International Journal of Engineering, Management and Humanities (IJEMH)
Volume 5, Issue 3, May.-June, 2024 pp: 214-220 ISSN: 2584-2145
www.ijemh.com

The Shortest Path Algorithm of the Modified Dijkstra and Avoid Obstacles

Nabeel Naeem Hasan[1]
[1]Republic of Iraq, Ministry of Education, Open Educational College, Studies Muthanna Centre. Department of Mathematic

Date of Submission: 12-06-2024 Date of Acceptance: 25-06-2024

Abstract

It is advantageous to optimize the pipeline network path in order to lower building costs. The main problem in path optimization is finding the shortest path for large amounts of data. The shortest path issue for large data sets cannot be resolved by Dijkstra's algorithm. To handle large amounts of data when determining the pipeline network's shortest path, Dijkstra's algorithm has been adjusted. Through simulation data, the suggested modified Dijkstra's algorithm (MDA) for pipeline system linking is validated and verified in this work with the purpose of finding the shortest paths that may avoid barriers for large data sizes.

Keywords: Pipeline, Shortest path, and modified Dijkstra algorithm

I. Introduction

The shortest path problem is said to apply to pipeline network design. In an effort to increase profits, businesses and agencies try to design pipelines with the shortest path in order to lower construction or operating costs or shorten the time it takes for various products, such as solid, gas, or liquid fluids, to arrive at their destinations [1, 2]. In this context, the use of shortest path algorithms is essential. Dijkstra's algorithm is one technique used to solve shortest path problems. By calculating the distance between the charts and a specified location, the technique is used to determine the charts' shortest path. Numerous applications, such as Google Maps and Internet networks, used it to determine the quickest route that might be taken from one place to another.

II. Dijkstra's algorithm

One of the most well-known shortest path algorithms is Dijkstra's algorithm, which is recognized for its ability to resolve shortest path problems for connected graphs with numerous sources and a nonnegative weight value for a single destination.

It calculates the shortest path between the beginning and ending points of a connected chart for each vertex in the graph. At the beginning, the mark is equal to zero. However, the distance between the beginning point and other points is unknown because the marks in other vertices equal infinite. Furthermore, it is necessary to identify if a vertex has been visited or not.Once the algorithm has traversed every vertex on the chart, it will come to an end.

2.1. Steps of Dijkstra's Algorithm

Dijkstra's algorithm involves the following steps:
1. Label the starting node (choose the node with the lowest node number if there are many nodes at that moment).
2. Take into account the node that has the most recent boxed label.
3. Decide which temporary label on the network is the least expensive.
4. Continue from Steps 2 and 3 until you reach the destination node.
5. Traverse the network by taking the shortest path possible from the starting node to the destination node.

III. Modified Dijkstra's Algorithm

The goal of this study is to determine the pipeline network's shortest path, which will minimize building costs. Every well refers to its source, while the main station alludes to its goal. This work presents the results of several adjustments to the original Dijkstra's algorithm, which leads to a modified Dijkstra's algorithm that deals with large data sizes quickly while avoiding any obstructions on the path connecting any two places. A thorough explanation of the suggested algorithm known as the Modified Dijkstra's Algorithm (MDA) is given in this section [3].

The most well-known shortest path algorithm is probably Dijkstra's algorithm. Some cases of modification to the original Dijkstra's algorithm have been covered in previous research.

There are situations where finding the shortest path between two nodes can be more expensive when effort and time are crucial. A modified Dijkstra algorithm was proposed by Gbadamosi and Aremu (2020) as an alternative to the traditional algorithm [4-6]. Creating a file with the vertices, edges, and shifting probabilities between edges was part of the technique. A random number generation approach was used to create the probability. A 40-node diagram was used to test the technique, and the findings suggested the shortest path.

A better way to handle the problem of determining the maximum load path is proposed by Wei et al. (2019) as a modified version of Dijkstra's algorithm [7-9]. On the other hand, the maximum load path problem cannot be solved using Dijkstra's technique for determining the shortest path.

Since the original Dijkstra's algorithm cannot handle situations like these, it was modified in this work to solve the shortest path issue, which occasionally involves obstacles of various kinds [10-12].

