Optimization of Logistics Route Based on Dijkstra

Xin Zhang, Yan Chen and Taoying Li
Transportation Management College
Dalian Maritime University
NO.1, Linghai Road, Dalian 116026, P.R.China
dlmu_zhangxin@163.com

Abstract—**In large or medium retail enterprises, the downstream of the supply chain is usually composed by distribution centers or various warehouses. Since the distance of each distribution route is large, so the optimization of path is very important for the cost control and profit maximization of the enterprise. This paper uses Dijkstra Algorithm to solve the shortest path problem, and uses the method of map labeling, simplifying the solving process Finally through the abstracting model of the problem, and designing of structure and combining with the process oriented C language, the paper realizes this function.**

Keywords: Dijkstra algorithm; map labeling; optimization of path

I. INTRODUCTION

Transportation is not only one of the main functions in logistics process, but also is the central activity of business [1]. Nowadays, socialization and specialization of the production are continuously improved. The production and consumption of all material cannot be done without transportation. Especially the development of electronic commerce which brings large challenge to the transportation .So the rationalization of the transportation path plays a significant role in logistics. In this paper, in order to realize the shortest path, the problem of the path from distribution center to each warehouse is transformed into the shortest path problem in graph theory.

The emergence of the earliest graphs can be traced back to 1736. The famous mathematician Euler uses it to solve the classic Seven Bridges of Konigsberg [2]. In the graphs, the relationship between each node is arbitrary .That is to say, each node and other nodes are likely to be relevant. The knowledge of graph has been widely used in real life. For example, the location problem, the critical path in the project, and minimum spanning tree to build the lowest price network, etc. The problem of the shortest path is one of the most typical problems in graph theory, Including Johnson algorithms, Dijkstra algorithm, Floyd algorithm, etc. Firstly this paper uses Dijsktra algorithm to model practical problems. Secondly, this paper uses C program language to make this function come true. Solving the problem of selecting the shortest distribution path, this paper helps to reduce the cost of logistics transportation and improves beneficial result.

II. RELATED CONCEPTS

1) *The graph*: a graph G is defined as an ordered two-tuples (V, E), denoted as G=<V, E>, among them:

V is a finite no empty set, its elements are called the G nodes, usually expressed as $V=\{v_1,v_2,\cdots,v_n\}$;

The elements of E are called the G side, denoted as $E=(v_i, v_j)$, while E is called the G edge set.

2) *Connected graph*[3]: with undirected graph G=<V, E>, if a channel between u and v(u,v∈V)exists , V is connected.

3)*Network Analysis*: qualitative and quantitative analysis of the network, so as to seek optimal scheme to achieve a certain goal.

4) *Adjacency matrix*[4]: if the graph is a weighted graph, it is expressed as

$$A[i][j]=\begin{cases} W_{ij},\ \text{when} <v_i,v_j>\in E \\ \infty,\quad \text{conversely.} \end{cases} \tag{1}$$

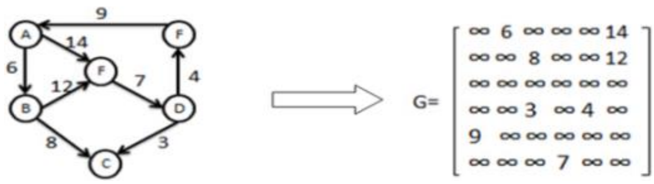


Figure 1: Adjacency matrix of the directed graph

III. THE BASIC IDEA OF DIJKSTRA

The Dijkstra algorithm is described as follows [5]:

G= (V, E) was a weighted directed graph. The set V is divided into two groups. The first group is the set of S that includes nodes already sought the shortest path (At the beginning ,there is only one node in S. after computing a shortest path each time, the corresponding node will be added in set S, until all the nodes are added to the S). The second group is set U whose nodes are yet to be determined. After finding a shortest path for a node, the shortest length of other nodes will be modified. Through several iterations, it ensures that each node finally gets the shortest path.

The algorithm steps:

1) Initially, set S only contains the source V;

2) Choose a node K from set U, distance from K to V is the shortest path;

3) Take K as the new consideration , modify the shortest path of each node.

4) Repeat steps (2) and (3) until all nodes are contained in S.

978-1-4799-8353-7 /15/$31.00 ©2015 IEEE

This work was supported by the National Natural Science Foundation of China (Grant No. 71271034), and the National Social Science Foundation of China (Grant No.15CGL031), and the Fundamental Research Funds for the Central Universities (Grant No. 3132014307, 3132014080, and 3132015048), and the Natural Science Foundation of Liaoning Province (Grant No.2014025015)

IV. ALGORITHM APPLICATION

A. Putting forward the problem

One distribution center issues the vehicles to deliver goods for five warehouses in this area. Each vehicle is responsible for a warehouse. So how to choose the path for each vehicle to make the each length of path is the shortest?

