LONTAR KOMPUTER VOL. 9, NO. 3 DECEMBER 2018 p-ISSN 2088-1541
DOI : 10.24843/LKJITI.2018.v09.i03.p07 e-ISSN 2541-5832
Accredited B by RISTEKDIKTI Decree No. 51/E/KPT/2017

The Comparison Determining Of Some Route Of Angkot In Bandung By Using Greedy Algorithm And Min Plus Algorithm

Eka Susilowati

Education of Mathematics, Universitas PGRI Adi Buana Surabaya
Surabaya, Indonesia
eka_s@unipasby.ac.id

Abstract

Bandung is one of the major cities in Indonesia. The lower middle class is greatly helped by public transportation. Angkot is transportation that is close to the people. However, public transportation services that are less organized can make people switch to using private transportation. This actually has a bad impact on traffic. Thus, there need to be improvements in public transportation in the city of Bandung. One-way roads in the city of Bandung are also the cause of many angkot routes. The choice of public transportation users to choose an efficient angkot route. Efficient here means a short path so that the travel time to the destination is minimal. In the previous article, the Cicaheum Ciroyom and Ujung Berung ITB angkot routes were obtained using the Greedy algorithm. In this discussion, the algorithm that can be used to determine angkot routes in Bandung is the Min-Plus algorithm. After being compared between the Greedy algorithm and the Min plus algorithm, the resulting angkot algorithm is better obtained by the Min Plus algorithm.

Keywords: *Min Plus Algebra, Bandung angkot route, shortest path, Greedy algorithm, Cicaheum terminal*

1. Introduction

The existence of angkot in Bandung actually give benefit to the community, especially the lower middle class and tourists. Bandung people still use public transportation as a means of transportation. Actually there are other transportation devices that can be used such as buses or taxis as a means of public transportation. However, there are considerations that people use public transportation to be an option. Angkot is used because of the low price.

However, the use of public transportation as a means of transportation has a weakness. Setting the streets in the city of Bandung is not simple with many one-way streets. In addition, the many routes from angkot that add to the confusion in the community. The problem that arises is how to choose the angkot route to reach the destination with minimal time and distance.

Research on min-plus algebra has been discussed by Watanabe [4]. Research on the shortest route using various algorithms has been discussed by Diana [5] and Fahim [6]. While determining the shortest route using the min plus algorithm has been discussed by Vivi [7] and Suprayitno [8]. Previous research on the Greedy algorithm. The discussion of the Greedy algorithm has been discussed in the Cahya Gunawan journal [9]. Cahya Gunawan explained the steps of route search using the Greedy algorithm with time and distance weighting. Cahya Gunawan [9] describes the route cicaheum ciroyom. While Shirley [10] explained the route of Ujung Berung ITB using the Greedy algorithm. In this study the algorithm discussed in the journal Rudhito [11] is used. The vertex weight used to obtain the shortest path is travel time. Min plus linear equation system as stated by Rudhito which will later be used to obtain the shortest route of cicaheum ciroyom angkot and berung end to ITB.

2. Research Methods

Methods and steps of research conducted in this study include studying the algebraic concept min plus. After this, we study the system of iterative linear equations min plus along with its properties and study the basic concepts of graph theory. We study the concept of CPM in finding the shortest path.Then, study the adoption and modification of forwarding calculation techniques and backward calculation techniques as in CPM using the algebraic approach min plus. Furthermore, learning the min algorithm plus its application in finding the shortest path and processing the travel time data of the angkot cicaheum ciroyom and ujung berung ITB to become the substance of route determination using the Min Plus algorithm. Next step, studying the Greedy algorithm related to its application in finding the shortest path and analyze the route generated using the Greedy algorithm and the Min Plus algorithm.

