Proceeding International Conference on Religion, Science and Education (2022) **1**, 639-644

Shortest Path Search for PDAM Pipeline Leaks Using Dijkstra Algorithm

Rahmat Irsyada[1,*], Hastie Audytra[2]
[1]Computer System, [2]Information System, Faculty of Science and Technology, Universitas Nahdlatul Ulama Sunan Giri,
Jalan Jendral Ahmad Yani No.10, Jambean, Sukorejo, Kec. Bojonegoro, Kab. Bojonegoro, Jawa Timur 62115, Indonesia. Tel. (0353) 887341.

Corresponding author*
irsyada.rahmat@unugiri.ac.id

Abstract: PDAM pipeline leaks are a major problem, both financially and in the short supply of water to customers. This problem resolves to immediately overcome the existing leak. But the problem is how to determine which parts of the control are closest so that improvements become optimal without wasting a lot of money. One of the methods used to solve the problem with the technique of solving algorithm problem, Dijkstra. The Dijkstra algorithm is an algorithm commonly used to search the shortest route. Based on the problem, the program made the shortest route search program for PDAM pipeline leaks using the Dijkstra algorithm.

Keywords: dijkstra, node_control, node_consument, PDAM.

Abbreviations: PDAM: Perusahaan Daerah Air Minum

Introduction

PDAM pipeline leaks are a major problem as they will harm both the PDAM and the customers. Water losses or non-revenue water (NWR) is a common problem for PDAMs in Indonesia. Most of the PDAMs in Indonesia have an NRW rate between 20% - 40%. NWR in PDAMs causes damages like financial loss, and major problems such as lack of water supply volume to the customers until the pressure harms the consumers and ultimately hurt the performance level of PDAM [1]

The problem of water leaks must be handled seriously by immediately repairing the leaking pipe. However, in repairing the leaking pipeline, it must be determined where the channel is repaired. PDAMs have controls in various places. When repairing the leaking pipeline at the consumer's houses, it should be selected from the nearest control section to optimize time and cost. The problem that occurs is how to determine the closest distance so that the nearest control can be selected.

Zhen reveals that optimal route search is always needed and used in route guidance systems such as urban target search and other route guidance implementations. Moreover, in repairing the leaking pipe of the PDAM, an optimal route search program can be used to find the nearest control section to fix it. According to Zhen [2], the resolution for route optimization is by using the Dijkstra algorithm. In addition, Liu [3] asserts that the algorithm Dijkstra is a technique that can overcome the problems of shortest path search from one node to every other node in the network. [4]

Studies on the problems of pipeline leaks have been conducted by many researchers such as detecting the location of pipeline leaks studied by Rianinda, et al. the research aims to find the location of PDAM pipeline leaks using SIGBEE/ IEEE 802.15.4 protocol and M2M platform. Another research conducted by Sudarsono, et al is to investigate the causes and locations of the pipeline leaks using thematic mapping of the distribution pipeline and GIS analysis. Nevertheless, the study

ISSN 2829-3355 | EISSN 2828-8467

had focused on determining the location of the pipeline leaks without determining which controls had to repair the leaks.[5][6][7]

Based on the problem, it will be made a program to determine the shortest route of consumer pipeline leaks by using the Dijkstra algorithm. From the consumer's nodes, it will be searched which control nodes have the shortest path; thus, it can be selected a control node that will repair the leak.

Materials and Methods

The method used in this program is Dijkstra. Dijkstra Algorithm is one of the variants of the greedy algorithm, which is one of the most popular algorithms in solving problems related to optimization problems. It is simple and straightforward. By the meaning that means greedy or vulturous, but not in a negative context. The greedy algorithm only thinks of the best solution to be taken at every step without thinking of future consequences. The principle of it is to take what you can get right now, and the decisions taken at each step cannot be changed anymore. In other words, the greedy algorithm attempts to make a selection of local optimum values at each step and hopes that these local optimum values lead to global optimum ones.[8]