B. Modeling problems

According to the route from the distribution center to each warehouse, abstract it to form a directed graph. Distribution center and each warehouse are expressed as the points. The distribution center is v_0, five warehouses are, in order, v_1, v_2 ,v_3, v_4, v_5.And the routes among them are represent as the edges of a graph. The distance of each route is represented as the weight of the edge. In this way, the problem is transformed as choose the shortest path by using Dijkstra.

C. Solution of the problem

The traffic network is abstracted as follows, v_0 is the distribution center, the others are the warehouses. All the weights of the edge between the nodes are the length of the route.

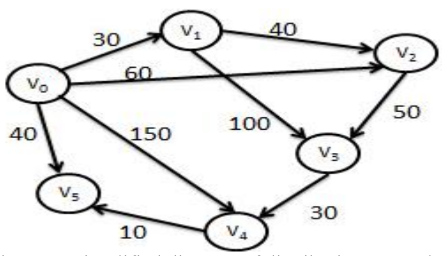


Figure 2: Simplified diagram of distribution network

1) Map labeling method[6] for solving

Write the initial value of path between each node;

According to the Dijkstra algorithm to solve problem(the value with "()"is the final result of path, in contrast, the value without"()"is the temporary result).

The detailed steps are as follows:

Step1: note length of the shortest path from v_0 to the nodes which is next to it.

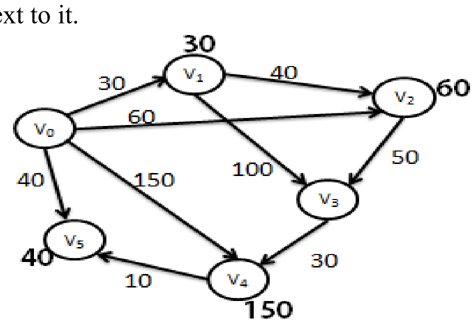


Figure 3: Initial route

Step2: find the minimal length, use parentheses to label.

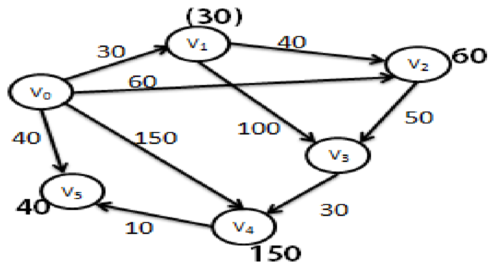


Figure 4: Route of finding the first shortest path

Step3: view v_1 as a new start, update the shortest path of nodes which is next to v_1.

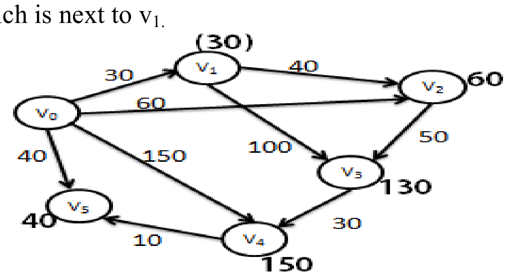


Figure 5: Route of first updating

Step4: repeat the step2.

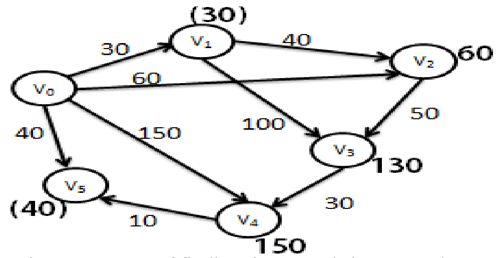


Figure 6: Route of finding the second shortest path

Step5: view v_5 as a new start, update the shortest path of nodes which is next to v_5.

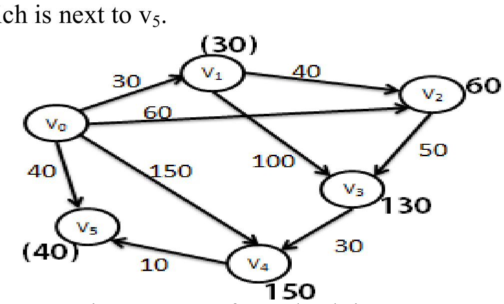


Figure 7: Route of second updating

Step6: repeat the step2

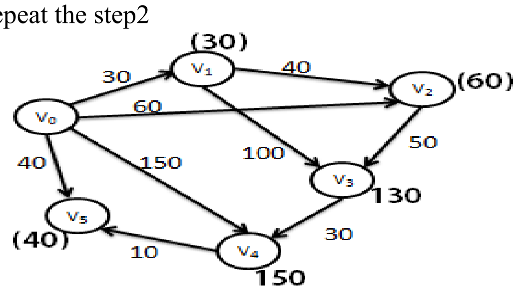


Figure 8: Route of finding the third shortest path

Step7: view v_2 as a new start, update the shortest path of nodes which is next to v_2.

