i S_i \quad \forall k\in K \tag{12}$$

$$A_j = \sum_{k\in K}\sum_{i\in W, i\neq j} X^k_{ij}\left(A_i + S_i + T^k_{ij}\right) \quad \forall j\in N \tag{13}$$

$$\sum_{j\in W} U^k_{ji} - \sum_{j\in W} U^k_{ij} = Z^k_i \quad \forall k\in K, i\in N \tag{14}$$

$$U^k_{ji}(n-1)X^k_{ij} \quad \forall k\in K, i\in A, j\in W \tag{15}$$

$$X^k_{ji}, Z^k_i \in \{0,1\} \quad \forall k\in K, i\in A, j\in W \tag{16}$$

$$U^k_{ij} \geqslant 0 \quad \forall k\in K, i\in A, j\in W \tag{17}$$

$$A_j \geqslant 0 \quad \forall j\in N \tag{18}$$

$$F_k \geqslant 0 \quad \forall k\in K \tag{19}$$

The decision of the model is the distribution of the shelf handling tasks on the AGVs and the order in which each AGV handles the shelves. The robot docking position indicates the robot's starting position, and we assume that the robot will return to the original starting position after completing all tasks. The objective function (1) minimizes the time to complete the task for the longest AGV, which is the time for the AGV to return to the robot park after completing all pallet handling tasks from the docking point, and can also be expressed as equations (2) and (3), which represent the time for the AGV to return to the robot park after completing tasks from the robot park; equations (4) and (5) ensure that each shelf is visited once. Equation (6) is the line flow balance constraint; Equation (7) is the service relationship between the AGV and the racks; Equation (8) is the distance time relationship between the AGV and the two racks; Equations (9) and (10) represent the distance time relationship between the AGV and the pallets and the dock; Equation (11) represents the total time required for the AGV to carry the pallets to and from the work platform is equal to the time required for the AGV to transport the pallets to the work platform Equation (12) indicates that the time required for the AGV to complete all tasks and return to the robot dock is equal to the sum of the time it takes to travel between the racks and the total time required to carry the racks to and from the platform; Equation (13) indicates that the time for the racks to start moving is equal to the total time required for the AGV to start moving the last pallet and to return to the picking platform. and the total time required to travel between the two racks. This constraint is only to illustrate the relationship between these times and can be excluded from the model solution; Eqs. (14) and (15) are elimination subcycle constraints; Eq. (16) represents 0-1 variables; Eqs. (17)-(19) represent integer variables. This problem can be reduced to the Assignment Problem, which is a difficult NP problem[3].

3.2. Scheduling modeling

According to related studies, when the routing strategy is determined and the traffic load exceeds a certain determined value per unit time, the network traffic flow quickly falls into a congested state. The biggest reason for local congestion is the uneven traffic flow distribution, where some sections have reached the bottleneck state, while some sections are in a non-saturated or sparse traffic state. Therefore avoiding excessive load per unit node is the key to solve congestion. In order to avoid excessive load in a certain area, AGV path planning needs to be considered for load balancing[].

According to the previous introduction, when the traditional *A** algorithm is used, i.e., load balancing is not considered, there will be a high load area concentrated near the exit, and the load in the row near the exit is similar to the bell-shaped curve distribution, and the load in the left and right direction of the exit row near the middle is also significantly higher than other areas. The high load area has a higher chance of congestion, which will cause inefficient operation of the AGV system and affect the overall AGV system operation. The essence of the AGV path planning method under consideration of load balancing is to take the load factor into account in the actual path cost based on the traditional *A** algorithm, i.e., the new cost function g(x) is improved by introducing the load factor, which can be expressed by equation (20).

