IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⁜ Impact Factor 8.066 ⁜ Peer-reviewed / Refereed journal ⁜ Vol. 10, Issue 5, May 2023

DOI: 10.17148/IARJSET.2023.10540

Reliability and Prevention of Wireless Sensor Networks using Dijkstra's Algorithm

Dr.B.Rajesh Shyamala Devi[1], Sudha Naveen[2], Inkollu Venkata Pavan Kalyan Reddy[3]

Department of Electronics and Communication Engineering, Hindustan Institute of Technology and Science

Chennai, India[1]

Department of Electronics and Communication Engineering, Hindustan Institute of Technology and Science

Chennai, India[2]

Department of Electronics and Communication Engineering, Hindustan Institute of Technology and Science

Chennai, India[3]

Abstract—WSNs are employed extensively in a broad spectrum of fields, including surveillance, healthcare, and ecological monitoring. As WSNs are frequently deployed in harsh and distant areas where maintenance is challenging or impossible, reliability is a critical component. Therefore, it's critical to create WSNs that are dependable and efficient even under challenging circumstances. In this project, we suggest using Dijkstra's method to increase the WSNs' dependability. A renowned graph-based approach for determining the shortest path across two points in a network is called Dijkstra's algorithm. With the aid of this method, we can improve the energy efficiency and longevity of WSNs by optimizing their routing patterns. To increase the dependability of WSNs, we will additionally be looking at preventive strategies. The tactics used in these preventative strategies will include redundancy, fault-tolerance, and error correction. By putting these strategies into practice, we can stop network failures from happening and make sure it keeps working even in challenging circumstances. To determine the success of the suggested strategy, simulations will be used, and the results will be compared to those of other current routing protocols. The outcomes of this study will be beneficial for creating dependable WSNs which can be employed in a variety of scenarios.

Keywords—WSNs, Dijkstra's method, routing patterns, dependability, shortest path, longevity, and energy efficient.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) have become an important research topic in recent years due to their wide range of applications [5]. They are typically used in applications such as environmental surveillance, industrial engineering and control, healthcare monitoring, and military operations. WSNs composed of an ample of minute, low power consuming sensors which are deployed in a particular area and can communicate with each other to transmit data to a central node or base station. The sensors are usually battery-powered and have a limited lifetime, making it difficult to replace or recharge them in some applications. Therefore, it is essential to develop reliable WSNs that can operate under various conditions. [6] One of the most significant challenges in designing WSNs is ensuring their reliability. Due to the harsh environments and limited resources, WSNs are more susceptible to failures than traditional wired networks. In WSNs, the nodes are typically deployed in remote and inaccessible areas, which makes maintenance difficult. Therefore, it is necessary to develop reliable WSNs that can withstand harsh environmental conditions and provide accurate data.

To address this challenge, this project proposes the use of Dijkstra's algorithm to improve the reliability of WSNs. [7] To determine the most efficient path across a pair of nodes in a distributed system, experts utilize the popular trend line-based method Dijkstra's algorithm. By optimizing the routing paths using Dijkstra's algorithm, the energy consumption of WSNs can be reduced, and the network's lifetime can be increased.

The proposed approach aims to reduce the number of network packets and to minimize the energy consumption of nodes, which is essential for extending the lifetime of WSNs. The proposed approach also includes prevention methods such as redundancy, fault-tolerance, and error correction. Redundancy is the use of additional nodes to provide backup and ensure continuity of service in case of node failure. [8] Fault-tolerance is the ability of the network to continue functioning in the presence of faults or failures. Error correction is the process of detecting and correcting errors in the data transmitted by the nodes.

DOI: 10.17148/IARJSET.2023.10540

To evaluate the proposed approach, simulations will be conducted using the NS2 simulator. It is possible for academics to replicate the functioning of complicated networks using the well-known discrete event network simulator "NS2" and analyse their performance under various conditions. [9] In the simulation, the proposed approach will be compared with other existing routing protocols to demonstrate its effectiveness. In the simulated environment, the efficacy of the suggested strategy will be evaluated in terms of network lifetime, energy usage, ratio of packet delivery, and delay.

