

Formulation of Congestion Metric & Different Routing Techniques, Indian Railways

By

Palkesh Bhidodiya

Under the guidance of

Prof. Narayan Rangaraj & Prof. Madhu Belur



Interdisciplinary Programme in
Industrial Engineering and Operations Research
Indian Institute of Technology, Bombay
Mumbai 400076

Outline

- Introduction
- Problem statement
- Objectives
- Review of Important Topics
- Experimentation and Results
- Conclusion
- Future work
- References

Introduction

- Indian Railways is one of the largest railway networks in the world with a total of 1,26,366 track kilometres and operates 13,169 passenger trains and 8,479 freight trains on a daily basis [1].
- Both the passenger and freight trains share the same track resources on most of its routes.
- In the financial year 2021-2022, about 20 % of the revenue of Indian Railways was generated by passenger movement and 75.8 % from freight transport [3].

Introduction (contd.)

- India has a very high and growing demand for freight trains, hence expanding the capacity of current railway sections is necessary to meet this demand.
- Freight trains have a relatively lower priority on mixed-traffic rail networks and are scheduled during the time windows left over by passenger trains.

The Problem

Our Focus: Creating high-quality freight paths on a shared rail network.

Issues:

- When there is congestion due to planned and unplanned events, getting a good freight path will be challenging.

Objectives

- To analyze and quantify the congestion on the railway block sections.
- To identify available freight windows.
- To create freight paths on a shared rail network.
- To optimize the overall travel time of freight trains.
- To compute the shortest freight path and subsequent second & third shortest paths.

Review of Topics

Line Capacity

- Line capacity means the number of trains that can be run on a section in 24 hours.
- Line capacity is an important factor for planning freight trains on mixed traffic.

Scott's formula

$$\text{Line Capacity } C = 1440 / (T + t) \times E$$

Where T is the running time of the slowest train over the critical block section, t is the Block operation time and E is the Efficiency factor

Dijkstra's Algorithm

- Dijkstra's algorithm is an algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road maps[5]
- The algorithm makes the optimal choice at each step as it attempts to find the overall optimal way to solve the entire problem.
- Dijkstra's Algorithm works for both directed and undirected, positively weighted graphs.

Drawback :

- Dijkstra's algorithm cannot handle negative edges (negative weights).

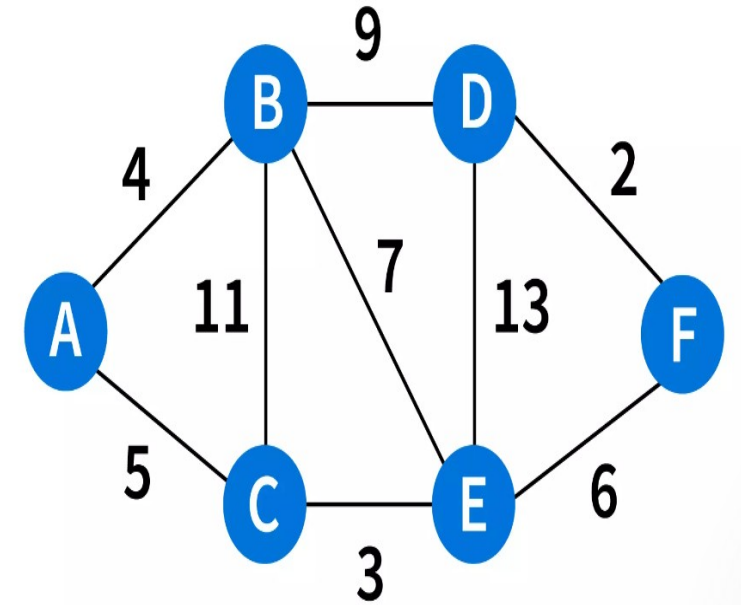


Fig.2 Example of network for finding the shortest path by Dijkstra's algorithm [5]

Yen's K Shortest Algorithms

- Yen's K shortest algorithms find K shortest path from origin to destination i.e. shortest path, 2nd shortest path...kth shortest path.
- By using Dijkstra, it finds the shortest path between 2 nodes. Then for the next shortest path, it prohibits the paths already considered as the shortest path & likewise.

