AMSE JOURNALS –2014-Series: Modelling D; Vol. 35; N° 1; pp 9-23
Submitted Nov. 2013; Revised Dec. 16, 2013; Accepted July 20, 2014

Solve the More General Travelling Salesman Problem

Caicai. Feng[1], J. Liang[1,2]

1. Department of Basic Sciences and Applied Technique, Guangdong University of Science and Technology, 523083 Guangdong P. R. China (57079540@qq.com)

2. School of Applied Mathematics, Guangdong University of Technology, 510006 Guangzhou P. R. China (LiangJP@126.com)

Abstract

In this paper, a more general traveling salesman problem has been discussed. In the traditional traveling salesman problem the graphs it discussed are only sides weighted graphs. The graphs discussed in paper [2] and in this paper are complete weighted graphs, that all nodes and sides are weighted. We obtain simple method to solve all the nodes weight on condition that all the sides weight and all the total work (for example: total cost) are known.

Key Words: Traveling salesman problem, solution, graph theory, total work.

1. Introduction

The traditional traveling salesman problem is the prolongation of the Hamiltonian problem. In paper [1], the graphs it discussed are only sides weighted graphs. In paper [2] and this paper, which discussed are complete weighted graphs, that all nodes and sides are weighted; In paper [2], it discuss how to find the best Nearest Neighbor Circuit such that the total work is the minimum, when the nodes weight and sides weight are known. In this paper, it discusses how to solve all the nodes weight when all the sides weight and all the total work are known.

Yet no effective method is known for the TSP. There is a well-known algorithm called Nearest Neighbor Algorithm. It's simple: While a salesman is in a city, the next city he need to go to is the one, that is the nearest and he hasn't been to yet, and so on until all the cities are visited. The method of the Nearest Neighbor Algorithm is: In a n-order side weighted undirected complete graph, we can freely choose any node as the starting node x_1 of the Hamiltonian Circuit, and then find the nearest node to x_1 as the second node x_2. The nearest node to x_2 is as the third node x_3 and so on until all the nodes are chosen. In order to express convenient, we can call the circuit, obtained by Nearest Neighbor Algorithm, the Nearest Neighbor Circuit. And we can call the Nearest Neighbor Circuit, with node v_i as its starting node, the v_i - Nearest Neighbor Circuit. We can call the side weight $\overline{a}_{ik}$, of the side between node v_k and the next node, the following nearest neighbor side weight of v_k in the v_i - Nearest Neighbor Circuit.

In the traditional traveling salesman problem the graphs it discussed are only sides weighted graphs. The graphs we discussed in paper [2] and in this paper are complete weighted graphs, that all nodes and sides are weighted. In this paper, we obtain simple method to solve all the nodes weight, on condition that all the sides weight and all the total work (for example: total cost) are known. It is the main result of this paper and is very important in simulation problem, for example in the urban planning administration or in the modern physical distribution management.

2. Main Results

At first place we give the method:

Method 2.1 Conside in a n-order complete weighted undirected graph, there are n nodes $\left(v_1, v_2, \cdots, v_n\right)$. Let k_{ij} means the side weight between v_i and v_j, and all the k_{ij} are known, where $i, j \in 1,2,\cdots,n$. Let $x_i, i \in 1,2,\cdots,n$ means the node weight of the node v_i and all the x_i are unknown.

In the v_i - Nearest Neighbor Circuit, we call the product $\overline{a}_{ij} x_j$ the work of node v_j, which the $\overline{a}_{ij}$ is the following nearest neighbor side weight of v_j. And we call the sum $\sum_{j=1}^{n} \overline{a}_{ij} x_j$ the total work of all the nodes $\left(v_1, v_2, \cdots, v_n\right)$ in the v_i - Nearest Neighbor Ciricuit.

Step 1 By Nearest Neighbor Algorithm, we let all the nodes v_i, $i \in 1,2,\cdots,n$ as starting node separately, and find out all the $\overline{a}_{ij}$ of v_j, $j = 1,2,\cdots,n$, $i = 1,2,\cdots,n$. Therefore we can get n v_i ($i \in 1,2,\cdots,n$)- Nearest Neighbor Ciricuits and their total work:

$$\begin{cases} \bar{a}_{11}x_1 + \bar{a}_{12}x_2 + \cdots + \bar{a}_{1n}x_n = \sum_{j=1}^{n} \bar{a}_{1j}x_j \\ \cdots\cdots\cdots \\ \bar{a}_{i1}x_1 + \bar{a}_{i2}x_2 + \cdots + \bar{a}_{in}x_n = \sum_{j=1}^{n} \bar{a}_{ij}x_j \\ \cdots\cdots\cdots \\ \bar{a}_{n1}x_1 + \bar{a}_{n2}x_2 + \cdots + \bar{a}_{nn}x_n = \sum_{j=1}^{n} \bar{a}_{nj}x_j \end{cases} \qquad (1)$$