One of the significant advantages of using Dijkstra's algorithm in WSNs is that it can reduce the number of network packets, which is essential for extending the lifetime of WSNs. The algorithm optimizes the routing paths, which decreases the nodes' utilization of energy and the amount of data packets transmitted. [10] This approach can also reduce network congestion, which can result in a higher packet delivery ratio and lower delay. Another advantage of using Dijkstra's algorithm is its scalability.

The algorithm is suitable for large-scale networks and can handle a high number of nodes. This feature makes it an ideal choice for WSNs, which typically consist of a large number of nodes. [11] In addition, the algorithm can be easily integrated with other routing protocols to enhance their performance. In conclusion, this project proposes the use of Dijkstra's algorithm to improve the reliability of WSNs. The algorithm optimizes the routing paths, which reduces energy consumption, packet loss, and delay, thereby increasing the network lifetime. The proposed approach also includes prevention methods such as redundancy, fault-tolerance, and error correction to prevent failures and ensure continuity of service. The proposed approach will be evaluated using simulations in the NS2 simulator, which will demonstrate its effectiveness in improving the reliability of WSNs.

II. LITERATURE SURVEY

The monitoring of many circumstances, including smart cities, healthcare, and military operations, is done via wireless sensor networks (WSNs). To provide a high level of service in WSNs, network coverage optimization is a crucial goal. Ant colony optimization methods are one method for improving coverage. [1] The application of ant colony optimization techniques in WSNs has been examined in a number of research. One such study suggested an algorithm for coverage optimization based on ant colonies. The method maximizes the shortest distance between any two sensors to maximize coverage while using ants to choose the optimum sites for sensor nodes. The findings from simulations demonstrated that in terms of network lifespan and coverage, the suggested method performed better than the current coverage optimization techniques.

An improved ant colony optimization technique was put out in another work to address nonlinear resource levelling issues. [2] The method tries to maximize resource utilization in WSNs by effectively distributing resources like bandwidth and energy. Ant colony optimization is a technique the program uses to find the best distribution of resources. According to the findings, the suggested algorithm outperforms existing techniques in terms of convergence rate and solution quality.

[3] A fault-tolerant embedding approach for node failure in airborne tactical network virtualization was suggested in third research. The technique incorporates virtual nodes into the network to increase the dependability of WSNs. In order to make sure that the network stays linked in the event of node failures, the method employs ant colony optimization to choose the optimal placements for virtual nodes. According to the simulation findings, the suggested technique performs better in terms of network connection than the currently used embedding strategies.

A strategy to increase the effectiveness of [4] ant colony optimization algorithms for resolving big Traveling Salesman Problem (TSP) cases was put out in fourth research. The suggested approach uses a local search algorithm to raise the calibre of the results produced by ant colony optimization. The simulation results shown that, in terms of solution quality and computational efficiency, the suggested strategy performs better than the alternatives.

According to the literature review, Dijkstra's technique was observed to be far superior in terms of reliability and failure mitigation, even though Ant Colony Optimization (ACO) algorithms have shown promise in optimizing coverage and resource allocation in Wireless Sensor Networks (WSNs). This is extremely important to our research since it uses Dijkstra's algorithm to increase the dependability and prevent failures in WSNs. As a result, our choice of algorithm is consistent with the results of earlier research and is probably going to increase functionality and longevity of networks.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⵋ Impact Factor 8.066 ⵋ Peer-reviewed / Refereed journal ⵋ Vol. 10, Issue 5, May 2023

DOI: 10.17148/IARJSET.2023.10540

III. METHODOLOGY

The commonly used graph discovery technique Dijkstra's algorithm determines the shortest link between two nodes in a graph. In this study, the shortest possible path amongst nodes in a wireless sensor network (WSN) will be determined using Dijkstra's method. [12] This algorithm's effectiveness, accuracy, and relatively easy implementation make it ideal for our needs.

To implement Dijkstra's algorithm in our WSN, we will be using the NS2 simulator. [13] NS2 is a popular network simulator that allows us to model and simulate various types of networks, including wireless sensor networks. Using NS2, we can create a simulation environment that accurately reflects the behaviour and characteristics of our WSN. The proposed block diagram is shown below in Fig. 1.

