A Fast and Reliable Dijkstra Algorithm for Online Shortest Path

Ms. Manas Kumar Satapathy, Mr. Sandip Kumar Bala

Department of Master in Computer Application, College Of Engineering Bhubaneswar, Odisha, INDIA.

ABSTRACT

There are several contexts in which the shortest path problem arises. Dijkstra's algorithm is a well-known shortest path algorithm. The ideas of network analysis with traffic problems are acknowledged in this research.A city's traffic situation can fluctuate at times, and a lot of requests typically come in that need to be resolved very away. The different issues with the current modified Dijkstra's shortest path methods are taken into consideration when developing the algorithm. Multiple parameters were incorporated into the MDSP algorithm in place of a single parameter to determine the valid shortest path for road networks. When the suggested method is compared to other existing algorithms, it becomes clear that the proposed algorithm is more accurate.

Key Words: MDSP, GPS, Geographic information systems, MDSP

1. INTRODUCTION

Shortest path computation is a crucial component of contemporary automobile navigation systems and has been thoroughly researched. This study makes use of the shortest path algorithm to assist in illustrating the most efficient route between two points. The problem of finding the shortest way given traffic condition is used to compute the shortest path and alternate path. It also provides alternative paths and the amount of traffic.

This is crucial to navigation systems because it facilitates rational decision-making and time-saving choices . Dijkstra's Algorithm is used to solve the shortest path issue of a graph with non-negative edge costs, yielding the shortest path tree. The majority of network connection protocols, including routing protocols, use this technique. The procedure begins by calculating the costs of the shortest path between a particular vertex in the graph and a target vertex; after the shortest path to the destination vertex has been identified, the algorithm stops .

The increasing popularity of GPS and location-based services has rekindled interest in creating a shortest path algorithm that is both scalable and extremely quick in order to provide users with a reliable route across road networks.

Finding the shortest route or separation between two points is among the most essential and significant major issues with road networks.

Many people commonly run into numerous issues when organizing travels in their own cars. In the recent past, numerous programs have been created to solve problems by determining the most effective path for road networks. Previous research indicates that a number of shortest path algorithms were created to determine the appropriate path for road networks. However, the issue persists.

In order to give travelers using road networks a better option, the research's goal is to suggest a new shortest path algorithm.

Geographic information systems (GIS) technology has advanced to the point that it can now be used to dispatch emergency vehicles, such as fire trucks and ambulances, and identify the quickest route. Since finding the quickest path is a difficult task and a link on a real road network in a city tends to have varying amounts of congestion throughout different times of the day .

Therefore, it is only possible to identify the fastest route in real time. There are situations where figuring out the quickest path only takes a few seconds. Furthermore, because numerous applications are involved in determining the shortest path over the road networks, determining the shortest paths on a big network can be computationally highly challenging when huge road networks are engaged in an application.

2. RELATED WORK& BACKGROUND

Because of the self-nature of the interactions, providing navigational aid to such a client type introduces additional obstacles not seen by guiding systems . All Navigation Indoor Models that offer direction in unfamiliar indoor situations include these algorithms. Path planning for a "intelligent map" makes use of Dijkstra's shortest path methods, which are based on a novel data structure called a "cactus tree" that displays the links between various items that depict an interior environment. Since the current "positioning and tracking" systems cannot provide the right position information that is primarily required by this sort of application, it is necessary to build an application for the visually impaired, as this research has found. We observed that the nature of transfer requires additional expenses.

This is the best-path problem for public transit networks, going from one edge to its neighboring edge . To Keep the dispersed transfer-related data in lists of indirect neighboring edges; this creates a space-saving storage structure. As a result, it creates a novel shortest path technique to address the transit issue of the data model and the storage of complicated network graphs.

As demonstrated by algorithm analysis, we provide a prior for a simple graph that depends on the Dijkstra's algorithm in terms of both space and time. Due to the daily growth in traffic, the intricate road network must perform a non-trivial operation to discover a better path from one site to another [7]. Numerous search techniques have been put out to address the Dijkstra's algorithm is the most well-known solution to the shortest path problem. In this work, We investigate the dependence on a few key cities using both heuristic and ignorant search techniques [8]. Routing algorithms are effectively utilized to reduce trip distance and transportation expenses.

The suggested model is evaluated using a sample dataset, and various working and traffic scenarios are simulated. The single source shortest path problem is currently solved most frequently using the Dijkstra's shortest path algorithm. The running time for determining a path between two vertices in graph figure 1 G(V, E), where V is the set of vertices and E is the set of edges, varies depending on the data structure that is utilized. Although there are several data structures that might marginally increase the time complexity—like the Fibonacci heap, which can purchase time complexity of O(V*log(V))—this project employs binary heap to perform Dijkstra's method.

