2024 5th International Conference on Big Data & Artificial Intelligence & Software Engineering (ICBASE) | 979-8-3315-0661-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICBASE63199.2024.10762725

Research on Multi-robot Path Planning Based on Optimal Energy Consumption

Yang Wang
School of Software Engineering
Chengdu University of Information Technology
Chengdu, China
*josuke77@outlook.com

Jinchuan Kang
School of Software Engineering
Chengdu University of Information Technology
Chengdu, China
799436326@qq.com

Yuqi Liu
School of Software Engineering
Chengdu University of Information Technology
Chengdu, China
1581719680@qq.com

Qi Tang
School of Software Engineering
Chengdu University of Information Technology
Chengdu, China
79578327@qq.com

***Abstract*—Aiming at the problem of energy consumption optimization in multi-robot path planning, this paper presents a multi-robot path planning algorithm based on optimal energy consumption. First, we improve the traditional A* algorithm to improve the efficiency and stability of multi-robot task execution through a three-step strategy, including queue priority, conflict avoidance and parking scheduling. Then, puts forward a new OECA* (Optimal Energy Consumption A*) algorithm, the algorithm integrates the multiple factors combination model of energy consumption, in order to more accurately estimate the energy consumption in the process of robot path planning in. Then, this paper verifies the performance of the improved A* algorithm and OECA* algorithm through comparative experiments, and the results show that OECA* algorithm has a good advantage in energy consumption. The research in this paper provides an efficient solution to the path planning and scheduling problem of multi-robot systems.**

Keywords-component; A* algorithm; OECA* algorithm; Multi-robot system; Path planning; Multi-factor combined energy consumption model

I. INTRODUCTION

Intelligent mobile robots are more and more widely used in all walks of life, such as warehouse handling robots, orchard robots [1], family service robots [2], medical service robots [3] and rescue and disaster relief robots [4]. Multi-robot path planning (MRPP), also known as Multi-Agent Path Finding (MAPF), is tasked to study the problem of simultaneous path planning of multiple robots in a discrete environment [5]. The goal of MAPF is to enable these robots to simultaneously get from their respective starting positions to their target positions while avoiding collisions.

At present, the research on multi-robot path planning mainly focuses on solving the collision problem of multiple robots during operation. Sharon [6] proposed the incremental cost of tree search algorithm (Increasing Cost Tree Search, ICTS), first introduced the concept of two layers of the search. The top level of ICTS is responsible for assigning a cost value to each robot, while the lower level conducts path search under the cost constraints set by the top level. In the two-layer solution framework, Sharon [7] proposed a complete and optimal Conflict-Based Search algorithm (CBS). Thayne [8] combination of ICTS and CBS algorithm, proposed the Conflict cost tree search algorithm (Conflict – Based Increasing Cost Search CBICS), effectively solve the problem of a symmetric block; Phillips [9] proposed a planning algorithm based on safety interval, which effectively improved the speed of path planning by determining conflict-free time Windows. Yakovlev [10] and others based on SIPP algorithm proposed weighted path planning interval (Weight SIPP with Duplicate States, Wd - SIPP); Li [11] shows estimated CBS algorithm (Explicit Estimation CBS, EECBS), the algorithm is based on the CBS bounded sub-optimal algorithm; Li [12] proposed an algorithm based on priority-basedsearch (PBS), which dynamically adjusts Priority ordering through path cost and priority depth search, and gradually restricts conflicts to find conflict-free paths.

The path planning algorithm of robots with energy consumption as the research direction mainly focuses on a single robot. Liu [13] proposed a single robot path planning method with optimal energy consumption with the goal of minimizing energy consumption of wheeled mobile robots. Lian [14] proposes an improved heuristic path planning algorithm based on the discharge characteristics of lithium batteries, which optimizes the energy consumption paths of multiple AGVs by converting energy consumption into the occupation time of the path network. Hu [15] proposed a high-precision mapping method, which provided the implementation basis for robot path optimization. Jiang [16] proposed an optimal node A* algorithm for job path planning of wheeled mobile robots with optimal energy consumption by using the optimal node search method. This energy optimization algorithm is often difficult to apply when dealing with complex map scenes with dozens of robots.

To solve the above problems, in order to avoid the influence of robot energy reserve on the efficiency of path planning, this paper proposes A multi-robot path planning algorithm based on optimal energy consumption based on an improved A* algorithm, taking energy consumption in the path planning process as a priority factor and integrating the mobile robot energy consumption model.

