Locating Multi-Warehouse Logistics

Ayesha Misbah[1], Dr. Aishwarya P[2]
Department of Computer Science Engineering
[1] Student ,Atria Institute of Technology, Bangalore, Karnataka, India
[2] Head of Department, Atria Institute of Technology, Bangalore, Karnataka, India.

Abstract- *This paper proposes a heuristic algorithm based on Dijkstra and Divide-and-Conquer Algorithm. Firstly, the sub-problem blocks are constructed. Existing methods to solve this problem, such as Genetic Algorithm or Simulated Annealing Algorithm, require long-term adjustment to get optimal parameters which need redesign when data set change, therefore these methods are not well universal applicability. Then, each block is used to calculate the appropriate warehouses. Warehouse location is extremely important in the whole logistics system. Finally, re-calculate the whole graph to get the final solution based on previous warehouse locations. Simulation results show that the algorithm in this paper can greatly reduce the computation time while get right solution, moreover, there are good solutions for different graphs, and general applicability is strong.*

Keywords- Warehouse location; Dijkstra; Divide-and Conquer Algorithms; heuristic algorithm.

I. INTRODUCTION

In choosing the right warehouse location can make all the difference in how effective, efficient, and profitable a company is. Leasing or purchasing a warehouse is a major decision, and a choosing the right location can significantly enhance a company's ability to compete and effectively serve customers.

Unfortunately, all the methods mentioned above have flaws that are inevitable. For example, the location of the warehouse calculated by the Gravity Method is always impossible due to the natural limitation. A great number of data is needed in the Chromatography Analysis to get a right result, with the weights hard to determine. In addition, the main idea of the Genetic Algorithm is to select the initial population, screen through the fitness algorithm, then cross reorganize with the crossover rate, and finally find the optimal individual fitness to a given threshold after mutating through variation. However, the Genetic Algorithm are usually related to low efficiency, time-consuming, premature convergence and difficulties in controlling parameters. Additionally, the main idea of Simulated Annealing Warehouse location is one of the most important issues of the logistics system. Particularly, the location of the multi-warehouse is most common for nowadays companies on a certain scale have several warehouses to store goods. The location of warehouse is always closely related to the construction cost of the warehouse, the routes of diversity, after-sale services and even the competitiveness of the enterprise itself. Therefore, it is one of the significant issues of logistics firms.

Algorithm is to search random walk in given intervals, and make the random walk gradually converged to optimal solution with the method of Metropolis sampling criterion. However, the capacity of this method to search the whole is limited for it is easily affected by the parameters.

To overcome the shortcomings, such as time consuming and poor universal applicability, of the previous studies, an algorithm on the foundation of Dijkstra and Divide-and Conquer Algorithm is proposed in this pater to solve multi-warehouse location.

There is still a long way to go to get a set of relatively good parameter for both the Simulated Annealing Algorithm and Genetic Algorithm, moreover, once large-scale data set adjustment is needed, the parameters have to be redesigned, which is hard to guarantee universal applicability.
However, the Genetic Algorithm are usually related to low efficiency, time-consuming, premature convergence and difficulties in controlling parameters. Additionally, the main idea of Simulated Annealing Algorithm is to search random walk in given intervals, and make the random walk gradually converged to optimal solution with the method of Metropolis sampling criterion. However, the capacity of this method to search the whole is limited for it is easily affected by the parameters.

The mathematical model and the theoretical basis of the algorithm will be introduced in the second part. The following part proposes the algorithms, respectively, to locate the warehouses and to plan the route from the warehouse to the consumer node.

Moreover, the toll standard and maximum traffic vehicle number of every expressway is also known, the demands of each area are also clear, and so is construction cost of the warehouse. The requirement is to minimize total cost of building warehouses and delivering the goods.

There is, at most, one edge between net-nodes, the upper and lower edge of the total capacity is independent, and so is unit price. The dotted lines connecting the net-nodes and the consumer-node is called consumer edge, representing the area's demands. Both consumer nodes and consumer edge are here just for convenience.

II. DESCRIPTION OF THE PROBLEM

Company X is a well-known producer of product Y in M province. Company X, to meet the demands, plans to build several warehouses in M province. According to the market research, it is known that this province is made up of N areas, two of which are connected by the known expressway.