$$g(x) = l(x) + \alpha \log d(x) \tag{20}$$

where l(x) denotes the actual distance from the initial node to node x and load(x) tables the load of node x. α is the conversion factor to convert load to load cost, and in subsequent experiments, different values of α are tested to find the optimal α value. When α=0 that is the case when load balancing is not considered. The principle of the simulation experiment is as follows: Keeping the starting and ending points of 4000 AGV tasks unchanged, the comparison examines the load of the road network with different load coefficients considering the load balancing case, and when the load coefficient is 0 is the traditional *A** algorithm, that is, no load balancing is considered. In the implementation step, the first AGV uses the path planning method with the improved *A** algorithm considering load balancing to carry out path planning, and then updates the load data in the road network according to the results of this path planning, and on the basis of this updated road network load, the subsequent AGVs use the same method to carry out path planning and update the road network load in turn until all AGVs

ISSN: 2664-9640

finish path planning and update the road network. until all the AGVs complete the route planning and update the road network. The experiment is conducted in a one-way multiple-input multiple-output road network model. In equation (20), let α = 0 for the case of no load balancing. In the 10×10 unidirectional multi-entry and multi-exit road network model, the values of Figure 1 (a) and Figure 1 (b) are set for the experiments[4]. The improved A^* algorithm of load balancing has a better load balancing effect in the one-way multi-entry-multi-exit road network model.

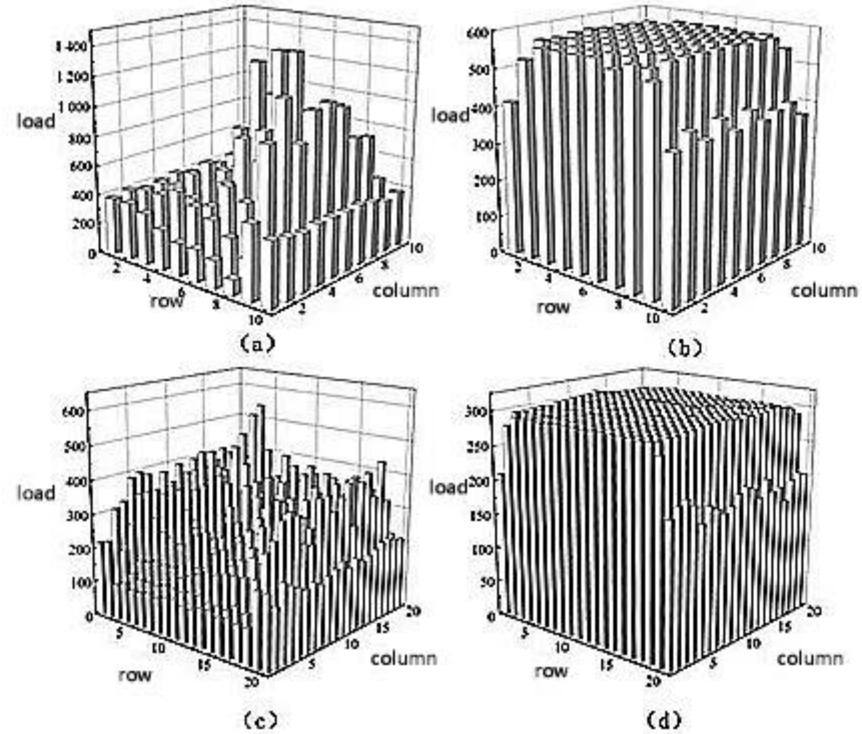

Figure 1 Simulation of load-bearing diagram

Fig. (a) Area load of 10×10 unidirectional multi-entry and multi-exit network with α = 0

Figure (b) 10×10 single-way multiple-input multiple-output network area load under α = 1

Figure (c) 20×20 two-way multi-input and multi-output network area load under α = 0

Figure (d) 20×20 two-way multiple-input multiple-output network area load under α = 10

