□研究論文□
韓國軍事運營分析學會誌
第 25 卷, 第 2 號, 1999. 12. 31.

Turn Penalty Algorithm for the Shortest Path Model with Fixed Charges

Choi, Seok Cheol*

Abstract

In this paper, we consider the shortest path network problem with fixed charges. A turn penalty algorithm for the shortest path problem with fixed charges or turn penalties is presented, which is using the next node comparison method. The algorithm described here is designed to determine the shortest route in the shortest path network problem including turn penalties. Additionally, the way to simplify the computation for the shortest path problem with turn penalties was pursued.

* 국방대학원 무기체계학과

1. INTRODUCTION

This article addresses a problem of network design, which we call a fixed charge problem. The shortest path problem with fixed charges or turn penalties is a variation on the standard shortest path problem which determines the minimum cost or minimum distance route from a given source node s to a given sink node t. Conventional shortest route algorithms are based on the following fundamental observations: "If the shortest route between node s and node t passes through node k, then that segment of the route from node s to node k is also the shortest route to node k.

In addition, the route from node k to node t is the shortest route between those two nodes, [10]."

However, in a network with turn penalties the shortest route from node s to node t through an intermediate node k may not include the shortest route from node s to node k or from node k to node t. Hence, the shortest path problem with turn penalties can not be directly solved using conventional shortest path network procedures, [10]. In other word, the fixed charge problem is more difficult to solve. In order to solve this kind of problem, a pseudo network procedure, which is used in modification to the basic problem structure, is utilized, [7, 10]. An extension of the standard shortest path problem, or the shortest problem when some arcs are closed for traveling during specified periods of time, was considered in Halpern and Priess, [2].

The fixed charge or turn penalty model has applications in problems of distribution, communication and transportation. The shortest path problem including turn penalties is particularly important for transportation networks representing the central business district of a metropolitan area. A left turn takes more time than a right turn does at some intersections because of the complexity of traffic movement in the downtown area. In this case the node can be assumed to have turn penalty on the transportation network, when a driver wants to make a turn to the left at the intersection.

The article is organized in the order of the following topics: a turn penalty algorithm for the feasible solution(heuristic) for the fixed charge problem and reporting on the results of the empirical study for all phases of the complete algorithm(including computational simplifications).

2. A HEURISTIC PROCEDURE

In this section a heuristic procedure is proposed. This heuristic enables one to achieve a good solution to the fixed charge problem. In this turn penalty algorithm, the underlying idea is to compare the distance from node 1 to node j using the next node passed node j. The following notation will be used in the description of the turn penalty algorithm:

D(i, j)=the distance from node i to node j;

L(i, ki)=the distance from node 1 to node i on the ki-th way to reach to node i;

L(j, kj)=the distance from node 1 to node j on the kj-th way to reach to node j;

P(j, kj)=the sequence of nodes associated with L(j, kj);

TP(m, i, j)=the turn penalty at node i from node m to node j;

N=the maximum number of ways to reach a node in a network.

The turn penalty algorithm is as follows:

a. Define the distance matrix, where elements are D(i, j), the length from node i to node j, if this exists.

b. Assign a label to every node in the network.

Step 0: Assign a label L(i, ki)=∞, where ki is 1, 2,..., N, and a path P(i, ki)=0 to each node j in the network.

Define TP(m, i, j) for turn penalties at node i.

Set L(1, 1)=0.

Step 1: For each node i,

if the number of L(i, ki)$<\infty$ is equal to n, set ki=n, where ki is the number of ways reached to node i.

Step 2: For each node j, which has a distance from node i less than ∞,

if ki=1, L(j, 1)$\geq\infty$,

compute L(j, 1)=L(i, 1)+D(i, j) and set P(j, 1)=i.

If ki=1, L(j, kj)$<\infty$, increase kj by 1and compute

L(j, kj+1)=L(i, 1)+D(i, j) and set P(j, kj+1)=i.

If ki=n, L(j, 1)$\geq\infty$, compute

L(j, kj)=L(i, ki)+D(i, j),

and set P(j, kj)=i, where n=2, 3,..., N, ki=1, 2,..., n, kj=1, 2,..., n.