Because $x_i, i \in 1,2,\cdots,n$ are unknown, so (1) is n-variables linear equations with variables $x_1, x_2, \cdots, x_n$. Let $W_i = \sum_{j=1}^{n} \bar{a}_{ij}x_j, i \in 1,2,\cdots,n$. Then the (1) can be:

$$\begin{cases} \bar{a}_{11}x_1 + \bar{a}_{12}x_2 + \cdots + \bar{a}_{1n}x_n = W_1 \\ \cdots\cdots\cdots \\ \bar{a}_{i1}x_1 + \bar{a}_{i2}x_2 + \cdots + \bar{a}_{in}x_n = W_i \\ \cdots\cdots\cdots \\ \bar{a}_{n1}x_1 + \bar{a}_{n2}x_2 + \cdots + \bar{a}_{nn}x_n = W_n \end{cases} \qquad (2)$$

Step 2 If $W_1, W_2, \cdots, W_n$ are known, solve the n-variables linear equations (2).

$$\text{Let } A = \begin{bmatrix} \bar{a}_{11} & \bar{a}_{12} & \cdots & \cdots & \bar{a}_{1n} \\ & & \cdots & \cdots & \\ \bar{a}_{i1} & \bar{a}_{i2} & \cdots & \cdots & \bar{a}_{in} \\ & & \cdots & \cdots & \\ \bar{a}_{n1} & \bar{a}_{n2} & \cdots & \cdots & \bar{a}_{nn} \end{bmatrix}_{n \times n}$$

be the coefficient matrix and

$$\bar{A} = \begin{bmatrix} \bar{a}_{11} & \bar{a}_{12} & \cdots & \bar{a}_{1n} & W_1 \\ & & \cdots & & \\ \bar{a}_{i1} & \bar{a}_{i2} & \cdots & \bar{a}_{in} & W_i \\ & & \cdots & & \\ \bar{a}_{n1} & \bar{a}_{n2} & \cdots & \bar{a}_{nn} & W_n \end{bmatrix}_{n \times (n+1)}$$

be the augmented matrix.

1) If $rank(\bar{A}) \neq rank(A)$, then this linear equations (2) has no solution.

2) If $rank(\bar{A}) = rank(A) = n$, then this linear equations (2) has unique solution. It shows that

when all the total work $W_i, i = 1,2,\cdots,n$ are known, all the nodes weight $x_i, i = 1,2,\cdots,n$ can be solved and are unique.

3) If $rank(\overline{A}) = rank(A) < n$, then this linear equations (2) has infinitely many solutions and has $n - rank(A)$ unknown freedom variables. It shows that when all the total work $W_i, i = 1,2,\cdots,n$ are known, all the nodes weight $x_i, i = 1,2,\cdots,n$ can be solved by $n - rank(A)$ of them.

The background of example 2.1: The container transport as an advanced modernization transport way is very popular and very important in the world trade. But for various reasons, it brings up the empty container transport problem. The empty container transport problem means the transportation of empty container include not only marine transportation but also land transportation. In 2011, there are 61 million empty standard containers transported in the global main routes. The transportation costs of those empty containers are about 36.6 billion dollars. It means that 19% of industry incomes are consumed by the empty container transport. So the study or research of the empty container transport is very important to scientific community and relevant companies.

Because of the unbalanced development of economy, different types of empty containers are piled up or required in different ports. There are different average unit transportation costs by different types of empty containers. And because the distances between ports are different, so there are different average unit transportation costs to different ports even by the same type. The transportation costs equals to the average unit transportation costs multiply by transportation quantity.

Example 2.1 There is a container transportation company going to arrange the transportation of empty containers in the main ports v_1, v_2, v_3, v_4, v_5 on condition that total transportation cost are given.

