ANALYSIS OPTIMATION ALGORITHMS AT SHORTEST PATH IN AREA SURABAYA

[1]MAS NURUL HAMIDAH

Department of Informatics Engineering, University of Bhayangkara Surabaya

Jalan Ahmad Yani 114, Surabaya 60231, Indonesia

e-mail: [1]nurul@ubhara.ac.id

ABSTRACT

The shortest path is one case that often in mobile applications now, and there are several optimization algorithms supporting solution shortest path, it can use the greedy algorithm and the dynamic programming algorithms , of course, both of them have optimization methods are different, the case studies will be taken is the area around Surabaya which will evaluate the performance of the optimization algorithm at a certain point in the point area of Surabaya, the calculation by the two algorithms are, it will conclude appropriate optimization algorithm to get to certain areas in Surabaya area. From the evaluation results of two optimization algorithms are greedy and dynamic programming in the can the optimum solution is to distance the RSI jemur sari heading RSAL is solution the same optimum passed node A-C-E-G-H with total shortest distance that is 1.99 km, while the optimum solution royal plaza toward marvell city also same produces result between greedy algorithm and dynamic programming algorithms that is 1-2-5-7-9 with total cost 2,57 km shortest distance.

***Keyword** : greedy algorithms, dynamic Programming algorithms*

1. INTRODUCTION

One of the problems in a city is a congestion , so the people in the city would require the optimum solution to their destination, they must know the shortest distance to get their destination. Because of the time in a city would be invaluable given the many citizens who have a solid activity and requires a fast time, let alone for example, an employee of the delivery order which is very minimum time that an order to its destination in accordance with the specified time estimates.

Searching the shortest path is one of the solutions for the urban areas, especially in the area major cities, and the solution of the shortest path searching with optimization algorithm, that algorithm Djikstra, greedy algorithm, branch and bound algorithm and dynamic programming algorithm.

in this paper, there are two optimization algorithms used are greedy algorithm and dynamic programming algorithm. Greedy algorithm has a more simple method of dynamic programming method [1] because in the method only view at the closest distance among others, in contrast to dynamic programming algorithm [2], which tend to be more detail in any calculations. In this case the optimum solution is sought is in the area of Surabaya.

2. SHORTEST PATH

Shortest path is one of the cases that use optimization algorithms, because it takes a minimum optimization at the completion of the search the shortest path, searching distance the optimum solution with the value of the minimum, while the shortest distance searcht is in the area of Surabaya for two cases, namely the distance between RSI Jemur Sari to RSAL, and the distance between the Royal

plaza to marvell city. The optimization algorithms used in this journal is a greedy algorithm and dynamic programming algorithm.

2.1 Greedy Algorithms

Greedy algorithm is the most simple algorithm than the others because greedy is the meaning "voracious" the greedy algorithms has only focus the most shortest distance to be selected , with no other choice.

the steps of this method are :

1. the best distance search can obtained at the time
2. Hope that by choosing a local optimum at every step to reach the global optimum. Greedy algorithm assumes that a local optimum is part of the global optimum.

The elements of greedy algorithms are :

1. Set of candidate, *C.*
2. Set of solution , *S*
3. selection function
4. feasible
5. objective function

2.2 Dinamic Programming Algorithms

Dynamic programming is one of the optimization algorithm for the completion of the shortest distance are discussed in this journal. In this algorithm often used for more complex issues. In the completion of premises of this algorithm is to use some calculation phase, where in the stages are interrelated, in contrast to the greedy algorithm that will only make one decision in determining the shortest distance, this algorithm has some of the decisions of the best, of course, is based on several stages already in the count.

Method used are :

1. stage (k) is proses choose of destination node (at here are 4 stage).
2. Status (s) associated with each node in the graph
3. Rekuens relation declare shortest path from status *s* to x_4 in stage *k :*

$$f_1(s) = c_{x1s} \qquad \text{(Basis)} \qquad (2.1)$$

$$f_k(s, xk) = \{ c_{x_{ks}} + f_{k-1}(x_k) \} \qquad \text{(Rekuens)} \qquad (2.1)$$

k = 2,3,4

Where :

a. x_k : desission variable on stage k (k=2,3,4)
b. $c_{x_{ks}}$: cost from s to x_k
c. $f_k(s,x_k)$ *:line cost total from s* to x_k
d. $f_k(s)$ *: minimum cost from* $f_k(s,x_k)$