The updated Dijkstra's algorithm follows a multi-step process [13]. Following the establishment of the first well, the suggested algorithm looks for the second closest well and verifies that it complies with the following requirements: no loop connecting any well to itself, a one-way path connecting the locations, and no closed-loop path. Furthermore, the distance will be penalized by an infinite value if there is an obstruction between the two sites [14]. As a result, the system starts looking for a different route that gets around this impediment and displays the best option [15].

2.2. Steps of Modified Dijkstra's Algorithm

The MDA algorithm consists of the following steps:

Step 1: Establish the starting point for the first well.

Step 2: Compute the distances between every neighborhood point at the same time to find the second closest well.

• The distance will be taxed by an infinite value if there is an obstruction separating the two points. As a result, the algorithm starts looking for a different path that gets around the obstruction.

• Use the Euclidean distance formula to find the distance between two points if there isn't an obstruction.

• Determine which well is the second closest by calculating the separations between other parallel sites.

Step 3: By simultaneously calculating the distance between each neighborhood point, the algorithm keeps looking for the third well. The procedure is the same as it was in step 2.

Step 4: Look for the following requirements for constraints:

• No self-circuit for any well

• There is only one direction on the path connecting the points—not two.

• No path with a closed loop.

Step 5: Finish the search that connects every point to the terminus, which stands for the main station.

Step 6: Determine the total cost and distance that is the shortest.

Step 7: Display the best answer and halt the process.

IV. Simulation

This section will cover the Modified Dijkstra's method (MDA) for simulating random data. Employing the Google Earth application, random points were chosen, and their x- and y-axis coordinates which indicate the point's location were manually entered to create the random data, as illustrated in Figure 1. Data was set to 100, 500, 1000, 1500, and 2000 points to assess how well the MDA algorithm performed in determining the minimal total cost and demonstrating the speed at which the algorithm was implemented to produce the shortest path. The MDA method produces the best results for all datasets as far as total cost and running time, according to the comparison results in Table 1. However, GA required the longest run time to yield the optimal result. When compared to ACO, GA was able to find an improved option for the number of points 100 and 500 in terms of overall cost. In the meanwhile, when compared to GA, the ACO algorithm produced a better result for the numbers of points 1000, 1500, and 2000.

The results of the simulation demonstrated that the shortest path algorithm, which is a modified version of Dijkstra's algorithm, may drastically lower both the overall cost and computation time. Table 1 & Figure 2 show how well this technique works with arbitrarily large amounts of data. (Where RT stands for run time and TC stands for total cost.)

Table 1: Modified Dijkstra's Algorithm

No of Points	MDA		ACO		GA	
	TC ($)	RT (S)	TC ($)	RT (S)	TC ($)	RT (S)
100	771571599	1.59	721437429	38.57	638062487	968.55
500	1633525161	8.48	1672900349	250.29	1672712697	1157.93
1000	2248467589	23.65	2553898130	809.96	3408797284	1562.63
1500	3210539720	43.47	3728123743	1617.38	6932349303	1980.05
2000	3830218716	67.05	4591406791	1671.61	10306520252	2329.03

