A Multi-AGV scheduling planning method based on improved GA

Yi Zhang, Mengsi Li
China Chongqing Information Accessibility and Service Robot Technology Research Center, Chongqing University of Posts and Telecommunications, Chongqing 400065, China
E-mail: 1375428906@qq.com

Abstract: The path planning of Multi-AGV does not especially consider the power consumption. Therefore, a new Multi-AGV scheduling planning method based on improved GA (Genetic Algorithm) is proposed. The improvement of this algorithm is showed in three aspects: first, in order to ensure that AGVs have the power to perform the task, the operation of power evaluation is added after each iteration is completed; second, the objective function will have two constraints: first, the minimum total power consumption for all AGV; second, the maximum power consumption of a single AGV is minimal; finally, by changing the mutation operator of the GA, the convergence speed and the convergence of the population are improved. The simulation results show that the improved GA scheduling results are more reasonable, and under the same conditions, AGVs power consumption will be smaller.

1. Introduction
Multi-objective optimization has been successfully applied in many fields, including engineering [1], transportation [2], and logistics [3]. The multi-objective optimization problem is to find an optimal solution among the conflicting targets to satisfy all the objective functions. AGV (Automated Guided Vehicle) has played an increasingly important role in the logistics industry due to the high efficiency of transporting materials in the workshop warehouse. However, the application of the Multi-AGV system still faces several important problems: the number of AGVs [4], path planning [5], constraints imposed [6], and so on.

Waldemarm. opolski [7] took the amount of resources of the manufacturing system and the number of AGVs as optimization objects, and optimized the number and cost of AGVs by software simulation. Smolic-Rocak et al. [8] proposed a dynamic path planning method for multi-AGV operation. The path planning of AGVs depends on the current task volume and task priority of the running AGV. Liu Yubang et al. [9] established a multi-objective mathematical model and combined it with two adaptive genetic algorithms (Aga) and multiple adaptive genetic algorithms (Maga) to optimize AGV by considering the charging task and the variable speed of AGV Task scheduling. This paper will improve the genetic algorithm based on electricity and mutation operators to accelerate the convergence rate and convergence of the population while considering energy consumption.

2. AGV path planning problem model
2.1. Environmental layout
In a workshop, there are multiple AGVs for material distribution. There are M cars and N work
stations (M < N). For the distribution of workstation. The assumptions as follows: 1) The starting position of each AGV is the warehouse; 2) There is only one warehouse and charging area, and both of them are the workstation 0; 3) When each AGV completes the task and the power is lower than the threshold, it needs to go to charge; 4) Only one AGV is required for each workstation in a single task.

2.2. Problem Model
In this section, the problem of path planning for AGV is expressed by a mathematical model. The definitions of its index, parameters, and variables are as follows:

1) Index
i, j—Index at both ends of the path between two workstation, i=0,1,2,...,N; j=0,1,2,...,N.
k—Index corresponding to AGV, k = 1,2, ..., M.
l—Index of the workstation that needs AGV distribution, l = 1,2, ..., Lk; 0 < Lk < N.

2) parameters
R_{ij}—the shortest path between workstation i and j.
C_{ij}—The length of the path segment R_{ij}.
Υ—Power conversion coefficient corresponding to AGV driving distance.
E_k—k-th AGV power consumption value.
ME_k—Maximum allowable power consumption of AGV in this plan.
iniE_k—the initial power value of k-th AGV.
Tk—k-th AGV battery threshold.

3) Variable
X_{ijk} = \{1, \text{the } k-th \text{ AGV passes the path } R_{ij}; 0, \text{otherwise}\} (1)
Y_{ki} = \{1, \text{the } k-th \text{ AGV goes to the } i-th \text{ station}; 0, \text{otherwise}\} (2)

\begin{align*}
E_k &= \sum_{i=0}^{L_k} \sum_{j=0}^{L_k} C_{ij} X_{ijk} \quad (3)
\end{align*}

4) Objective function
\[ F = \min \left( \sum_{k=1}^{M} E_k \right) \cap \min (\max (E_k)) \quad (4) \]

