International Journal of Information System & Technology
Akreditasi No. 158/E/KPT/2021 | Vol. 7, No. 5, (2024), pp. 333-340

Determination of The Closest Path Using The Greedy Algorithm

Mhd Furqan[1], Rifki Mahsyaf Adha[2], Armansyah[3]*
[1,2,3]Departement of Computer Science, Faculty of Science and Technology,
Universitas Islam Negeri Sumatera Utara, Indonesia
Email: [1]mfurqan@uinsu.ac.id, [2]rifkimahsyaf@gmail.com,
[3]armansyah@uinsu.ac.id

Abstract

Several alternate routes are displayed by the greedy algorithm, which is widely used in the closest travel route search application. This study employs the greedy method, which sets up a route map to quickly determine the shortest path. The goal of this study is to find the shortest path using a greedy algorithm. By using a greedy algorithm system to find the closest point to which the user's selection is made, the study's eight times with different points on the graph can be seen in the user's position. In an attempt to find the best solution, the greedy algorithm—which is renowned for its simplicity and effectiveness—iteratively chooses the best option available at each step. The greedy algorithm frequently gives priority to proximity when it comes to travel route optimization, and it might not always produce the shortest path overall. However, it's a well-liked option for some applications due to its quickness and simplicity of implementation. Notwithstanding its drawbacks, the greedy algorithm can offer insightful solutions for optimization and route planning issues. Users can make decisions more quickly and possibly find alternate routes they might not have otherwise thought of by using this algorithm to find the closest point in a travel route search application. The study's conclusions also emphasize how crucial it is to take user convenience and preferences into account when developing route planning systems. Future studies could look into ways to improve the greedy algorithm's performance and fix its drawbacks, like adding more heuristics or combining it with other optimization strategies. Overall, this study's findings validate the greedy algorithm's efficacy as a workable choice for locating the closest point in travel route search applications, providing consumers with a dependable and approachable navigational aid.

Keywords: *Greedy Algorithm; rute; node; graph*

1. Introduction

Algorithms can solve various computer science problems by combining several simple, random ways to construct a solution. The algorithm is divided into several approaches by taking an easy way to solve the problem. The Greedy algorithm is a fast approach to finding solutions and simple implementations of search methods with easy calculations [1]. One issue is that various studies focus on searching for the shortest route at different locations. The shortest route is the choice in solving a problem by determining from the starting point to the reference point [2]. The shortest route search saves time in getting to the destination. The Greedy Algorithm forms a step solution, and at each step will have many choices and possibilities that can be followed by exploration with this algorithm. The next step decision is to take the most profitable in the current situation [3].

A large area with many highways often makes it difficult for someone to find the most optimal route in terms of distance and time taken to travel from one place to another. Currently, searching for the closest route with the most optimum is a problem, and it is essential to overcome it so that time and money are well-spent when traveling. It is even essential for shipping and distribution companies to find a solution to this problem. This problem can be a reference for improving quality in helping users to know and choose the closest path with the most optimum route. The greedy algorithm is one of the methods that will help find the path with the closest route to reach a destination [4].

ISSN : 2580-7250

The best route provides a choice of the closest path to help users reach their destination quickly. Efficiency is one way to solve the problem in stages. Juniar (2015) researched the application of greedy algorithms to develop production schedules that produce optimal solutions in the convection industry. Another study conducted by [5] Oktaviana & Naufal (2017) implemented a greedy algorithm in room optimization in preparing class schedules and proved effective in helping to arrange rooms using a greedy algorithm. According to [6] Xu *et al*. (2021), the closest and optimal route efficiency can use a suitable greedy algorithm at each stage to achieve better efficiency and effective performance. Problems in finding the closest route become a basis for making effective alternatives using greedy algorithms. A method that aims to provide solutions that can be solved with the help of information and communication technology in the form of applications that have implemented graph search algorithms [7].

2. Research Methodology

2.1. Greedy Algorithm

The greedy algorithm solves a problem to find a solution step by step. The solution is the best choice, and each step achieves the best possible outcome on a global level. The optimization problem in the greedy algorithm includes several elements, which are the set of candidate solutions at each step. One candidate is taken from the set. The next element is a set of solutions where the set of candidates is selected as a solution. The elements of the selection function select the most likely candidate to take the optimal solution at a step that is never considered again. Elements of the feasibility function with a feasible label in providing a feasible solution, namely, the candidate has been formed and does not violate the existing route. The objective function in a greedy algorithm maximizes or minimizes the value of a solution in a solution [7].

