

A Logistics Enterprises Vehicle Scheduling Method Based on Genetic Algorithm

Yan Chen^{1, a}, Xiaoliang Song¹, Xiaowei Han^{1, b}, Dongzheng Wang^{2, c}

¹College of information Engineering Shenyang University, Shenyang, 110044, China

²Faculty of Electronic Information and Electrical Engineering, Dalian University of Technology, Dalian, 116024, China

email: ^asycy@163.com, ^bhxw69@163.com, ^c 13998879310@163.com

Keywords: Logistics enterprise. Genetic algorithm. Vehicle scheduling

Abstract. This paper deals with the genetic algorithm to solve the vehicle scheduling problem of logistics enterprises. It established the mathematical model of vehicle scheduling problem first. Then determined the method of population initialization and the representation of fitness function, in consideration of GA's premature convergence to local values and other defects for genetic algorithm itself, this paper improved the genetic algorithm mutation probability. Finally, it carried out the simulation experiment, and the result shows that the improved genetic algorithm has a good effect on improving the algorithm quality and search efficiency.

Introduction

Vehicle is one of the most important business of logistics enterprises, the vehicle scheduling problem is the core issue related to logistics distribution efficiency and cost control. Optimizing vehicle scheduling program and selecting the appropriate transport route can reduce the operation cost of the enterprise, accelerate the speed of response to customer demand, improve service quality and enhance customers' satisfaction. To select the appropriate distribution scheme can not only reduce the distribution range, but also reduce fuel costs and vehicle consumption, improve the income of logistics operation, then it can bring huge benefits for the company.

There are a variety of vehicle scheduling methods, you can choose different methods according to the different layouts of customers' desired goods, distribution center sites and traffic routes. Simple transportation can use the directional car scheduling method, loop scheduling method, cross scheduling method and so on. When the transport task is heavy and the transportation network is complex, linear programming methods in Tacticians can be used to dispatch the vehicle operation legitimately, such as the shortest path method, the tabular method and the diagram method.

Vehicle scheduling algorithm is divided into exact algorithm and heuristic algorithm. Exact algorithm includes branch and bound method, cutting plane method, network flow algorithm, dynamic programming algorithm. The precise algorithm can get the optimal solution when the problem scale is small. But it is difficult to solve the problem in time when the real problem is of high complexity. The heuristic algorithm includes initial solution, exchange improved method and improved algorithm. For the vehicle scheduling problem with a large number of customers, the alternative vehicle routing solutions grow rapidly, it is impossible to select classic exact algorithm to solve such large-scale vehicle scheduling problem. So, using heuristic algorithm to solve this problem has become an important research direction. Genetic algorithm provides an important method for solving this problem, but the standard genetic algorithm has the disadvantages of slow convergence rate and premature. This paper studied the distribution problems in one direction of single station and multiple model, and improved genetic algorithm.

The code design of genetic algorithm to solve the vehicle scheduling problem

To optimize the vehicle distribution problems of logistics enterprises through genetic algorithm, the most critical and basic work is how to express the solution and how to evaluate it, the representation methods and evaluation methods of solution directly determine the degree of difficulty to realize the algorithm and the algorithm performance.

If all customers are able to fall to the vehicle distribution plan for a solution, infeasible path number M of this distribution program can be set as 0, which means this solution is a viable solution; If the clients in the latter are not able to fall to the vehicle delivery plan, then M will be set as 1, which means this solution is a infeasible solution. Here we can punish the function according to the number and penalty weight of infeasible path, this penalty weight need to be a relatively large positive number according to the value of target function, so we can get the evaluation value of each infeasible path which can be calculated from formula (1). When the value of E decreases, the quality of the solution will increase. Penalty function is used to deal with evaluation of constraint solutions.

$$E = Z + M \times P_w \quad (1)$$

①**Coding method.** Using the coding method arranged directly by customers.

②**Fitness function.** Using the formula (2) as the fitness evaluation function. Performing a decoding operation on an individual first, to get the infeasible path number- M and the objective function value- Z of the distribution scheme by decoding operation and put M and Z into the formula, then you can get the fitness of the individual- f .

$$f = 1/E = 1/(Z + M * P_w) \quad (2)$$

③**Selection strategy.** The best individuals are preserved and they are combined with the roulette wheel when making a selection strategy in this paper. When dealing with a specific selection strategy, the N individuals of each generation is put in a descending order according to the size of the fitness first, the individual with best fitness that is with optimal performance is put in the first place, then the optimal one is save, copied directly to the next generation and put in the first position of the next generation. In the next generation, the other $N-1$ individuals need to be generated with the roulette wheel selection method based on fitness.

④**Crossover operator.** In this paper, crossover operator adopted the OX cross method. In other words, in the actual calculation process, the size of random probability determines whether cross happens or not.

⑤**Mutation Operator.** This program designed for mutation technique for several times, which means that in the actual calculation process, the size of random probability determines whether mutation happens or not. If the generated random probability is greater than mutation probability, then mutation happens. When it happens, this program will adopt the method for generating random number to produce the exchange number J , the individual gene which needs mutation should be exchanged for J times. In the exchange process, the positions of the exchanged genes are randomly generated.

⑥**Termination Criteria.** The program terminates the execution of the program by the way of making evolutionary algebra.