3. Result and Discussion

3.1. Basic Theory

3.1.1. Min Plus Algebra

In general, min-plus algebra is analog with max-plus algebra. When given a set $\mathbb{R}_{\varepsilon} = \mathbb{R} \cup \{\varepsilon\}$ with $\mathbb{R}$ is a set of real numbers and $\varepsilon = -\infty$. Operations are defined as follows:

$$a \oplus b = \min\{a, b\} \tag{1}$$

$$a \otimes b = a + b \tag{2}$$

For all $a, b \in \mathbb{R}_{\varepsilon}$. The set $(\mathbb{R}_{\varepsilon}, \oplus, \otimes)$ is a commutative idempotent semiring with neutral elements $= 0$ and unit elements $\varepsilon = +\infty$. The set $(\mathbb{R}_{\varepsilon}, \oplus, \otimes)$ is called min-plus algebra, then it is notated $\mathbb{R}_{\min}$. Relations $\leq_m$ are defined on $x \leq_m y \Leftrightarrow x \oplus y = x$.

3.1.2. Basic Concept of Graph Theory

Graphs can be represented in the form of images consisting of dots labeled representing dots and curves or segments that represent the edge (edge). This curve connects dots.

A path in a directed graph G is a row of arcs $(i_1, i_2), (i_2, i_3), \ldots, (i_{l-1}, i_l)$ with $(i_k, i_{k+1}) \in E, k = 1, 2, \ldots, l-1$. The path can be interpreted $i_1 \to i_2 \to \ldots \to i_l$. A directed graph $G = (V, E)$ with $V = 1, 2, \ldots, n$ is said to be strongly connected if for each $i, j \in V$, $i \neq j$, there is a path from i to j. A graph that does not contain a circuit is called a noncyclic graph.

A directed graph $G = (V, E)$ is said to be of a weighted if the arc $(j, i) \in E$ is related to a real number a_{ij}. Real numbers a_{ij} are said to be arc weights $(j, i) \in E$. The weight of the path is defined as the number of weights of the arcs that makes up the path. The shortest path is the path with the minimum weight among other trajectories.

3.1.3. Applied Min Plus Algebra on the Shortest Path Problem

A matrix $A \in \mathbb{R}_{\min}^{n \times n}$ is said to be semi-definite if all circuits $G = (V, E)$ have a non-positive weight. For semidefinite matrices $A \in \mathbb{R}_{\min}^{n \times n}$ defined $A^* = E \oplus A \oplus A^{\otimes 2} \oplus \ldots \oplus A^{\otimes n} \oplus A^{\otimes n+1} \oplus \ldots$ Next, the set $\mathbb{R}_{\min}^{n}$ is a set

$\{\mathbf{x}=[x_1,x_2,\ldots,x_n]^T \mid x_i \in \mathbb{R}_{\min}, i=1,2,\ldots,n\}$ The following is given a directed graph of weight connected to a strong cyclic definition.

Definition 1

A unidirectional trajectory network S is a weighted directed graph connected to a strong cyclic $S=(V,A)$ with $V=\{1,2,\ldots,n\}$ which meets the conditions: if $(i,j)\in A$ so $i<j$.

A network with travel time weights can be modeled to a direct weighted graph. This graph can be represented as a matrix of min-plus algebra. In a directional network, the point states are the intersection, while the arc states is a path, the weight of the arc indicates the travel time so that the weight in the network is non-negative.

The shortest path analysis is done by analyzing and modifying the forward and backward calculation techniques in the CPM method on the analysis of critical paths on the project network using a system of linear equations, min plus.

The existence of the unity of the solution of the system of linear equations min plus is the same as the existence of the unity of the system of linear max plus equations (Bacceli). Given $A\in \mathbb{R}_{\min}^{n\times n}$ and $\mathbf{b}\in \mathbb{R}_{\min}^{n}$. If A semidefinite then the vector $x^{\bullet}=A^{*}\otimes\mathbf{b}$ is the solution of the system $\mathbf{x}=A\otimes\mathbf{x}\oplus\mathbf{b}$.

