

Navigating Airports Using Graph Algorithms

Data Structures & Algorithms

CSL2020

Instructor : Dip Sankar Banerjee Sir

Mentor : – Jainan Nareshkumar Tandel

BINAY SUMAN

B22CS019

Problem Statement

1

Finding the shortest distance between airports in a network of routes. Utilizing datasets containing airports and route information. Implementing Dijkstra's, Floyd Warshall's, and Bellman-Ford's algorithms. And also comparing their outputs.

- Efficient route planning is crucial for fast travel and passengers satisfaction in aviation.
- Comparing the outputs of Dijkstra's, Bellman-Ford, and Floyd-Warshall's algorithms using accurate data allows for a comprehensive evaluation of their performance and suitability for different scenarios, leading to informed decision-making in route selection.

Existing Situation

Some Research papers have been published in this Regard.

According to GeeksforGeeks,

- Bellman-Ford is the faster Algorithm when the graph is unweighted.

As per “medium.com”

- The Bellman-Ford algorithm is more versatile .
- Dijkstra’s Algorithm is more efficient .

Overall, It is seen as not easy to answer, so which is the better algorithm? It is a common perception that Dijkstra is more efficient and Bellman is better for handling negative cycles, and both of these are better than the Floyd Algorithm.

Limitations of Algorithms

Dijkstra's

- Limited to graphs with non-negative edge weights.
- Inefficient for graphs with negative edge weights or cycles.
- Requires a priority queue for efficient implementation, which can be memory-intensive and challenging to implement in certain scenarios.

Bellman-Ford

- It may need help finding the shortest path in negative-weight cycles.

Floyd Warshall

- Memory-intensive nature due to the need to store the distance matrix.