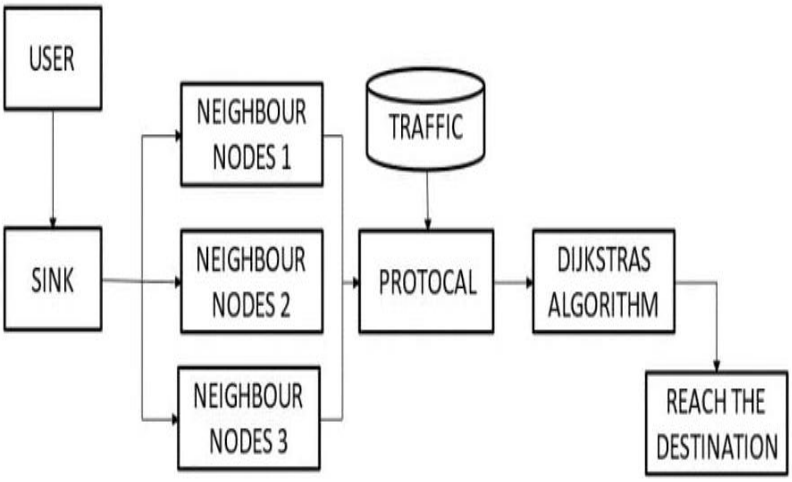

Fig. 1. Block diagram of the proposed model

The first step in implementing Dijkstra's algorithm is to construct a graph that represents the WSN. [14] In our case, the nodes in the WSN will be represented as vertices in the graph, and the edges between the vertices will represent the communication links between the nodes. The weights assigned to each edge will represent the cost of transmitting data between the nodes. All the nodes 50 in count in our case are initialized with their costs as shown in Fig. 2.

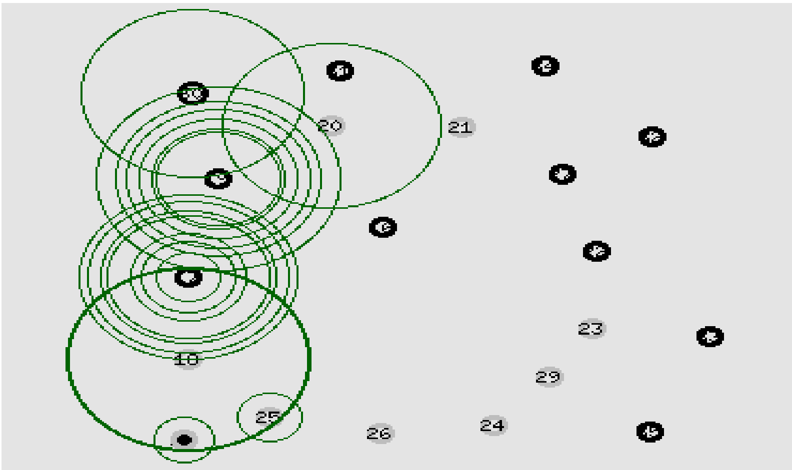

Fig. 2. Initializing the nodes

After creating the graph, we can start using Dijkstra's method to determine the shortest route between the network's nodes. The plot with the source and destination along with their cost are allocated as shown in Fig. 3. In order to find the shortest path.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ∺ Impact Factor 8.066 ∺ Peer-reviewed / Refereed journal ∺ Vol. 10, Issue 5, May 2023

DOI: 10.17148/IARJSET.2023.10540

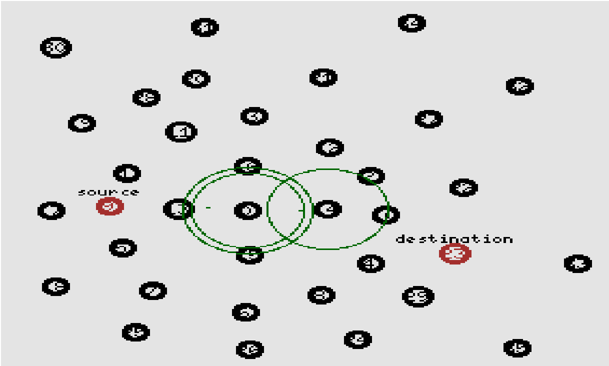

Fig. 3. Allocation of source and destination

This entails creating a table that has a collection of unvisited nodes as well as the distance between each node and the source node for each node. [2] The node with the least distance value is then chosen repeatedly and designated as visited. We continue the procedure until all nodes have been visited, updating the distance table to reflect the recently found shortest pathways mentioned as attack nodes as shown in Fig. 4. and Fig. 5.

