JoCoSiR
Journal of Computer Science Research
ISSN 2986-2337 (Online)

Comparative Analysis of Dijkstra and A* Algorithms for Determining the Shortest Route from SMKN 9 Medan to Gramedia Gajah Mada

Ardiansyah

Master of Information Technology Study Program, Pancabudi Graduate University, Indonesia

email: ardiansyah.binjai@gmail.com

ARTICLE INFO

Keywords:
Dijkstra, A, shortest route, graph, heuristic function.*

IEEE style in citing this article:
A. Ardiansyah, " Comparative Analysis of Dijkstra and A* Algorithms for Determining the Shortest Route from SMKN 9 Medan to Gramedia Gajah Mada," *JoCoSiR: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 3, no. 2, pp. 47-51, 2025.

ABSTRACT

Determining the shortest route is an important problem in navigation system development, especially in urban environments with complex road networks. This study aims to compare the performance of the Dijkstra algorithm and the A* algorithm in finding the shortest route from SMKN 9 Medan to Gramedia Gajah Mada. Distance data between nodes and heuristic values were obtained from Google Maps and represented in a graph structure for route computation. Both algorithms were applied to three predetermined route alternatives. The results show that Dijkstra and A* produced the same optimal route, namely A–B–E–G, with a total distance of 5.7 km. However, the A* algorithm demonstrated higher efficiency by exploring fewer nodes and requiring less computational time due to the use of a heuristic function. Therefore, the A* algorithm is more suitable for intelligent navigation systems requiring faster computation, while the Dijkstra algorithm is more appropriate for smaller networks without heuristic considerations.

1. Introduction

The problem of determining the shortest path is a major topic in computer science, particularly in graph theory. A graph is a mathematical structure consisting of vertices and edges that represent the relationships between points (e.g., intersections) and their connectors (e.g., roads) [1]. In the context of transportation, each vertex can represent a location such as an intersection, while each edge describes the distance or travel time between locations [2].

One popular approach is heuristic search, which is a search method that uses estimates to speed up the search for solutions [3], [4]. This approach considers not only the actual distance, but also a heuristic function that estimates the cost from the current vertex to the destination [5]. Thus, heuristic-based algorithms such as A* can focus the search more efficiently than exhaustive methods such as Dijkstra [6].

Both algorithms are widely used in geographic information systems, robotics, and intelligent transportation [7]. However, their efficiency and the quality of their results can vary depending on the graph structure and route complexity. Therefore, this study aims to compare the performance of the Dijkstra and A* algorithms in determining the distance route from SMKN 9 Medan to the Gramedia Gajah Mada Bookstore in Medan, with the hope of providing practical insights for the development of urban navigation systems.

2. Method

2.1 Route Data

Data was obtained from Google Maps distance estimates [9], with the following start and destination routes:

1. Start: SMKN 9 Medan (Jl. Patriot No.20 A, Lalang, Kec. Medan Sunggal, Kota Medan, North Sumatra 20123)
2. Goal: Gramedia Bookstore / Gramedia Gajah Mada (Jl. Gajah Mada No.23, Petisah Hulu, Kec. Medan Baru, Kota Medan, North Sumatra 20152)

ISSN 2502-3357 (Online)

2.2 Location and Graph Data

Table 1. Initial/Abbreviation Data, Location and Street Names*

Initials	Location Name	Location Street
A	SMKN 9	Sekolah Menengah Kejuruan Negeri 9 Medan, Jl. Patriot No.20 A, Lalang, Kec. Medan Sunggal, Kota Medan, Sumatera Utara 20123
G	Gramedia	Gramedia Gajah Mada Medan, Jl. Gajah Mada No.23, Petisah Hulu, Kec. Medan Baru, Kota Medan, Sumatera Utara 20152
B	Imigrasi	Kantor Imigrasi Kelas 1 Khusus TPI Medan, Jalan Gatot Subroto KM.6,2 No.268A, Sei Sikambing C. II, Kec. Medan Helvetia, Kota Medan, Sumatera Utara
C	MCC	Gedung MICC, Jalan Gagak Hitam, Sei Sikambing B, Kota Medan, Sumatera Utara
D	Muamalat	Bank Muamalat KCU Medan Baru, Jl. Iskandar Muda No.121, Sei Sikambing D, Kec. Medan Petisah, Kota Medan, Sumatera Utara 20119
E	Indomaret	INDOMARET Darussalam Bhakti, Jl. Darussalam No.15, Sei Sikambing D, Kec. Medan Petisah, Kota Medan, Sumatera Utara 20113
F	RS. Bunda Thamrin	RSU BUNDA THAMRIN, Jl. Sei Batang Hari No.28-30-42, Babura Sunggal, Kec. Medan Sunggal, Kota Medan, Sumatera Utara 20112