Suppose there are container ships $S_i\ (i = 1,2,3,4,5)$ in the ports $v_i\ (i = 1,2,3,4,5)$. Every week, the container transportation company sent ships S_i to transport the empty containers starting from the ports v_i, and visit all the other ports and return to the starting port. Each port is visited only once. Suppose the ships S_i's total transportation cost W_i $(i = 1,2,3,4,5)$ are known. Suppose there are T_i types $(i = 1,2,3,4,5)$ of empty containers, which are required by other ports and ready for shipment, piled up in port v_i. The average unit transportation costs (dollars/per empty container) between the ports v_i and v_j of T_i types $(i = 1,2,3,4,5)$ are known, as show in table 1. But the quantities x_i of type T_i are unknown.

Suppose the empty containers will be transported to the port, to which the average unit transportation cost is the minimum. And the empty containers in previous port would be unloaded in the next port. Suppose the quantities x_i are stable and inexhaustible. That means if the previous x_i is carried away, then the next x_i quantity empty containers are immediately ready for shipment instead of.

On condition that the ships S_i's total transportation cost W_i are known, how to arrange the quantity x_i $(i = 1,2,3,4,5)$?

Table 1 The Average Unit Transportation Costs (dollars/per empty container)

Port	Type	The average unit transportation costs (dollars/per empty container)				
		To ports v_1	**To ports v_2**	**To ports v_3**	**To ports v_4**	**To ports v_5**
v_1	T_1	**0**	**13**	**10**	**11**	**15**
v_2	T_2	**13**	**0**	**12**	**6**	**9**
v_3	T_3	**10**	**12**	**0**	**14**	**7**
v_4	T_4	**11**	**6**	**14**	**0**	**8**
v_5	T_5	**15**	**9**	**7**	**8**	**0**

The solution is: According to the question, construct a 5-order complete weighted undirected graph with nodes v_1, v_2, v_3, v_4, v_5 as shown in the Fig.1. Suppose the side weights, which are known, means the average unit transportation costs (dollars/per empty container) between the ports v_i $(i = 1,2,3,4,5)$. Suppose the nodes weights x_i, $i = 1,2,\cdots,5$, which are unknown, means the quantity of type T_i of empty containers. So the question "On condition that the ships S_i's total transportation cost W_i are known, how to arrange the quantity x_i $(i = 1,2,3,4,5)$?" equals to the question "On condition that the total work W_i are known, how to arrange the nodes weight x_i $(i = 1,2,3,4,5)$?"

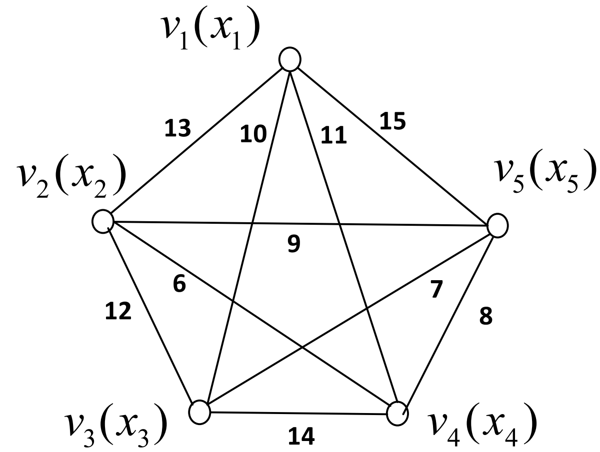

Fig.1. 5-order Complete Weighted Undirected Graph

Step 1. By Nearest Neighbor Algorithm, we let all the nodes as starting node separately. So we can get 5 v_i ($i = 1,2,\cdots,5$)-Nearest Neighbor Circuit, and their total work (the total transportation cost).

1) Let the node v_1 as the starting node. Because of $\min\{13,10,11,15\} = 10$, the nearest node to v_1 is v_3. Then the side v_1v_3 is as the first side of the v_1 - Nearest Neighbor Circuit, as shown in the Fig.2. Therefore, we can get the following nearest neighbor side weight of v_1: $\overline{a}_{11} = 10$ and the work (the transportation cost) of v_1: $10x_1$, in the v_1 - Nearest Neighbor Circuit.

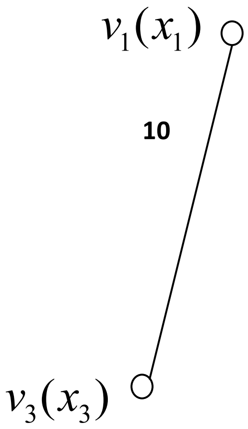

Fig.2. $v_1 \underset{10}{-} v_3$ of the v_1 - Nearest Neighbor Circuit

To the node v_3, because of $\min\{12,7,14\} = 7$, so the nearest node is v_5, besides v_1. Then the side v_3v_5 is as the second side of the v_1 - Nearest Neighbor Circuit, as shown in the Fig.3. Therefore, we can get the following nearest neighbor side weight of v_3: $\overline{a}_{13} = 7$ and the work (the transportation cost) of v_3: $7x_3$, in the v_1 - Nearest Neighbor Ciricuit.