Table 1. Type of Algorithm

Input	Algorithms have input or initial conditions before implementation, which can be a variable value taken from a custom set.
Output	After an algorithm is implemented, it will generate an output reflecting the final conditions based on the input value processed through the algorithm.
Definiteness	The steps written in the algorithm are defined clearly so that it is easy for algorithm users to do
Finiteness	Every algorithm should produce an outcome or output. Subsequently, each initial condition or input given is subjected to further steps.
Effectiveness	The algorithm has specific time intervals for each step, resulting in a suitable solution at the end, as expected.
Generality	The steps of the algorithm are applied to any set of inputs that fit a given problem, not just to a particular set.

The Greedy Algorithm focuses on completing the solution one step at a time. When it reaches a node, it considers it as a potential destination. It selects the next node based on the shortest distance between nodes, not necessarily toward the final destination. This process continues until it reaches the destination point [8].

2.2. Graph

The graph is a diagram that contains specific information when interpreted precisely [3]. In Graph G, elements are defined as set order (,) and denoted as follows G = (,). The is a non-empty set of a node and is the set of edges that connects a pair of nodes and node E. Nodes in a graph can be denoted by letters of the alphabet such as a, b, c, d, …. with natural numbers like 1, 2, 3, 4, …. or can be combined from both [9]. The problem of finding the shortest path in a graph is one optimization problem. When searching for the shortest path, a weighted graph is used. This means that each edge in the graph is assigned

a value or weight. The weights on the sides of the graph can represent the distance between cities, message delivery time, development costs [10].

The assumption used here is that all weights are positive. The shortest path is the path that is passed from one node to another node magnitude or value on the side that sums up finally from the initial node to the final node the smallest. The shortest path is the minimum path required to reach one place from another. The graph used is weighted, where each side is given a value or weight [7].

2.2. Method

The method used to support this research with several data collection stages and hardware and software. Applying greedy algorithms to mobile applications uses the Javascript programming language. It uses NodeJs and Expo as development tools with a smooth implementation process that requires some attribute data for x and y coordinates and attribute data in distance data for each location. The software used is the Microsoft Windows 10 Operating System, and sublime text 3.

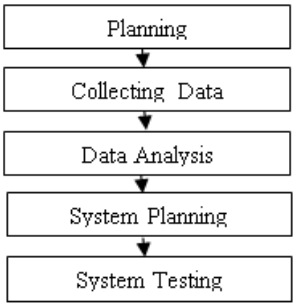

Figure 1. Research Stages

2.3. Algorithm Design

The application algorithm is to be implemented into the application by executing the instructions in the application in stages so that the application can be used appropriately and efficiently by the user. The user must perform the following algorithm in running the application to activate GPS on the smartphone, and the user logs into the application, the users input the destination location. After the user enters the destination data, the data will be processed, and then the users will see the closest route along with the distance and travel graph.

2.4. Flowchart Greedy

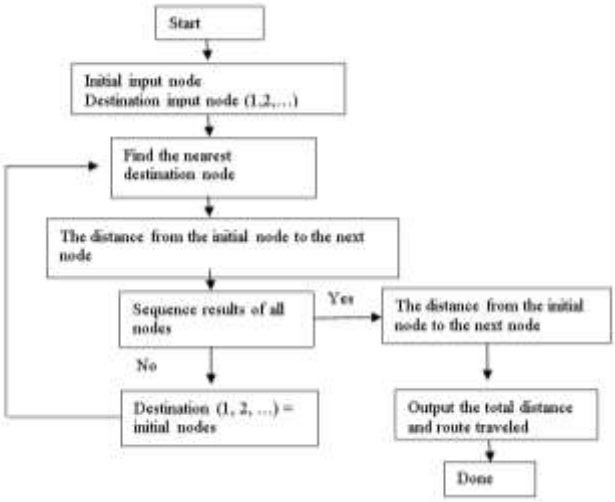

Figure 2. Flowchart Greedy Algorithm

3. Result and Discussion

3.1. Result

The results of the study of the closest route search application of a mobile-based greedy algorithm. The results of the study are described in table 1 as follows.

Table 2. Location Data

Location	***Mark***
Gudang pupuk PT. Mekar Trawali	*Point 1*
Toko Pupuk NPK	*Point 2*
Toko Pupuk Sembiring	*Point 3*
Toko Pupuk Nutrive	*Point 4*
Distributor Pupuk organik Cair Sawit	*Point 5*
Distribusi pupuk Nasa Medan	*Point 6*
Toko Pupuk Tani Jaya	*Point 7*
Agen Resmi Pupuk NASA Medan	*Point 8*
Tok Pupuk Organik NASA	*Point 9*
Toko pupuk Praninta	*Point 10*

The location point for each mark is adjusted and implemented based on a mobile application by displaying test results from the nearest route search application that has been made as material for implementing the Greedy Algorithm to explain the appearance of the display on the nearest route search application applied to the Greedy Algorithm.