II. Path Planning Algorithm Based on Optimal Energy Consumption

A. Improved A* Algorithm

A* algorithm is the most effective direct search method for solving the shortest path in static road network. The algorithm uses a heuristic function to find the end point with the least cost from the starting point. The general form of the estimation function for node n is:

$$f(n) = g(n) + h(n) \tag{1}$$

Where: n is the current node; $f(n)$ is the total cost; $g(n)$ is the actual cost from the starting point to the current node; $h(n)$ is the estimated cost from the current node to the target node.

Compared to other search algorithms, the A* algorithm is faster because it uses heuristic functions to reduce the number of search paths. This makes the A* algorithm very efficient when dealing with large-scale search problems. However, when the traditional A* algorithm is applied to the path planning of multiple robots, because the A* algorithm is a greedy algorithm, each robot tends to find the shortest path, which may lead to path conflicts.

In order to solve the above problems, A three-step strategy A* algorithm is proposed to improve the A* algorithm. In the initial stage, all robots will join the queue according to their priority. Each robot will generate an initial path using the A* algorithm and set priorities based on the importance of the task or the length of the path. As determined by Equation (2):

$$P_i = \alpha \cdot TaskPriority_i + \beta \cdot PathLength_i \tag{2}$$

Where, P_i represents the priority of the i th robot, $TaskPriority_i$ is the importance of the task, $PathLength_i$ is the initial path length generated by the A* algorithm, and α and β are the weights that balance the influence of task priority and path length.

In the process of performing tasks, the robot will encounter the situation of path conflict or crossing in the process of traveling. At this time, the algorithm will detect the potential collision and avoid the conflict by dynamically adjusting the path or making some robots wait to ensure the smooth progress of the task. The specific Equation is as follows:

$$Collision_{ij}(t) = \begin{cases} 1 & R_i(t) = R_j(t), t \in T \\ 0 & \text{Or else} \end{cases} \tag{3}$$

$$AdjustPath(R_i, t) = \begin{cases} Replan(R_i) & \textit{If a conflict is detected} \\ Wait(R_i) & \textit{If can't replan} \end{cases} \tag{4}$$

Where $R_i(t)$ represents the position of robot R_i at time t. If robot R_i and R_j are in the same position at the same time t, a conflict occurs. The robot can avoid conflicts by replanning the path $Replan(R_i)$ or $Wait(R_i)$.

Toward the end of the mission, the robot needs to find the nearest parking spot or make a temporary stop if no parking space is available. Parking selection and temporary parking strategies are as follows:

$$D_i = min\left(d(R_i, P_j) \mid P_j \in AvaileParkingPoints\right) \tag{5}$$

$$ParkingDecision(R_i) = \begin{cases} Park\ at\ P_j \\ TemporaryStop(R_i) \end{cases} \tag{6}$$

Where, D_i represents the distance between robot R_i and the nearest parking point P_j, and $d(R_i, P_j)$ is the distance function between robot R_i and parking point P_j. If the parking distance found is less than the threshold value, the robot will park at the parking point; Otherwise, the robot will $TemporaryStop(R_i)$.

This three-step strategy decomposes the A* algorithm into a multi-stage path planning and adjustment process, and combines the elements of task scheduling, path conflict detection and dynamic adjustment to improve the adaptability of the algorithm in multi-robot systems.

B. Multi-factor Combined Energy Consumption Model

The A* algorithm applied in MAPF problem takes the time cost as the optimization goal, and does not consider the energy consumption in the process of multi-robot movement as a factor, which will lead to unnecessary energy consumption. Therefore, a multi-factor combined energy consumption model is proposed in this paper. The model takes into account the load, task execution, and energy consumption required to charge during task execution, and adjusts the total energy consumption according to task complexity and load size. The model Equation is as follows:

$$E_{total} = \omega_1 \cdot E_{load} + \omega_2 \cdot E_{task} + \omega_3 \cdot E_{charge} \tag{7}$$

Where: ω_1, ω_2, ω_3 is the weight coefficient of each energy consumption factor, E_{total} is the total energy consumption of the robot during task execution, E_{load} is the energy consumption of load handling, E_{task} is the energy consumption of task execution, and E_{charge} is the energy consumed by the robot in order to recharge the electricity during the execution of the task. Equation (7) can be decomposed into:

$$E_{load} = C_{load} \times W \times D + f(W) \times C_{terrain} \times D \tag{8}$$

$$E_{task} = \sum_{i=1}^{n} (C_{task,i} \times T_i) + \sum_{j=1}^{m} (C_{operation,j} \times n_j) \tag{9}$$

$$E_{charge} = \left\lceil \frac{T \cdot r_{battery}}{C_{battery}} \right\rceil \cdot \frac{C_{battery}}{\eta} \tag{10}$$

In Equation (8), C_{load} is the load handling energy constant per unit weight of the robot. As the load weight W increases, and the load weight is proportional to the distance D traveled, the robot requires additional energy to overcome the gravitational effect. The load distribution function $f(W)$ is used to represent the distribution state of the load on the robot and the influence of its dynamic change. The terrain-dependent energy consumption constant $C_{terrain}$ describes the additional energy consumption of the robot as it moves over different terrains.