According to Noto, the Dijkstra method is an algorithm aimed at finding the most optimal path. This algorithm is the searches of the shortest path and minimum cost, which cause the search region to expand concentrically so that it has weaknesses in the efficiency, which is weak, and the search time, which is longer when the existing node is so large. Moreover, Fuhao reveals that the weakness of the Dijkstra algorithm is that if the number of the nodes is very big then it will occupy a lot of CPU memory.[9]

Steps of algorithm Dijkstra according to Noto are as follows: [10][11]

1. Step 0 – Mark the starting point
2. Step 1 – Calculate the movement costs for movement from the starting point to each node connected to the starting point and mark the node which has the smallest value.
3. Step 2 – Calculate the movement costs for movement between the starting point and each node connected to the marked node and mark the node which has the smallest value.
4. Step 3 – Repeat step 2 until the destination is marked.

   The value obtained is the minimum cost, the movement to the destination. In addition, the previous node will be stored in the memory when marking the node that allows getting the shortest path to the destination.

This program is created using PHP programming language and implements the Dijkstra algorithm. The program is the shortest route search program for PDAM pipeline leaks. The program consists of two kinds of nodes where the first node is a control node named "B" with a blue node indicator, while the other node is a consumer node named "A" with an orange node indicator. The total node controls in this program are 20 nodes, while the number of node consumers is 30 nodes. This program will then release the closest distance from the consumer node that experiences pipeline leak to the control node that will fix it. The purpose of making this program is to facilitate the determination of which parts of the control should fix the pipeline leaks from the consumers. The determination is based on the closest distance as it is considered to be optimal without wasting a lot of money.

Mapping

Programming begins with the planning process. The first planning is to make mapping of consumer node location and control nodes along with the distance. This mapping will simplify the process of making the program convert it into an adjacency matrix used for mathematical calculations until for mapping the consumer node location and control nodes in the program created. It will be conducted a scaling 1: 100 on the mapping of the program. Hence, when the real distance is 100m, then the distance in the program is 1. Mapping the consumer node location and control nodes is shown in Figure 1.[12]

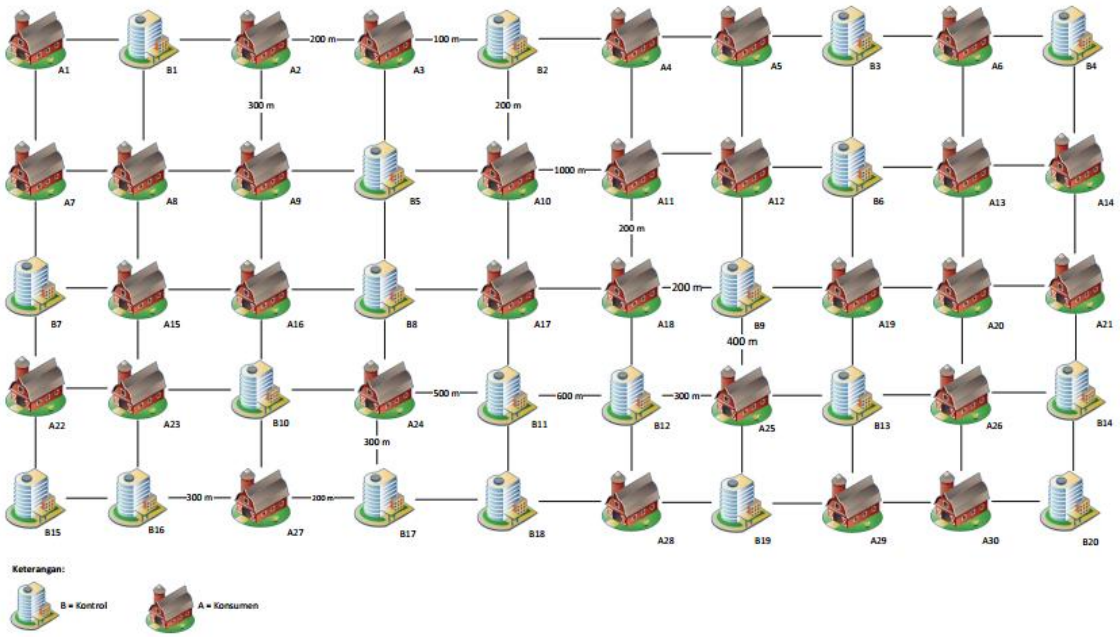

Figure 1. Mapping of consumer locations and PDAM controls.