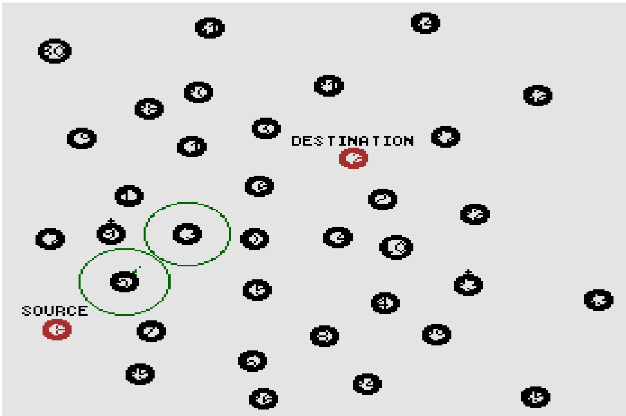

Fig. 4. Multiple Iterations

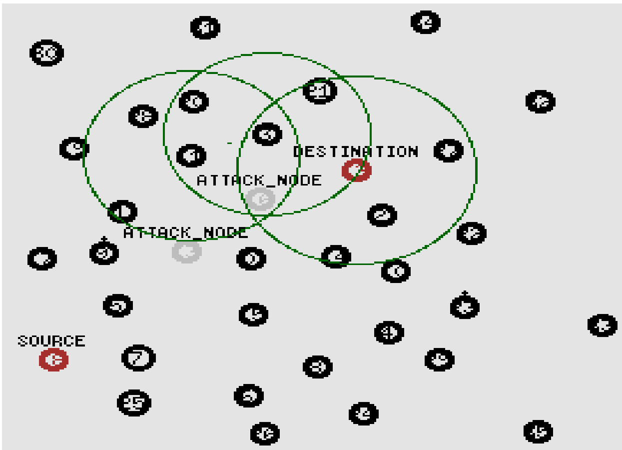

Fig. 5. Shortest path attack nodes

In our simulation, we will be using Dijkstra's algorithm to optimize the routing of data packets between nodes in the network. [7] By finding the shortest path between nodes, we can reduce the latency and energy consumption associated with transmitting data over longer distances. We will also be using NS2 to evaluate the performance of our WSN under various conditions, such as varying numbers of nodes, varying data rates, and varying traffic loads. Overall, the combination of Dijkstra's algorithm and NS2 provides a powerful tool for optimizing the performance of wireless sensor networks. By accurately modelling the behaviour of the network and applying efficient algorithms like Dijkstra's, we can improve the reliability, efficiency, and longevity of our WSN.

IARJSET ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

ISO 3297:2007 Certified ⋇ Impact Factor 8.066 ⋇ Peer-reviewed / Refereed journal ⋇ Vol. 10, Issue 5, May 2023

DOI: 10.17148/IARJSET.2023.10540

IV. RESULTS AND DISCUSSION

Based on the graphs, it is evident that Dijkstra's algorithm outperforms ACO in terms of energy consumption, packet delivery ratio, and delay. The red line in all three graphs represents ACO, while the green line represents Dijkstra's algorithm.

In the first graph shown in Fig. 6., which represents energy consumption, the green line (Dijkstra's) is consistently lower than the red line (ACO), indicating that Dijkstra's algorithm is more energy efficient. This is an essential aspect of WSNs as the nodes often have limited energy resources, and conserving energy is crucial to extend the network's lifetime.