Implementation

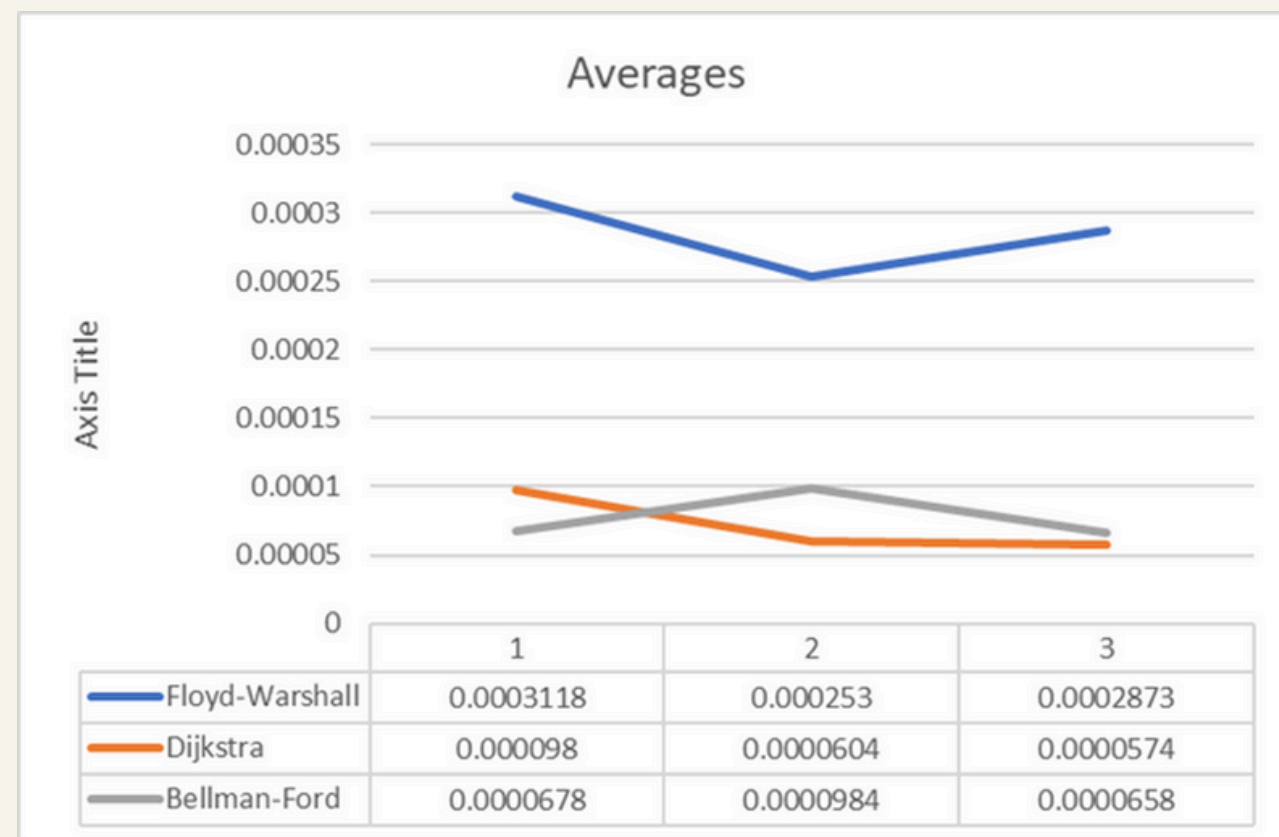
4

Github Repo Link: <https://github.com/binay4/DSPAproject>

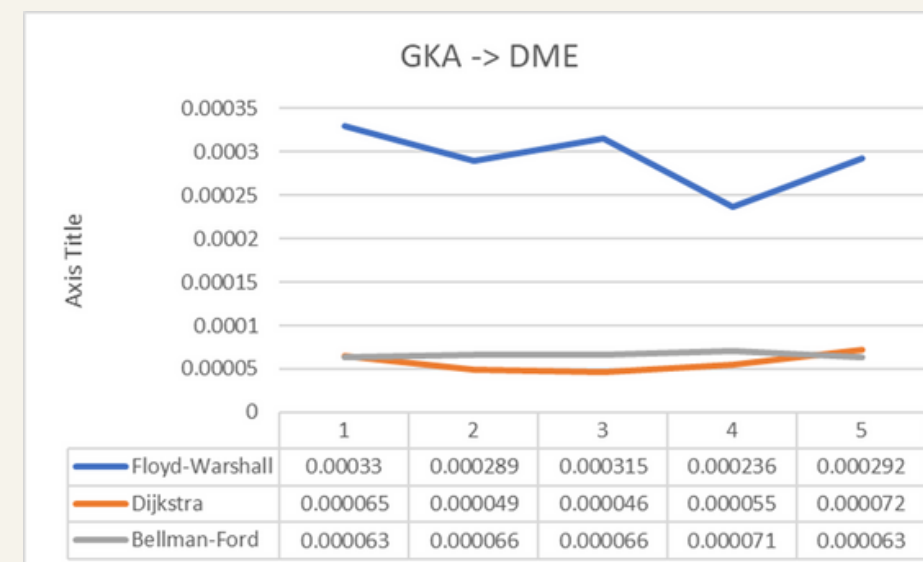
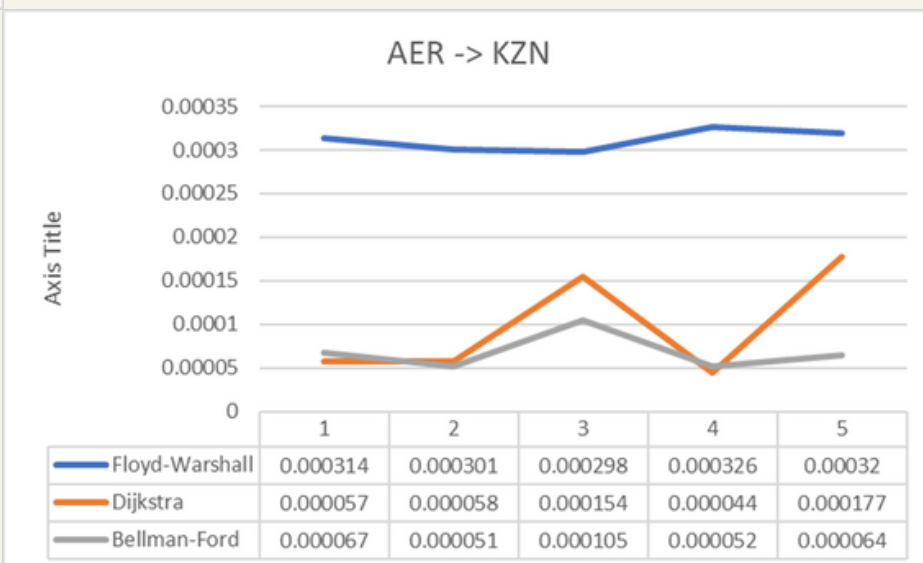
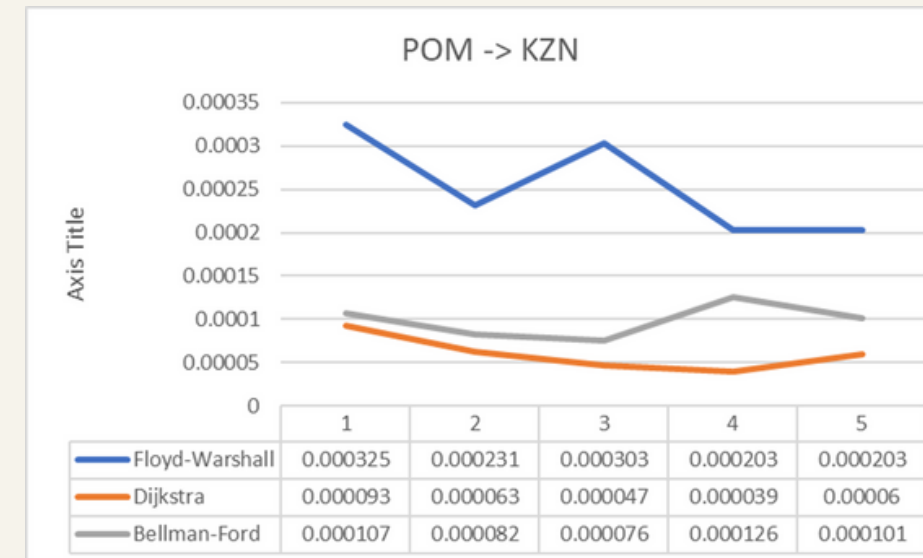
Data structures: Used a graph data structure to represent the network of airports and routes and adjacency lists and matrices to store the connections between airports.