Adjacency Matrix

Calculations using the algorithm Dijkstra are quite complicated. Therefore, the calculation with the Dijkstra algorithm is usually started by making an adjacency matrix. This matrix is a mapping transformation containing nodes and distances into a table containing distances in numerical form which will make it easier to calculate mathematically. The adjacency matrix is shown in Table 1.

Table 1. Adjacency Matrix.

	A2	A3	A9	A10	A11	A18	A24	A25	A27	B2	B9	B11	B12	B16	B17
A2	0	2	3	0	0	0	0	0	0	0	0	0	0	0	0
A3	2	0	0	0	0	0	0	0	0	1	0	0	0	0	0
A9	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A10	0	0	0	0	10	0	0	0	0	2	0	0	0	0	0
A11	0	0	0	10	0	2	0	0	0	0	0	0	0	0	0
A18	0	0	0	0	2	0	0	0	0	0	2	0	0	0	0
A24	0	0	0	0	0	0	0	0	0	0	0	5	0	0	3
A25	0	0	0	0	0	0	0	0	0	0	4	0	3	0	0
A27	0	0	0	0	0	0	0	0	0	0	0	0	0	3	2
B2	0	1	0	2	0	0	0	0	0	0	0	0	0	0	0
B9	0	0	0	0	0	2	0	4	0	0	0	0	0	0	0
B11	0	0	0	0	0	0	5	0	0	0	0	0	6	0	0
B12	0	0	0	0	0	0	0	3	0	0	0	6	0	0	0
B16	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0
B17	0	0	0	0	0	0	3	0	2	0	0	0	0	0	0

Algorithm

The algorithm of the shortest route determination program of consumer's pipeline leaks to the control using algorithm Dijkstra, a technique of problem-solving, is as follows:

1. Start

2. Initialization of variables such as consumers, controls, and locations of the consumers.
3. Declaring the consumer nodes and control nodes into arrays
4. Combining consumer and control matrices
5. Declaring the distance
6. Entering the distance between nodes
7. Selecting the consumer nodes which experience the leak
8. Declaring the smallest path and cost
9. Arranging the paths using loop
10. There is an option to see the next nodes in the path that have been compiled using a recursive
    a. If the next node is not a control node then the cost is added by the distance and returns.
    b. If the next node is a control node then the path is found.
11. The overall paths found are collected
12. There is an option to determine the shortest path
    a. If the cost is smaller than the smallest cost, then the smallest cost is changed with the cost which will go back to the path of discovery.
    b. If the cost is greater than the smallest cost, then the smallest cost path is revealed.
13. The path and the smallest cost are shown
14. End

Flowchart

An overview of the algorithm of the shortest route determination program of consumer's pipeline leaks to the control using algorithm Dijkstra, the technique of problem-solving.[13] in the form of the flowchart is shown in Figure 2.