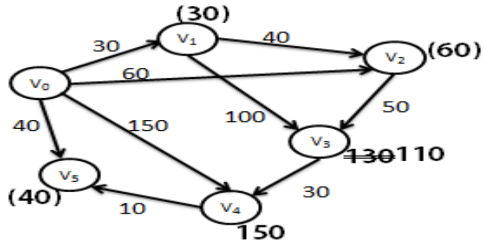


Figure 9: Route of third updating

Step8: repeat the step2.

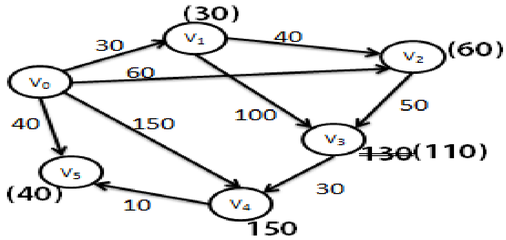


Figure 10: Route of finding the fourth shortest path

Step9: view v_3 as a new start, update the shortest path of nodes which is next to v_3.

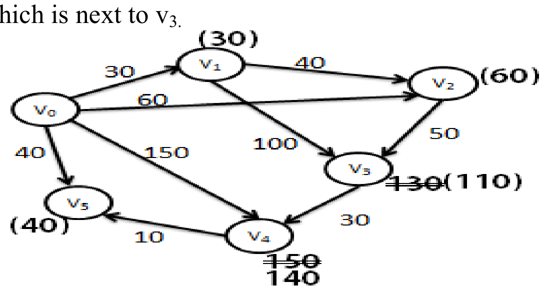
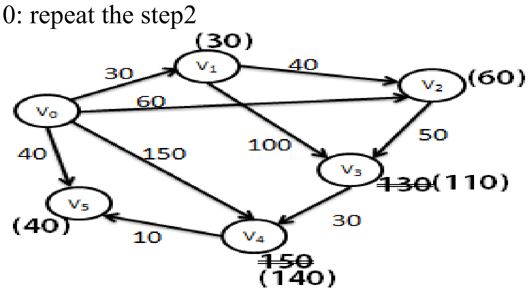


Figure 11: Route of fourth updating

Step10: repeat the step2



Picture 12: Route of all the shortest paths

2) Summarized as follows:

TABLE 1 SOLVING PROCESS

step	S set	U set
1	Select source Vo now $S=\{V_0\}$ The shortest path V_0->V_0, Regard V_0 as a start，to find from V_0	$U=<V_1,V_2,V_3,V_4>$ V_0-> V_1=30 V_0-> V_2=60 V_0-> V_3=∞ V_0-> V_4=150 V_0-> V_5=40 Discover that V_0-> V_1=30 is the shortest path
2	Add V1 into S, now $S=\{V_0,V_1\}$ the shortest path:V_0->V_0=0, V_0->V_1=30 regard V_1 as a new middle node, find the shortest path from V_0->V_1=30	$U=\{V_2,V_3,V_4,V_5\}$ V_0-> V_1-> V_2=70 V_0-> V_1-> V_3=130 V_0-> V_1-> V_4=∞ V_0-> V_1-> V_5=∞ Discover that V_0-> V_5=40 is the shortest
3	Add V_5 into S ,now $S=\{V_0,V_1, V_5\}$ The shortest path:V_0->V_0=0, V_0->V_1=30, V_0->V_5=40 Regard V_5 as a new middle node，find the shortest path from V_0->V_5=40	$U=\{ V_2,V_3,V_4\}$ V_0->V_5->other=∞ To sum up , V_0->V_2=60 is the shortest
4	Add V2 into S, now $S=\{ V_0,V_1, V_5，V_2\}$ Regard V_2 as a new middle node，find the shortest path from V_0-> V_2=60	$U=\{V_3,V_4\}$ V_0->V_2-> V_3=110 V_0->V_2->other=∞ To sum up, V_0->V_2-> V_3=110 is the shortest
5	Add V_3 into S, now $S=\{ V_0,V_1,V_5,V_2,V_3\}$,regard V_3 as a new middle node，find the shortest path from V_0->V_2-> V_3=110	$U=\{V4\}$ V_0->V_2-> V_3-> V_4=140 V_0->V_2-> V_3->other To sum up V_0->V_2-> V_3->V_4=140is the shortest
6	Add V_4 into S，now $S=\{ V_0,V_1,V_5,V_2,V_3, V_4\}$	U={}is empty，checking is over