In Equation (9), n is the number of decomposition steps of the task, $C_{task,i}$ is the unit time energy consumption constant of the step i task execution, and T_i is the time of the step i task execution. m is the number of classes of operations. $C_{operation,j}$ is the unit operating energy consumption constant of class j operations (such as grasping, assembly, etc.). n_j is the number of times a class j operation is executed.

In Equation (10), $\lceil \cdot \rceil$ represents rounding up to ensure that the charging demand can be met during each task execution. The battery life $T_{battery}$ depends on the capacity $C_{battery}$ of the robot battery and the energy consumption rate $r_{battery}$ per unit time. η is the charge efficiency, usually $0 < \eta \le 1$.

This complex combined model takes into account load handling, task execution, environmental impact, communication, and the energy required to charge during task execution. The model reflects the various influencing factors in the real scene in more detail, helps to calculate the total energy consumption in the robot system more accurately, and provides a basis for path planning and task assignment.

C. OECA* Algorithm

In summary, based on the improved A* algorithm, the multi-factor combined energy consumption model is introduced to construct the OECA* algorithm. The specific OECA * algorithm is shown in Fig. 1.

Step1: For each robot r_i, calculate the initial path p_i using A* and assign a priority based on task importance and path length. Insert each robot into the priority queue. Initialize energy consumption E_{r_i} for each robot.

Step2: While the queue is not empty, retrieve the highest-priority robot and check for potential collisions. If a collision is detected, either replan the path or make the robot wait. Accumulate energy consumption for path re-planning, waiting, or movement.

Step3: Once the robot reaches its goal, check if a parking spot is available. If parking is available, park the robot; otherwise, temporarily stop it. Add energy consumption for parking or temporary stop.

Step4: After each robot completes its task, add its energy consumption E_{r_i} to the total energy E_{total}.

Step5: The final paths for all robots P_{paths} and the total energy consumption E_{total}.

Figure 1. Optimal Energy Consumption A* algorithm.

III. SIMULATION EXPERIMENT AND ANALYSIS

A. Comparative Analysis of A* Algorithm

In order to verify the feasibility and effectiveness of the improved A* algorithm in this paper, the traditional A* algorithm, the algorithm in the [17] and the improved A* algorithm in this paper are respectively compared in the experimental environment in Fig. 2. The specific data are shown in Table I.

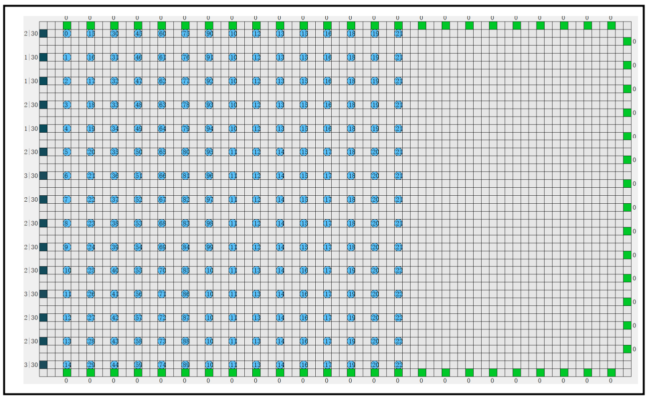

Figure 2. Simulated warehouse map.

TABLE I. COMPARISON OF EACH ALGORITHM PATH

Algorithm	Experimental data		
	Number of Goods	***Number of Robots***	***Completion Time***
Traditional A* Algorithm	8000	225	72

Algorithm	*Experimental data*		
	Number of Goods	***Number of Robots***	***Completion Time***
Reference [17] Algorithm	8000	225	62
Improved A* Algorithm	8000	225	53
Traditional A* Algorithm	8000	100	198
Reference [17] Algorithm	8000	100	156
Improved A* Algorithm	8000	100	123

According to the simulation results, when the number of robots is 225, compared with the traditional A* algorithm, the algorithm proposed in [17] saves about 13.89% of the time. In this paper, the improved A* algorithm further optimizes the search direction on the basis of the traditional A* algorithm, with higher search efficiency and minimum energy consumption, saving about 26.39% of time compared with the traditional A* path and about 14.51% of time compared with the [17]. When the number of robots is 100, the improved A* algorithm in this paper still maintains superior performance. While ensuring higher search efficiency, it saves about 37.87% of time compared with the traditional A* algorithm, and about 21.15% of time compared with the [17]. In summary, the improved A* algorithm proposed in this paper can plan more efficient global paths for robots.