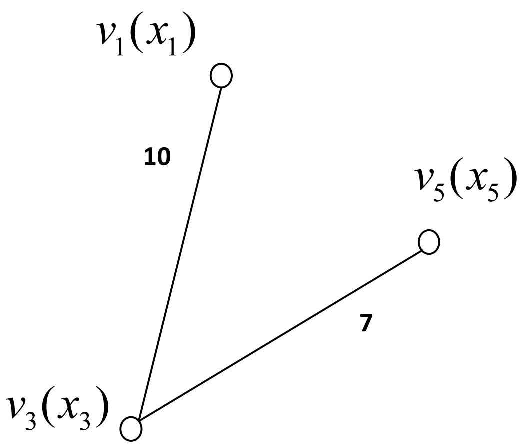

Fig.3. $v_1 \underset{10}{-} v_3 \underset{7}{-} v_5$ of the v_1 - Nearest Neighbor Circuit

To the node v_5, because $\min\{9,8\} = 8$, so the nearest node is v_4, besides v_1 and v_3. Then the side v_5v_4 is as the third side of the v_1 - Nearest Neighbor Circuit, as shown in the Fig.4. Therefore, we can get the following nearest neighbor side weight of v_5: $\overline{a}_{15} = 8$ and the work (the transportation cost) of v_5: $8x_5$, in the v_1 - Nearest Neighbor Circuit.

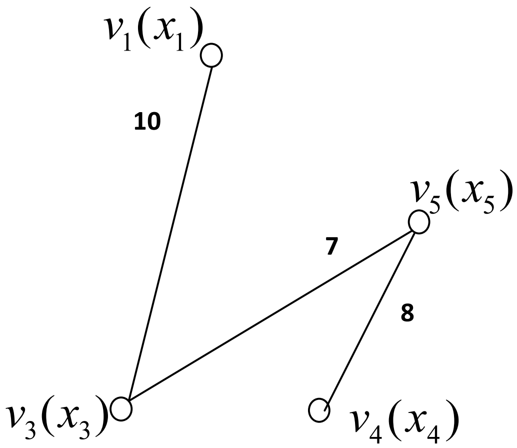

Fig.4. $v_1 \underset{10}{-} v_3 \underset{7}{-} v_5 \underset{8}{-} v_4$ of the v_1 - Nearest Neighbor Circuit

To the node v_4, the nearest node is v_2, besides v_1, v_3 and v_5. Then the side v_4v_2 is as the fouth side of the v_1 - Nearest Neighbor Circuit, as shown in the Fig.5. Therefore, we can get the following nearest neighbor side weight of v_4: $\overline{a}_{14} = 6$ and the work (the transportation cost) of v_4: $6x_4$, in the v_1 - Nearest Neighbor Ciricuit.

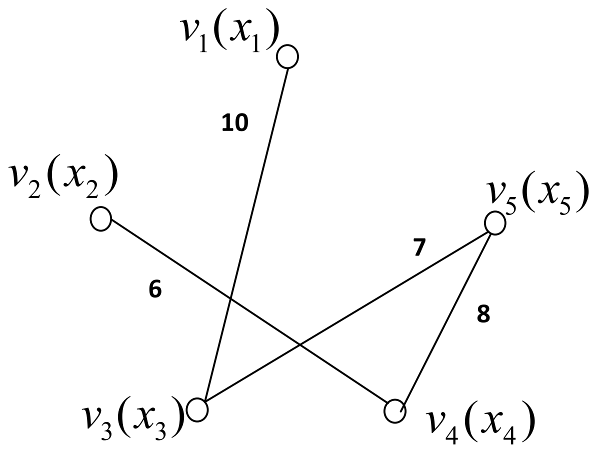

Fig.5. $v_1 \underset{10}{-} v_3 \underset{7}{-} v_5 \underset{8}{-} v_4 \underset{6}{-} v_2$ of the v_1 - Nearest Neighbor Circuit