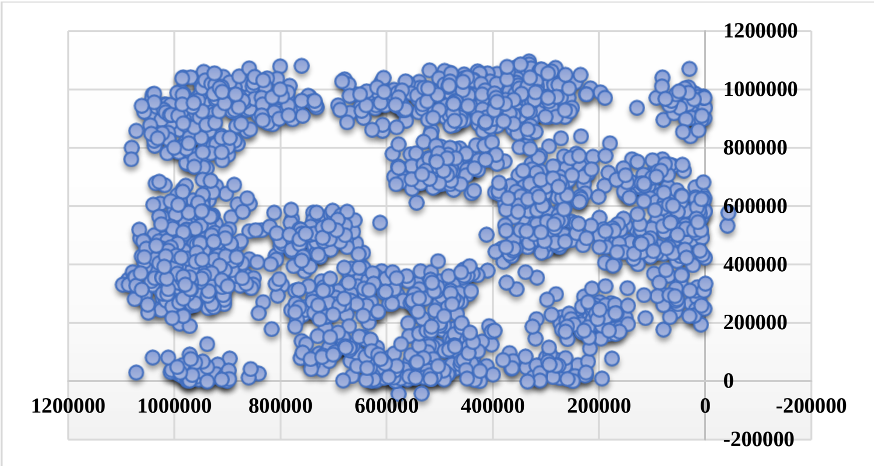

Figure 1: The locations of 2000 random points

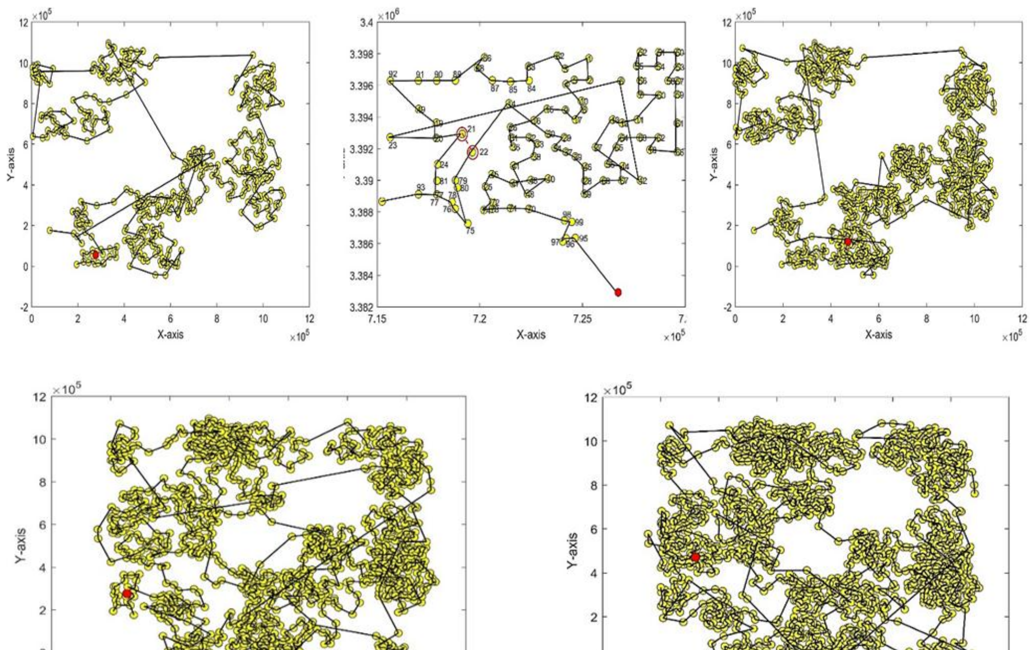

Figure 2: Using the MDA algorithm for varying numbers of random data points, the final shortest path of each scenario

International Journal of Engineering, Management and Humanities (IJEMH)
Volume 5, Issue 3, May.-June, 2024 pp: 214-220 ISSN: 2584-2145
www.ijemh.com

(a) Shortest path with the MDA algorithm for 100 poin. (b) Shortest path with the MDA algorithm for 500 point. (c) Shortest path with the MDA algorithm for 1000 point. (d) Shortest path with the MDA algorithm for 1500 point.
(e) Shortest path with the MDA algorithm for 2000 point

V. Performance of MDA Algorithm to handle obstacles:

Overcoming the difficulties that emerge in the network of pipelines is one of the issues that earlier research has not been able to resolve. No novel algorithm to tackle the hurdles could be proposed by any earlier study [5] [6]. The suggested MDA in this study was able to successfully avoid the pipeline network's obstructions. Thus, the performance of the MDA algorithm in managing various barriers is examined in this part. To do this, random barriers have first been injected into the original dataset. Table 2 presents five examples, with a data set of 100 points, of obstacles that arose along a predefined path.

The results of the suggested MDA method are displayed in Table 2, where the overall distance increases as the number of obstacles increases. Even though there seemed to be more barriers in the pipeline network, the algorithm was still able to generate the shortest path in less run time. The locations of the barriers and the alternative route designed to avoid them are shown in Figures 3 through 7.