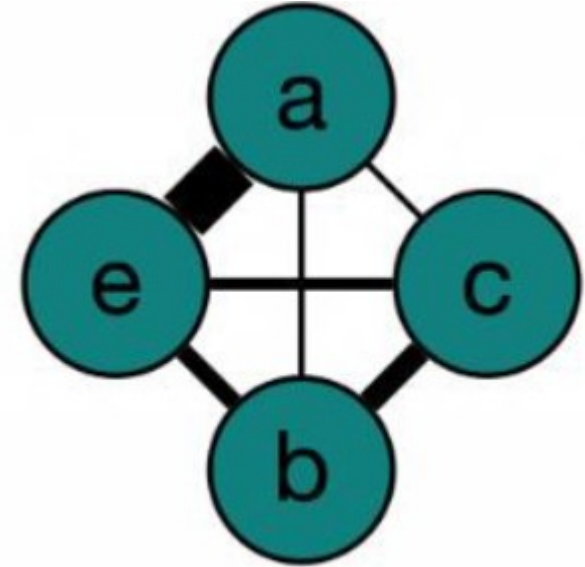


Fig. 3 Example of network for finding the shortest path by K shortest algorithm [8].

Alternate route

- Route choice is the process, in which a route planner has to choose a route from different alternatives. The level of the route is also a factor in the route selection process, in addition to the available routes[9].
- There may be cases where a planner has to plan which route to follow if one route is taking some amount of time and another alternate route is taking slightly less time but the distance is more. so, in such a situation, the planner has to decide whether to take that alternate path or not.
- Major construction, maintenance, and traffic incidents, these events are commonly termed planned and unplanned events.
- Each of these occurrences results in a decrease in the capacity of railway sections and, possibly, an increase in demand, thus necessitating the consideration of alternate route implementation.

Work on a similar idea of routing

- Google Maps supports its traffic views and faster route suggestions on two different types of information
- Real-time data sent by sensors and smartphones that report how fast cars are moving at that exact moment and factual data about the average time it takes to travel a specific section of road at specific times on specific days[6].
- Few BMW car models have a system called Real Time Traffic, in which Predictive Routing actively monitors traffic conditions along regularly travelled routes.
- Based on current traffic conditions and the traffic environment, alternative routes and departure times are automatically suggested [11].

Experimentation & Results

Dataset

- In this work, we focus on the six GQD routes of Indian Railways that connect New Delhi, Mumbai, Chennai, and Kolkata.
- These principal routes carry about 58% of freight traffic and 52% of passenger traffic and account for around 15% of the network, approximately 9100 route kilometres [2].
- A schematic diagram of these six routes can be seen in Fig. 4.
- In Dataset trains are clustered on day basis

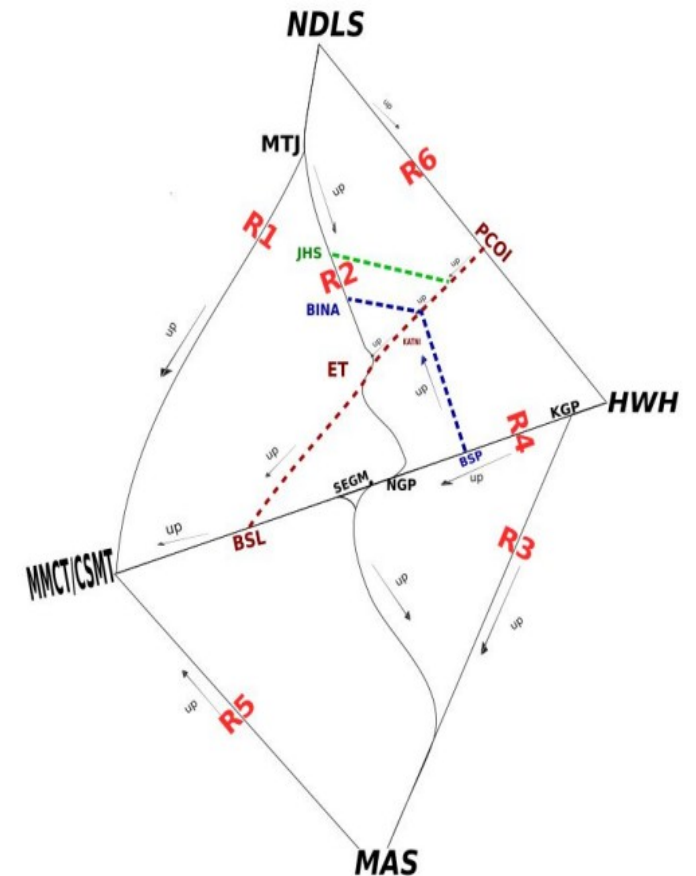


Fig.4 GQD routes of Indian Railways under consideration

Heatmap of occupancies on the Block section

- We can able to comment on how much occupancy a block section will face at a particular time period, the time window is user defined the user can give it 30 minutes, 1 hour or any other time period.
- Heatmap also help us in deciding whether some section/route is congested or not.
- Timetabling can also be done by visualizing heatmap and decision support. The planner can schedule trains based on track availability and historical demand pattern.