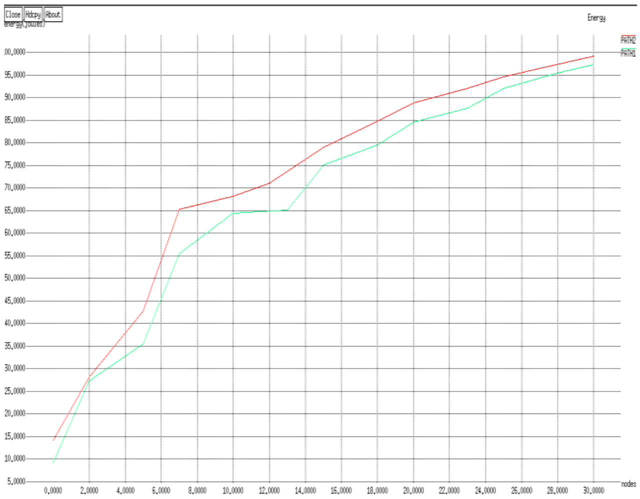

Fig. 6. Energy consumption graph

The second graph shown in Fig. 7 represents the packet delivery ratio, where the green line (Dijkstra's) is consistently higher than the red line (ACO), indicating that Dijkstra's algorithm is more reliable in delivering packets. Packet delivery ratio is a critical performance metric as the sensor nodes often send data to a central base station, and data loss can cause significant problems in the network.

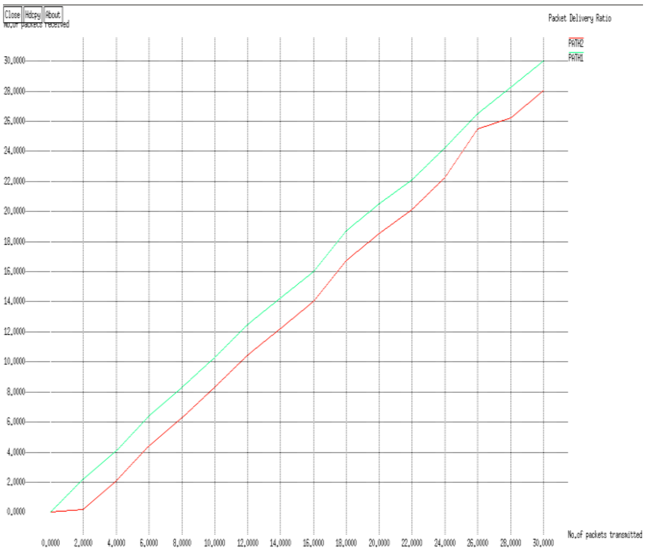

Fig. 7. Packet delivery ratio

The third graph shown in Fig. 8 represents delay, where the green line (Dijkstra's) is consistently lower than the red line (ACO), indicating that Dijkstra's algorithm has less delay in packet transmission. This means that Dijkstra's algorithm is more suitable for real-time applications that require low latency, such as video streaming or voice communication.

DOI: 10.17148/IARJSET.2023.10540

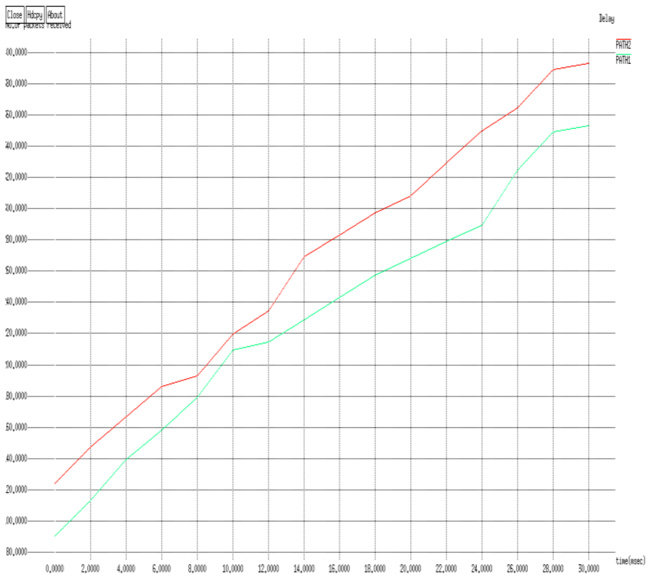

Fig. 8. Delay

Overall, the results suggest that Dijkstra's algorithm is a more effective and efficient approach for resource allocation and routing in WSNs than ACO. The superior performance of Dijkstra's algorithm can be attributed to its deterministic nature and the fact that it considers the shortest path only, whereas ACO is a stochastic algorithm that explores multiple paths simultaneously, which can result in suboptimal solutions.