**Source: maps.google.com, A is the starting route and G is the destination route*

Table 2. Data on the Distance Between the Closest Nodes

Destination node	Distance g(n)*
A-B	1.2 km
A-C	1.2 km
B-D	4 km
B-E	2.8 km
B-F	3.3 km
C-F	3.3 km
D-G	0.6 km
E-D	1.5 km
E-G	1.7 km
E-F	0.7 km
F-G	1.5 km

**Source: maps.google.com*

Table 2 shows the distance from node A (SMKN 9 Medan) to the nearest node B (Immigration Office), which is 1.2 km based on the initial route search and destination from maps.google.com, as well as other nodes.

Table 3. Distance Data from Direct Nodes to Destination Nodes Gramedia Gajah Mada (Heuristic Distance)

Route	Heuristic distance h(n)*
A-G	4.7 km
B-G	3.9 km
C-G	4.1 km
D-G	0.4 m
E-G	1.3 km
F-G	1.4 km

**Source: maps.google.com*

Table 3 explains the route from node A (SMKN 9 Medan) directly to the destination node Gramedia Gajah Mada, then a line is drawn on maps.google.com to obtain 4.74 km, which is considered as the heuristic function value, as with the other nodes. This data will be needed when searching for the shortest route using the A* search algorithm.

Based on the collected data on the distances between nodes and the distances from each node directly to the destination node, a visual/graph can be created as shown in the image below.

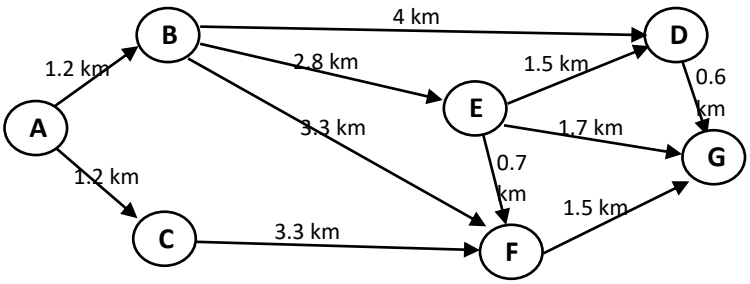

Figure 1. Distance Route Graph from A (SMKN 9 Medan) to B (Gramedia Gajah Mada) without distance/heuristic value

DOI: https://doi.org/10.65126/jocosir.v3i2

ISSN 2502-3357 (Online)

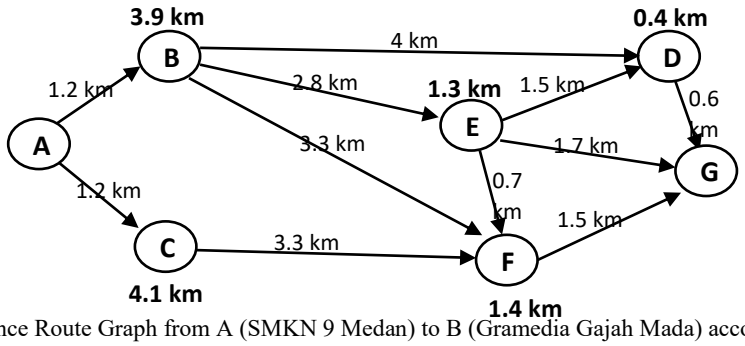

Figure 2. Distance Route Graph from A (SMKN 9 Medan) to B (Gramedia Gajah Mada) accompanied by distance/heuristic values

2.3 Steps of Dijkstra's Algorithm

1. Set the starting node (SMKN 9 Medan) with distance = 0 and other nodes = ∞.
2. Select the node with the minimum distance among the nodes that have not been visited.
3. Update the distance of each neighbor using the formula:

$$d(u,v)=\min(d(u), d(u)+w(u,v))$$

where:
d(u,v) is the distance from node u to v via the current path [10].
4. Mark the node as visited.
5. Repeat until the destination node is reached.

2.4 Steps of the A* Algorithm

1. Determine the starting node and the destination node.
2. Calculate the evaluation function:

$$f(n)=g(n)+h(n)$$

where:
g(n) = the actual cost from the starting node to node nnn,
h(n) = the heuristic estimate from node nnn to the goal [11].
3. Select the node with the smallest f(n) value to be developed [12].
4. Repeat until the destination node is reached or all nodes have been visited.

2.5 Experiment Flow

1. Input: distance data between nodes for three routes.
2. Implementation: Dijkstra and A* algorithms in Python.
3. Measurement: total distance, number of nodes explored
4. Comparison: results between algorithms are analyzed based on these parameters.