3. EXPERIMENTAL PROCESS

A. Proposed method

The calculation of the shortest path or distance between two sites is a fundamental and significant topic in road networks. Many people usually run into a lot of issues while organizing their own car travel. In recent times, numerous programs have been created to resolve issues by determining the most effective path for road networks.

Previous research indicates that a number of shortest path algorithms were created to determine the best route for road networks. However, the issue persists.

Therefore, it is necessary to provide a new shortest path algorithm to give drivers using the road networks a better option .

Geographic information systems (GIS) technology has advanced to the point that it can now be used to dispatch emergency vehicles, such as fire trucks or ambulances, and identify the quickest route.

Since the shortest path is not always obvious and a link on a real road network in a city may have varying amounts of congestion at different times of the day.

Therefore, it is only possible to identify the fastest route in real time. There are situations where figuring out the quickest path only takes a few seconds. Furthermore, because numerous applications are involved in determining the shortest path over the road networks, determining the shortest paths on a big network can be computationally highly challenging when large road networks are included in an application .

Many shortest path algorithms, including Dijkstra's algorithm, the Floyd Warshall algorithm, the Bellman-Ford algorithm, a search algorithm, and Johnson's algorithm, were created and published in earlier literature.

The shortest path methods that are now in use have been thoroughly examined. Ultimately, it was found that the best method for determining the shortest path in road networks is Dijkstra's shortest path algorithm. However, several changes must be made to the current Dijkstra's shortest path algorithm in order to increase performance, identify a legitimate shortest path, and lower computing complexity. For this reason, the Modified Dijkstra's Shortest Path algorithm (MDSP), a brand-new algorithm, is put forth.

The Modified Dijkstra's Shortest Path algorithm (MDSP) is a suggested algorithm. Rather than relying only on one parameter, this system looked at a number of them to determine the legitimate shortest path. By calculating the MDSP algorithm's nodes and time complexity, its efficiency is evaluated in terms of the shortest path. The first for loop has an O(V) run time. The heap's Extract_Min is logV for each iteration of the while loop. The inner for loop has an O(E) execution time since it iterates across each node that is near to the current node. As a result, this algorithm's temporal complexity is O ((V + E) *log(V) = O (E* log(V)). This algorithm's accuracy is amply demonstrated in [5]. The implemented algorithm's running time will become longer and longer as a graph's node count rises. A city's road network typically contains more than 10^4 nodes. The fast-shortest path algorithm gains favor. Mohring et al. examined the various Dijkstra's shortest path algorithms that are currently in use.

They discovered that using the Dijkstra's algorithm as it is now.

4. CONCLUSION

The shortest path problem for road networks is the subject of extensive research. This research suggested a novel shortest path algorithm called MDSP, which incorporates numerous characteristics. The numerous issues with the current modified Dijkstra's shortest path methods are taken into consideration when developing the algorithm. To identify the valid shortest path for road networks, this MDSP technique used numerous parameters instead of only one. The MDSP algorithm results demonstrate that the suggested algorithm effectively determines the road network's shortest path with the least amount of computational complexity.

5. REFERENCE

1. Brunel, E., Delling, D., Gemsa, A., and Wagner, D., “Space-efficient SHARC-routing”, 9th International Symposium on Experimental Algorithms (SEA). 2012, 47-5858.

2. Bauer, R. and Delling, D. 2009., “SHARC: Fast and robust unidirectional routing”, ACM Journal of Experimental Algorithmics 14. Announced at ALENEX 2008. SSRG International Journal of Computer Science and Engineering (SSRG – IJCSE) – Volume 5 Issue 12 – December 2018 ISSN: 2348 – 8387 http://www.internationaljournalssrg.org Page 27

3. Bauer, R., Delling, D., Sanders, P., Schieferdecker, D., Schultes, D., and Wagner, D.," Combining hierarchical and goal-directed speed-up techniques for Dijkstra's algorithm", ACM Journal of Experimental Algorithmics, 2010, pp 34-42.

4. Thomas H. Cormen, Charles E. Lieserson, Ronald L. Rivest, Clifford Stein, "Introduction to Algorithms", Prentice Hall of India, 2009.

5. Anany Levitin, "Introduction to the design & analysis of algorithms", Pearson Education, Second Edition, 2009.

6. D.Delling, A. V. Goldberg, T. Pajor, and R. F. Werneck. Customizable route planning. In SEA, pages 376-387, 2011.

7. D.Delling, A. V. Goldberg, and R. F. Werneck," Hub label compression", SEA, pages 18-29, 2013.