Simplification of Dijkstra's algorithm to find the Single Source Shortest path for N vertices using Path Matrix

[1]Pooja Singh,[2]Prof. Sanjeev Kumar Singh
[1]M.tech Scholar,[2]Head of Department
[1,2]Department of Computer Science and Engineering
[1,2] VIET Dadri G.B. Nagar U.P.

Abstract-In this paper we proposed optimal solution for finding single source & shortest path for N vertices.Dijkstra's algorithm wastes a lot of time and storage space in applications. This paper optimizes this algorithm using the path matrix. This algorithm can determine whether a path from source vertex to other vertices is either directly or through more than one intermediate vertex. In other words by using path matrix it can test reach ability of source vertex to destination vertex in a graph.

***Keywords*-Dijkstra's Algorithm, Directed Graph, Shortest Path, Path matrix.**

.

Introduction-The single-source shortest path problem in which we have to find shortest paths from a source vertex *v* to all other vertices in the graph. There is a graph have given as an input which have its edges weighted. It will calculate the shortest weighted graph or the path of less distance. There may be two types of conditions that-

- Source node and final node is given.
- Source node is given but final node is not given.

There are different types of shortest path algorithm to find the shortest path of directed graph. The most efficient algorithm used to find the shortest path between source vertex and destination vertex is Dijkstra's algorithm but this algorithm has increased time complexity and space. This paper has proposed optimization of Dijkstra's algorithm to find the Single Source & Shortest path for N vertices using Path Matrix

Dijkstra's Algorithm

Dijkstra's algorithm finds the shortest path from the source vertex to each other vertex in the graph(G). This algorithm used for finding the shortest paths from a single node to a single destination node by stopping the algorithm once the shortest path to the destination node has been determined.

In this paper we implement Dijkstra's algorithm step by step with the help of example. A weighted simple graph with positive weights and graph has vertices $V_0, V_1, \ldots V_n$ is given as an input.

For $i \longleftarrow 1$ to n

Label $V_0 \longleftarrow 0$

Label $V_i \longleftarrow \infty$

$S \longleftarrow$ an empty set

while destination vertex $\notin S$

begin

u $\leftarrow$ a vertex of graph which is not already in S

label(u) $\leftarrow$ minimum

S $\leftarrow$ S ∪ {u} {add u in set S}

For all the Vertices V which is not already in S

do If label (u) + w(u ,v) < label(v)

then label (v) $\leftarrow$ label (u)+ w(u,v)

end

{ label of destination vertex is found which is shortest }

Let us solve this single source shortest path problem using Dijkstra's Algorithm. According to this algorithm the label Source vertex is zero and label of all other vertices are ∞ .

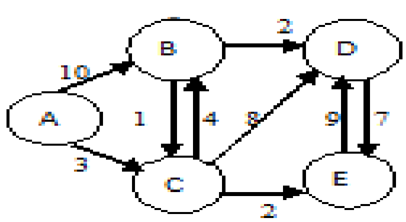

Fig.1

Vertex label	A	B	C	D	E
	0	∞	∞	∞	∞

Adjacent of vertex A=B & C

Dijkstra's algorithm gives us the following formula

dij= Minimum of {old label of j, old label of i + wij}

Where d is distance and w is weight.

d_{AB} = Minimum of {old label of B, old label of A +w_{AB} }

=minimum of {∞, o+10}

= minimum of {0,10 }

= 10

d_{AC} = Minimum of {old label of C, old label of A +W_{AC}}

= minimum of {∞,0+3}

= minimum of $\{\infty,3\}$

= 3

Vertex label

A	B	C	D	E
0	10	3	∞	∞

Now lowest label vertex from vertices B ,C,D, E is C.So we have processed the vertex C.

d_{CB}= Minimum of {old label of B, old label of C+W_{CB}}

= minimum of {10,3+4}

= minimum of {10,7}

= 7

d_{CD}= Minimum of {old label of D, old label of C +W_{CD}}

= minimum of $\{\infty,3+8\}$

= minimum of $\{\infty,11\}$

= 11

d_{CE}= Minimum of {old label of E, old label of C+W_{CE}}

= minimum of $\{\infty,3+2\}$

= minimum of $\{\infty,5\}$

= 5

Vertex label

A	B	C	D	E
0	7	3	11	5

Now old vertex with minimum label among B,D,E is E. So we further processing with vertex E.

d_{ED}= Minimum of {old label of D , old label of E+W_{ED}}

= minimum of {11,5+9}

= minimum of {11,14}

= 11

Vertex label

A	B	C	D	E
0	7	3	11	5

Now old vertex with minimum label among B,D is B. So we have processed the vertex B.

d_{BC}= Minimum of {old label of C, old label of B+W_{BC}}

= minimum of {3,7+9}

= minimum of {3,16}

= 3

d_{BD}= Minimum of {old label of D, old label of B+W_{BD}}

= minimum of {11,7+2}

= minimum of {11,9}

= 9

Vertex label

A	B	C	D	E
0	7	3	9	5

Now we have only vertex D.So we further processing with vertex D.