Theorem 2

Given a network path in the direction n of the point and A is a networked weighted direct graph weighting matrix of the network. Vector at the earliest starting point i can be passed by

$$x^e=(E\oplus A\oplus\ldots\oplus A^{\otimes^{n-1}})\otimes b^e \tag{3}$$

Where $\mathbf{b^e}=\left[0,\varepsilon,\ldots,\varepsilon\right]^T$. Furthermore, x_n^e is a minimum time to traverse the network.

Theorem 3

Given a direct path network with n the point and A is a networked weighted direct graph weighting matrix of the network. Vector at the time of a solution of the latest point given by

$$\mathbf{x^l}=-((A^T)^*\otimes\mathbf{b^l}) \tag{4}$$

Where $\mathbf{b^l}=\left[\varepsilon,\varepsilon,\ldots,-x_n^e\right]^T$. The two theorems above become the basis for calculating minimum time determination. Determination of minimum time is done by first determining the star operation on the matrix of min-plus algebra (A. Rudhito). The next step, determining the critical path through CPM, with the min-plus algebra approach, the critical path can be searched by requiring $x_i^l=x_i^e$.

3.1.4. Min Plus Algorithm

The Min Plus algorithm is an algorithm that adopts the calculation technique available on CPM. The calculation technique adopted is forward calculation and backward calculation technique. From the calculation technique, combined with the linear iterative equation system min plus. If described in steps per step, the min plus algorithm can be described as follows:

The Min Plus algorithm has the following calculation steps such as enter a min-plus matrix, $n\times n$ which is a matrix that corresponds to a trajectory graph. Then count forward such as counting $A^2,A^3,\ldots,A^n$ and A^+, counting E and A^*. After this, create a matrix B. Calculate vectors when starting the earliest $ESi=A^*\times B$. Create a matrix when it starts at the earliest $MESi$. Create a matrix when it starts at the earliest $MESj$. Create a matrix at the fastest completion $MECj$ Countdown such as counting A_2 and A_2^+, counting E_2 and A_2^*. Create a matrix B_2. Calculate the slowest completion vector $LCj = -(A_2^* \times B_2)$

3.1.5. Greedy Algorithm

Greedy Algorithm is a problem-solving step by step and is one method to solve optimization problems. Determining the solution using the Greedy algorithm is described in the following steps:

a. There are many choices that need to be explored at each step of the solution. Therefore, every step must be concluded the best decision in determining the choice. The decision that has been taken in a step cannot be changed again in the next step.
b. The approach used in the Greedy algorithm is to make visible choices that provide the best acquisition solution that is by making the locally optimum choice at each step expected to provide a global optimum solution.

The way the Greedy algorithm works:

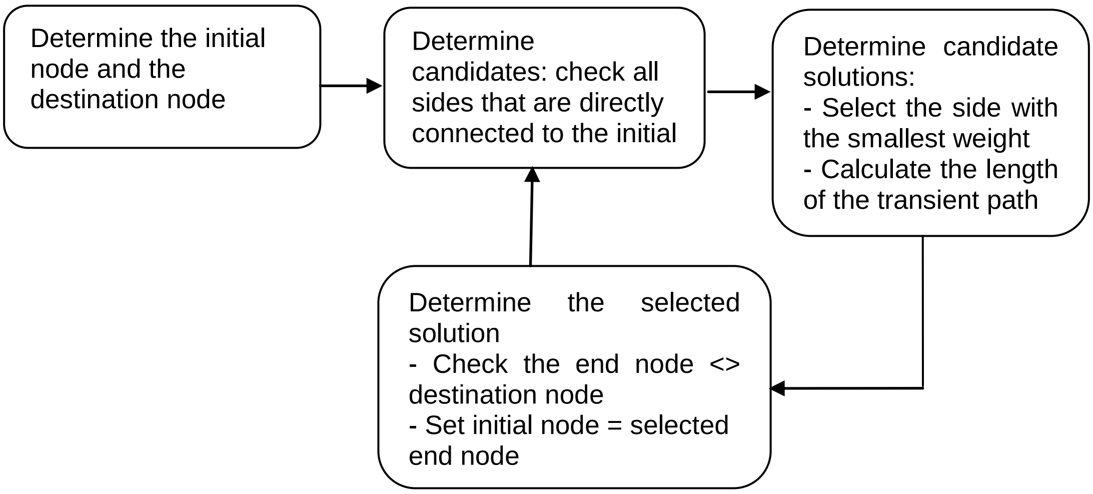
Determine the initial node and the destination node
Determine candidates: check all sides that are directly connected to the initial
Determine candidate solutions:
- Select the side with the smallest weight
- Calculate the length of the transient path
Determine the selected solution
- Check the end node <> destination node
- Set initial node = selected end node