Reset L(j, 1)=min{L(j, kj)}, where kj=1, 2,..., n.

Drop L(j, kj), where kj=2, 3,..., n.

If ki=n, L(j, kj)<∞, increase kj by 1 and compute

L(j, kj+1)=L(i, ki)+D(i, j) and set P(j, kj+1)=i, where n=2,3,..., N, ki=1,2..., n., kj+1=N-kj,..., N.

Step 3: If there is a turn penalty from node m through node i to node j,

L(j, kj)=L(j, kj)+TP(m, i, j).

Otherwise, L(j, kj)=L(j, kj).

Step 4: Increase i by 1.

If i is equal to the total number of nodes, stop.

Otherwise, go to Step 1.

3. COMPUTATIONAL RESULTS AND DISCUSSION

3.1 Shortest Path with Turn Penalty

The fixed charge problem arises when certain additional penalties or cost incurred in traversing through one or more nodes in a network. To demonstrate the application of turn penalty algorithm in the shortest path problem with turn penalties, let us consider the directed, rectangular grid network of Figure 3.1.

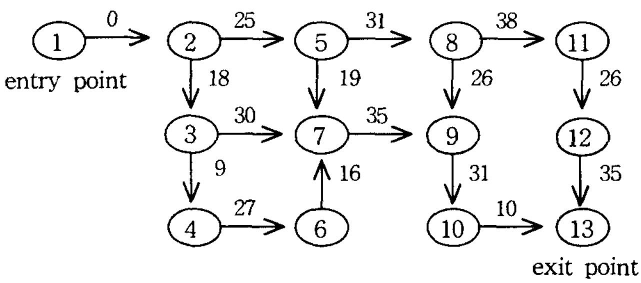

Figure 3.1 Sample Network with Turn Penalty

This example problem was selected in order to check the procedure and results of a computer program by hand-calculation. This directed network represents a segment of city traffic intersections leading from an entry point at node 1 to an exit point at node 13. The distance of traversing each arc is shown beside each arc. Additionally, it is assumed that any turn will incur each fixed charge for each turn and fixed charges or turn penalties can vary from node to node as shown in Table 3.1. The turn penalty is defined by using the three-dimensional array such as TP(i, j, k). We will discuss this turn penalty later.

Table 3.1 Turn Penalty Value

No.	Node i	Node j	Node k	Penalty
1	1	2	3	1
2	2	3	7	2
3	2	5	7	1
4	3	4	6	2
5	4	6	7	1
6	5	7	9	1
7	5	8	9	2
8	6	7	9	1
9	7	9	10	2
10	8	11	12	1
11	9	10	13	2

It is desired to identify the most economical or shortest path to travel from node 1 to node 13. Since the number of nodes is 13(n=13), the number of iterations for this algorithm is equal to 12 which is the number of nodes minus one. Initially the distance matrix is defined, where D(i, j) is a distance from node i to node j in a directed network. If the arc(i, j) does not exist, the infinite number is assigned for computation. The original distance matrix for the example problem is as shown in Table 3.2.

For clarity, the infinite number elements of the matrix have been omitted.

(1) Adding Turn Penalty

There are two kinds of the fixed charge problem in a network:

Table 3.2 Original Distance Matrix

i \ j	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0											
2		0	18		25								
3			0	9			30						
4				0		27							
5					0		19	31					
6						0	16						
7							0		35				
8								0	26		38		
9									0	31			
10										0			10
11											0	26	
12												0	35
13													0

One of the fixed charge problem occurs when sending flow along an arc involves a fixed cost for using the arc regardless of any turn as long as the amount of flow is positive. This fixed charge or fixed cost flow problem was considered in Hochbaum and Seger, [3]. The above model can be used to solve optimization problems in such areas as machine scheduling, metal processing, financial budgeting, aircraft routing, warehousing, catering, etc. [6]. In this case the fixed charge can be represented by using two-dimensional array such as FC(i, j) which is a fixed cost from node i to node j.