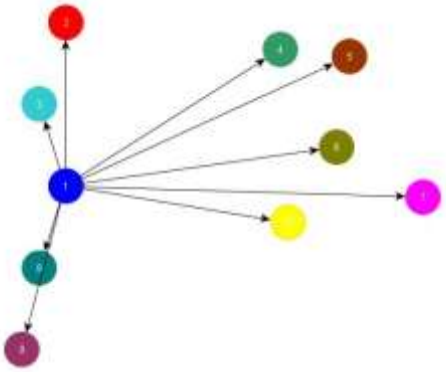

Figure 3. First Graph

The test results show that the user's initial position is in the PT Mekar Trawali fertilizer warehouse, with position 1 as the starting point. Then the system will apply a Greedy Algorithm to determine the closest point to which the user will go, and the closest point is obtained, namely point three. So the first user goal is point three, Toko Pupuk Sembiring.

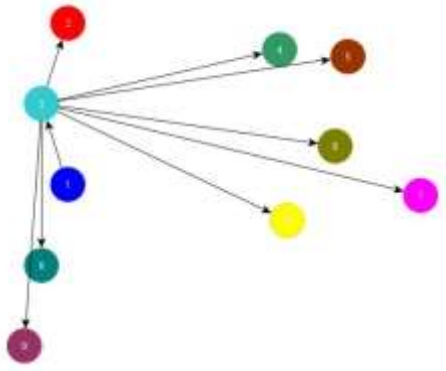

Figure 4. Second Graph

Based on the test above, the user current position is at point 3, Toko Pupuk Sembiring. Then the system will apply a Greedy Algorithm to determine the closest point to which the next user will go and the closest point is obtained to point 2. So, the second destination is Toko Pupuk NPK.

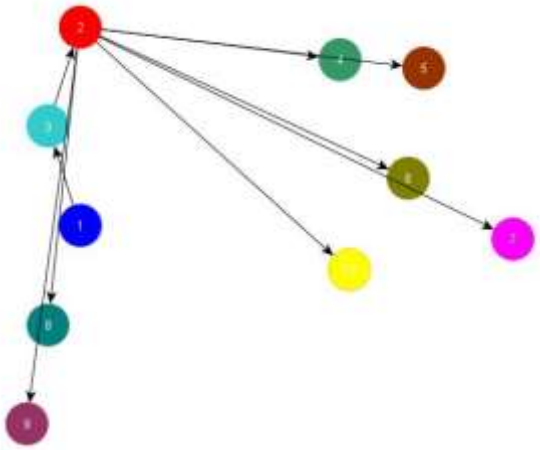

Figure 5. Third Graph

Based on the test above, the user current position is at point 2, Toko Pupuk NPK. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 4. So, the third destination is Toko Pupuk Nutrive.

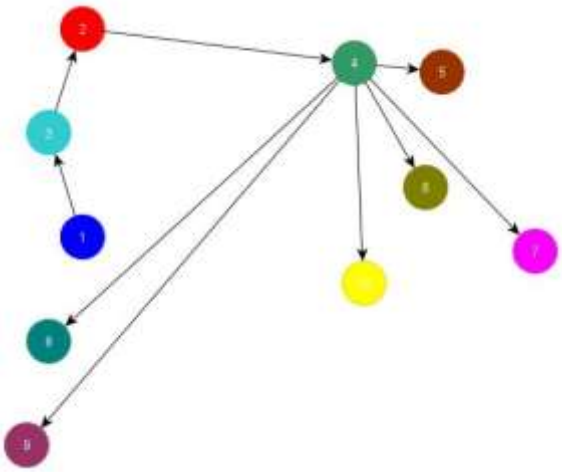

Figure 6. Fourth Graph

Based on the test above, the user current position is at point 4, Toko Pupuk Nutrive. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 5. So, the fourth destination is Distributor Pupuk Organik Cair Sawit.

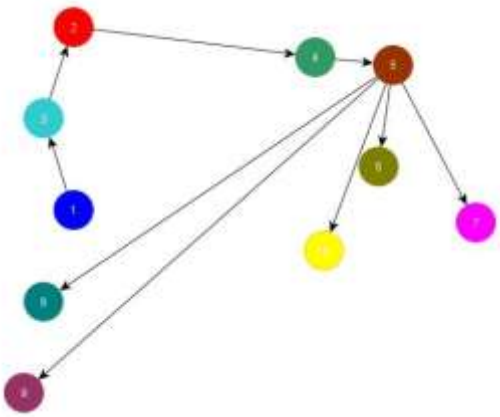

Figure 7. Fifth Graph

Based on the test above, the user current position is at point 5, Distributor Pupuk Organik Cair Sawit. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 6. So, the fifth destination is Distributor Pupuk Nasa Medan.