Freight Corridors

- Identifying freight windows (duration greater than 30 minutes) by extracting free time slots at block sections, time windows representing passenger train paths are not breached.
- There are some possible potential freight windows available on each section of the railway which can be used for freight train operations.
- In these available freight windows, no other train is scheduled.
- Benefit of using freight corridors is that the speed of freight trains can be increased and on its path, it'll face a few halts.

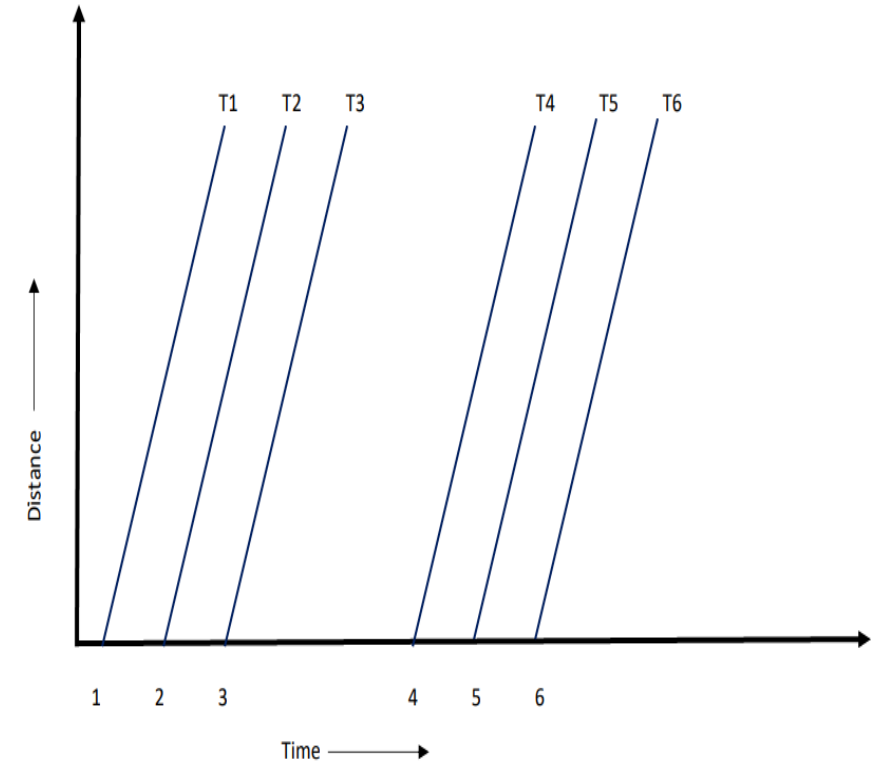


Fig.6 Distance and Time graph for Trains

Freight Corridors (contd.)

Algorithm-2 To Determine Freight Window of a block sections

Require: Input = Block section list, occupancy of block section

Occupancy list = List of occupancy at block section

N = length of occupancy list

Initialize empty list = []