Table 2: Analysis of obstacles appeared on the path of 100 points

No. of Point	Number of obstacles	Total Distance	Cost ($)	Run Time (s)
100	1- (21,22)	123310.83	12,331,083	1.44
	2- (21,22),(74.72)	124120.03	12,412,003	1.44
	3- (21,22),(74.72),(22,14)	126752.61	12,675,261	1.43
	4- (21,22),(74,72),(22,14),(86,89)	128181.57	12,818,157	1.49
	5-(21,22),(74,72),(22,14),(86,89),(67,44)	124421.00	12,442,100	1.42

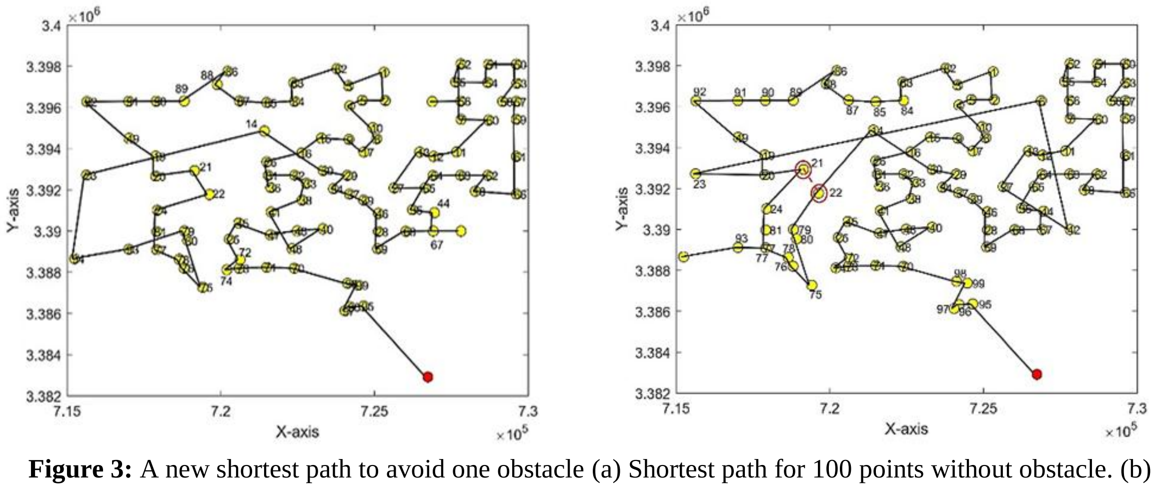

Figure 3: A new shortest path to avoid one obstacle (a) Shortest path for 100 points without obstacle. (b) Shortest path for 100 points with one obstacle. Total distance (123310.83).

International Journal of Engineering, Management and Humanities (IJEMH)
Volume 5, Issue 3, May.-June, 2024 pp: 214-220 ISSN: 2584-2145
www.ijemh.com