Link v_2 to v_1. Then the side v_2v_1 is as the fifth side of the v_1 - Nearest Neighbor Circuit.Therefore, we can get the following nearest neighbor side weight of v_2: $\overline{a}_{12} = 13$ and the work (the transportation cost) of v_2 : $13x_2$, in the v_1 - Nearest Neighbor Circuit. So the v_1 - Nearest Neighbor Circuit is finished like that: $v_1 \underset{10}{-} v_3 \underset{7}{-} v_5 \underset{8}{-} v_4 \underset{6}{-} v_2 \underset{13}{-} v_1$, as shown in the Fig.6 and its total work (the total transportation cost) is:

$$10x_1 + 7x_3 + 8x_5 + 6x_4 + 13x_2 = W_1 \ ,$$

That is

$$10x_1 + 13x_2 + 7x_3 + 6x_4 + 8x_5 = W_1 \qquad (3)$$

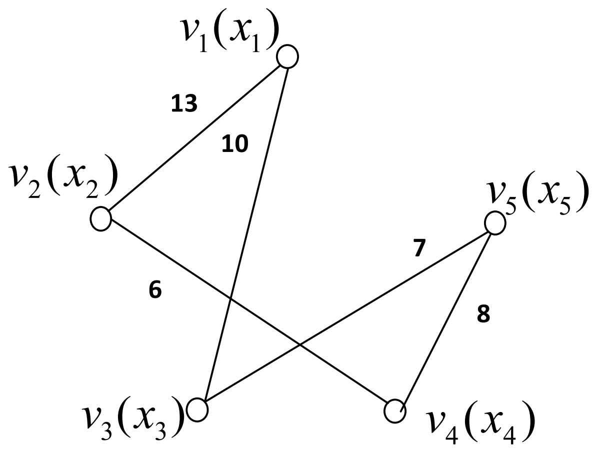

Fig.6. $v_1 \underset{10}{-} v_3 \underset{7}{-} v_5 \underset{8}{-} v_4 \underset{6}{-} v_2 \underset{13}{-} v_1$ Of the v_1 - Nearest Neighbor Circuit

2) In the same way, let the node v_2 as the starting node. Therefore, we can get $\bar{a}_{21} = 6$, $\bar{a}_{22} = 8$, $\bar{a}_{23} = 7$, $\bar{a}_{24} = 10$, $\bar{a}_{25} = 13$. So the v_2 - Nearest Neighbor Circuit is like that: $v_2 \underset{6}{-} v_4 \underset{8}{-} v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{13}{-} v_2$, as shown in the Fig.7 and its total work (the total transportation cost) is:

$$6x_2 + 8x_4 + 7x_5 + 10x_3 + 13x_1 = W_2 \ ,$$

That is

$$13x_1 + 6x_2 + 10x_3 + 8x_4 + 7x_5 = W_2 \qquad (4)$$

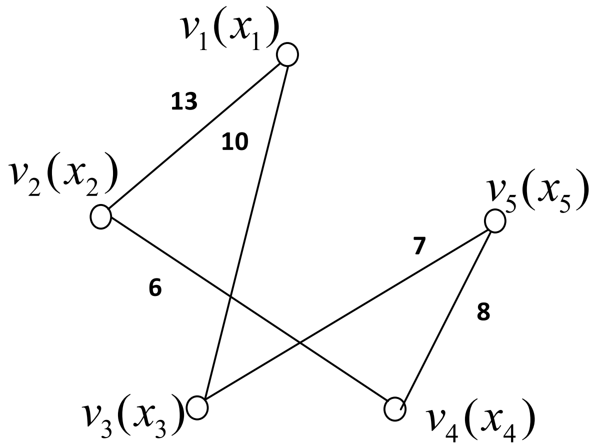

Fig.7. $v_2 \underset{6}{-} v_4 \underset{8}{-} v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{13}{-} v_2$ of the v_2 - Nearest Neighbor Circuit