Algorithms: Selected well-suited and common algorithms to handle the problem. Used both types of algorithms, to find the shortest path between a single source and all other airports, used Dijkstra and to find the shortest path between all pairs of vertices, used Floyd-Warshall.

Time complexity:



Results & Analysis



- In terms of time complexity, Dijkstra's algorithm and Bellman-ford seems to be comparatively same and efficient compared to Floyd-Warshall.
- Floyd-Warshall consistently performs slower than Dijkstra and Bellman-Ford algorithms in the given scenarios.

	Dijkstra's	Bellman Ford	Floyd Marshall
Time Complexity	$O(n^2)$	$O(ne)$	$O(n^3)$

- n -> number of vertices
- e -> number of connections

Conclusion

7

Finally, to conclude our implementation of all three algorithms, Dijkstra, Bellman-Ford, and Floyd-Warshall, It is seen that both the Dijkstra and Bellman are very closely matched and outperform each other at several occasions, so it would be unfair to say that any one algorithm is faster than other. Still, the Floyd algorithm is the slowest among the three. This might have been because it uses three nested loops to iterate over all pairs of vertices. We have used any graph with negative cycles, which is when the bellman algorithm is asit to shine, so in comparison of time complexities, Dijkstra has a better consists at outperforming other algorithm any graph with negative cycles, which is when the bellman algorithm is asit to shine, so in comparison of Time complexities, Dijkstra has a better consists at outperforming other algorithms, and look at the point that no one cycles were considered It could be stated that Dijkstra is a super algorithm if negative cycles are not considered.

Summary:

By applying graph algorithms, the research optimised aircraft route planning. The research implemented and compared Dijkstra's, Bellman-Ford's, and Floyd-Warshall's algorithms to assess their performance and application. Using accurate data and thorough research, route selection improved client travel experiences. The project's findings provide the groundwork for aircraft route planning enhancements.



REFERENCES

8

Dijkstra's Algorithm

- [stackExchange](#)
- [Mastering](#)
- [/QCode](#)
- [stackoverflow](#)

Floyd Warshall Algorithm

- [stackoverflow](#)
- [Codesnippets](#)
- [pieces](#)
- [GitHub](#)

Bellman-Ford Algorithm

- [programiz](#)
 - [geeks for geeks](#)
- 

THANK YOU