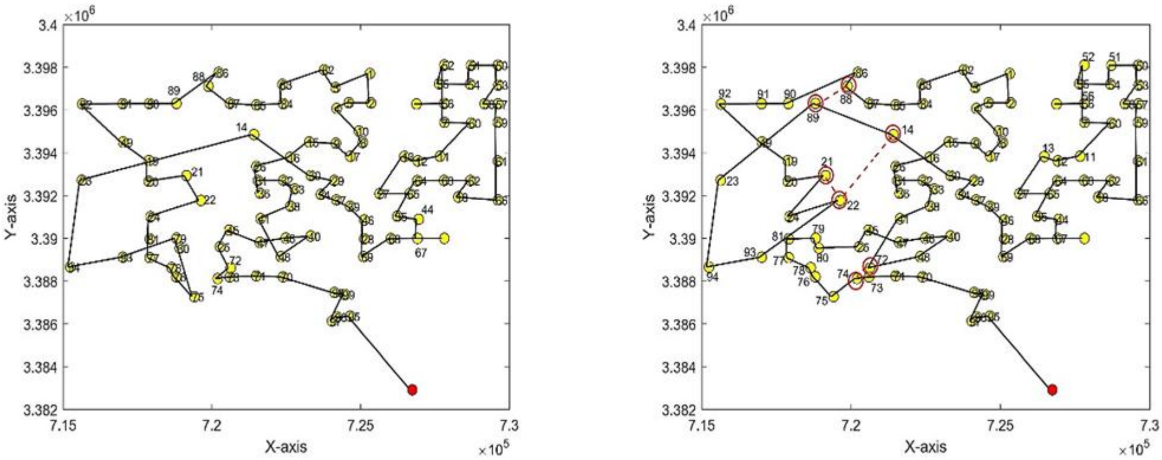

Figure 4: A new shortest path to avoid two obstacles(a) Shortest path for 100 points without obstacle. Total distance (123310.83)(b) Shortest path for 100 points with two obstacles. Total distance (124120.03).

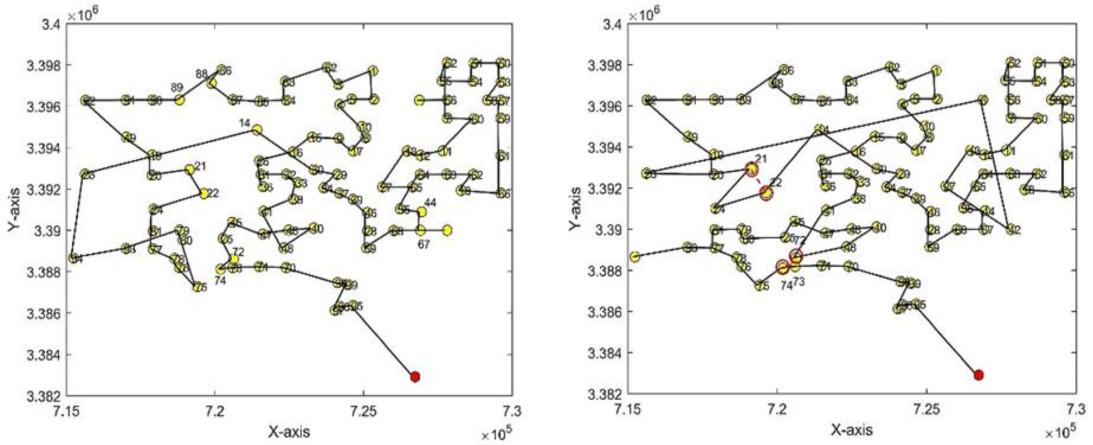

Figure 5: A new shortest path to avoid three obstacles(a) Shortest path for 100 points without obstacle. Total distance (123310.83)(b) Shortest path for 100 points with three obstacles. Total distance (126752.61)

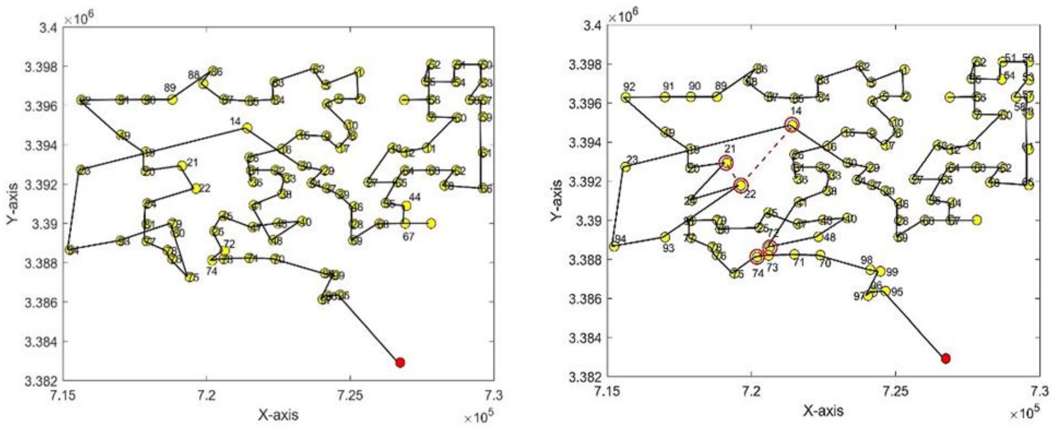

Figure 6: A new shortest path to avoid four obstacles.(a) Shortest path for 100 points without obstacle. Total distance (123310.83).(b) Shortest path for 100 points with four obstacles. Total distance (128181.57)