3) In the same way, let the node v_3 as the starting node. Therefore, we can get $\overline{a}_{31} = 7$, $\overline{a}_{32} = 8$, $\overline{a}_{33} = 6$, $\overline{a}_{34} = 13$, $\overline{a}_{35} = 10$. So the v_3 - Nearest Neighbor Circuit is like that: $v_3 \underset{7}{-} v_5 \underset{8}{-} v_4 \underset{6}{-} v_2 \underset{13}{-} v_1 \underset{10}{-} v_3$, as shown in the Fig.8 and its total work (the total transportation cost) is:

$$7x_3 + 8x_5 + 6x_4 + 13x_2 + 10x_1 = W_3 \ ,$$

That is

$$10x_1 + 13x_2 + 7x_3 + 6x_4 + 8x_5 = W_3 \qquad (5)$$

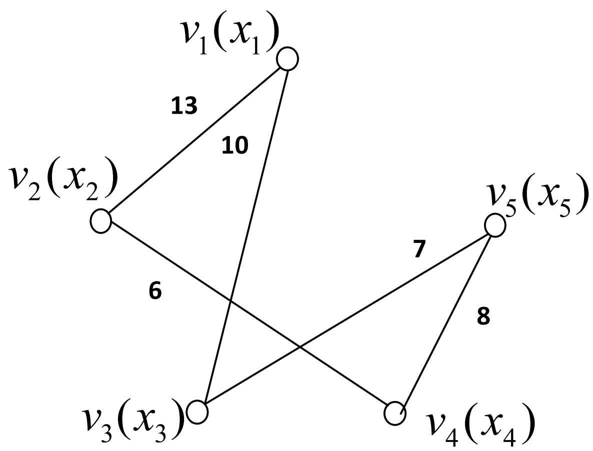

Fig.8. $v_3 \underset{7}{-} v_5 \underset{8}{-} v_4 \underset{6}{-} v_2 \underset{13}{-} v_1 \underset{10}{-} v_3$ of the v_3 - Nearest Neighbor Circuit

4) In the same way, let the node v_4 as the starting node. Therefore, we can get $\overline{a}_{41} = 11$,

$\bar{a}_{42} = 9$, $\bar{a}_{43} = 10$, $\bar{a}_{44} = 6$, $\bar{a}_{45} = 7$. So the v_4 - Nearest Neighbor Circuit is like that $v_4 \underset{6}{-} v_2 \underset{9}{-} v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{11}{-} v_4$, as shown in the Fig.9 and its total work (the total transportation cost) is:

$$6x_4 + 9x_2 + 7x_5 + 10x_3 + 11x_1 = W_4 \ ,$$

That is

$$11x_1 + 9x_2 + 10x_3 + 6x_4 + 7x_5 = W_4 \qquad (6)$$

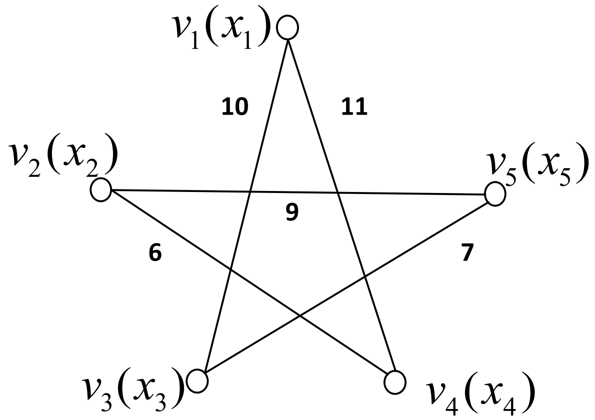

Fig.9. $v_4 \underset{6}{-} v_2 \underset{9}{-} v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{11}{-} v_4$ of the v_4 - Nearest Neighbor Circuit

5) In the same way, let the node v_5 as the starting node. Therefore, we can get $\bar{a}_{51} = 11$, $\bar{a}_{52} = 9$, $\bar{a}_{53} = 10$, $\bar{a}_{54} = 6$, $\bar{a}_{55} = 7$. So the v_5 - Nearest Neighbor Circuit is like that $v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{11}{-} v_4 \underset{6}{-} v_2 \underset{9}{-} v_5$, as shown in the Fig.10 and its total work (the total transportation cost) is:

$$7x_5 + 10x_3 + 11x_1 + 6x_4 + 9x_2 = W_5 \ ,$$

That is

$$11x_1 + 9x_2 + 10x_3 + 6x_4 + 7x_5 = W_5 \qquad (7)$$

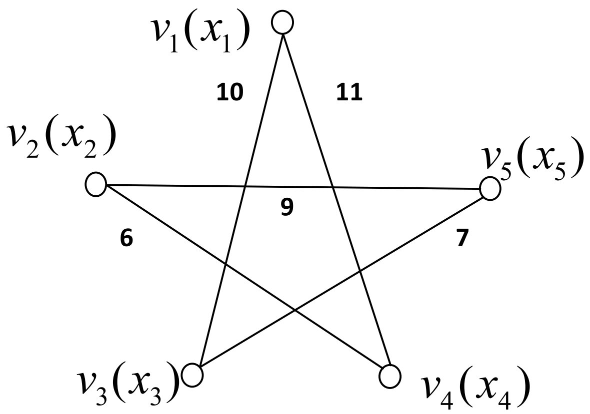

Fig.10. $v_5 \underset{7}{-} v_3 \underset{10}{-} v_1 \underset{11}{-} v_4 \underset{6}{-} v_2 \underset{9}{-} v_5$ of the v_5- Nearest Neighbor Circuit