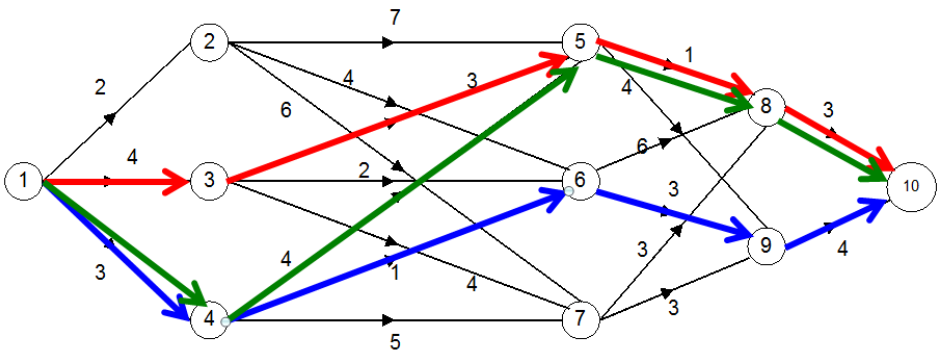

figure 1 Shortest path on dynamic programming algorithms

3. RESULT AND ANALYSIS

3.1 Data Collection

Data collection will get from reference literature and also through google maps to be known point coordinates at any point to be used. The data used only two cases: A case for the calculation of the shortest distance from RSI Jemur Sari to RSAL and for case B for the calculation of the shortest distance from the Royal plaza to Marvell City. Below is a table of nodes are used:

Table 3.1 determination node for Case A

Node	The Place
A	Rumah Sakit Jemur Sari
B	Hotel Luminor
C	UIN Sunan Ampel
D	Plasa Marina
E	JX Expo
F	Gudang Antik
G	Giant
H	RSAL

Table 3.2 determination node for Case B

Node	The Place
1	Royal plasa
2	RSI A. Yani
3	Stasiun Wonokromo
4	Joyoboyo
5	Hotel Novotel
6	Masjid Alfalah
7	AJBS
8	KFC
9	Marvel City

To obtain data on the distance of each node or the destination from google maps. For more details can be seen in the following figure:

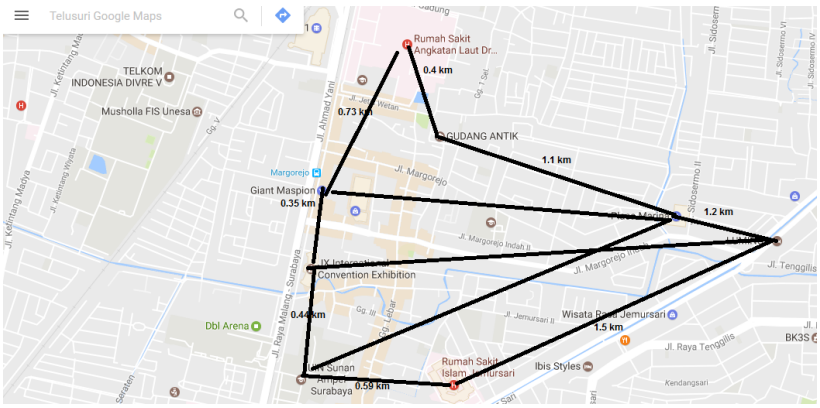

figure 2. the Map RSI Jemur to RSAL

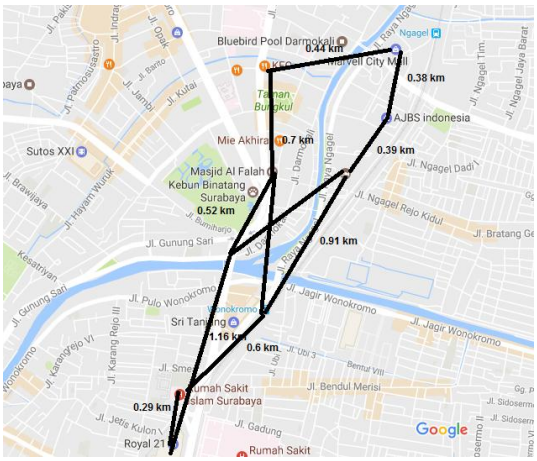

figure 3 the Map Royal to Marvell City

3.2 Design shortest path

The design of shortest path search to determine nodes will through at the time search will refer to the data that have been obtained through google maps, so they can know distance of each node, for more details can be seen in the following figure:

