****

**CSE246: Algorithm**

**Sec-3**

**Group: 08**

**Project Report –**

**A Comparative Study on the Finding Shortest Route Algorithms**

**Submitted to-**

Md Mostofa Kamal Rasel

PHD, Assistant Professor

Department of Computer Science & Engineering

East West University

**Submitted by:**

(2019-2-60-051) Orpa Numan

(2019-2-60-040) Md Saiful

**Submission Date: 20/09/2021**

**A Comparative Study on the Finding Shortest Route Algorithms**

**Introduction**

A special person who is a Journalist. As their work is to cover the news of specified area. He generally stays in the capital Dhaka. But because of his job he sometimes has to stay out of his home also. So, when any call arrives, he has to reach that place to cover the news as soon as possible. So, he always needs to find the shortest route to reach faster. But his current location and destination is not specified. As it is not specified so, he need to know all possible shortest route in that city or out of the city so that he can cover the news.

Suppose, the journalist generally stays in Dhanmondi and he covers the news of capital but sometimes he needs to go out of his area to cover some special type of news. As, he has a special type of application so, his director has given him a job to inform other journalist the shortest route to go one place to another.

The capital consists of E bidirectional roads, each linking two of the V intersections, conveniently numbered from 1 to V. Each of the next E roads contains weight W (1 ≤ W ≤ 1000).A journalist moves from Source S to destination D where, Source and destination is not equal(S≠D). So, 1≤S, D ≤V. Also, the journalist Q times queries to inform other journalists the shortest route from source to destination. Where Q constrains of 1≤Q≤. E, V constrains of 1≤E, V≤500.

**Applications**

Some real-life applications to find shortest path,

* **Fire service:** There work is to reach at destination as soon as possible to save the life. So, there are many stations of fire service office all around the country. When a call arrives at main station, they transfer the call to nearest possible fire service office using shortest path algorithm.
* **Digital mapping services of Google map:** When we need to go from one place to another then we need the shortest path to reach destination as fast as possible. As there are many routes from source to destination, Google map shows us the minimum distance. So, they used shortest path finding algorithm to give the proper route.
* **Robotic Path:** In current world robots and drones has been used in various purposes. The drones/robots which are automated and are used to deliver the packages to a specific location or used for a task are loaded with this algorithm module so that when the source and destination is known, the robot/drone moves in the ordered direction by following the shortest path to keep delivering the package in a minimum amount of time.
* **Distance-vector routing protocol:** A version of shortest path finding algorithm is used in the distance-vector routing protocol. This protocol decides how to route packets of data on a network. The distance equation to decide weights in the network is the number of routers a certain path must go through to reach its destination.

**Algorithms**

In this problem we can see that here each road has certain weights. So, BFS can’t be used here as it only works for unweighted graphs. So, at first, we try to solve this problem using Dijkstra algorithm. But we faced some pros and cons to solved it. Then we tried to solve it using Bellman Ford algorithm but also here we face some pros and cons during the solution. Then finally we tried to solve it using Floyd Warshall algorithm. Also, here are some pros and cons but so far this is the best solution for our problem.

Let’s see how each algorithm works,

**Dijkstra:** Dijkstra's algorithm makes use of weights of the edges for finding the path that minimizes the total distance (weight) among the source node and all other nodes. This algorithm is also known as the single-source shortest path algorithm. It is important to note that Dijkstra’s algorithm is only applicable when all weights are positive because, during the execution, the weights of the edges are added to find the shortest path. Though, we are only considering the positive weights so it is not the actual problem. But the main problem arises when there are multiple sources but Dijkstra only works for single source at a time. If we try to solve it using Dijkstra then we need to run this Queries(Q) times then the complexity becomes much higher.

Time Complexity:

Single Query

* O((V+E)\*logE) if we use a min heap/priority queue
* O((V+E)\*logV) if we use a Fibonacci heap

Memory Complexity:

* O(V+E)

Mutiple query

* O(Q(V+E)\*logE)) if we use a min heap/priority queue
* O(Q(V+E)\*logV)) if we use a Fibonacci heap

Memory Complexity:

* O(V+E)

**Bellman Ford:** Bellman Ford algorithm also makes use of weights of the edges for finding the path that minimizes the total distance (weight) among the source node and all other nodes. It is similar to Dijkstra but it can work with graphs in which edges can have negative weights. But we are not concerned about negative edges. It only works as single source algorithm but the main problem arises when we try to solve our problem. Then we have to run the algorithm Queries (Q) times and the complexity becomes much higher.

Complexity: Single source

* Bellman Ford runs V rounds of relaxation, V-1 rounds for finding the shortest paths overall O(VE)

Memory Complexity:

* O(V+E) for adjacency matrix

Multiple Source

* Q times queries, Bellman Ford runs V rounds of relaxation, V-1 rounds for finding the shortest paths overall O(QVE).

Memory Complexity:

* O(V+E) for adjacency matrix

**Floyd Warshall:** Floyd–Warshall algorithm is an algorithm for finding shortest paths in a directed and undirected weighted graph with positive or negative edge weights. A single execution of the algorithm will find the lengths of shortest route between all pairs of nodes. The complexity of this algorithm is much higher than Bellman Ford and Dijksta. But Here we need to run the algorithm once only as it finds out the shortest route between all pair of nodes. So, with respect to other algorithm the complexity remains much lower.

Complexity: Single query

Time Complexity: O()

Memory Complexity: O()

Multiple Query

Time Complexity: O()

Memory Complexity: O()

**Complexity comparison table for inputs,**

|  |  |  |  |
| --- | --- | --- | --- |
| Algorithm | Complexity | Single Query Time | Multiple Query Time |
| Input | Worst case | V=500, E=500, Q=1 | V=500, E=500, Q=1000000 |
| Dijkstra | (V+ElogV) | .00005s | 50s |
| Bellman Ford | (VE) | 0.0025s | 2500s |
| Floyd Warshall | () | 1.25s | 1.25s |

**Conclusion:** By observing all the algorithm of finding shortest route we saw that for single query Dijkstra is better option and for multiple source Floyd Warshall is better option. But in our problem, we are working for multiple query so, using Floyd Warshall we got the better result with minimum complexity with respect to other algorithms.