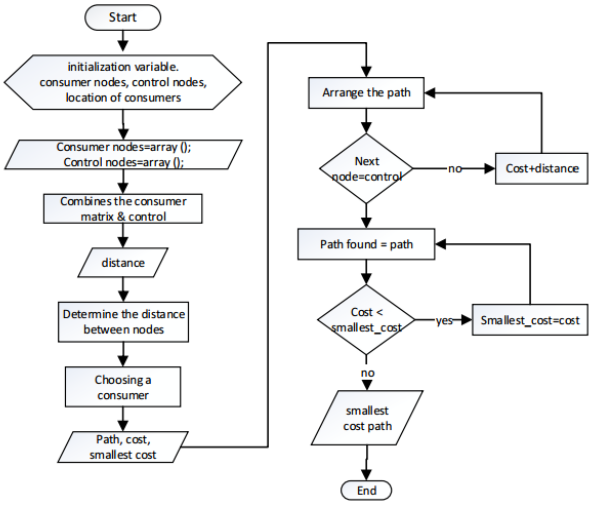

Figure 2. Flowchart of the shortest route search program of PDAM pipeline leaks

Results and Discussion

The result of this research is a program to determine the shortest route of pipeline leaks from the consumer node to the control node by using the Dijkstra algorithm. From the consumer nodes, it will be sought the control nodes that have the shortest path to handle the pipeline leaks of the consumers.

The mapping program of pipeline leaks is shown in Figure 3. The mapping program shows the overview of the PDAM pipeline route consisting of consumer nodes (orange color), node control (blue color), and distances of each node. This program will ask the users to select where the leak is located on the consumer node where there will be an output of which control nodes should fix the leak; the result is obtained from the consideration of the nearest distance of the pipeline leaks.

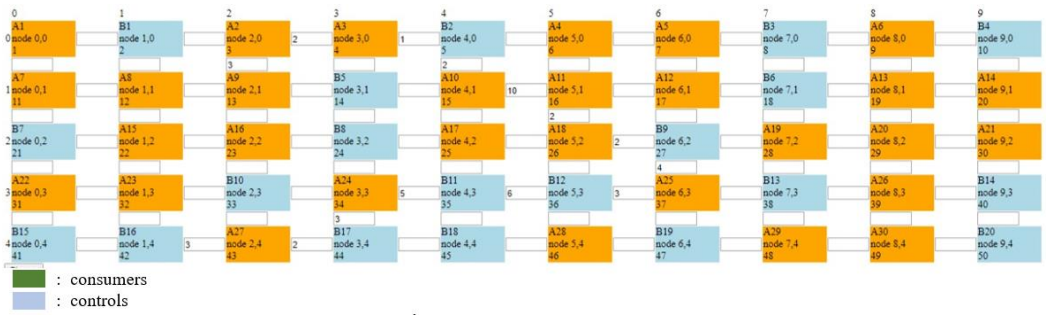

Figure 3. Mapping in program

From the mapping, it is determined the pipeline leak is located on consumer A11. The result of the program is that the control node that should be selected to fix the leakage is node B9, whose cost is 4 that goes into the *optimal* cluster. While the control node that should not be chosen to fix the pipeline leak is node B16 as it has the highest cost that goes into the *not optimal* cluster. Hence, the program results are shown in Figure 4.

Figure 4. Results of program

Discussion

The shortest route search program of PDAM pipeline leak using algorithm Dijkstra results in the shortest route which is B9 control with cost equal to 4, and the longest route which is B16 control with cost equal to 24. Thus, the control that should fix the pipeline leak is B9 control with route A11 - A18 - B9. To see the truth of the program that has been made, then it is checked by making manual calculations that will be compared with the results of the program

Manual calculation of the shortest route of pipeline leak by using the algorithm Dijkstra is done by marking the starting point which is the point of pipeline leaks and then calculating every possibility towards the control nodes, then selecting it by choosing the path that has the shortest route. The control node with the shortest node is selected to address the consumer's pipeline leaks.

The calculation begins by creating an adjacency matrix which will simplify the calculation because it displays the distance of each node in the table form. The adjacency matrix is shown in Table 1. After the adjacency matrix is made, then the calculation is conducted to try every path that exists from the consumer nodes that experience the leak to all control nodes. The results of manual calculations are shown in Table 2.

Table 2. Results of manual calculation.

