Mathematics and Computer Science
2022; 7(6): 113-117
http://www.sciencepublishinggroup.com/j/mcs
doi: 10.11648/j.mcs.20220706.12
ISSN: 2575-6036 (Print); ISSN: 2575-6028 (Online)

An Alternative Method of Floyd-Warshall Algorithm Is Proposed for Solving Shortest Path Problems Using Triangular Procedure

Md Zahidul Islam*, Md Asadujjaman, Mahabub Rahman

Department of Mathematics, University of Dhaka, Dhaka, Bangladesh

Email address:
mdzahidul-2015318460@math.du.ac.bd (Md Zahidul Islam), asad132@du.ac.bd (Md Asadujjaman),
mahabub-2015317100@math.du.ac.bd (Mahabub Rahman)

*Corresponding author

To cite this article:
Md Zahidul Islam, Md Asadujjaman, Mahabub Rahman. An Alternative Method of Floyd-Warshall Algorithm Is Proposed for Solving Shortest Path Problems Using Triangular Procedure. *Mathematics and Computer Science*. Vol. 7, No. 6, 2022, pp. 113-117.
doi: 10.11648/j.mcs.20220706.12

Received: October 25, 2022; **Accepted**: November 18, 2022; **Published**: December 8, 2022

Abstract: This paper presents a new approach for finding the shortest path compared with the Floyd-Warshall algorithm which is mostly used for determining the shortest path between every pair of nodes of network modeling problems. Finding the smallest route through a road network is one of the innumerable real-world applications of the shortest path problem. In this paper, we will discuss the existing Floyd-Warshall algorithm. Then we will explain our new approach to solving shortest-path problems. The method is developed based on a right-angle triangle. This technique of solving the shortest path problem brings out the same result as the existing Floyd-Warshall algorithm. Additionally, we demonstrate that the foundation of our algorithm is simpler to comprehend, which could be helpful for instructional reasons. An example verifies our algorithm and demonstrates how it is used. We hope this paper will give the reader an idea of the network modeling problems and their efficacy, enumerate the benefits gained, and identify areas for further improvement.

Keywords: Network Programming, Shortest Path Problem, Floyd–Warshall Algorithm, Triangular Algorithm

1. Introduction

Operation Research (OR) is a way that helps to make better decision in complex scenarios. You'll come to appreciate the power of OR techniques in solving real-world problems and applications [1]. Application of OR technique spread over various fields in engineering, management and public system [2-3]. The Shortest Path Problem is an indispensable and well-known problem in operations research, which is related to finding a path between two nodes of a network such that the sum of the weights (distance, time, cost etc.) of its connecting edges is minimized [3]. Finding the quickest route via a road network is one of the numerous real-world applications of the shortest path problem. The nodes often represent as the locations and the edge represents the distance required to travel. There are two kinds of network, with cycle and without cycle. Both scenarios have algorithms with a guaranteed optimal outcome. Optimization is an effort to obtain the best results by considering the constraints, and limitations that exist on a problem [9, 14] In case of without cycle there is a source & a sink. But for a network with cycle there is no source & sink. Every node can be a source or a sink. In this work we will discuss the shortest path problem with cycles on a network. First, we will discuss the Floyd–Warshall algorithm that finds both the shortest routes and the minimum cost between every pair of nodes on this network [13], and then develop a new algorithm for this problem that reduces the required computational effort of the Floyd–Warshall algorithm. Besides, the understanding of our proposed algorithm is much easier than that of the currently available Floyd-Warshall algorithms. The work is organized as follows. Section 2 provides some basic definition. Section 3 provides a summary on the Floyd–Warshall algorithm. In Section 4, we have discussed our proposed algorithm. Section 5, a brief illustration that supports our algorithm. The work ends with the conclusion [4].

2. Definitions

2.1. Network

A network consists of a set of nodes linked by arcs [7]. The modulator for expressing a network is (N, A). Where N is the node A is the arc. The network is described as N = {1,2,3,4} & A = {(1,2), (1,3), (2,3), (3,4), (4,2)} [4].

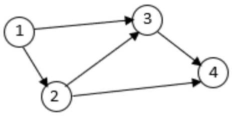

Figure 1. *Network.*

2.2. Weighted Network

In a network if each arc is associated with a number which represent some type of flow, distance, cost etc between two nodes then the network is called weighted network.