Network Structure Model implies particular areas, expressways between them as well as the toll standard and capacity of each expressway. The nodes surrounded by solid lines represent area are called net-nodes, and they are all numbered. The solid line connected one area to another which represents the expressway is called the edge. The nodes surrounded by dotted lines represent area are called consumer-nodes, and they are all numbered, too. Moreover, the numbers in brackets indicate the highway toll standard (unit price) and the maximum number of vehicles the expressway can hold (total capacity). Thus, a undirected graph with weights is constructed G=(V, E,C) the net-nodes is V={v0, v1, v2, K, vn} the edge is E={e0, e1, e2, K, em} and the consumer nodes is C={c0, c1, c2, K, cn} the cost and total capacity of each edge is different. As shown in Fig.2-1, net-nodes v0 and v1 are connected, with a total capacity of 16, unit price of 2. If 10 units of v0□v1 are occupied, there are still 16 available units of v1□v0 Different consumer nodes represent different demands. As shown in Fig.2-1, the demand of the consumer node 0 c is 28.

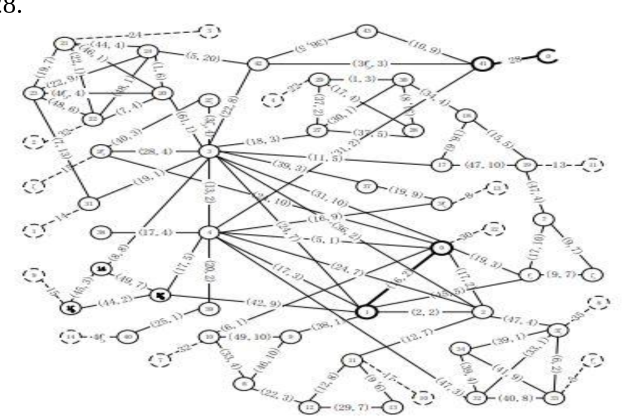


Figure 2-1 A small case map

Route (c,j) represents the j-th route from warehouse to the consumer-node ci , v0 represents a net-node representing a warehouse. v1 to vn-1 represents the net-nodes passing by, and vn represents the net-node that the consumer-node is only connected to w (vp, vq) represents the unit price of each edge and the unit price can only be positive.

Table - The route table of case

Routes	Price sum	Capacity	Cost sum
$Route(c_0,1)=\{v_{39},v_4,v_4,v_{41}\}$	$Price(c_0,1)=4$	$Capacity(c_0,1)=5$	$Cost(c_0,1)=20$
$Route(c_0,2)=\{v_3,v_4,v_{41}\}$	$Price(c_0,2)=4$	$Capacity(c_0,2)=13$	$Cost(c_0,2)=52$
$Route(c_0,3)=\{v_{40},v_{39},v_4,v_{41}\}$	$Price(c_0,3)=5$	$Capacity(c_0,2)=10$	$Cost(c_0,2)=50$
$Route(c_1,1)=\{v_1,v_{20},v_{25}\}$	$Price(c_1,1)=2$	$Capacity(c_1,1)=21$	$Cost(c_1,1)=42$

The Divide-and-Conquer Algorithm is broadly used in programming. The basic point of this algorithm is to split problem which is difficult to be solved at once into several sub-problems which is easier but essentially the same as the original problem, and then to solve them separately. In general, warehouse location is always calculated in hundreds of nodes or even more. It is of no efficiency to calculate the cost of the warehouse plan directly through the exhaustive method for the time it will take is n! The number is too large and the efficiency is too low. But the artificial division cannot meet the universal applicability. Therefore, the algorithm proposed in this paper is divided into three steps:

a) Deleting edges in the construction of sub-problem blocks. The demands of the various consumer nodes will be put together, the nodes and the edge of the network will be deleted in accordance with certain rules in the algorithm proposed in this paper. So that the remaining node with a great demand, called the patrol nodes, cannot be fully satisfied by other edges. The nodes focusing on their own demand to patrol-node are called block-consumer-nodes. A sub-problem is made up of the block which is made up by their own block consumer-nodes and the patrol-nodes.