B. OECA Algorithm Simulation Analysis*

In order to verify the effectiveness of OECA* algorithm in reducing energy consumption of multi-robot systems, the warehouse map of AGV is adopted in this paper, as shown in Fig. 3. The traditional A* algorithm, improved A* algorithm and OECA* algorithm were used to solve the test examples respectively, and the replenishment mechanism was added in the process of the experiment. Under different number of robots, different completion time of cargo handling was obtained. According to the experimental data, the completion time diagram of the algorithm is drawn, as shown in Fig. 4. As can be seen from the figure, with the increase in the number of robots, the completion time of OECA* algorithm decreases compared with the traditional A* algorithm and the improved A* algorithm.

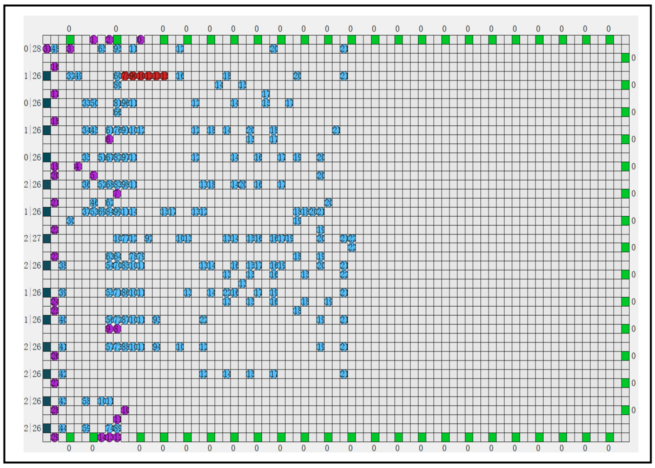

Figure 3. AGV warehouse map.

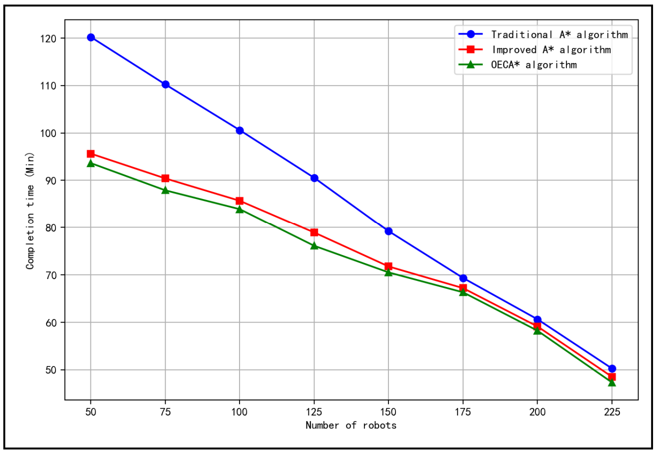

Figure 4. Robot energy consumption relationship.

The specific data is shown in Table II. When the number of robots is 125, the completion time of OECA* algorithm is reduced by 10.3% and 2.3% respectively compared with the other two algorithms. When the number of robots is 225, the OECA* algorithm reduces the completion time by 20.3% and 5.3%, respectively, compared with the other two algorithms. It can be seen that the algorithm planning task is completed well, and it can adapt to the complex environment well, and further reduce energy consumption.

TABLE II. ALGORITHM ENERGY COMPARISON

Number of robots	**Completion Time**		
	Traditional A* Algorithm	***Improved A* Algorithm***	***OECA* algorithm***
50	120.21	95.64	93.64
75	110.23	90.42	87.91
100	100.59	85.7	83.93
125	90.55	78.93	76.12
150	79.21	71.76	70.51
175	69.34	67.2	66.32
200	60.63	59.12	58.23
225	50.27	48.52	47.33

IV. CONCLUSIONS

Aiming at the path planning requirements of multi-robot with limited energy, a new OECA* algorithm based on global energy optimization is proposed in this paper, and its feasibility and effectiveness are verified by simulation. The main work is as follows:

To improve the A* algorithm, A three-step strategy A* algorithm is proposed, which uses the queuing algorithm at the beginning of operation, dynamically adjusts the path or makes some robots wait during operation to avoid conflicts and ensure the smooth progress of tasks. At the end of operation, robots need to find the nearest parking point or temporarily park when there is no available parking space.