Experiments are conducted in the two-way multiple-input multiple-output road network model. The same experiments were conducted in the 20×20 bidirectional multiple-input multiple-output road network model by setting α values to 0 and 10, respectively. The load situation of the road network under the two road network models is shown in Figure 1 (c) and Figure 1 (d), respectively. From the comparison, it can be seen that the improved A^* algorithm of load balancing has better load balancing effect in the two-way multi-entry and multi-exit road network model. It can be seen that under both road network models, the regional load of the road network gradually tends to be stable as the value of α increases. After the value of α increases to a specific value, the regional load in the middle row is basically the same except for the entrance and exit rows. Since the entrance and exit rows are only determined by the randomly generated tasks, the load on the road network in the near entrance and near exit rows is almost unchanged when load balancing is considered and not considered. From the regional load in the middle of the road network, it can be seen that the A^* algorithm improved by load balancing in both unidirectional and bidirectional road network models has a better load balancing effect[5].

ISSN: 2664-9640

3.3. Model solving

The Matlab program was written and the simulation results are shown in Figure 2.

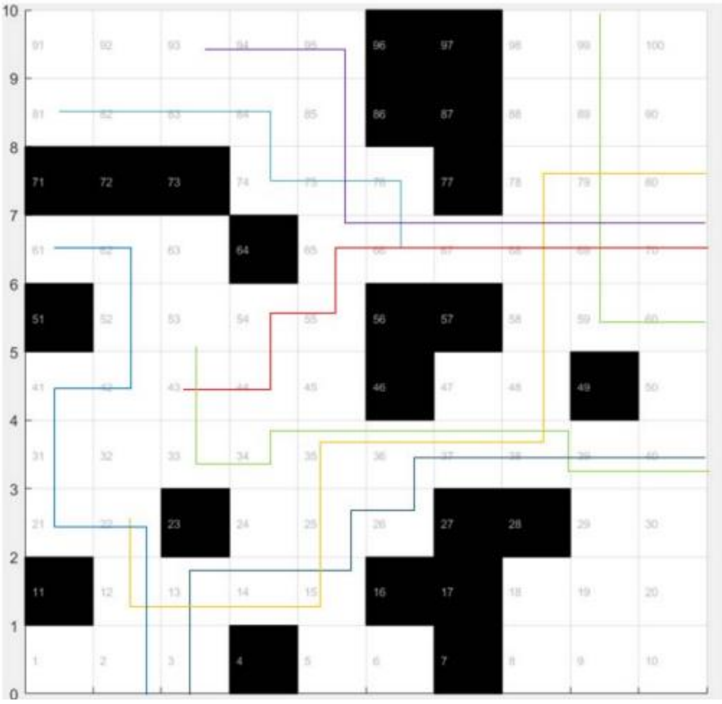

Figure 2 Simulation results

4. Conclusion

This paper provides a good and concise theoretical support for the design and application of AGV systems, taking into account the dynamic scheduling characteristics, a clear analysis process, a simple model, and a clear description of the whole scheduling process. For the solution of the scheduling model of AGV system, we propose a time-multi-objective planning model and an ant algorithm to optimize the scheduling results, combining the advantages of both, which not only shortens the solution time but also makes the solution results as optimal as possible. However, when building the scheduling model, the acceleration and deceleration processes of AGVs are neglected, which may have some influence on the final total completion time.

References

[1] Sun H. Research and System Implementation of Movement Scheduling Algorithm for Multi-Handling Robot of Warehouse Goods[D]. Chengdu University of Information Technology, 2018.

[2] Chen Shengjun. Logistics Sorting System Based on Multiple AGV Scheduling[D]. Shenzhen University, 2017.

[3] Yuan Yang, Ye Feng, Lai Yizong, et al. Multi-AGV path planning combining load balancing and $A*$ algorithm[J]. Computational Engineering and Applications, 56(05):251-256, 2020.

[4] Qiu YJ. Dynamic cyclic classification method to solve RGV scheduling problem[J]. Electronic World, 2021, (17):29-30.

[5] Wang ZY. Research on path planning and scheduling algorithms for multi-AGV systems[D]. Beijing University of Posts and Telecommunications, 2019.