Therefore, formulas (3)-(7) are 5-variables linear equations (8) with variables x_1, x_2, x_3, x_4, x_5.

$$\begin{cases} 10x_1 + 13x_2 + 7x_3 + 6x_4 + 8x_5 = W_1 \\ 13x_1 + 6x_2 + 10x_3 + 8x_4 + 7x_5 = W_2 \\ 10x_1 + 13x_2 + 7x_3 + 6x_4 + 8x_5 = W_3 \\ 11x_1 + 9x_2 + 10x_3 + 6x_4 + 7x_5 = W_4 \\ 11x_1 + 9x_2 + 10x_3 + 6x_4 + 7x_5 = W_5 \end{cases} \tag{8}$$

We can see that, in fact, the v_1-Nearest Neighbor Circuit, the v_2-Nearest Neighbor Circuit and the v_3-Nearest Neighbor Circuit are the same circuit although by different starting nodes v_1, v_2, v_3. The v_4-Nearest Neighbor Circuit and the v_5-Nearest Neighbor Circuit are the same circuit although by different starting nodes v_4, v_5.

Step 2 Because W_1, W_2, $\cdots$, W_5 are known, solve this 5-variables linear equations (8). Let

$$A = \begin{bmatrix} 10 & 13 & 7 & 6 & 8 \\ 13 & 6 & 10 & 8 & 7 \\ 10 & 13 & 7 & 6 & 8 \\ 11 & 9 & 10 & 6 & 7 \\ 11 & 9 & 10 & 6 & 7 \end{bmatrix}_{5\times 5}$$

be the coefficient matrix and

$$\overline{A} = \begin{bmatrix} 10 & 13 & 7 & 6 & 8 & W_1 \\ 13 & 6 & 10 & 8 & 7 & W_2 \\ 10 & 13 & 7 & 6 & 8 & W_3 \\ 11 & 9 & 10 & 6 & 7 & W_4 \\ 11 & 9 & 10 & 6 & 7 & W_5 \end{bmatrix}_{5\times 6}$$

be the augmented matrix. Then

$$\overline{A} = \begin{bmatrix} 10 & 13 & 7 & 6 & 8 & W_1 \\ 13 & 6 & 10 & 8 & 7 & W_2 \\ 10 & 13 & 7 & 6 & 8 & W_3 \\ 11 & 9 & 10 & 6 & 7 & W_4 \\ 11 & 9 & 10 & 6 & 7 & W_5 \end{bmatrix}_{5\times 6}$$

$$\xrightarrow{linlear \quad transformation} \begin{bmatrix} 1 & 0 & 0 & \frac{188}{203} & \frac{93}{203} & \frac{30W_1 + 67W_2 - 88W_4}{203} \\ 0 & 1 & 0 & -\frac{10}{203} & \frac{62}{203} & \frac{20W_1 - 23W_2 + 9W_4}{203} \\ 0 & 0 & 1 & -\frac{76}{203} & -\frac{16}{203} & \frac{-51W_1 - 53W_2 + 109W_4}{203} \\ 0 & 0 & 0 & 0 & 0 & W_3 - W_1 \\ 0 & 0 & 0 & 0 & 0 & W_5 - W_4 \end{bmatrix}_{5\times 6}$$

1) Because $rank(A) = 3$, $n = 5$, then $rank(A) < n$, so this linear equations (8) has no unique solution.

2) When $W_3 - W_1 \neq 0$ or $W_5 - W_4 \neq 0$, that is $W_3 \neq W_1$ or $W_5 \neq W_4$, then there is $rank(A) = 3 \neq rank(\overline{A})$, so this linear equations (8) has no solution.