This paper presents a multi-factor combined energy consumption model. The model takes into account the energy consumption of the load, task execution and cooperative work of the robots, and adjusts the total energy consumption according to the task complexity and load size. Based on the improved A* algorithm, a multi-factor combined energy consumption model is introduced to construct OECA* algorithm.

The OECA* algorithm in this paper solves the path planning problem of multi-robot in complex environment, and has certain practical value. However, the communication delay problem will occur during the actual operation of the robot. In the future, various kinds of delay problems existing in the actual robot will be studied to adapt to the practical application scenario.

REFERENCES

[1] BI S,WANG Y H.Inter-line pose estimation and fruit tree location method for orchard robot[J].Transactions of the Chinese Society for Agricultural Machinery,2021,52(8):16-26.(in Chinese)

[2] GAO B.Research on key technologies of indoor intelligent sweeping robot[D].Chongqing:Chongqing University,2017.(in Chinese)

[3] YAO B,ZHANG J X,DAI Y,et al.Research on decoupling method of multi-dimensional force sensor used in minimally invasive surgical robot[J].Chinese Journal of Scientific Instrument,2020,41(1):147-153.(in Chinese)

[4] QIU Y,SHEN L Y,HU W J,et al.Shape rebuilding and positioning method of search and rescue robot endoscope in ruin crack[J].Chinese Journal of Scientific Instrument,2015,36 (12):2782-2789.(in Chinese)

[5] Stern R, Sturtevant N, Felner A, et al. Multi-agent pathfinding: Definitions, variants, and benchmarks[C]// Proceedings of the International Symposium on Combinatorial Search. USA: AAAI Press, 2019, 10(1): 151-158.

[6] Sharon G, Stern R, Goldenberg M, et al. The increasing cost tree search for optimal multi-agent pathfinding[J]. Artificial intelligence, 2013, 195: 470-495.

[7] Sharon G, Stern R, Felner A, et al. Conflict-based search for optimal multi-agent pathfinding[J]. Artificial intelligence, 2015, 219: 40-66.

[8] Walker T T, Sturtevant N R, Felner A, et al. Conflict-based increasing cost search[C]//Proceedings of the International Conference on Automated Planning and Scheduling. 2021, 31: 385-395.

[9] Phillips M, Likhachev M. Sipp: Safe interval path planning for dynamic environments[C]//2011 IEEE international conference on robotics and automation. IEEE, 2011: 5628-5635.

[10] Yakovlev K, Andreychuk A, Stern R. Revisiting bounded-suboptimal safe interval path planning[C]//Proceedings of the International Conference on Automated Planning and Scheduling. 2020, 30: 300-304.

[11] Li J, Ruml W, Koenig S. Eecbs: A bounded-suboptimal search for multi-agent path finding[C]//Proceedings of the AAAI conference on artificial intelligence. 2021, 35(14): 12353-12362.

[12] Li J, Lin E, Vu H L, et al. Intersection coordination with priority-based search for autonomous vehicles[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2023, 37(10): 11578-11585.

[13] Liu S, Sun D. Minimizing energy consumption of wheeled mobile robots via optimal motion planning[J]. IEEE/ASME Transactions on Mechatronics, 2013, 19(2): 401-411.

[14] Lian Y, Zhang L, Xie W, et al. An improved heuristic path planning algorithm for minimizing energy consumption in distributed multi-AGV systems[C]//2020 International Symposium on Autonomous Systems (ISAS). IEEE, 2020: 70-75.

[15] Hu K, Chen Z, Kang H, et al. 3D vision technologies for a self-developed structural external crack damage recognition robot[J]. Automation in Construction, 2024, 159: 105262.

[16] JIANG Lei, JIA Wenyou, LIU Li, LIANG Lidong, WEI Wentao. An Optimal Node A* Algorithm for Path Planning of Wheeled Mobile Robot Under Optimal Energy Consumption[J]. Mechanical Science and Technology for Aerospace Engineering, 2023, 42(6): 949-955. doi: 10.13433/j.cnki.1003-8728.20220038

[17] Gu Qing, Dou Fengqian, Ma Fei. Energy Optimal Path Planning of Electric Vehicle Based on Improved A * Algorithm[J]. Transactions of the Chinese Society for Agricultural Machinery,2015,46(12):316-322.