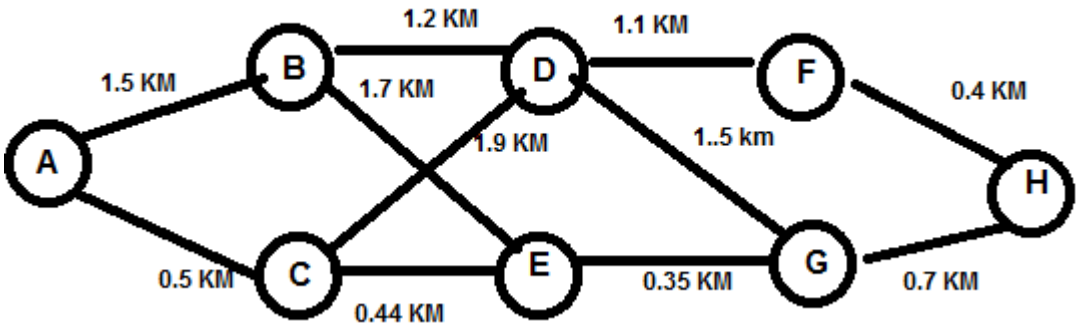

Figure 4. Design shortest path for case A nodes

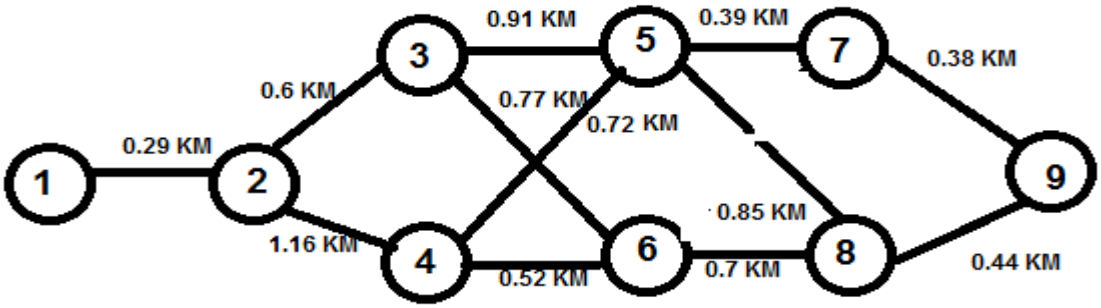

Figure 5. Design of shortest path for case B nodes

3.3 The calculation result

1. Case A

1.1 Greedy Algorithms

For case A calculation with refer to figure 4 . obtanable calculation as:

Calculation route 1 :
A – B – C – E – H with cost 1.5 + 1.2+ 1.1 + 0.4 = 4.2
Calculation route 2 :
A – B – D – G –H with cost 1.5 + 1.7 +1.5 +0.7 = 4.9
Calculation route 3 :
A – C – D – E – H with cost 0.5 + 1.9 + 1.1 + 0.4 = 3.9
Calculation route 4 :
A – C – E – G – with cost 0.5 + 0.44 + 0.35 + 0.7 = 1.99

Then solution optimum of shortest distance is 1.99 km with the route A – C – E – G – H, the places are start from RSI Jemur Sari – UIN Sunan Ampel – JX Expo – Giant - RSAL

1.2 Dynamic programming Algorithms

Stage 1

	Solution optimum	
s	$f_1(s)$	x_1
B	1.5	A
C	0.5	A

Stage 2 $f_2(s) = \min_{s_2}\{c_{x_2 s} + f_1(x_2)\}$

x2 / s	$f_2(s) = \min_{s_2}\{c_{x_2 s} + f_1(x_2)\}$		Solution optimum	
	B	*C*	$f_2(s)$	X_2
D	2.7	2.2	2.2	C
E	3.2	0.9	0.9	C

Stage 3 $f_3(s) = \min_{s_3}\{c_{x_3 s} + f_2(x_3)\}$

X_3 / s	$f_3(s) = \min_{s_3}\{c_{x_3 s} + f_2(x_3)\}$		Solution optimum	
	D	E	$f_3(s)$	X_3
F	3.3	-	3.3	D
G	3.7	1.25	1.25	E