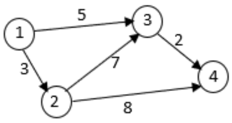

Figure 2. *Weighted Network.*

2.3. Directed Network

If an arc allows a positive flow in one direction and zero flow in opposite direction is called directed arcs & a directed network has directed arcs [8].

2.4. Path of a Network

A path is a sequence of distinct arcs that join two nodes through other nodes regardless of direction of flow in each other [5].

2.5. Connected Network

A connected network is such that every two distinct nodes are linked by at least one path [5].

2.6. Cycle

A path form a cycle if it connects a node to itself through other nodes [5].

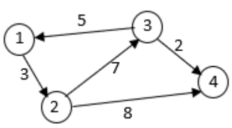

Figure 3. *Cycle.*

2.7. Shortest Path of a Network Problem

Shortest path problem is the problem of finding a path between two nodes in a graph such that the sum of the weight of its constituent edges is minimized [6].

3. The Floyd-Warshall Algorithm

Floyd-Warshall algorithm probably one of the best algorithm for finding the shortest path between a pair of nodes of a network [11, 15]. The algorithm represents a network of n nodes as a square matrix with n rows and n columns [12]. Given a network $N(V, A)$ where $V = \{1,2,3,\cdots\cdots,n\}$ and arc set.

$A = \{(i,k): i,k \in V\}$. This algorithm is established on four steps procedure. First, two square matrices D_j *and* R_j for $j = 0,1,2\cdots\cdots,n$ holding the shortest route from source to sink between every two arbitrary nodes i and k respectively. The algorithm is simple but it requires lot of calculations. For a network with n nodes Floyd-Warshall algorithm requires D_j *and* R_j matrices to be calculated for $n+1$ times starting from D_0 *and* R_0. Here is the explanations of Floyd-Warshall algorithm [3]. For details one may search [4-5].

Algorithm:

Step 01. Set D_j *and* R_j as two square $n \times n$ matrices, where j is the number of stage and n is the total number of nodes of the network.

Step 02. For $j = 0$ calculate D_0 *and* R_0

$$D_0 = [d_{ik}]$$

where

$$d_{ik} = d_{ik} \text{ } if \text{ } there \text{ } is \text{ } a \text{ } direct \text{ } route \text{ } from \text{ } i \text{ } to \text{ } k$$

$$d_{ik} = \infty \text{ } if \text{ } there \text{ } is \text{ } no \text{ } direct \text{ } route \text{ } from \text{ } i \text{ } to \text{ } k$$

$$d_{ik} = 0 \text{ } if, i = k$$

$$R_0 = [r_{ik}]$$

where

$$r_{ik} = k \text{ } if \text{ } there \text{ } is \text{ } a \text{ } direct \text{ } route \text{ } from \text{ } i \text{ } to \text{ } k$$

$$r_{ik} = - \text{ } if \text{ } there \text{ } is \text{ } no \text{ } direct \text{ } route \text{ } from \text{ } i \text{ } to \text{ } k$$

$$r_{ik} = - \text{ } if, i = k$$

Step 03. For the rest of $j = 1,2,\ldots\ldots.n$ calculate the D_j *and* R_j matrices as follows From now, we derive the entities of the D_j *and* R_j matrices on the basis of the most recent previous matrices [5].

$That \text{ } is, the \text{ } D_{j-1}$ *and* R_{j-1} matrices.

$$D_j = [d_{ik}]$$

where

$$d_{ik} = d_{ik} \text{ } if \text{ } i = k$$

Otherwise

$$d_{ik} = min(d_{ik}, d_{ij} + d_{jk})$$

$$R_j = [r_{ik}]$$

where

$$r_{ik} = k \ if \ i = k, i = j, k = j$$

$$r_{ik} = k \ if \ d_{ik} < d_{ij} + d_{jk}$$

$$r_{ik} = j \ if \ d_{ik} > d_{ij} + d_{jk}$$

Step 04. Repeat Step 3 until we get D_j *and* R_j

4. Algorithm of the Proposed Method (Triangular Algorithm)

Now we will present our main beneficiation of this work. We will introduce a new algorithm that alleviates the amount of calculations of Floyd-warshall algorithm. This is a right angle triangular graphical approach that's why we named it Triangular Algorithm. The concept is too easy to understand. This process could be an advantage for educational purposes. The main idea of this algorithm is set of right angle triangle. The algorithm is given below.