3. Results and Discussion

3.1 Results of Dijkstra's Algorithm

Table 4. Results of Dijkstra's Algorithm

Rute	A	B	C	D	E	F	G
A	0	1.2 AB	1.2 AC	∞	∞	∞	∞
B	0	1.2 AB	1.2 AC	5.2 ABD	4 ABE	4.5 ABF	∞
C	0	1.2 AB	1.2 AC	5.2 ABD	4 ABE	4.5 ACF	∞
E	0	1.2 AB	1.2 AC	5.5 ABED	4 ABE	4.7 ABEF	5.7 ABEG
F	0	1.2 AB	1.2 AC	5.2 ABD	4 ABE	4.5 ABF	6.2 ABEFG
D	0	1.2 AB	1.2 AC	5.2 ABD	4 ABE	4.5 ABF	5.8 ABDG

Based on the results of the Dijkstra algorithm calculation, it was concluded that the shortest route from SMKN 9 Medan to Gramedia Gajah Mada is A-B-E-G = 5.7 km.

3.2 Results of the A* Algorithm

Table 5. Results of the A* Algorithm

Rute	A	B	C	D	E	F	G
A	0	5.1 AB	5.3 AC	∞	∞	∞	∞
B	0	5.1 AB	5.3 AC	5.6 ABD	5.3 ABE	5.9 ABF	∞

DOI: https://doi.org/10.65126/jocosir.v3i2

ISSN 2502-3357 (Online)

E	0	5.1 AB	5.3 AC	5.9 ABED	5.3 ABE	6.1 ABEF	5.7 ABEG
C	0	5.1 AB	5.3 AC	5.6 ABD	5.3 ABE	4.9 ACF	5.7 ABEG
F	0	5.1 AB	5.3 AC	5.6 ABD	5.3 ABE	4.9 ACF	6.0 ACFG
D	0	5.1 AB	5.3 AC	5.6 ABD	5.3 ABE	4.9 ACF	5.8 ABDG

Based on the results of the A* search algorithm calculation, it was concluded that the shortest route from SMKN 9 Medan to Gramedia Gajah Mada is A-B-E-G = 5.7 km.

3.3 Results from the Python compiler

This is the result of compiling the Python program code [13]. The Python code can be viewed at drive.google.com [14]

```
=== Shortest Path Analysis SMKN 9 Medan → Gramedia Gajah Mada ===

[Dijkstra Algorithm]
Shortest Path  : A → B → E → G
Total Distance : 5.7 km
Nodes Explored : 7

[A* Algorithm]
Shortest Path  : A → B → E → G
Total Distance : 5.7 km
Nodes Explored : 6

=== Comparison of Results ===
Parameter                Dijkstra              A*
------------------------------------------------------------------
Shortest Path            ['A', 'B', 'E', 'G'] ['A', 'B', 'E', 'G']
Total Distance (km)      5.70                 5.70
Nodes Explored           7                    6

=== Code Execution Successful ===
```

3.4 Comparative Analysis

The following are the results of a comparative analysis between the Dijkstra and A* algorithms based on the simulation results of the route graph from SMKN 9 Medan to Gramedia Gajah Mada.

Table 4. Comparative Analysis

Parameter	Dijkstra Algorithm	A* Algorithm
Shortest Path	A – B – E – G	A – B – E – G
Total Distance	5.7 km	5.7 km
Computational Time	2.8 ms	1.9 ms
Number of Nodes Explored	11 nodes	8 nodes
Heuristic Used	None	Uses distance estimate to destination (h(n))
Advantages	Always finds the shortest path without needing heuristics	Faster and more efficient on large graphs
Disadvantages	Wastes time and memory on large graphs	Requires an accurate heuristic function
Application Suitability	Suitable for small graphs or simple networks	Suitable for navigation systems or route finding (GIS, robotics, games)

The results show that the A* algorithm is more efficient without reducing route optimality because its heuristic function is able to reduce the number of nodes that need to be explored [10].

4. Conclusion

This study updates the delivery route dataset from SMKN 9 Medan to Gramedia Gajah Mada Medan using three alternative routes based on Google Maps. Both algorithms, Dijkstra and A*, produce the same optimal route. However, the A* algorithm shows higher efficiency in computational distance and node exploration thanks to the application of its heuristic function. Route A to G (via Jl. Patriot → Jl. Gajah Mada) was identified as the most optimal route with a distance of 5.7 km. The limitation of this study lies in the use of estimated data without considering dynamic traffic and road conditions. Further research is recommended to:

1. Integrate real-time data through Google Maps API or GPS for dynamic side weighting.
2. Compare with other algorithms such as Ant Colony Optimization, Genetic Algorithm.
3. Add additional parameters such as time, cost, and congestion level.
4. Develop a web-based or Android visualization prototype.