Stage 4 $f_4(s) = \min_{s4}\{c_{x_4 s} + f_3(x_4)\}$

X_4 / s	$f_4(s) = \min_{s4}\{c_{x_4 s} + f_3(x_4)\}$		Solution optimum	
	F	G	$f_4(s)$	*X4*
H	3.7	1.95	1.95	G

Then the optimum solution for greedy algorithms and dynamic algorithms can follows on figure 6 :

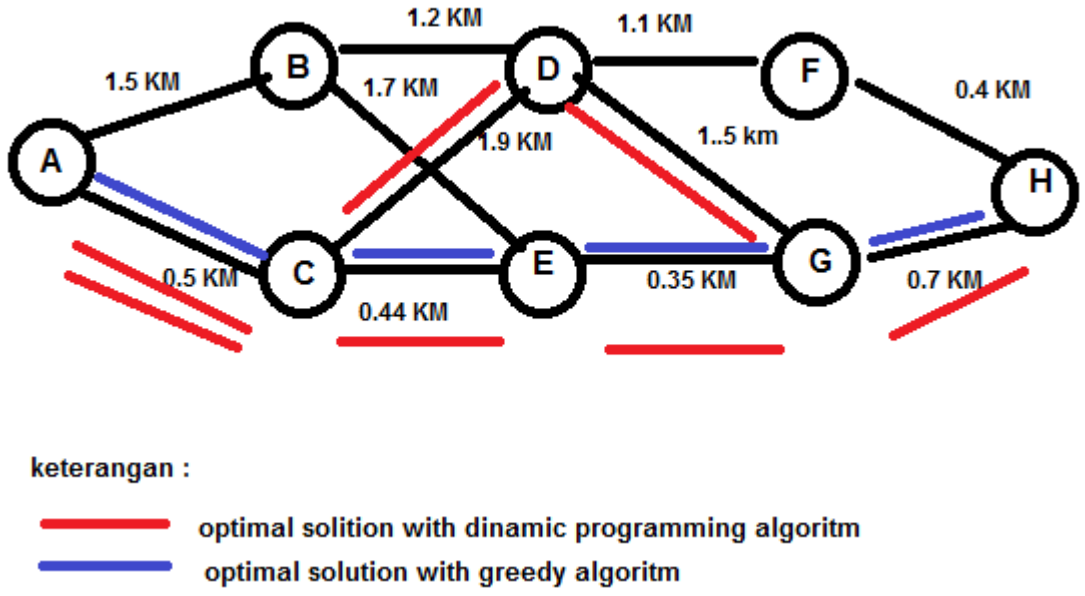

Figure 6. Optimal Solution with algorithms both for case A

2. Problem B

2.1 Greedy algorithms

For case A calculation with refer to figure 5 . obtanable calculation as:
Calculation route 1
1 – 2 – 3 – 5 – 7 – 9 with cost 0.29 + 0.6 + 0.91+ 0.39 + 0.38 = 2.57
Calculation route 2
1 – 2 – 3- 6- 8- 9 with cost 0.wj + 0.6 + 0.77 + 0.7 + 0.44 = 2.8
Calculation route 3
1 – 2 – 4 – 5 – 7 – 9 with cost 0.29 + 1.16 +0.52 + 0.72 + 0.39 + 0.38 = 3.46
Calculation route 4
1 – 2 – 4 – 6 – 7 – with cost 0.29 + 1.16 + 0.52 + 0.85 + 0.38 = 3.2
Calculation route 5
1 – 2 – 4 – 6 – 8 – 9 with cost 0.29 + 1.16 + 0.52 + 0.7 +0.44 = 3.11

Then solution optimum of shortest distance 2.57 km with the route 1 – 2 – 3 – 5 – 7 – 9, and the places are start from Royal plaza – RSI A. Yani – Stasiun Wonokromo – Hotel Novotel – AJBS – Marvell City

2.2 Dynamic programming algorithms.

Stage 1

	Solution optimum	
s	$f_1(s)$	x_1
2	0.29	1

Stage 2 $f_2(s) = \min_{s_2}\{c_{x_2 s} + f_1(x_2)\}$

x2 / s	$f_2(s) = \min_{s_2}\{c_{x_2 s} + f_1(x_2)\}$	Solution optimum

	2	$f_2(s)$	X_2
3	0.89	0.89	2
4	1,18	1.18	2

Stage 3 $f_3(s) = \min_{s_3}\{c_{x_3 s} + f_2(x_3)\}$

X_3 / s	$f_3(s) = \min_{s_3}\{c_{x_3 s} + f_2(x_3)\}$		Solution optimum	
	3	4	$f_3(s)$	X_3
5	1.8	1.9	1.8	3
6	1.66	1.7	1.66	3