3) result as follows:

The shortest path between V_0 to V_1： V_0-> V_1=30

The shortest path between V_0 to V_2： V_0-> V_2=60

The shortest path between V_0 to V_3：

V_0->V_2-> V_3=110

The shortest path between V_0 to V_4：

V_0->V_2-> V_3-> V_4=140

The shortest path between V_0 to V_5： V_0-> V_5=40

D. Realization of Dijkstra algorithm

Core code as follow:

```
void Dijkstra( MGraph N, int V0, PathMatrix path,ShortPathLength dist)
  /* path[v] represents the shortest path from V0 to V;dist[V] is the length of the shortest path* */
  /* when final[V] equals 1,we find the shortest path from V0 to V*/
  {
  int v,w,i,k,min;
  int final[MaxSize];        /*where the shortest path from v0 to this node has been found */
  for(v=0;v<N.vexnum;v++)/*Initialize the shortest path from v0 to v*/
  {
  final[v]=0;
  dist[v]=N.arc[v0][v].adj;
  for(w=0;w<N.vexnum;w++)
     path[v][w]=0;
     if(dist[v]<INFINITY) /*if there is the shortest path from v0 to v, initialized as the distance from v0 to v*/
     {
     path[v][ v0]=1;
     path[v0][v]=1;
     }
  }
  dist[v0]=0;      /*distance from v0 to v0 is zero*/
  final[v0]=1;    /*v0 is included in set S*/
  for(i=1;i<N.vexnum;i++)
  {
  min=INFINITY;
  for(w=0;w<N.vexnum;w++)
   {
   if(!final[w]&&dist[w]<min) /*find node whose distance from
```

```
v0 is the shortest*/
        {
          v=w;
        min=dist[w];
        }
      final[v]=1;
      }
      for(w=0;w<N.vexnum;w++)    /*update the shortest path from
v0 to nodes which is not belong to S*/
        {
      if(!final[w]&&min<INFINITY&&N.arc[v][w].adj<INFINIT
Y&&(min+N.arc[v][w].adj<dist[w]))
      {
      dist[w]=min+N.arc[v][w].adj;
      for(k=0;k<N.vexnum;k++)
      path[w][k]=path[v][k];
      path[w][w]=1;
      }
      }
    }
    }
```

V. CONCLUTION

In logistics, the shortest path is one of the most common problems in medium or large enterprises. It is not only the general geographical shortest distance. It can be extended to other metrics, such as time, cost, line capacity and so on[7]. Accordingly, the shortest path can also become the problem of quickest path, the problem of minimum cost, etc. In addition, there are many factors may effect distribution path. Such as traffic congestion, traffic light quantity etc. when considering these factors to determine the distribution route, we can set weight for each factor ,and quantify them as value on each route ,finally, transform to the shortest path problem. This paper uses the knowledge of graph to modeling distribution path, and combines with the C language to realize this function. So it plays a auxiliary role in the selection of the actual logistics path to reduce logistics cost and improve income of the enterprise.

VI. ACKNOWLEDGE

The authors are very grateful to the editors and reviewers for their valuable comments and suggestions. This work was supported by the National Natural Science Foundation of China (Grant No. 71271034), and the National Social Science Foundation of China (Grant No.15CGL031), and the Fundamental Research Funds for the Central Universities (Grant No. 3132014307, 3132014080, and 3132015048), and the Natural Science Foundation of Liaoning Province (Grant No.2014025015) .

VII. RESERENCE

[1]Gensheng Liu. Fei Su. Di Zhao. Modeling and optimization of multi objective optimal path based on Dijkstra algorithm [J]. Journal of Chizhou Teachers College, 2007, 21,(3):17-22

[2]Dayou Liu. Qiangyuan Yu. BoYang .Sehngsheng Wang. Li Jiang. Data Structure(Second Edition)[M]. BeiJing: Higher Education Press,2010: 147-162

[3]Wanling Qu. Suyun Geng. Liang Zhang. Discrete mathematics[M]. BeiJing: Higher Education Press,2008:269-271

[4] Rui Chen. Etc. Zero basic to learn data structure[M]. BeiJing: machinery industry Press,2010:182-191

[5]Weimin Yan. Weimin WU. Data structure (C language version)[M]. BeiJing. Tsinghua University Press,2002:122-146

[6]Jie Chen. Feng Lu. The realization and efficiency analysis of the shortest path labeling algorithm in traffic network [J]. Journal of Image and Graphics, 2005, 10(9):1134-1138

[7]Hanhua Qu. Lina Wang. Jie Zhou. Application of Dijkstra algorithm in urban road network[J]. The knowledge and technology of computer, 2007, (22):984-986