V. CONCLUSION

In conclusion, the results of the study indicate that Dijkstra's algorithm is a better choice for resource allocation and routing in WSNs than ACO. The proposed system is simulated under various circumstances which has 50 nodes and excelled with its performance in terms of Energy, Latency (Delay) and Packet Delivery ratio. It is crucial to remember, nevertheless, that the effectiveness of the algorithms may differ based on the particular application and network conditions. This work paves a way to make reliable data transfer at faster speeds with help of the Dijkstra's algorithm. This approach can be adapted in hybrid data transfer tools which is a reliable and Easy to incorporate with various forms of signals. Further research is needed to investigate the algorithms' performance in different scenarios and optimize them further to enhance their performance.

REFERENCES

[1] M. Khoshrangbaf, V. K. Akram, and M. Challenger, “Ant colony based coverage optimization in wireless sensor networks,” in Communication Papers of the 17th Conference on Computer Science and Intelligence Systems, 2022.

[2] J.-Q. Geng, L.-P. Weng, and S.-H. Liu, “An improved ant colony optimization algorithm for nonlinear resource-leveling problems,” Comput. Math. Appl., vol. 61, no. 8, pp. 2300–2305, 2011.

[3] J. Miao, N. Lv, Q. Gao, K. Chen, and X. Wang, “Fault-tolerant embedding algorithm for node failure in airborne tactical network virtualization,” IEEE Access, vol. 10, pp. 60558–60571, 2022.

[4] R. Skinderowicz, “Improving Ant Colony Optimization efficiency for solving large TSP instances,” Appl. Soft Comput., vol. 120, no. 108653, p. 108653, 2022.

[5] J. Lei and X. Tian, “Application of an improved ant colony algorithm in routing of wireless sensor networks,” Chem. Eng. Trans.

[6] V. M. Mualuko, P. K. Kihato, and V. Oduol, “Routing optimization for wireless sensor networks using fuzzy ant colony,” Ripublication.com. [Online]. Available: https://www.ripublication.com/ijaer17/ijaerv12n21_151.pdf. [Accessed: 24-Apr-2023].

[7] R. Misra and C. Mandal, “Ant-aggregation: ant colony algorithm for optimal data aggregation in wireless sensor networks,” in 2006 IFIP International Conference on Wireless and Optical Communications Networks, 2006, p. 5 pp. – 5.

[8] J. He, Z. Teng, and F. Zhang, “Research on power control routing algorithm for wireless sensor networks based on ant colony optimization,” 2022.

[9] S. K. Sharma and B. L. Pal, “Shortest path searching for road network using A* algorithm,” Ijcsmc.com. [Online]. Available: https://ijcsmc.com/docs/papers/July2015/V4I7201599a23.pdf. [Accessed: 25-Apr-2023].

DOI: 10.17148/IARJSET.2023.10540

[10] S. O. James and A. S. Chandran, "Comparitive analysis on shortest path computation in gps systems," Ijcrt.org. [Online]. Available: https://ijcrt.org/papers/IJCRT2009225.pdf. [Accessed: 25-Apr-2023].
[11] Vadivu, "Shortest Path of a Graph using Centrality Measures," Int. J. Adv. Trends Comput. Sci. Eng., vol. 9, no. 3, pp. 3898–3903, 2020.
[12] M. Zhou and N. Gao, "Research on Optimal Path based on Dijkstra Algorithms," in Proceedings of the 3rd International Conference on Mechatronics Engineering and Information Technology (ICMEIT 2019), 2019, pp. 884–892.
[13] A. Madkour, W. G. Aref, F. U. Rehman, M. A. Rahman, and S. Basalamah, "A survey of shortest-path algorithms," arXiv [cs.DS], 2017.
[14] X. Yin and J. Yang, "Shortest paths based Web service selection in Internet of things," J. Sens., vol. 2014, pp. 1–10, 2014.