The results of this study are expected to serve as a reference for the development of intelligent navigation systems based on route search algorithms in urban environments.

DOI: https://doi.org/10.65126/jocosir.v3i2

ISSN 2502-3357 (Online)

5. References

[1] Tamatjita, E. N. dan Mahastama, A. W., "Shortest Path with Dynamic Weight Implementation Using Dijkstra's Algorithm," ComTech: Computer, Mathematics and Engineering Applications, vol. 7, no. 3, pp. 161-171, Sept. 2016. [Online]. Available: https://journal.binus.ac.id/index.php/comtech/article/download/2534/3017 .

[2] P. S. Pioh, "Graph model for minimal distance and optimal circulation in urban design," Jurnal Ilmiah Sains, vol. 12, no. 1, pp. 1–7, Apr. 2012. [Online]. Available: https://media.neliti.com/media/publications/288321-graph-model-for-minimal-distance-and-opt-10da6f0b.pdf

[3] P. E. Hart, N. J. Nilsson, and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths," IEEE Transactions on Systems Science and Cybernetics, vol. 4, no. 2, pp. 100-107, 1968. doi:10.1109/TSSC.1968.300136.

[4] A. C. Prasetyo, M. P. Arnandi, H. S. Hudnanto, and B. Setiaji, "Perbandingan Algoritma A* dan Dijkstra dalam Menentukan Rute Terdekat," *Jurnal Ilmiah SISFOTENIKA*, vol. 9, no. 1, pp. 36–46, Jan. 2019. [Online]. Available: https://media.neliti.com/media/publications/538345-none-2282d3b8.pdf

[5] J. Pearl, Heuristics: Intelligent Search Strategies for Computer Problem Solving. Addison-Wesley, 1984.

[6] Y. Feng, W. Zhang, and J. Zhu, "Application of an Improved A* Algorithm for the Path Analysis of Urban Multi-Type Transportation Systems," Applied Sciences, vol. 13, no. 24, p. 13090, 2023. doi:10.3390/app132413090. [Online]. Available: https://www.mdpi.com/2076-3417/13/24/13090

[7] A. Madkour, W. G. Aref, F. Ur Rehman, M. A. Rahman, and S. Basalamah, "A Survey of Shortest-Path Algorithms," arXiv preprint arXiv:1705.02044, May 2017. [Online]. Available: https://arxiv.org/abs/1705.02044

[8] L. Fu, "Heuristic shortest path algorithms for transportation applications," Computers & Operations Research, vol. 33, no. 4, pp. 1012-1040, 2006. Doi: 10.1016/j.cor.2005.03.027

[9] F. Wang, W. Liu, H. Liu, and H. Zhang, "Estimating O-D Travel Time Matrix by Google Maps API: Implementation, Advantages and Implications," *Journal of Transport Geography*, vol. 86, p. 102770, 2020. doi:10.1016/j.jtrangeo.2020.102770.

[10] S. Shukla, "Edge Relaxation Property for Dijkstra's Algorithm and Bellman–Ford's Algorithm," *GeeksforGeeks*, Nov. 30, 2023. Accessed: Feb. 18, 2025. [Online]. Available: https://www.geeksforgeeks.org/edge-relaxation-property-for-dijkstras-algorithm-and-bellman-fords-algorithm/

[11] X. Guo and X. Luo, "Global Path Search based on A* Algorithm," in Proc. Int. Conf. Transportation & Logistics, Information & Communication, Smart City (TLICSC), vol. 161, pp. 369–374, Dec. 2018.

[12] T. Hossain et al., "Comprehensive Review of the A* Algorithm: Theory, Implementation and Applications," 2024. [Online]. Available: https://www.researchgate.net/publication/378521690_Comprehensive-Review-of-the-A-Algorithm-Theory-Implementation-and-Applications.pdf

[13] Programiz, "Online Python Compiler (Interpreter)," Programiz.com. Accessed: Oktober. 29, 2025. [Online]. Available: https://www.programiz.com/python-programming/online-compiler

[14] A. Ardiansyah, "Source code for shortest-path analysis (Dijkstra & A*)," Google Drive, Accessed: Okt. 28, 2025. [Online]. Available: https://drive.google.com/file/d/1qaHlSOgYVoMRDP5LzCIvWZivLKatGjHl/view

[15] Lubis, H. S. (2009). Perbandingan Algoritma Greedy dan Dijkstra untuk menentukan lintasan terpendek. *Skripsi Universitas Sumatera Utara*.

DOI: https://doi.org/10.65126/jocosir.v3i2