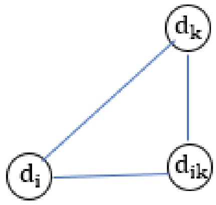

Figure 4. *Constructing a Right Angle Triangle for Triangular Algorithm.*

Algorithm:

Step 01: Set D_j *and* R_j as two square $n \times n$ matrices, where n is the number of nodes & j is the stage number.

Step 02: Derive the D_0 *and* R_0 matrices by following step 02 of the Floyd-Warshall algorithm.

Step 03: For remaining $j = 1,2,\cdots\cdots,n$ calculate the D_j matrices applying one of the following rule.

(a) If an ∞ exist in any row or/and in any column of the D_j matrix, the remaining entities of that row or that column respectively, next to it will not change. Thus they can be substituted with their values from the D_{j-1} matrix for $j \geq 1$.

(b) If (a) does not result in a complete D_j matrix, derive those remaining entities by drawing a sets of triangle as Figure 4 illustrates (For each d_{ik}, $\forall_{i,k}$ in the D_j matrix forms right angle triangle).

(c) Derives the R_j matrix as follows. The R_j matrix is derived depending on D_j matrix. If an entity does not change the R_j matrix will not change. On the other hand, if an entity changes in the D_j matrix, its corresponding R_j matrix can be substituted with j.

Step 04: Repeat step 03 until D_n and R_n are yielded.

5. An Example

In this part, we will discuss on triangular algorithm by solving an example. We will solve the problem with both algorithm (the Floyd-Warshall algorithm and the Triangular algorithm) and we will show that the result is same for both the algorithm. The network problem is given below (Figure 5).

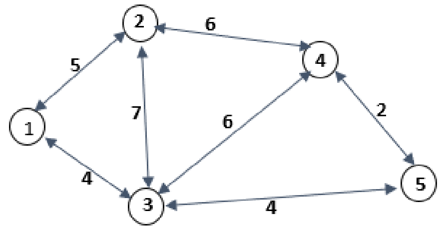

Figure 5. *Example of the Network Problem.*

First we will solve the problem using Floyd-Warshall algorithm, then we will apply Triangular algorithm. It is trivial to show that following the Floyd-Warshall algorithm, D_5 *and* R_5 would be

$$D_5 = \begin{bmatrix} 0 & 5 & 4 & 10 & 8 \\ 5 & 0 & 7 & 6 & 8 \\ 4 & 7 & 0 & 6 & 4 \\ 10 & 6 & 6 & 0 & 2 \\ 8 & 8 & 4 & 2 & 0 \end{bmatrix} R_5 = \begin{bmatrix} - & 2 & 3 & 3 & 3 \\ 1 & - & 3 & 4 & 4 \\ 1 & 2 & - & 4 & 5 \\ 2 & 2 & 3 & - & 5 \\ 3 & 4 & 3 & 4 & - \end{bmatrix}$$

However, a complete calculation for deriving only D_1 *and* R_1 is given below.

From step 3 of Floyd-Warshall algorithm D_1 would result in the following values. $For\ j = 1$

$$d_{23} = min\ (d_{23,}, d_{21} + d_{13}) = min\ (7,5 + 4) = 7$$

$$d_{24} = min\ (d_{24,}, d_{21} + d_{14}) = min\ (6,5 + \infty) = 6$$

$$d_{25} = min\ (d_{25,}, d_{21} + d_{15}) = min\ (\infty, 5 + \infty) = \infty$$

$$d_{32} = min\ (d_{32,}, d_{31} + d_{12}) = min\ (7,4 + 5) = 7$$

$$d_{34} = min\ (d_{34,}, d_{31} + d_{14}) = min\ (6,4 + \infty) = 6$$

$$d_{35} = min\ (d_{35,}, d_{31} + d_{15}) = min\ (\infty, 4 + \infty) = \infty$$

$$d_{42} = min\ (d_{42,}, d_{41} + d_{12}) = min\ (6, \infty + 5) = \infty$$

$$d_{43} = min\ (d_{43,}, d_{41} + d_{13}) = min\ (6, \infty + 4) = 6$$

$$d_{45} = min\ (d_{45,}, d_{41} + d_{15}) = min\ (2, \infty + \infty) = 2$$

$$d_{52} = min\left(d_{52,}, d_{51} + d_{15}\right) = min(\infty, \infty + 5) = \infty$$