Figure 1. Greedy algorithm process flow

The Greedy algorithm is based on the transfer of edges per edge and every step was taken does not have consequences for the future. The Greedy Algorithm does not operate in its entirety against all the alternative solutions that exist and some Greedy problems do not always succeed in providing solutions that are truly optimum but provide solutions that are near optimum. Optimization problems that are solved using the Greedy algorithm are composed of elements, namely the candidate set, the set of solutions, the selection function, the feasibility function and objective functions.

3.2. Result

3.2.1. Angkot Route by Using Min-Plus Algorithm

Angkot routes in Bandung can be described using graph theory. This depiction is carried out as one of the steps to analyze angkot routes in Bandung using the min plus algorithm. The form of angkot route graph in Bandung is a collection of nodes connected to the edge. The weight provided in the edge indicates the travel time from the bus stop to the bus stop. The following form is used to describe the angkot route graph in Bandung.

○ is a node that symbolizes the stop.

p-ISSN 2088-1541
e-ISSN 2541-5832

→ is an edge which symbolizes the direction of the angkot route.

The shortest path is searched if it does not load the circuit on the track image. This path image is represented by a graph. In order to meet the requirements of the track sought using the MATLAB program above, look for the path that fulfilled these requirements.

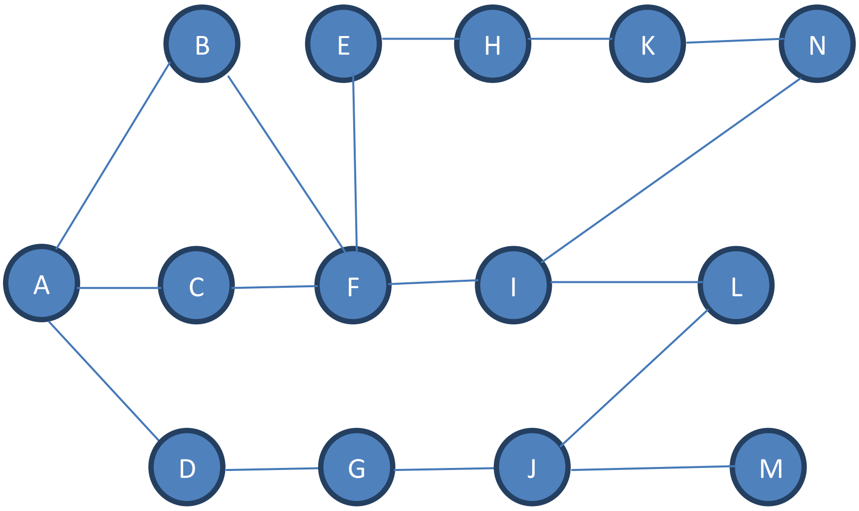

Figure 2. Cicaheum terminal line to Jl. Dipatiukur

Explanation :