while *i* is smaller than N **do**

if occupancy *i* is zero

 add time slot of *i* till continuously occupancy is zero in N

end while

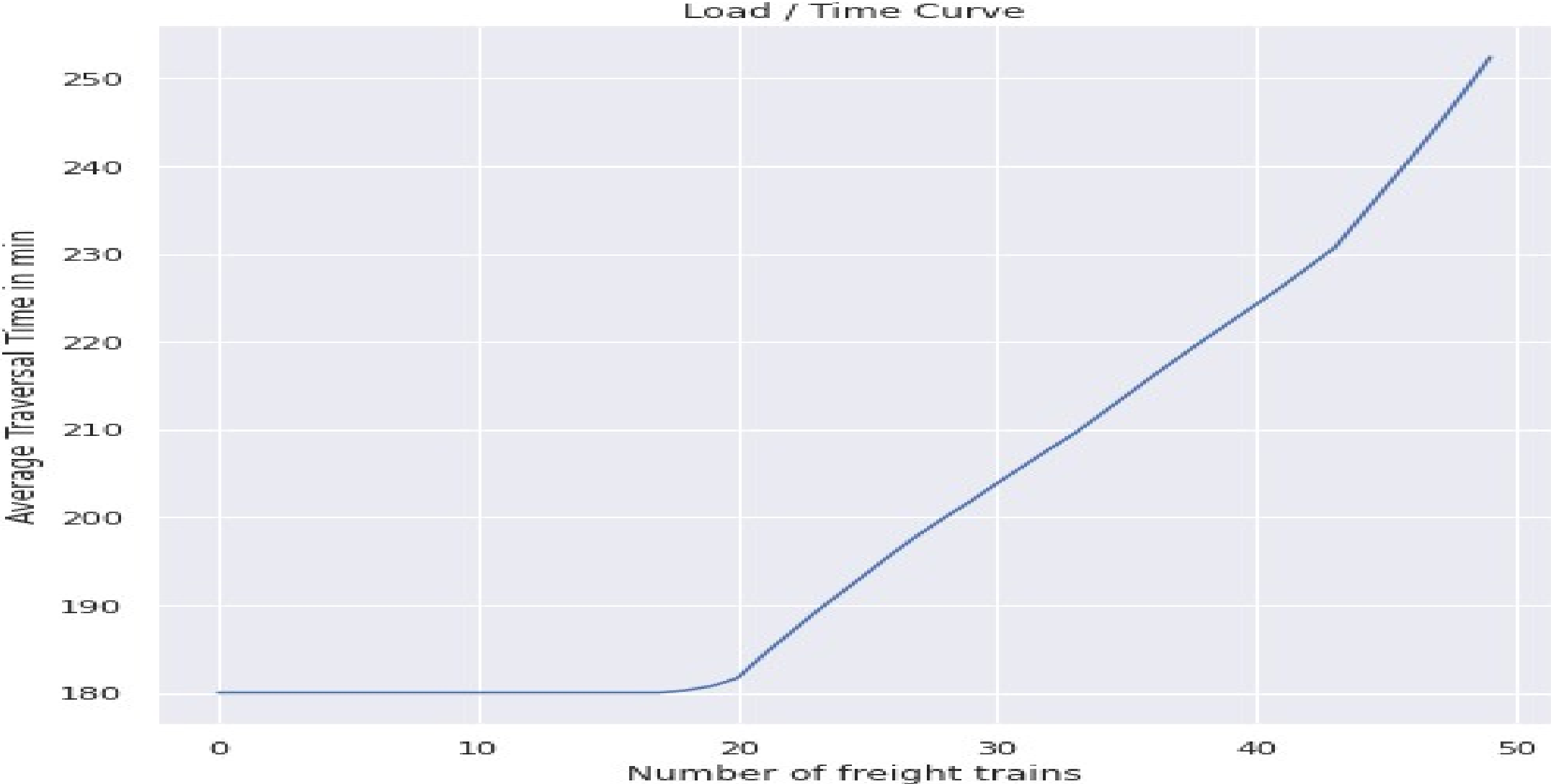
return list of freight windows

Freight Corridors (contd.)

Option	Start Time of corridor	End Time of corridor	Total Time
1st Window	0.30 am	3.00 am	2.30 hrs.
2nd Window	6.00 am	7.30 am	1.30 hrs.
3rd Window	8.30 am	10.30 am	2.00 hrs.
4th Window	1.30 pm	2.00 pm	0.30 hrs.
5th Window	4.00 pm	5.00 pm	1.00 hrs.
6th Window	7.00 pm	9.00 pm	2.00 hrs.

Table.1 Available Freight Windows at Balharshah

Load and Time curve



Scheduling of Freight Trains

- Scheduling is done in the identified freight windows and the number of trains to be fired is decided based on the length of the corridor with a safety headway.
- Scheduling is not done for the whole network in one pass, crew change is planned at intermediate stations where adequate infrastructure is available.
- After scheduling trains in between crew change points, then connect the whole path, to know the exact timing of reaching the destination or any certain station.

Total Time of Travel

Algorithm-3 Computing Total Time taken by freight train in freight path

Require: Input = Block section list, starting time, ending time and delay of train at block sections

N = Length of block sections in freight path

Initialize Temp = 0 and Total time = 0

for i in each N **do**

 add in temp by subtracting ending time of **i** and starting time of **i**

do for all the block sections

end **for**

for i in each N **do**

 compute Total time by adding temp of **i** and delay of **i**

end **for**

return Total time of freight train

Algorithm-3 Computing Total Time taken by freight train in
freight path

Total Delay Faced

Algorithm-4 Computing Total Delay of freight train in freight path

Require: Input = Block section list, starting and ending time of train at block sections

N = Length of block sections in freight path

Initialize delay=0

for i in each N **do**

if starting time of (i+1) is greater than ending time of i **do**

 add in delay by subtracting starting time of (i+1) and ending time of i

else do

 add delay by computing, 1440 subtract (ending time of i addition starting time of (i+1))

do for all the block sections

end for

return delay of the train

Algorithm-4 Computing Total Delay of freight train in freight path

Total Time and Delay Faced

S.No	Options form Each window	Start Time at Balharshah	End Time at Swegram	Total Time	Total Delay Faced
1	1 st	0.30 am	03.40 am	3.10 hrs	00.10 hrs.
2	2 nd	6.00 am	9.00 am	3.00 hrs.	00.00 hrs.
3	3 rd	8.30 am	11.50 am	3.00 hrs.	00.20 hrs.
4	4 th	1.30 pm	6.00 pm	4.30 hrs.	1.30 hrs.
5	5 th	4.00 pm	7.00 pm	3.00 hrs.	00.00 hrs.
6	6 th	7.00 pm	10.00 pm	3.00 hrs.	00.00 hrs.