International Journal of Engineering, Management and Humanities (IJEMH)
Volume 5, Issue 3, May.-June, 2024 pp: 214-220 ISSN: 2584-2145
www.ijemh.com

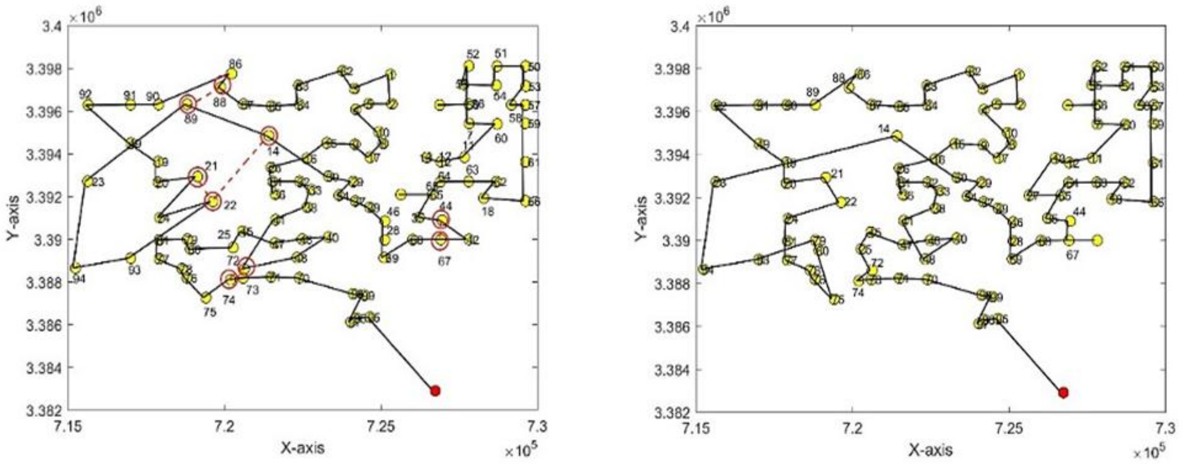

Figure 7: A new shortest path to avoid five obstacles.(a) Shortest path for 100 points without obstacle. Total distance (123310.83),(b) Shortest path for 100 points with five obstacles. Total distance (124421.00).

VI. Conclusions

To minimize the cost of building the pipeline network, this study suggests modifying Dijkstra's algorithm to find the pipeline's shortest path. The total cost and total duration needed for implementing the algorithm are the most crucial evaluation criteria. After running several simulations, it was discovered that the two most significant factors influencing this optimization were the amount of data and the distance that stood between the starting and destination positions. A randomized data network consisting of 2000 locations that needed to be connected via the shortest path was subjected to the new technique. The outcomes of the experiment demonstrated that the altered algorithm could find outstanding results in a record-breaking two minutes. However, GA required the longest possible run time to yield the optimal result. While compared to GA, the ACO algorithm produced a better total cost solution for the numbers of points 1000, 1500, and 2000. In the meantime, when compared to ACO, GA found the optimal solution for the numbers 100 and 500.Among the traditional greedy algorithms is Dijkstra's algorithm. It was used to find the graph's shortest path; however, the path discovered by Dijkstra's algorithm might be more expensive to build or might have obstructions that raise construction costs. Based on this, a shortened path calculation for huge data sets has been developed by modifying the traditional Dijkstra's algorithm. The outcomes demonstrated that the altered algorithm performs exactly as predicted.

References

[1]. Almaalei, N. N., &Razali, S. N. A. M. (2019). Performance Evaluation of Different Short Path Algorithms to Improve Oil-Gas Pipelines. International Journal of Advanced Computer Science and Applications, 10(11), 1-6.