A :	Terminal Cicaheum
B :	Antapani
C :	Jl. Ahmad Yani
D :	Jl. K.H. Hasan Mustopa
E :	Jl. Sukabumi
F :	Jl. Jakarta
G :	Jl. Surapati
H :	Jl. Laswi
I :	Jl.Supratman
J :	Jl.Panata Yuda
K :	Jl. Riau
L :	Jl. Diponegoro
M :	Jl. Dipatiukur
N :	Jl. Taman Pramuka

Table 1. Travel time and Distance Cicaheum terminal line to Jl. Dipatiukur

From	To	Travel time(minutes)	Distance(km)
A	B	4,5	1,6
A	C	6,5	3,7
A	D	3,5	1,3
B	F	6,5	2,4
C	F	1	0,45
F	E	1	0,8
D	G	7,5	2,4
E	H	6	1,8
F	I	5	1,4

G	J	6	1,8
H	K	6,5	2,3
K	N	4,5	1,3
N	I	2	1
I	L	6,5	2,6
L	J	4,5	1,6
J	M	2	0,85

Input in program MATLAB is adjacency matrix. The adjacency matrix of figure 2 is

$$\begin{pmatrix}
\infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
4,5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
6,5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
3,5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & 1 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & 6,5 & 1 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & 7,5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & 6 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & 5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & \infty & 6 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & \infty & \infty & 6,5 & \infty & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 6,5 & \infty & \infty & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 4,5 & \infty & \infty & \infty \\
\infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & 2 & \infty & \infty & \infty & \infty
\end{pmatrix}$$

The final result of the course of the program uses MATLAB, it looks like the time vector starts the fastest through the path (i, j), $\mathbf{x}^{\mathbf{e}}$ and the time is past the pass, at the latest (i, j), $\mathbf{x}^{l}$.

$$\mathbf{x}^{e} = \begin{pmatrix} 0 \\ 4,5 \\ 6,5 \\ 3,5 \\ 8,5 \\ 7,5 \\ 11 \\ 14,5 \\ 12,5 \\ 17 \\ 21 \\ 19 \\ 25,5 \\ 19 \end{pmatrix} \text{ dan } \mathbf{x}^{l} = \begin{pmatrix} 0 \\ -\varepsilon \\ -\varepsilon \\ 3,5 \\ -\varepsilon \\ -\varepsilon \\ 11 \\ -\varepsilon \\ -\varepsilon \\ 17 \\ -\varepsilon \\ -\varepsilon \\ -\varepsilon \\ 19 \end{pmatrix}$$

The shortest path is selected if $\mathbf{x}_i^e = \mathbf{x}_i^j$. Figure 2 is a graph representation of the Cicaheum terminal line to Jl. Dipatiukur. From the results above, the minimum travel time is 19 minutes. The shortest paths obtained are (A, D), (D, G), (G, J), (J, M).,

The route from terminal Cicaheum to Jl. Dipatiukur is

Terminal Cicaheum – Jl. K.H. Hasan Mustopa – Jl. Surapati – Jl. Panata Yuda – Jl. Dipatiukur