5) Constraints
\[ \sum_{k=1}^{M} Y_{ki} = \begin{cases} M, & i = 0 \\ 1, & i = 1,...,L_k \end{cases} \quad (5) \]
\[ \sum_{i=0}^{L_k} X_{ijk} = Y_{kj}, j = 0,...,L_k, k = 1,...,M \quad (6) \]
\[ \sum_{j=0}^{L_k} X_{ijk} = Y_{ki}, i = 0,...,L_k, k = 1,...,M \quad (7) \]
\[ \begin{cases} \text{Planning succeeded, } & E_k \leq ME_k \\ \text{Replanning, otherwise} \end{cases} \quad (8) \]

In the above model, [10] deduced the coefficient Y_t for converting energy consumption into time. Assuming that the AGV consumes power only during operation, and travels at a fixed speed v during the mission, Y_t = Y_t × v. The above constraint (5) means that the starting point of all AGVs must be workstation 0, and all stations can only pass one AGV; constraint (6) means that the start of each path is at workstation 0; constraint (7) means that the end of each path is at workstation 0; constraint (8) means that each AGV must require sufficient power to complete the task.
3. Algorithm Design
Because this paper proposed the constraint of AGV energy consumption, the proposed improvement is to add an energy consumption judgment operation based on the results of the GA. Then the objective function is combined with the double constraint of minimum total energy consumption and minimum single AGV maximum energy consumption, and the mutation operator of the genetic algorithm is changed to accelerate the algorithm’s convergence speed.

The newly added constraints need to determine whether the scheduling result matches the remaining power of each AGV. The specific judgment operation steps are as follows: Step 1: Enter the maximum allowable power consumption of all AGVs (ME_1, ME_2, ..., ME_k); Step 2: Find out the power consumption value corresponding to the chromosome (E_1, E_2, ..., E_k); Step 3: Sort the results of steps 1 and 2 in ascending order: ME = sort([ME_1, ME_2, ..., ME_k]), EH = sort([E_1, E_2, ..., E_k]); Step 4: If EH\leq ME, the result is satisfied, otherwise the result is discarded.

The above steps explain the process of judging the matching between the solution result of the current algorithm and the amount of AGV power, and solve the possible situation of insufficient power of the AGV during the execution of the task. This judgment is added to the genetic algorithm.

3.1. Gene coding
The coding methods of genetic algorithm include binary coding, real number coding, permutation coding, etc. This paper uses permutation coding.

3.2. Population initialization
Population size is very important for the convergence of GA. The population size is related to the variable N, and the appropriate population size should be controlled between 4N and 6N [11].

3.3. Fitness Function
This paper uses the exponential fitness function in [12]. Referring to [13], we chose a fitness function with exponential transformations as:
\[ f = \alpha \times \exp(\beta \times E) \]  
(9)

Where E(=E_1 + E_2 +...+E_k) is one of the parent populations; \( \alpha \) and \( \beta \) are arithmetic constants; the smaller the \( \alpha \), the greater the genetic probability of the individual with the greatest fitness.

3.4. Select operation
The purpose of selection is to inherit the best individuals as much as possible, but in order to prevent the population from converging too quickly to fall into a local optimum, non-excellent individuals must also be considered when selecting. The selection of roulette[14] is for the entire population, and the chances of being selected with high fitness are high, and the chances of being selected with low fitness are small. Therefore, this article uses the roulette selection method.

3.5. Crossover operations
Crossover refers to the process in which two chromosomes exchange some genes with each other to form a new individual. After the cross operation, new descendants are generated, and the basic characteristics of the parent are inherited. This article chooses cyclic crossover.

3.6. Mutation operation
The traditional mutation operation is to exchange genes randomly to ensure the diversity of piggyback belts, but the convergence rate is relatively slow. In order to improve the convergence speed of the population, the improved mutation operation is as follows:

A task is randomly selected from the AGVs with the largest energy consumption required to complete the task among all chromosomes, and then assigned to the AGV with the least energy consumption, which can effectively improve the population convergence speed.
4. Example simulation

In this example simulation, all constraints and requirements are met. Set up 40 workstations, 5 AGVs perform tasks, and the power threshold of all AGVs $T_k = 15\%$ (the percentage of power, $k = 1, 2, \ldots, 5$).

Experiment 1: The initial power value of 2-th AGV $\text{iniE}2 = 100\%$, and the remaining AGVs is 40%.

As shown in Figure 1, after the 3500 generations of the two algorithms, the total distance of the improved GA is 63.6282, the traditional GA is 76.8208. The total distance and total energy consumption of the improved GA are smaller than the traditional GA, and the number of iterations is less. In addition, the power consumption of 1-th AGV and 4-th AGV in the traditional GA exceeds the maximum allowable power consumption. In particular, the 4-th AGV may fail to complete the task.