$$d_{53} = min\ (d_{53,}, d_{51} + d_{15}) = min\ (4, \infty + 4) = 4$$

$$d_{54} = min\ (d_{54,}, d_{51} + d_{15}) = min\ (2, \infty + \infty) = 2$$

$$Thus\ D_1 = \begin{bmatrix} 0 & 5 & 4 & \infty & \infty \\ 5 & 0 & 7 & 6 & \infty \\ 4 & 7 & 0 & 6 & 4 \\ \infty & 6 & 6 & 0 & 2 \\ \infty & \infty & 4 & 2 & 0 \end{bmatrix} and\ R_1 = \begin{bmatrix} - & 2 & 3 & 4 & 5 \\ 1 & - & 3 & 4 & 5 \\ 1 & 2 & - & 4 & 5 \\ 1 & 2 & 3 & - & 5 \\ 1 & 2 & 3 & 4 & - \end{bmatrix}$$

By continuing these process we will get,

$$D_5 = \begin{bmatrix} 0 & 5 & 4 & 10 & 8 \\ 5 & 0 & 7 & 6 & 8 \\ 4 & 7 & 0 & 6 & 4 \\ 10 & 6 & 6 & 0 & 2 \\ 8 & 8 & 4 & 2 & 0 \end{bmatrix} \; and \; R_5 = \begin{bmatrix} - & 2 & 3 & 3 & 3 \\ 1 & - & 3 & 4 & 4 \\ 1 & 2 & - & 4 & 5 \\ 2 & 2 & 3 & - & 5 \\ 3 & 4 & 3 & 4 & - \end{bmatrix}$$

We showed how the Floyd-Warshall algorithm works. Now we continue with the example by explaining the triangular algorithm. As $n = 5$ we need two square 5×5 matrices D_j and R_j for six stages, Starting from D_0 and R_0.

$$Here\ D_0 = \begin{bmatrix} 0 & 5 & 4 & \infty & \infty \\ 5 & 0 & 7 & 6 & \infty \\ 4 & 7 & 0 & 6 & 4 \\ \infty & 6 & 6 & 0 & 2 \\ \infty & \infty & 4 & 2 & 0 \end{bmatrix} \text{ and } R_0 = \begin{bmatrix} - & 2 & 3 & 4 & 5 \\ 1 & - & 3 & 4 & 5 \\ 1 & 2 & - & 4 & 5 \\ 1 & 2 & 3 & - & 5 \\ 1 & 2 & 3 & 4 & - \end{bmatrix}$$

Following the Triangular algorithm to derive the D_1 and R_1matrices, since $j = 1$ the firs row and first column will remain same. The diagonal elements remain zero. From step 03 (a) there are ∞ in the first row and column. So we need to calculate only d_{23}, d_{32} only.

$$D_0 = \begin{bmatrix} 0 & 5 & 4 & \infty & \infty \\ 5 & 0 & 7 & 6 & \infty \\ 4 & 7 & 0 & 6 & 4 \\ \infty & 6 & 6 & 0 & 2 \\ \infty & \infty & 4 & 2 & 0 \end{bmatrix}$$

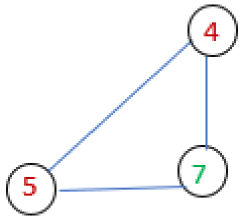

Figure 6. *Constructed Triangle for Calculating Entity* d_{23}.

$$D_0 = \begin{bmatrix} 0 & 5 & 4 & \infty & \infty \\ 5 & 0 & 7 & 6 & \infty \\ 4 & 7 & 0 & 6 & 4 \\ \infty & 6 & 6 & 0 & 2 \\ \infty & \infty & 4 & 2 & 0 \end{bmatrix}$$

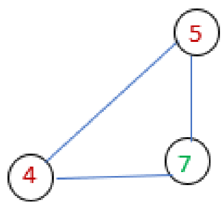

Figure 7. *Constructed Triangle for Calculating Entity* d_{32}.