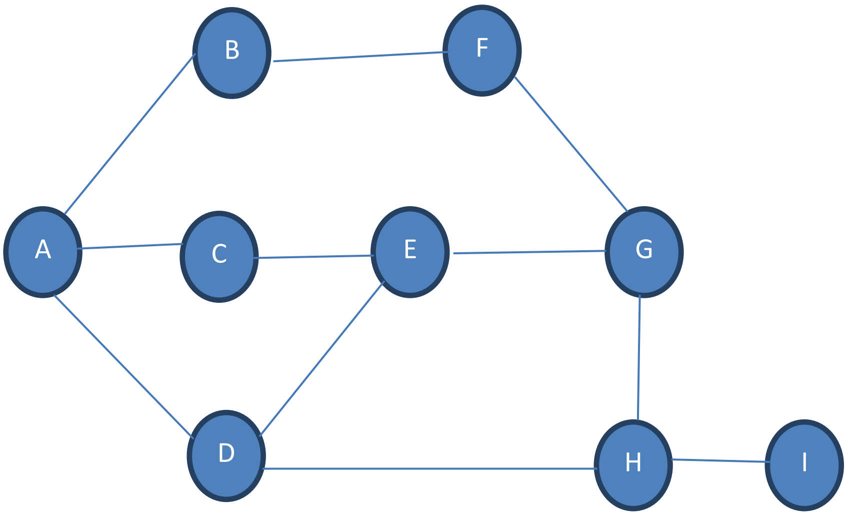

Figure 3. Ujung Berung bus stop to ITB

Explanation :
A : Ujungberung
B : Gedebage/ Soekarno – Hatta
C : Antapani
D : Terminal Cicaheum
E : Cicadas
F : Kiaracondong
G : Supratman
H : Gasibu
I : Siliwangi/ Sabuga ITB

Table 2. Travel time and Distance Ujung Berung bus stop to ITB

From	To	Travel time(minutes)	Distance(km)
A	B	6,5	7
A	C	10	5
A	D	10	4,5
B	F	17	8
C	E	6	2
D	E	4,5	2
D	H	11	5
F	G	8	2,5
E	G	7	2,5
G	H	6	3,5
H	I	4,5	1,5

Input in program MATLAB is adjacency matrix. The adjacency matrix of figure 3 is

$$\begin{pmatrix} \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 6{,}5 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 10 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ 10 & \infty & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & 6 & 4{,}5 & \infty & \infty & \infty & \infty & \infty \\ \infty & 17 & \infty & \infty & \infty & \infty & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & 7 & 8 & \infty & \infty & \infty \\ \infty & \infty & \infty & 11 & \infty & \infty & 6 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & \infty & \infty & 4{,}5 & \infty \end{pmatrix}$$

The final result of the course of the program uses MATLAB, it looks like the time vector starts the fastest through the path (i, j), $\mathbf{x}^{\mathbf{e}}$ and the time is past the pass, at the latest (i, j), $\mathbf{x}^{l}$.

$$\mathbf{x}^{e} = \begin{pmatrix} 0 \\ 6{,}5 \\ 10 \\ 10 \\ 14{,}5 \\ 23{,}5 \\ 21{,}5 \\ 21 \\ 25{,}5 \end{pmatrix} \text{ dan } \mathbf{x}^{e} = \begin{pmatrix} 0 \\ -10 \\ 2 \\ 10 \\ 8 \\ 7 \\ 15 \\ 21 \\ 25{,}5 \end{pmatrix}$$

Figure 3 is a graph representation of the Ujung Berung bus stop to ITB. From the results above, the minimum travel time is 25.5 minutes. The shortest paths obtained are (A, D), (D, H), (H, I).

The angkot route from Ujung Berung to the ITB terminal uses the Min Plus algorithm for the travel time i.e.

Ujung Berung - Cicaheum - Gasibu - Siliwangi / Sabuga Terminal ITB.

3.2.2. Angkot Route By Using Greedy Algorithm

Based on the paper "Search Simulation Cicaheum Ciroyom Angkot Route in Bandung Using Greedy Algorithms" (Cahya Gunawan, [1]) explained the angkot route from Cicaheum terminal to Dipatiukur road. The route obtained using the Greedy algorithm is Cicaheum Terminal - Jl. K.H. Hasan Mustopa - Jl. Surapati - Jl. Panata Yuda - Jl. Dipatiukur if searching for angkot routes with distance and time weights.

According to the paper "The Use of the Greedy Algorithm in Determining the Path of Angkot in Bandung" (Shirley [2]) describes the angkot route from Ujung Berung to ITB using the Greedy algorithm. The results of the search for angkot routes using the Greedy algorithm on prices for the Ujung Berung route to ITB are

1. Ujung Berung – Antapani – Cicadas – Supratman – Gasibu – Siliwangi.
2. Ujung Berung – Gedebage/ Soekarno Hatta – Kiaracondong – Supratman – Gasibu – Siliwangi/ ITB
3. Ujung Berung – Antapani – Cicadas – Supratman – Gasibu – Siliwangi/ITB.
4. Ujung Berung – Terminal Cicaheum – Cicadas – Supratman – Gasibu – Siliwangi/ITB.
5. Ujung Berung – Terminal Cicaheum – Gasibu – Siliwangi/ITB.