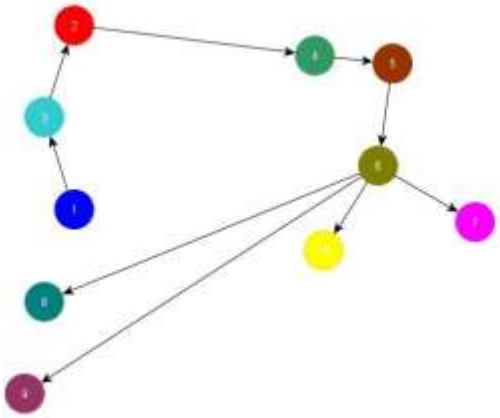

Figure 8. Sixth Graph

Based on the test above, the user current position is at point 6, Distributor Pupuk Nasa Medan. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 7. So, the sixth destination is Toko Pupuk Tani Jaya.

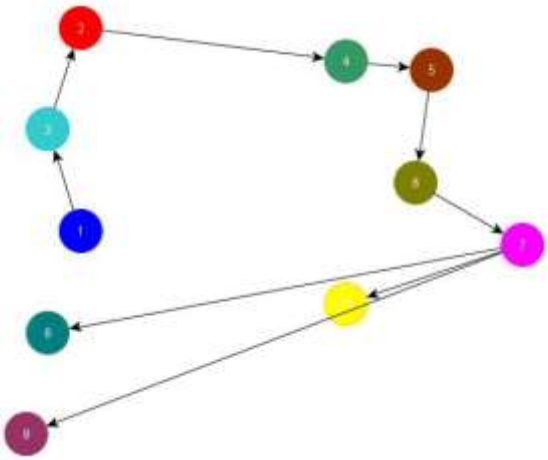

Figure 9. Seventh Graph

Based on the test above, the user current position is at point 7, Toko Pupuk Tani Jaya. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 10. So, the seventh destination is Toko Pupuk Praninta.

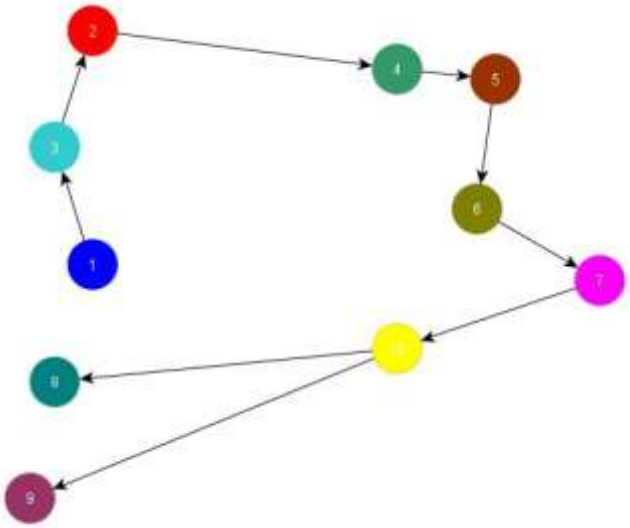

Figure 10. Eigth Graph

Based on the test above, the user current position is at point 10, Toko Pupuk Praninta. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 8. So, the eigth destination is Agen Resmi Pupuk Nasa Medan.

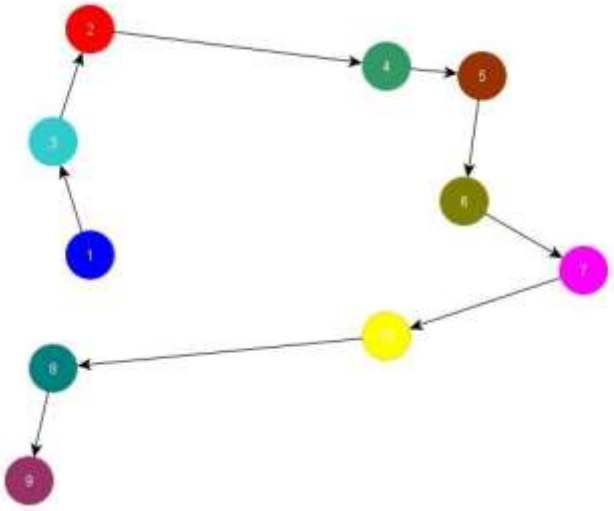

Figure 11. Ninth Graph

Based on the test above, the user current position is at point 8, Agen Resmi Pupuk Nasa Medan. Then the system will apply a Greedy Algorithm to determine the closest point is obtained to point 9. So, the nineth destination is Toko Pupuk Organik Nasa.