The other one of the fixed charge problems occurs when any turn will incur each fixed charge at a specified node. In this case it is important to remember the previous node and the future node at the present node, because the turn penalty depends on these three nodes. As previously mentioned, this turn penalty can be represented by using three-dimensional array such as TP(i, j, k) which is a turn penalty from node i through node j to node k.

As can be seen in the turn penalty value in Table 3.1, we use the three-dimensional array for the turn penalty for each turn. When determining the distance from node 1 to node k, if there is any turn from node 1 through node j to node k, we add the turn penalty value to the distance from node 1 to node k. If there is no turn at a node, only the distance from node j to node k will be considered for the total distance from node 1 to node k through an intermediate node j.

(2) Finding Shortest Path with Turn Penalty

The application of this turn penalty algorithm to the example network with fixed charges gives the following intermediate results.

(a) The total distance from node 1 to node j and the way to reach to node j are as shown in Table 3.3.

Table 3.3 Distance and Way from Node 1 to Node j

j \ kj	1	2	3	4	5
1	0				
2	0				
3	19				
4	28				
5	25				
6	57				
7	51	45	74		
8	56				
9	86	81	110	84	
10	119	114	143	115	
11	94				
12	122				
13	130	125	154	126	157

For clarity, the infinite number elements are omitted in this table.

(b) The paths from node i to node j associated with the total distance in Table 3.3. are as shown in Table 3.4.

For instance the route to go from node 1 to node 8 is only one way with the distance of 56. The corresponding path is 1-2-5-8.

However, the route to go from node 1 to node 7, in which there are three ways, has a minimum distance of 45.

In conventional shortest path network procedures, we can compare the distances and choose the minimum distance of 45 from node 1 to node 7. In the fixed

Table 3.4 Path from Node i to Node j

j \ kj	1	2	3	4	5
1	0	0	0	0	0
2	1	0	0	0	0
3	2	0	0	0	0
4	3	0	0	0	0
5	2	0	0	0	0
6	4	0	0	0	0
7	3	5	6	0	0
8	5	0	0	0	0
9	7	7	7	8	0
10	9	9	9	9	0
11	8	0	0	0	0
12	11	0	0	0	0
13	10	10	10	10	12

charge problem, we can not select the minimum distance now, because paths from node 5 to node 7 and from node 6 to node 7 have turn penalties at node 7, but the path from node 3 to node 7 does not have turn penalties at node 7. Therefore, we can compare the total distance from node 1 to node 7 at node 9 and select the minimum distance of 81 which can be a candidate for the shortest path. This is why we can call the algorithm the next node comparison method. The corresponding path is 1-2-5-7. We can ignore the other path from node 1 to node 7.

There are two different paths from node 1 to node 9:

The first path is from node 1 through node 7 to node 9 with the distance of 81 (Path : 1-2-5-7-9);

The second path is from node 1 through node 8 to node 9 with the distance of 84 (Path : 1-2-5-8-9).

At node 10 we can choose the minimum distance of 114 between the distance of 114(The corresponding path is 1-2-5-7-9-10) and 115(The corresponding path is 1-2-5-8-9-10), because of the same reason stated above. The path with the distance of 115 will be dropped for the next computation.

Finally at node 13, we can compare the distances from node 1 through node 9 to node 13 and from node 1 through node 11 to node 13. The minimum distance from node 1 to node 13 is equal to 125. In order to find the corresponding sequence of nodes, refer to the matrix in Table 3.4 and proceed as follows: The value of P(13, 2) associated with the value of L(13, 2) is equal to 10, indicating that node 10 is the previous node of node 13, which is the exit point. Now we obtain the value of P(10, 2) to identify the previous node of node 10. This value is equal to 9. Similarly, the value P(9, 2)=7 indicates that node 7 is the previous node of node 9. Also we can find P(7, 2)=5, and P(5, 1)=2, and also P(2, 1)=1 in the same way. Therefore, the shortest path from node 1 to node 13 is 1-2-5-7-9-10-13 with the distance of 125.

3.2 Computational Simplifications

It is very important to number the nodes particularly for this kind of shortest path problem, because the order in which the nodes are numbered affects the computational efficiency of turn penalty algorithm in a computer program, as well as Floyd's algorithm and the double sweep method. The distance matrix and adjacency matrix with an upper triangular matrix which is as shown in Table 3.5, can simplify computational procedure.