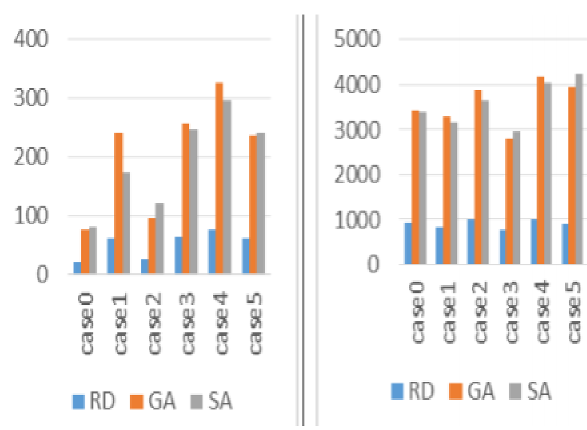
b) Solving the sub-problem. In the block, the patrol-nodes and block-consumer-nodes represent the prepared warehouses. Different warehouses will be picked out with the arrangement of combinations. And then the heap optimized Dijkstra algorithm will be used to calculate the total cost of possible plans (including building the warehouses and the corresponding transport routes). Then the one with the lowest cost will be recorded as our selection.

c) Further calculations on the basis of the previous results. Since the method of the previous step is easy to fall into the local optimum, the calculated path is discarded, and only the warehouse nodes are used as the reference nodes. The whole problem is recalculated by the heap optimized Dijkstra algorithm to make the final decision.

III. EXPERIMENTAL DISCUSSIONS

In this paper, the time complexity of the RD algorithm is determined. If the warehouse location plan is not found at the end of the algorithm iteration, then it is judged as a failure. The time complexity of the GA algorithm and the SA algorithm is uncertain, if there is no warehouse location plan for running more than 100s, then the algorithm is recomputed. Experimental results as follow.

Figure shows three different sizes of data sets from large to small. And the results show that the difference between RD algorithm and GA or SA algorithm is about 2%. Particularly, in small-scale date sets there is little difference. Furthermore, the time spent on the RD algorithm in this paper is only about 1/4 of that on GA or SA. Compared with the traditional Genetic algorithm and Simulated Annealing algorithm, the algorithm in this paper saves a lot of time while ensuring the results even with longer-term adjustment parameters. What's more, the algorithm in this paper is directly calculated, so it is of better universal applicability.



IV. CONCLUSION

In this paper, the order of the consumption nodes in the block is calculated according to the serial number, and it can be considered in random order in the future. A heuristic algorithm based on Dijkstra and Divide-and Conquer Algorithm is proposed in this paper to discuss the he problem of warehouse location. this algorithm not only saves a lot of time but also guarantees the correctness. Moreover, other finding routes algorithms that are easier and less time-consuming are looking forward to be applied.

REFERENCES

[1] Yingfeng Zhang, Renbin Feng, Xingfang FU. Application of Genetic Algorithms in Selecting Accurate Freight Site. Electrical Automation, 2004, 26 (3):16-18.

[2] Liu G, Ramakrishnan K G. A* Prune: an algorithm for finding K shortest paths subject to multiple constraints[C] INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. IEEE, 2001, 2: 743- 749.

[3] Biao Yang, Yipeng Zhou. A Study on the Location of Warehouse Based on Mixed-integer Programming. Logistics Sci-Tec, 2015, 38 (4):118-120.

[4] Qingfang Ruan. The Study and Application of the Warehouse Selected Location Gravity Model Approach in the Logistics Professional Teaching. Logistics Sci-Tech, 2008, 31 (8) :84-86.

[5] Zhenya Zhao, Guoxian He. Optimization of Emergency Logistic Warehouse Location Based on Simulation Annealing Algorithm. Journal of Dalian Jiaotong university, 2010, 31 (3):102-106.

[6] WJ Baumol, P Wolfe. A Warehouse-Location Problem. Operations Research, 1958, 6 (2):252-263.

[7] J Perl, MS Daskin. A warehouse location-routing problem. Transportation Research Part B Methodological, 1985, 19 (5):381-396.

[8] JE Beasley. An algorithm for solving large capacitated warehouse location problems. European Journal of Operational Research, 1988, 33 (3):314-325.
[9] Chuanwei Fang, Xiaoyi Wang. Discrete Warehouse Location Based on Genetic Algorithm. Journal of Shenzhen Polytechnic, 2004, 3 (3):34- 38.
[10] Ronald H. Ballou. Dynamic Warehouse Location Analysis. Journal of Marketing Research, 1968, 5 (3):271-276.
[11] Liu G, Ramakrishnan K G. A* Prune: an algorithm for finding K shortest paths subject to multiple constraints[C] INFOCOM 2001. Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. IEEE, 2001, 2: 743- 749.