To calculate d_{23}, as $j = 1$, we leave d_{11} element and construct a right angle triangle in D_0 starting at d_{23} collaborating with the elements of first row and first column as shown in the (figure 6). Thus $d_{23} = min\ (7{,}5 + 4) = 7$. Similarly $d_{32} = min\ (7{,}4 + 5) = 7$ (Figure 7).

$$Thus\ D_1 = \begin{bmatrix} 0 & 5 & 4 & \infty & \infty \\ 5 & 0 & 7 & 6 & \infty \\ 4 & 7 & 0 & 6 & 4 \\ \infty & 6 & 6 & 0 & 2 \\ \infty & \infty & 4 & 2 & 0 \end{bmatrix}$$

$$It\ is\ trivial\ to\ show\ that\ R_1 = \begin{bmatrix} - & 2 & 3 & 4 & 5 \\ 1 & - & 3 & 4 & 5 \\ 1 & 2 & - & 4 & 5 \\ 1 & 2 & 3 & - & 5 \\ 1 & 2 & 3 & 4 & - \end{bmatrix}$$

Continuing this process we will stop at

$$D_5 = \begin{bmatrix} 0 & 5 & 4 & 10 & 8 \\ 5 & 0 & 7 & 6 & 8 \\ 4 & 7 & 0 & 6 & 4 \\ 10 & 6 & 6 & 0 & 2 \\ 8 & 8 & 4 & 2 & 0 \end{bmatrix} \; and \; R_5 = \begin{bmatrix} - & 2 & 3 & 3 & 3 \\ 1 & - & 3 & 4 & 4 \\ 1 & 2 & - & 4 & 5 \\ 2 & 2 & 3 & - & 5 \\ 3 & 4 & 3 & 4 & - \end{bmatrix}$$

That is the same as the Floyd-Warshall algorithm's outcome, which allows us to determine the shortest path between any two network nodes. Again we can say that the Triangular algorithm will minimize the amount of calculation considerably.

6. Conclusion

The goal of this paper is to develop a technique to solve the shortest path problems of a network model. So, we introduced a new algorithm for finding the shortest path of any pair of nodes. We have discussed the Floyd-Warshall algorithm first then we described our new approach (Triangular Algorithm). The Floyd–Warshall algorithm and the Triangular algorithm have exactly the same performance in deriving the D_0 and the R_0 matrices. Also we have got similar result for D_n and the R_n matrices. However, the corresponding matrices are derived by the triangular algorithm more faster because there are fewer calculations involved. The possibility of future research based on this paper lies in the development of our proposed procedure.

References

[1] Paul A. Jensen, Operations Research Models and Methods, Wiley (1865).

[2] Gupta, P. K., D. S. Hira, Problem in Operation Research, S. Chand & Company Ltd, Ram Nagar.

[3] Introduction to Operation Research by prof. G. Srinivasan, IIT Madras.

[4] Operation Research: An introduction by Hamdy A. Taha, ninth edition 2011.

[5] R. E. Bellman, On a routing problem, Quarterly of Applied Mathematics 16 (1) (1958) 87–90.

[6] E. W. Dijkstra, A note on two problems in connection with graphs, Numeriskche Mathematik 1 (1959) 269–271.

[7] L. R. Ford, D. R. Fulkerson, Flows in Networks, Princeton University Press, Princeton, NJ, 1962.

[8] G. Gallo, S. Pallotino, Shortest paths algorithms, Annals of Operations Research 13 (1988) 3–79.

[9] B. V. Cherkassky, A. V. Goldberg, T. Radzik, Shortest paths algorithms: theory and experimental evaluation, Mathematical Programming 73 (2) (1996) 129–174.

[10] R. W. Floyd, Algorithm 97, Communications of the ACM 5–6 (1962) 345.

[11] S. Warshall, A theorem on boolean matrices, Journal of the ACM 9 (1962) 11–12Rn are yielded.

[12] Asghar Aini, Amir Salehipur, Speeding up Floyd-Warshall algorithm for the cycled shortest path Problems Shahid Sattari University, Iran.

[13] Hiller F. S., Liberman G. J., Introduction to Operations Research, The McGraw-Hill Companies, Inc., 2011.

[14] Prim and Floyd-Warshall Comparative Algorithms in Shortest Path Problem, Zuhri Ramadhan, Faculty of Science and Technology, Universitas Pembangunan Panca Budi, Medan, Indonesia, 2018.

[15] Improving The Floyd-Warshall All Pairs Shortest Paths Algorithm, Ismail H. Toroslu, Ankara, Terkey-2021.