In this case we need to take care of the arc (i, j) where j is greater than i. In other words, we can ignore the arc(i, j) where i is greater than j because the arc(i, j) does not exist from node i to node j in this directed network.

The adjacency matrix can be formed as an upper triangular matrix in this kind of the directed network as shown in Figure 3.1. The reason is that this is a directed network and there is no arc from node i to node j, where i is greater than j.

4. CONCLUSION

4.1 Conclusion

In this paper, we examined the turn penalty algorithm for determining the shortest path in the shortest path problem with turn penalties in a directed network. The procedure that we presented, based on new comparison method, corresponds well.

The fixed charge problem with zero fixed cost or zero turn penalty which is a

Table 3.5 Upper Triangular Adjacency Matrix

i \ j	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	1	0	0	0	0	0	0	0	0
3	0	0	0	1	0	0	1	0	0	0	0	0	0
4	0	0	0	0	0	1	0	0	0	0	0	0	0
5	0	0	0	0	0	0	1	1	0	0	0	0	0
6	0	0	0	0	0	0	1	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	1	0	0	0	0
8	0	0	0	0	0	0	0	0	1	0	1	0	0
9	0	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	1
11	0	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0	0	0	0	0	0	0	0	1
13	0	0	0	0	0	0	0	0	0	0	0	0	0

regular shortest path problem can be solved by this turn penalty algorithm without any changes. When the path reached to a node from node 1 is less than or equal to 5, the problems are easily solvable by the FORTRAN program which we have written. However, when the way reached to a node from node 1 is greater than 5, the program should be modified to solve the problems.

4.2 Areas of Further Research

This work is a new trial which determine the shortest path in the shortest path problem with turn penalties by a modified regular shortest path algorithm. However, my work was restricted to the directed network problem. Thus, a possible extension of this research would be to extend this result to an undirected network problem for general application.

Additionally, in this paper the algorithm was applied to the single source and the single destination case. Another area for future research can be the case of the single source and the multiple destinations, or the case of the multiple sources and the single destination.

REFERENCES

[1] Grinold, R. C., "Calculating Maximal Flows in a Network with Positive Gains," Operations Research, 21, pp. 528-541, 1973.

[2] Halpern J. and Priess I., "Shortest Path with Time Constraints on Movement and Parking," Networks, 4, 1974, pp. 241-253.

[3] Hochbaum, Dorit S. and Seger, Arie, "Analysis of a Flow Problem with Fixed Charges," Networks, 19, 1989, pp. 291-312.

[4] Jarvis, J. J., and A. M. Jezior, "Maximal Flow with Gains through a Special Network," Operations Research, 20, pp. 678-688, 1972.

[5] Jensen, P. A., and G. Bhaumik, "A Computationally Efficient Algorithm for the Network with Gains Problem," Paper presented at the 45th National Meeting of ORSA, Boston, April 1974.

[6] Jewell, W. J., "Optimal Flow Through Networks with Gains," Operations Research, Vol. 10, No. 4, 1962, pp. 476-499.

[7] Kibby, F. R. and R. B. Potts, "The Minimum Route Problem for Networks with Turn Penalties and Prohibitions," Trans. Res., 3, 1969, pp. 397-408.

[8] Maurras, J. F., "Optimization of the Flow through Networks with Gains," Mathematical Programming, 3, pp. 135-144, 1972.

[9] Minieka, E. T., "Optimal Flow in a Network with Gains," INFOR, 10, pp. 171-178, 1972.

[10] Phillips, Don T. and Alberto Garcia-Diaz, "Fundamentals of Network Analysis," Prentice-Hall, 1981, pp. 68-72.

[11] Truemper, K., "An Efficient Scaling Procedure for Gains Networks," Networks, 6, pp. 151-160, 1976.

[12] Truemper, K., "Optimal Flows in Nonlinear Gains," Networks, 8, pp. 17-36, 1978.

[99년 9월 2일 접수, 99년 11월 12일 최종수정]