Route	Path	Cost	Cluster
A11 – B9	A11 – A18 – B9	4	Optimal
A11 – B12	A11 – A18 – B9 – A25 – B12	11	Less optimal
A11 – B2	A11 – A10 – B2	12	
A11 – B11	A11 – A18 – B9 – A25 – B12 – B11	17	
A11 – B17	A11 – A18 – B9 – A25 – B12 – B11 – A24 – B17	19	
A11 – B16	A11 – A18 – B9 – A25 – B12 – B11 – A24 – B17 – A27 – B16	24	Not optimal

In the shortest route search of the PDAM pipeline leak using the algorithm Dijkstra, there are 3 kinds of clusters: (1) *optimal*, (2) *less optimal*, and (3) *not optimal*. The *optimal* cluster is the route that has the lowest cost, where the control node should be selected to fix the pipeline leaks as it has the lowest path distance. The *not optimal* cluster is the route that has the highest cost, where the node control should not be selected because it has a long path. While the *less optimal* cluster is the mid-route that has neither the lowest cost nor the highest cost.

Based on the manual calculations, the shortest distance is the control node B9 with a cost of 4. While the farthest distance is controlled node B16

with a cost of 24. Thus, the manual calculation and program calculation are in line with 100% accuracy, the shortest distance to the control node B9 with a cost of 4 in which the line is A11 - A18 - B9.

Conclusions

The shortest route search of PDAM consumers experiencing pipeline leaks to the fixing controls can be done with programs that implement algorithm Dijkstra search. Dijkstra algorithm is a method commonly used to find the shortest route. Hence, by using this program, it can be easily determined which control nodes are closest to the leaking consumer nodes so that repairing can be done optimally without wasting a lot of money. The results of this program have been compared with manual calculations and the results are in accordance with the theory of the Dijkstra algorithm.

Conflict of Interest: There are no conflicts of interest concerning the publication of this article.

References

[1] Aprian, Raden. 2010. Perbandingan Algoritma Dijkstra dan Algoritma Floyd-Warshall dalam Penentuan Lintasan Terpendek (Single Pair Shortest Path). Makalah If2251 Strategi Algoritmik

[2] Badan Pedukung Pengembangan Sistem Peyediaan Air Minum. 2015. Kierja PDAM 2015. Jakarta: Kementrian Pekerjaan Umum dan Perumahan Rakyat.

[3] Fuhao, Zhang. 2009. An Algorithm of shortest path on Dijkstra for huge data. Sixth International Conference on Fuzzy Systems and Knowledge Discovery.

[4] http://www.lintasbojonegoro.com/ini-sebab-pipa-pdam-rentan-bocor-saat-musim-kemarau/, diakses pada 14 Agustus 2020

[5] Jong Jek Siang. 2004. Matematika Diskrit dan Aplikasinya pada Ilmu Komputer, Yogyakarta: Andi

[6] Joni, Luh. 2010. Pencarian rute terpendek tempat wisata di bali dengan Menggunakan algoritma Dijkstra. Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010).

[7] Liu, Big, et all.. 1994. Finding the Shortest Route Using Cases, Knowledge, and Dijkstra's Algorithm.IEEE (0885-9000/94.S4.00).

[8] Novandi, Raden Aprian Diaz. 2007. Perbandingan algoritma dijkstra dan algoritma Floydwarshall dalam penentuan lintasan terpendek (single pair shortest path). ITB : strategi algoritmik tahun 2007.

[9] Noto, Masato dan Sato, Hiroaki. 2000. A method for the shortest path search by extended Dijkstra algorithm. IEEE (0-7803-6583-6).

[10] Sudarsono, dkk. 2013. Pemanfaatan Pemetaan Tematik untuk Analisis Kebocoran Jaringan Pipa Distribusi di PDAM Demak. Teknik Vol 34 No 3 Tahun 2013. ISSN 0852-1697.

[11] Rianinda, dkk. 2015. Perancangan Prototipe Sistem Pemantau Kebocoran pada Pipa Distribusi Air Menggunakan Protokol SIGBEE/IEEE 802.15.4 dan Platform M2M. e-Proceeding of Engineering Vol 2 No 2 Tahun 2015. ISSN : 255-9365.

[12] Zhen, Fan Yue, et all. 2010. An improved Dijkstra algorithm used on vehicle optimization route planning. 2010 2nd international conference on computer engineering and technology.