3.2. Discussions

In the shortest path, the data that will be used is usually the distance. The distance is used as a reference to compare one distance with another to get the shortest final conclusion. In this research, the authors made a case study on the problem of fertilizer distribution in the city of Medan. The data is used in the form of distance, which will later be used to find the nearest location. In the resulting graph above, it is known that the current user position is at each point. Then the selection process is carried out from one

point to another point. The graph shows that the final destination is at point 9, which means it is the end point of the user's destination. We get a graph of the user travel route starting from point 1 (starting location) to point 3 (first destination) to point 2 (second destination) to point 4 (third destination) to point 5 (fourth destination) to point 6 (fifth destination) to the 7 (sixth destination) to the point 10 (seventh destination) to the point 8 (eight destination) to the 9 (ninth destination).

4. Conclusion

The application of the greedy algorithm has proven to be effective in helping to find the desired routes because, in the way this algorithm works, it assumes routes or paths as networks with non-negative vertex weights, exact and fast. For tracing to every possible branch, a recursive procedure is used to speed up the program's running. Suggestions for further research are expected to be able to implement this algorithm to find the closest route to different cases. The greedy algorithm has been improved to find the shortest path in a more optimal way.

References

[1] P. Di. Thanh, H. Thi Thanh Binh, D. Di. Dac, N. Binh Long, and L. M. Hai Phong, "A Heuristic Based on Randomized Greedy Algorithms for the Clustered Shortest-Path Tree Problem," *2019 IEEE Congr. Evol. Comput. CEC 2019 - Proc.*, pp. 2915–2922, 2019, doi: 10.1109/CEC.2019.8790070.

[2] M. M. lakshmi and M. M. S. priya, "Shortest Path Algorithm and its Implementation," *Int. J. Math. Trends Technol.*, vol. 36, no. 2, pp. 82–85, 2016, doi: 10.14445/22315373/ijmtt-v36p511.

[3] Sabarudin, Raja, "Solusi Optimum Minmax 0/1 Knapsack Menggunakan Algoritma Greedy," *Jurnal Evolusi* , vol. 147, no.2 . March, pp. 11–40, 2016.

[4] A. Juniar, A. Juniar Jurusan Sistem Informasi, and S. Tinggi Manajemen Industri - Kementerian Perindustrian Jl, "Penerapan Algoritma Greedy pada Penjadwalan Produksi Single-Stage dengan Parallel Machine di Industri Konveksi," *OKTOBER 2015 IJCCS*, vol. 16, pp. 1–5.

[5] S. Oktaviana and A. Naufal, "Algoritma Greedy untuk Optimalisasi Ruangan dalam Penyusunan Jadwal Perkuliahan," 2017.

[6] U. M. Rifanti and B. Arifwidodo, "The implementation of floyd algorithm in determining the shortest routes for tourism transportation," *Regist. J. Ilm. Teknol. Sist. Inf.*, vol. 5, no. 2, pp. 140–151, Jul. 2019, doi: 10.26594/register.v5i2.1683.

[7] E. N. Hayati and A. Yohanes, "Pencarian Rute Terpendek Menggunakan Algoritma Greedy," *Semin. Nas. IENACO*, pp. 2337–4349, 2014.

[8] H. Azis, R. D. Mallongi, D. Lantara, and Y. Salim, "Comparison of Floyd-Warshall Algorithm and Greedy Algorithm in Determining the Shortest Route," *Proc. - 2nd East Indones. Conf. Comput. Inf. Technol. Internet Things Ind. EIConCIT 2018*, pp. 294–298, 2018, doi: 10.1109/EIConCIT.2018.8878582.

[9] S. Saifulloh and R. E. Fitriyani, "Analisa Pencarian Rute Tercepat Menuju Telaga Sarangan Menggunakan Algoritma Greedy," *J. Teknol. dan Ilmu Komput. Prima*, vol. 1, no. 1, pp. 1–5, 2018, doi: 10.34012/jutikomp.v1i1.231.

[10] Y. D. Mahendra, N. Nuryanto, and A. Burhanuddin, "Sistem Penentuan Jarak Terdekat Dalam Pengiriman Darah Di Pmi Kota Semarang Dengan Metode Algoritma Greedy," *J. Komtika*, vol. 2, no. 2, pp. 136–142, 2019, doi: 10.31603/komtika.v2i2.2601.