3) When $W_3 - W_1 = 0$ and $W_5 - W_4 = 0$. That is $W_1 = W_3$ and $W_4 = W_5$, then there is $rank(A) = rank(\overline{A}) = 3 < n$, so this linear equations (8) has infinitely many solutions and has $5 - 3 = 2$ unknown quantities of freedom. It shows on when all the total work (the total costs of container ships) $W_i \ (i = 1,2,3,4,5)$ are known and $\begin{cases} W_1 = W_3, \\ W_4 = W_5 \end{cases}$, all the nodes weight $x_i \ (i = 1,2,3,4,5)$ can be solved by two of them. So the container transportation company can

first satisfy two quantities of x_i $(i = 1,2,3,4,5)$ which are badly in needs of, then calculate the other three quantities. Suppose $W_1 = W_3 = 10000, W_2 = 11000$, $W_4 = W_5 = 10500$, and the demand priority order and the required quantities of A, B, C, D, E are: D=200, E=300, C=500, A=400, B=250. So, according to priority order, we can first satisfy the $x_4 = 200$ of D and $x_5 = 300$ of E, then solve the other $x_3 = 352$ of C, $x_1 = 233$ of A, $x_2 = 122$ of B by linear equations (8).

The main result of this example 2.1 is: It shows that when all of the total work (the total transportation costs of container ships) $W_i, i = 1,2,\cdots,5$ are known and $\begin{cases} W_1 = W_3, \\ W_4 = W_5 \end{cases}$, all the nodes weight (the quantities) $x_i, i = 1,2,\cdots,5$ can be solved by two of them. This is very important in simulation problem.

3 . Conclusions

1) In method 2.1,let all the nodes as starting node separately and find out all the $\overline{a}_{ij}$ $(i, j = 1,2,\cdots,n)$ by Nearest Neighbor Algorithm.Then we can get n v_i $(i \in 1,2,\cdots,n)$-Nearest Neighbor Ciricuits and n-variables linear equations with variables $x_1, x_2, \cdots, x_n$, because their total work are known. Then solve those n-variables linear equations. So the solutions $x_1, x_2, \cdots, x_n$ are solved.

2) In paper [1], the graphs it discussed are only sides weighted graphs. In paper [2] and this paper, which discussed are complete weighted graphs, that all nodes and sides are weighted; In paper [2], it discuss how to find the best Nearest Neighbor Circuit such that the total work is the minimum, when the nodes weight and sides weight are known. In this paper, it discuss how to solve all the nodes weight when all the sides weight and all the total work are known.

References

1. Xuecai Shao,Tongying Shen (2010) "Discrete Mathematics" [M], Bei Jing, Tsinghua University Press,pp.195-199.

2. Caicai Feng, J. Liang "The Solution of a More General Traveling Salesman Problem", AMSE Journals, Advances A-Mathematics. In press.

3. Hou MengShu,Liu DaiBo (2012) "A Novel Method for Solving the Multiple Traveling Salesmen Problem with Multiple Depots", Chinese Science Bulletin, Vol. 20, pp. 1886-1892.

4. Rongwei Gan,Qingshun Guo,Huiyou Chang,Yang Yi (2010) "Improved Ant Colony Optimization Algorithm for The Traveling Salesman Problems", Journal of Systems Engineering and Electronics, Vol. 8, pp. 329-333.

5. Heow Pueh Lee (2008) “ Solving Traveling Salesman Problems Using Generalized Chromosome Genetic Algorithm”, Progress in Natural Science, pp. 887-892.
6. J. Liang (2008). “The Limit Cycle of a Class Cubic System III”, Journal of Systems Science and Mathematical Science, Vol. 12, pp. 576-587.
7. Liang, Yanchun (2003) “Solving Traveling Salesman Problems by Genetic Algorithms”, Progress in Natural Science, Vol. 4, pp. 135-141.
8. Zhou, TieJun (2012) “A Multi-Agent Approach for Solving Traveling Salesman Problem”, Journal of Wuhan University, Vol. 57, pp. 1886-1892.
9. Jeongho Bang, Junghee Ryu (2012) “A Quantum Heuristic Algorithm for the traveling Salesman Problem”, Journal of the Koren Physical Society, Vol. 186, pp.1944-1949.
10. J. Liang (2012) “The Uniqueness of limit cycle in a class for quantic polynomial system”, AMSE Journals, Advances A-Mathematics, Vol. 49, pp.1-2.
12. Gilad Barach (2012) “Hugo Fort Information in the traveling Salesman Problem”, Applied Mathematics, Vol. 42, pp. 926-930.
13. J. Liang (2005) “The Limit cycle in a class of Quantic Polynomial System II”, AMSE Journals, Advances A-Mathematics, Vol. 30, No 2, pp.1-12.