Table.2 Details for Freight Path Balharshah to Sewagram

Conclusion

- In this study, a heuristic-based methodology is suggested for creating freight paths along the six GQD routes of the Indian Railway without interfering with the scheduled passenger paths.
- The main tool used for train path generation is Python Code.
- In terms of the number of freight paths generated and scalability to a large network, the proposed methodology is found to perform well.
- We also computed the freight path for network Balharshah to Sewagram, from available freight windows.

Future Work

- Implement shortest path algorithms to get the freight path which will utilize the capacity and resources better.
- Devise a Methodology to predict the routing of freight trains, based on real-time congestion and factual data about the average time it takes to travel a specific section at specific times on specific days.
- Identify potential windows where few passenger trains are planned.
- Cumulate the congestion on each path to determine the feasible routes set.
- Identify the alternate routes by using algorithms like the K-shortest path algorithm.

References

- [1] Ministry of Railways, Government of India, “Indian Railways Year Book 2019-20”. Available at: [yearbook201920englishfinalweb6670607.pdf \(indiarailinfo.com\)](http://yearbook201920englishfinalweb6670607.pdf)
- [2] Golden Quadrilateral & Golden Diagonal (GQ-GD), Journals of India, 2021. Available at: [Golden Quadrilateral & Golden Diagonal \(GQ-GD\) - JournalsOfIndia](http://GoldenQuadrilateral&GoldenDiagonal(GQ-GD)-JournalsOfIndia)
- [3] Distribution of Indian Railways revenue in financial year 2022, Statista, 2022. Available at: [India: railways revenue distribution by segment 2022 | Statista](http://India:railwaysrevenue distribution by segment 2022 | Statista)
- [4] Narayan Rangaraj, Sudarshan Pulapadi, Swapnesh Subramanian and Madhu N. Belur “Integrated timetabling for section and suburban railway operations”. Available at: [c19irt.pdf \(iitb.ac.in\)](http://c19irt.pdf(iitb.ac.in)) (Accessed: 13 October 2022).
- [5] Mehul Pandey “Dijkstra Algorithm Scaler Topics”, 2022. Available at: [Scaler Topics - Technopedia for Your Mastermind](http://ScalerTopics-TechnopediaforYourMastermind)

References (contd.)

- [6] Haria, Viddit, Yashvi Shah, Vanya Gangwar, Vansh Chandwaney, Tarang Jain, and Yash Dedhia. "The working of google maps, and the commercial usage of navigation systems." International Journal of Innovative Research in Technology 6, no. 5 (2019). Available at: [Academia.edu](https://www.academia.edu)
- [7] Bellman Ford Algorithm, From Wikipedia, the free encyclopedia. Available at: [Bellman-Ford algorithm - Wikipedia](https://en.wikipedia.org/wiki/Bellman-Ford_algorithm)(Accessed: 13 October 2022)
- [8] Mehul Gupta, Graph Analytics: Pathfinding algorithms using Neo4J, Medium Available at: [Graph Analytics: Pathfinding algorithms using Neo4J | by Mehul Gupta | Data Science in your pocket | Medium](https://medium.com/@mehulgupta1997/graph-analytics-pathfinding-algorithms-using-neo4j-by-mehul-gupta-data-science-in-your-pocket-medium) (Accessed: 14 October 2022)

References (contd.)

- [9] Alivand, Majid, Hartwig Hochmair, and Sivaramakrishnan Srinivasan. "Analyzing how travelers choose scenic routes using route choice models." *Computers, Environment and Urban Systems* 50 (2015): 41-52. Available at: [Analyzing how travelers choose scenic routes using route choice models - ScienceDirect](#) (Accessed: 15 October 2022).
- [10] Dijkstra Algorithm, From Wikipedia, the free encyclopedia. Available at: [Dijkstra's algorithm - Wikipedia](#) (Accessed: 13 October 2022).
- [11] BMW worldwide real-time traffic information services, eeneautomotive,2022 Available at: [- EENewsEurope ... \(eeneautomotive.com\)](#)

Thank You!