The angkot route from Ujung Berung to ITB obtained through the Greedy algorithm on the distance is Ujung Berung - Gedebage / Soekarno - Hatta - Kiaracondong - Supratman - Gasibu - Siliwangi / ITB.

3.2.3. Result Analysis

The angkot route from Ujung Berung to Siliwangi ITB uses the Greedy algorithm with a price weight is Ujung Berung - Antapani - Cicadas - Supratman - Gasibu - Siliwangi. While the angkot route from Ujung Berung to Siliwangi ITB uses the Greedy algorithm with distance weight is Ujung Berung - Gedebage / Soekarno Hatta - Kiaracondong - Supratman - Gasibu - Siliwangi. The use of the Greedy algorithm with price and distance weights produces different routes. Furthermore, the Min Plus algorithm is used, it turns out to produce different routes. The route generated using the Min Plus algorithm is Ujung Berung - Cicaheum Terminal - Gasibu - Siliwangi / Sabuga ITB. When we use different weight with the same algorithm, we find a different route. Moreover, we use a different algorithm, we also find a different route.

When viewed from the time weight, the Berung end route to ITB generated using the Min Plus algorithm, takes 22.5 minutes. It's different when using the Greedy algorithm. The time needed to use the Greedy algorithm with a price weight is 33.5 minutes. While the time needed to use a Greedy algorithm with distance weight is 31 minutes. This difference occurs due to the weight was taken. Greedy algorithm does not always provide optimal solutions. This is because searching the local maximum at each step without regard to the overall solution. If the user wants to make the time more efficient then the route taken with the Min Plus algorithm can be an option

4. Conclusion

Based on the discussion given in the previous chapters, the following conclusions can be drawn: Routes generated using the Min Plus algorithm first, the angkot route from terminal Cicaheum to Jl. Dipatiukur is Terminal Cicaheum – Jl. K.H. Hasan Mustopa – Jl. Surapati – Jl. Panata Yuda – Jl. Dipatiukur. Second, the angkot route from Ujung Berung to ITB is Ujung Berung – Terminal Cicaheum – Gasibu – Siliwangi/ Sabuga ITB. The route from Ujung Berung to ITB using Min Plus Algorithm and Greedy Algorithm is different. Time from Ujung Berung to ITB using Min Plus Algorithm is faster than the time from Ujung Berung to ITB using Greedy Algorithm.

When compared between the Min Plus algorithm and the Greedy algorithm, to determine the shortest path, it is more efficient to use the Min Plus algorithm. In most cases, the greedy algorithm will not produce the most optimal solution, as well as the greedy algorithm usually provides a solution that approaches the optimum value in a fairly fast time. Greedy algorithm does not always provide optimal solutions. This is because the local search is maximum at each step without regard to the overall solution. Min Plus Algorithm always regards to the overall solution because eusing PERT CPM Technique with forwarding technique and backward technique so regard to th overall solution.

References

[1] D. Ardiansyah, "Implementasi Algoritma Greedy Untuk Melakukan Graph Coloring: Studi Kasus Peta Provinsi Jawa Timur," *Jurnal Informatika,* vol. 4, pp. 440-448, 2010.

[2] H. A. Alvin Yuvianto, "Implementasi Algoritma Greedy Pada Pencarian Langkah Optimal Permainan Mahjong Solitaire," *Jurnal Rekayasa Sistem dan Teknologi Informasi,* vol. 1, pp. 226-231, 2017.