d_{DE}= Minimum of {old label of E, old label of D+W_{DE}}

= minimum of {5,11+7}

= minimum of {5,18}

= 5

Vertex label

A	B	C	D	E
0	7	3	9	5

Now Solution Graph is given below-

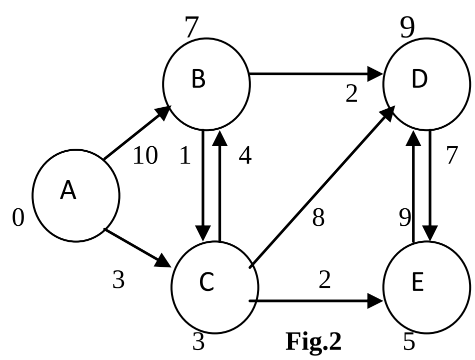

Fig.2

Simplification of Dijkstra's Algorithm to find single shortest path using Path Matrix

In this simplification we compute path matrix to finding single source shortest path from source vertex to destination vertex. This algorithm can determine a path form source vertex to destination vertex.

Path matrix-

Path matrix of any graph can be computed form its adjacent matrix A. This computation involves the computations of A1, A2, and A3.....An. To compute path matrix we performed union operation.

Input- A weighted simple graph with all positive weights.

Output- A shortest path from source vertex to destination vertex.

Let's find shortest path (single source) by computing path matrix in few steps-

Step 1- Let initialize the value of Source Vertex is 0 and all other Vertices value are ∞ and put this values in path matrix.

$$\Rightarrow \quad V0 \leftarrow 0$$

$$\Rightarrow \quad Vi \leftarrow \infty$$

Step 2-Put the distance value of adjacent vertices of source vertex as a weighted value of adjacent vertices in path matrix.

Step 3- To continue with a vertex which has lowest weight and find adjacent vertices of this lowest weighted vertex. Add this lowest weight with adjacent vertices distance from lowest weighted vertex. Placed this value as temporary weight of adjacent vertices. If adjacent vertex has already weight then we compare this weight with new temporary weight which weight is lowest placed at adjacent vertex as a new temporary weight of vertex.

$$\Rightarrow \quad \text{If label } (u) + w(u, v) < \text{label}(v)$$

$$\Rightarrow \quad \text{then label } (v) \leftarrow \text{label } (u) + w(u,v)$$

Step 4- Repeat Step 3 until all vertices are included and get the shortest path from source vertex to destination vertex as a solution.

Let start to implement this few steps to find single source shortest path by creating path matrix for a graph which is given below-

Problem is to find single source shortest path from source vertex A to destination vertex D.

\

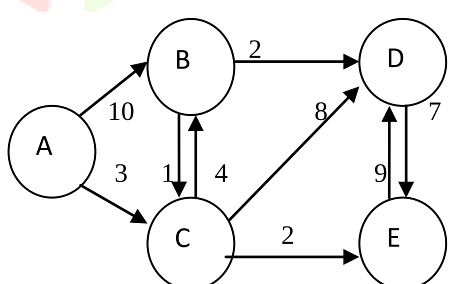

Fig.3

Path matrix is-

A	B	C	D	E	Visited Vertices
0	∞	∞	∞	∞	A
0	10	3	∞	∞	A,C
0	7	3	11	5	A,C,E
0	7	3	11	5	A,C,E,B
0	7	7	9	5	A,C,E,B,D

Solution – The single source shortest path from source vertex A to destination vertex is 9.

Conclusion

In this paper, we simplified Dijkstra's algorithm by using path matrix to find shortest path from source vertex to destination vertex. Path matrix reduces Iterations of Dijkstra's algorithm. Path matrix is a matrix which contains distance value from source vertex to other vertex. This simplification also can increase performance to find single source shortest path with few steps. Simplified dijkstra's algorithm reduces time complexity because traditional version of dijkstra's algorithm is more time taking.

References

[1] Anjali Jain ,U.Datta,Neelam.Joshi "Implemented Modification in Dijkstra's Algorithm to Find the Shortest Path for 'N' Nodes with Constraint" (Deptt. Of Computer Science &Engg. MPCT Gwalior) International Journal of Scientific Engineering and Applied Science (IJSEAS) – Volume-2, Issue 2, February 2016 ISSN: 2395-3470

[2] Charika Jain "Optimization of Dijkstra'sAlgorithm " (Amity School of Engg&Tech,Amity University Rajasthan) International Journal on Recent and Innovation Trends in Computing and Communication ISSN 2321 – 8169

[3] Arjun RK, Pooja Reddy, Sharma, M. Yamuna "Research on the optimization of dijkstra's algorithm and its applications" (School of Computer Science, Vellore Institute of Technology , Vellore, Tamil Nadu, (India))

[4] KairanbayMagzhan, Hajar Mat Jani "A Review And Evaluations Of Shortest Path Algorithms" (ISSN 2277-8616)

[5] Y. Cao, The Shortest path algorithm in data structures. Yibin University, vol. 6, 2007,pp.82 -84.