![AGV path map](image1)

Figure 1 AGV path map of the two algorithms in experiment 1

Figure 2 shows the total distance optimal solution iterative process of the two algorithms. The graph shows that the improved GA has faster convergence speed.

![Optimal solution](image2)

Figure 2 Total distance optimal solution iterative graph of two algorithms

Experiment 2: The initial charge values of all AGVs are 100%. Obtain that the optimal solution of the total path of the improved GA is small, and observe the iterative process of a single maximum path.

![Optimal solution](image3)

Figure 3 shows the single AGV maximum path iterative process of the two algorithms in Experiment 2. It can be seen from the figure that the improved GA not only converges faster, but also the single largest path is smaller, that is, the energy consumption is smaller.
5. Conclusion
This paper proposes an improved genetic algorithm to solve the scheduling problem of multiple AGVs. By adding power judgment operations, it is ensured that AGV can select tasks more reasonably and accurately, and that it can complete tasks with sufficient power. There is also a double constraint that the minimum total energy consumption and the minimum maximum energy consumption of a single path are added. The last improvement is an improved mutation operator of the GA. The results of two experiments show that the solution of the improved GA is more reasonable, and the convergence speed is faster and the energy consumption is smaller under the same conditions.

Acknowledgment
This work was supported by the National Natural Science Foundation of China. (51705065).

Reference
[1] Hu XS, Clara MM, Yang YL. Charging, power management, and battery degradation mitigation in plug-in hybrid electric vehicles: a unified cost-optimal approach. Mechanical Systems and Signal Processing. 2017; 87(17), 4–16.
[2] Xu G, Liang X, Yao S, et al. Multi-objective aerodynamic optimization of the streamlined shape of high-speed trains based on the Kriging model. PLoS One. 2017; 12(1): e0170803. https://doi.org/10.1371/journal.pone.0170803 PMID: 28129365
[3] Qu S, Ji Y. The worst-case weighted multi-objective game with an application to supply chain competitions. PLoS One. 2016, 11(1): e0147341.
[4] Vivaldini K, Rocha LF, Martarelli NJ, et al. Integrated tasks assignment and routing for the estimation of the optimal number of AGVs. The International Journal of Advanced Manufacturing Technology. 2016; 82(1): 719–736.
[5] Li B, Liu H, Xiao D, et al. Centralized and optimal motion planning for large-scale AGV systems: A generic approach. Advances in Engineering Software. 2017; 106: 33–46.
[6] Xin JB, Negenborn RR, Lodewijks G. Trajectory planning for AGVs in automated container terminals using avoidance constraints: a case study. IFAC Proceedings. 2014; 47(3): 9828–9833.
[7] Waldemar K, Opolski. Cost Optimization in Manufacturing System with Unidirectional AGVs[J]. Applied Mechanics and Materials. 2014, 3201(555).
[8] SMOLIC-ROCAK N, BOGDAN S, KOVACIC Z, et al. Time windows based dynamic routing in multi-AGV systems[J]. Transactions on Automation Science and Engineering, 2010, 7(1): 151-155
[9] Liu Yubang, Ji Shouwen, Su Zengrong, Guo Dong. Multi-objective AGV scheduling in an automatic sorting system of an unmanned (intelligent) warehouse by using two adaptive genetic algorithms and a multi-adaptive genetic algorithm.[J]. PloS one, 2019, 14(12).
[10] Mousavi M, Yap HJ, Musa SN, et al. Multi-objective AGV scheduling in an FMS using a hybrid of genetic algorithm and particle swarm optimization. Plos One 2017; 12(3): e0169817.
[11] Endo S, Konishi M, Yoshida M. Motion planning method for mobile robots in large scale
transportation systems by genetic algorithms. Trans Inst Syst Control Inform Eng. 2000; 45(13): 115–23.

[12] Ingber L. Very fast simulate annealing. Math Conput Modeling. 1989; 32(12): 967–973.

[13] Yang SQ, Yang JM, Sun C. Improved exponentiation scale transformation in application of genetic algorithm. Computer Engineering and Applications. 2014; 169(17): 8–11.

[14] Xia GM, Zeng JC. A stochastic particle swarm optimization algorithm based on the genetic algorithm of roulette wheel selection, Computer Engineering and Science. 2007; 29(6): 6–11.