[3] A. C. Dian Rachmawati, "Implementasi Algoritma Greedy Untuk Menyelesaikan Masalah Knapsack Problem," *Jurnal Sains dan Komputer,* vol. 12, pp. 185-192, 2013.

[4] Y. W. Sennosuke Watanabe, "Min Plus Algebra and Networks," *RIMS Kokyuroku Bessatsu,* pp. 41-54, 2014.

[5] M. S. K. I. S. Diana Okta Pugas, "Pencarian Rute Terpendek Menggunakan Algoritma Dijkstra dan Astar(A*) pada SIG berbasis Web Untuk Pemetaan Pariwisata kota Sawahlunto," *Transmisi,* vol. 13(1), pp. 27 - 32 , 2011.

[6] S. S. Kistosil Fahim, "Aplikasi Aljabar Max Plus Pada Pemodelan Dan Penjadwalan Busway Yang Diintegrasikan dengan Kereta Komuter," *Jurnal Teknik Pomits,* vol. I, pp. 1 - 6, 2013.

[7] P. B. R. N. I. D. Vivi Suwanti, "Penerapan Min Plus Algebra Pada Penentuan Rute Tercepat Distribusi Susu," *Limits,* vol. 14(2), pp. 103 - 112, 2017.

[8] S. H., "Correctness Proof of Min Plus Algebra for Network Shortest Paths Simultaneous Calculation," *Journal of Technology and Social Science (JTSS),* vol. 1(1), pp. 61 - 69, 2017.

[9] C. Gunawan, "Simulasi Pencarian Rute Angkot Cicaheum Ciroyom Kota Bandung Menggunakan Algoritma Greedy," 2012.

[10] Shirley, "Penggunaan Algoritma Greedy dalam Penentuan Jalur Angkot di Bandung," Institut Teknologi Bandung, Bandung, 2010.

[11] M. A. Rudhito, "Sistem Persamaan Linear Min Plus dan Penerapannya pada Masalah Lintasan Terpendek," in *Seminar Nasional Matematika dan Pendidikan Matematika Universitas Negeri Yogyakarta*, Yogyakarta, 2013.

[12] E. Horowitz, Computer Algorithms, 2nd Edition,, USA: SIlicon Press, 2008.

[13] M. A. Rudhito, Aljabar Max Plus dan Penerapannya, Yogyakarta: Universitas Sanata Dharma Press, 2016.

[14] Mustofa, "Sistem Persamaan Linear Pada Aljabar Min Plus," in *Seminar Nasional Penelitian, Pendidikan dan Penerapan MIPA Univeritas Negeri Yogyakarta*, Yogyakarta, 2011.

[15] Subiono, Aljabar Min Plus dan Terapannya, Version 3.0.0., Surabaya: Institut Teknologi Sepuluh November, 2015.

[16] H. S. Lubis, "Perbandingan Algoritma Greedy dan Dijkstra Untuk Menentukan Lintasan Terpendek," Universitas Sumatra Utara, Medan, 2009.

[17] I. T. P. Alamsyah, "Penerapan Algoritma Greedy Pada Mesin Penjual Otomatis(Vending Machine)," *Scientific Journal of Informatics,* vol. 1, pp. 201 - 209, 2014.

[18] A. Juniar, "Penerapan Algoritma Greedy Pada Penjadwalan Produksi Single-Stage dengan Parallel Machine di Industri Konveksi," *Jurnal SIFO Mikrosil,* vol. 16, pp. 175-184, 2015.

[19] P. Wahyuningsih, "Penerapan Algoritma Greedy Untuk Mendeteksi Aktivitas Lansia Pada Karpet Menggunakan Arduino Mega," *Jurnal Informatika Sains dan Teknologi,* vol. 3, pp. 51-60, 2018.

[20] D. Wiliam Aprilius, "Implementasi Algoritma MAX-MIN Ant System Pada Penjadwalan Mata Kuliah," *Jurnal ULTIMATICS,* vol. V, pp. 48-53, 2013.