Improvement of Genetic Algorithm

It is important to choose the crossover probability and mutation probability which has a decisive impact on the convergence of genetic algorithm. In the increasing process of P_c , the algorithm will generate new individuals more quickly. But, if the value of P_c is too large, it will destroy the genetic

pattern of the algorithm and can't produce the optimal solution. However, when the value of P_c is too small, the search process of the algorithm will be very slow, which is adverse to the calculation of the final optimal solution. When the mutation probability P_m is big, genetic algorithm which is superior originally will lose its superiority and become a simple method for random indexing. For specific problems, it requires specific analysis to determine the most appropriate values of P_c and P_m and requires a lot of work when analyzing their values, but it is uncertain to find the most suitable P_c and P_m at last. Srinivas et al. proposed an adaptive genetic algorithm (Adaptive GA, AGA), with changes in the calculation process, the values of P_c and P_m will change automatically with the change of individual fitness. The formula of adaptive genetic algorithm P_c and P_m are shown as follows:

$$P_c = \begin{cases} k_2, f_{avg} \geq f' \\ k_1 \frac{f_{max} - f'}{f_{max} - f_{avg}}, f_{avg} \leq f' \end{cases} \quad (3)$$

$$P_m = \begin{cases} k_4, f_{avg} \geq f \\ k_3 \frac{f_{max} - f}{f_{max} - f_{avg}}, f_{avg} \leq f \end{cases} \quad (4)$$

In formula (3) and formula (4), f_{max} represents the maximum fitness value in the population; f_{avg} represents the average fitness value of various groups; f' represents the larger value of fitness under the comparison of two individuals; f represents variability monomer of fitness value; The values of k_1, k_2, k_3, k_4 are between 0 and 1. From the above formulas, it is obvious that with the increase of fitness and its constantly approach to the maximum fitness value, the value of P_c and P_m will be gradually decreased along; Contrarily, when the value of fitness reaches to the maximum, the value of P_c and P_m will decrease to zero. The value of this dynamic evolution P_c and P_m is suitable for the late evolutionary population. Each individual in the population at the later stage of evolution will have higher fitness, at this time, big changes will undermine populations' excellent performance instead. But this adjustment method makes the early evolution of evolutionary process very slow, makes the early excellent individual plan does not change, which is not conducive to produce the optimal solution.

Quadratic function $y = 1 - x^2$ ($0 < x < 1$) can solve the above problems. Since this function's decline rate is slow in the early stage but it will turn fast in the later stage, it has the characteristics of continuous, monotonic and non-linear between 0 and 1, it is conducive to the adjustment of genetic algorithm adaptive mechanism.

$$P_c = \begin{cases} P_{c2} + (P_{c1} - P_{c2}) \left[1 - \left(\frac{f' - f_{avg}}{f_{max} - f_{avg}} \right)^2 \right], f' \geq f_{avg} \\ P_{c1}, f' < f_{avg} \end{cases} \quad (5)$$

$$p_m = \begin{cases} p_{m2} + (p_{m1} - p_{m2}) \left[1 - \left(\frac{f - f_{avg}}{f_{max} - f_{avg}} \right)^2 \right], & f \geq f_{avg} \\ p_{m1}, & f < f_{avg} \end{cases} \quad (6)$$

In the above formula, $p_{c1} = 0.9$, $p_{c2} = 0.6$, $p_{m1} = 0.1$, $p_{m2} = 0.01$. In this adaptive mechanism, in the early days of arithmetic operations, it ensures the evolutionary ability on a large degree, enhances the search ability of the algorithm. At the later stage, crossover probability decreases the damage to the excellent gene gradually. This improved adaptive mutation probability not only ensures the diversity of the population, but also ensures the convergence of the genetic algorithm.

Conclusions

This paper described a mathematical model of logistics enterprises vehicle scheduling problem first, and then used genetic algorithms to solve this problem. A simulation experiment was carried out after the improvement of mutation probability and genetic probability about genetic algorithm. There exists the disadvantage of the premature in unimproved genetic algorithm, improved genetic algorithm can adjust the range of the mutation probability and crossover probability automatically, which not only ensure the diversity of population, but also ensure the convergence of the algorithm to solve the precocious shortcomings. The experimental results showed that the improved algorithm is easier to get the optimal solution for logistics vehicle scheduling problem.

Acknowledgements

This work was supported by the project of science and technology special program of shenyang (Grant No.[2012]212).

References

- [1]H. Sun, X. Tan. *Study of genetic algorithm in optimization of vehicle dispatching* [J]. Computer engineering and applications 2010,46(24):246-248.
- [2]X. L. Ge,X. Wang,L. B. Xing. *Multi vehicle scheduling problem and the cloud genetic algorithm in dynamic demand* [J]. Journal of systems engineering,2012,27(24):821-832.
- [3]Novoa C, Storer R. *An approximate dynamic programming approach for the vehicle routing problem with stochastic demands*[J].European Journal of Operational Research,2009,196(2):509-515.
- [4]Branchini R M, Armentano V A. *Adaptive granular local search heuristic for a dynamic vehicle routing problem*[J]. Computers Operations Research,2009,36(11):2955-2968.
- [5]Muller J. *Approximative solutions to the bicriterion vehicle routing problem with time windows* [J]. European Journal of Operational Research,2010,202(1):223-231.
- [6]Brando J. *A tabu search algorithm for the heterogeneous fixed fleet vehicle routing problem*[J]. Computers Operations Research,2008,38(1):140-151.
- [7]Y. P. Li,T. Zhang. *A new genetic algorithm for solving TSP problem* [J]. Journal of Taiyuan University of Technology,2008,39(3):268-271.
- [8]J. L. Zhang,Y. W. Zhao. *Path optimization problem modeling and optimization of Multi vehicle dynamicdemand* [J]. Computer integrated manufacturing system,2010,16(3):543-550.