[2]. TurkiJalil, A., Emad Al Qurabiy, H., Hussain Dilfy, S., Oudah Meza, S., Aravindhan, S., M Kadhim, M., & M Aljeboree, A. (2021). CuO/ZrO2 nanocomposites: facile synthesis, characterization and photocatalytic degradation of tetracycline antibiotic. Journal of Nanostructures, 11(2), 333-346.

[3]. Almaalei, N. N., Razali, S. N. A., & Mohammed Alduais, N. A. (2018). An efficient algorithm to improve oil gas pipelines path. International Journal of Engineering & Technology, 7(4), 5412-5418.

[4]. Mezan, S. O., Al Absi, S. M., Jabbar, A. H., Roslan, M. S., &Agam, M. A. (2021). Synthesis and characterization of enhanced silica nanoparticle (SiO2) prepared from rice husk ash immobilized of 3-(chloropropyl) triethoxysilanea. Materials Today: Proceedings, 42, 2464-2468.

[5]. Magzhan, K., &Jani, H. M. (2013). A review and evaluations of shortest path algorithms. International journal of scientific & technology research, 2, 99-104.

[6]. Gbadamosi, O. A., &Aremu, D. R. (2020). Design of a Modified Dijkstra's Algorithm for finding alternate routes for shortest-path

problems with huge costs. IEEE International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), 10, 1-6.

[7]. Hasan Jabbar, A., Oudah Mezan, S., Alameri, A. A., Prakaash, A. S., Mohammed Baqir Al-Dhalimy, A., &Younes, A. (2024). Fe3O4@ Diamine-CuI Nanocomposite: A Novel and Highly Reusable Nanomagnetic Catalyst for Ecofriendly Synthesis of TriarylImidazoles. Polycyclic Aromatic Compounds, 44(2), 1309-1325.

[8]. Chu, F., & Chen, S. (2012). Optimal design of pipeline based on the shortest path. Physics Procedia. 2012 International Conference on Medical Physics and Biomedical Engineering, 33, 216–220.

[9]. Altajer, A. H., Efendi, S., Jabbar, A. H., Oudah Mezan, S., Thangavelu, L., M Kadhim, M., & F Alkai, A. M. (2021). Novel carbon quantum dots: Green and facile synthesis, characterization and its application in on-off-on fluorescent probes for ascorbic acid. Journal of Nanostructures, 11(2), 236-242.

[10]. Shahadat, M., Oves, M., Shalla, A. H., Ahammad, S. Z., Wazed Ali, S., &Sreekrishnan, T. R. (2020). Fabrication of polyaniline supported nanocomposites and their sensing application for detection of environmental pollutants. Modern Age Waste Water Problems: Solutions Using Applied Nanotechnology, 93-147.

[11]. Mezan, S. (2021). Silica/Polystyrene Nanocomposite (SiO2/PsNCs) using Sol-Gel Method Drug Delivery System: A Review. Design Engineering, 10175-10181.

[12]. Li, F., Liu, Q., Guo, X., & Xiao, J. (2015). A survey of optimization method for oil-gas pipeline network layout. International Conference on Mechatronics, Electronic, Industrial and Control Engineering, 257–260.

[13]. Mezan, S. O.Faraj Lakatan Kzar. Anwar Khairi Abed. (2024). Synthesis and Characterization of Zinc Oxide /polyaniline (ZnO/PANI) Nanocomposite for solar cell applicationInternational Journal of Engineering, Management and Humanities (IJEMH) Volume 5, Issue 3, May. -June, 2024 pp: 145-150 www.ijemh.com ISSN: 2584-2145.

[14]. Wei, K., Gao, Y., Zhang, W., & Lin, S. (2019, March). A Modified Dijkstra's Algorithm for Solving the Problem of Finding the Maximum Load Path. In 2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT), 10-13.

[15]. Mezan, S. O. Faraj Lakatan Kzar. Anwar Khairi Abed. (2024). The Effect of Annealing on the Structural, Optical and Electrical Properties of SnO2Thin Film Prepared by Chemical Pyrolysis. International Journal of Engineering, Management and Humanities (IJEMH) Volume 5, Issue 3, May. -June, 202 4 pp: 135-144 ISSN: 258