Stage 4 $f_4(s) = \min_{s4}\{c_{x_4 s} + f_3(x_4)\}$

X_4 / s	$f_4(s) = \min_{s4}\{c_{x_4 s} + f_3(x_4)\}$		Solution optimum	
	5	6	$f_4(s)$	*X4*
7	2.19	-	2.19	5
8	2.5	2.36	2.5	5

Stage 5 $f_5(s) = \min_{s5}\{c_{x_5 s} + f_4(x_5)\}$

X_5 / s	$f_5(s) = \min_{s5}\{c_{x_5 s} + f_4(x_5)\}$		Solution optimum	
	7	8	$f_5(s)$	*X5*
9	2.56	2.8	2.56	7

Then the optimum solution for greedy algorithms and dynamic algorithms can follows this figure :

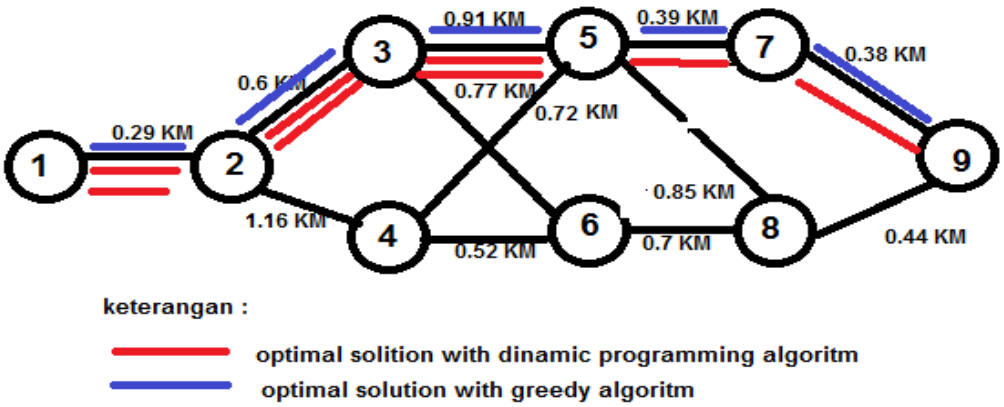

Figure 6. Optimal Solution with algorithms both for case B

3.3 Analysis :

The search of the shortest path algorithm using greedy algorithms and dynamic programming algorithms obtained equally good results in the case of A or case B, in the case of A obtained solution route to greedy algorithm A - C - E - G – H with the places are , the places are start from RSI Jemur Sari – UIN Sunan Ampel – JX Expo – Giant - RSAL and the cost for the case A is 1.99 km to the

solution route with A dynamic programming algorithm - C - E / D - G - H. While the problems and the greedy route Encryption 1-2 - 3-5 - 7-9 , the places are strat from Royal plaza – RSI A. Yani – Stasiun Wonokromo – Hotel Novotel – AJBS – Marvell City and the cost for the case B is 2:57 km but also for dynamic programming algorithm obtained the same route as the route on greedy algorithm. Although the method used to determine the different solutions shortest distance between two algorithms, in this journal showed similar results between the two algorithms.

4. CONCLUSION

This journal can be concluded that:

1. Greedy algorithm could provide a solution that is easiest to just look at the value that the shortest distance between the other distances.
2. The dynamic programming algorithm can provide solutions to some stage in the algorithms no only provide one solution but also could do with some solution paths.
3. Using the algorithms are solution to Case A or Case B are get same result with different methods.

REFERENCES

[1] Enty Nur Hayati, *shortest* p*ath with greedy algorithms, seminar national IENACO* 2014.

[2] Wahyu Jati, *Dynamic programming rick*, M.A 1997 A tutorial on Dynamic programming

[3] Audrey M. H, Indra K. R, and Purbandini, *HOTEL SEARCH SYSTEM BASED TRAVEL ROUTE Shortest TOURIST ATTRACTION TO CONSIDER USING THE GREEDY ALGORITHM*, journal of information system engineering and business intelligence vol 1 no 1 April 2015