DYNAMIC MINIMUM SPANNING TREE ROUTING FOR MULTI-AGENT SYSTEMS USING REAL-TIME GPS UPDATES

Elda Rayhana [1], M. Ikrar Yamin [2], Riadi Marta Dinata [3], Niko Purnomo [4]
[1] Study Program of Physics, Faculty of Applied Science and Technology, ISTN Jakarta
[2] Study Program of Electrical Engineering, Faculty of Applied Science and Technology, ISTN Jakarta
[3,4] Study Program of Informatics Engineering, Faculty of Applied Science and Technology, ISTN Jakarta
Jl. Moch. Kahfi II No. 30, Srengseng Sawah, Jagakarsa, South Jakarta 12640, Indonesia
adiarray@istn.ac.id

ABSTRACT

Infrastructure-less mobile networks are increasingly important for applications such as disaster response, autonomous coordination, and remote field operations. However, maintaining efficient communication routes in such environments is challenging because node mobility continuously changes network topology. This study addresses the problem of maintaining stable and efficient routing among mobile nodes with limited computational capability. The objective of this research is to design and evaluate a real-time routing system for mobile networks using Minimum Spanning Tree (MST) optimization. The proposed system integrates GPS-enabled ESP32 nodes with a centralized server that computes network topology using Kruskal's MST algorithm. A client–server architecture is implemented in which mobile nodes periodically send location updates to the server. The server constructs a weighted graph, computes the MST, and distributes routing instructions back to each node. Experiments were conducted under three mobility scenarios—static, linear motion, and random walk producing a total of 2,700 position updates. The results show that the system maintains server computation times below 10 ms and end-to-end latency between 187–223 ms. The Tree Stability Index reached 1.000 in static conditions, 0.947 during linear motion, and 0.683 in random mobility. These results demonstrate that the proposed approach provides efficient and responsive routing for mobile networks operating without communication infrastructure.

Keyword: *Minimum Spanning Tree, ESP32, Mobile Networks, GPS Routing, Infrastructure-Less Communication*

1. INTRODUCTION

The ability to establish autonomous and flexible communication networks has become essential in situations where conventional infrastructure is unavailable, damaged, or impractical to deploy. Critical scenarios such as disaster response, military field coordination, and logistical operations in remote environments require communication systems that can maintain connectivity despite constant changes in node positions and the absence of supporting infrastructure [1]. Mobile Ad Hoc Networks and Vehicular Ad Hoc Networks provide a foundation for such environments by enabling direct peer-to-peer communication among mobile units including vehicles, robots, or personnel [2]. However, maintaining stable and efficient routing within these networks remains a fundamental challenge due to unpredictable mobility patterns and dynamically shifting topologies [15].

Graph theory offers a rigorous mathematical framework for addressing topological problems in dynamic communication networks. One of its most fundamental concepts, the Minimum Spanning Tree, enables the construction of a communication backbone that connects all nodes with the minimum possible total edge weight, thereby reducing communication overhead and energy consumption [3], [4], [5]. Recent studies have shown that MST-based routing approaches can improve efficiency and reduce computational burden in highly dynamic routing environments, making them suitable for mobile robotic systems and vehicular coordination [6], [13], [19]. Nevertheless, many existing works remain limited to simulation-only evaluations or focus on simplified hardware trials that do not capture full real-time dynamics [14], [17].

Parallel advancements in the Internet of Things ecosystem have introduced a new generation of low-cost microcontrollers such as the ESP32, which offer integrated WiFi connectivity, low power consumption, and reliable performance suitable for field deployments [7], [8]. The integration of GPS modules with ESP32 devices has enabled real-time location tracking in a wide range of applications including vehicle monitoring, mobile robotics, and environmental sensing [9], [10]. Despite these developments, limited research has explored the seamless integration of MST-based dynamic topology optimization with real hardware systems capable of operating in infrastructure-less outdoor environments.

This research addresses the gap by designing and evaluating a dynamic routing system based on Minimum Spanning Tree computation for mobile ESP32 nodes equipped with GPS receivers. The proposed system follows a client-server architecture in which mobile nodes transmit their coordinates to a central server, which constructs a weighted undirected graph, computes its MST in real-time, and returns local routing instructions to each node. The system is intended to operate without reliance on cellular or internet infrastructure, making it suitable for field operations and emergency deployments. The research aims to evaluate three key aspects: the efficiency of bidirectional WiFi-tethered communication in maintaining low route-update latency, the computational performance and responsiveness of server-side MST calculation under dynamic mobility conditions, and the clarity of local route visualization using a simple LCD module.

By combining graph-based optimization with low-cost IoT hardware, this study provides a practical and replicable solution for multi-agent coordination in infrastructure-less environments. The findings contribute to the advancement of field-deployable mobile networks that are robust, affordable, and capable of real-time adaptation [12].

2. LITERATUR REVIEW

Literature review provides an overview of previous studies related to mobile ad hoc communication, graph-based routing optimization, and Internet of Things hardware used for mobile network implementations. The review highlights how Minimum Spanning Tree algorithms, GPS-enabled IoT devices, and wireless communication frameworks have been applied in dynamic networking environments. This section also identifies the research gap addressed by the proposed system.

2.1. Mobile Ad Hoc and Infrastructure-less Networks

Mobile Ad Hoc Networks (MANET) enable communication among mobile devices without relying on fixed infrastructure such as cellular towers or access points. Each node in a MANET functions both as a host and as a router, forwarding packets for other nodes in the network. Because node positions change dynamically, the network topology must constantly adapt to maintain connectivity and routing efficiency. Recent studies indicate that conventional routing protocols often experience scalability and stability problems when node mobility becomes highly dynamic [21]. Vehicular Ad Hoc Networks (VANET) represent a specialized form of MANET designed for communication between vehicles and roadside units. These networks are widely used in intelligent transportation systems and cooperative mobility applications. However, frequent topology changes caused by vehicle movement create challenges in maintaining stable routing structures and minimizing communication overhead [22].

2.2. Minimum Spanning Tree in Network Routing

Graph theory plays a significant role in designing efficient routing structures for dynamic networks. One commonly used structure is the Minimum Spanning Tree (MST), which connects all nodes in a graph while minimizing the total edge weight. In communication networks, the edge weight may represent physical distance, transmission cost, or signal strength. Recent studies show that MST-based routing approaches can reduce redundant communication links and improve energy efficiency in wireless networks. Algorithms such as Kruskal's and Prim's methods are frequently used to compute MST structures due to their relatively low computational complexity and deterministic outcomes. Several studies have applied MST techniques to wireless sensor networks and cooperative robotic systems to maintain efficient network topology under changing node positions [23]. However, many existing MST routing approaches are evaluated primarily through simulation rather than real-world implementations. Consequently, practical considerations such as hardware limitations, communication latency, and real-time processing constraints remain insufficiently explored.

2.3. Internet of Things Devices for Mobile Networking

The rapid development of the Internet of Things (IoT) has introduced compact microcontrollers capable of wireless communication and real-time sensing. Devices such as the ESP32 have become widely used due to their integrated WiFi capability, low power consumption, and relatively high processing performance compared with earlier microcontrollers. Several studies have utilized ESP32 devices for applications including environmental monitoring, smart transportation systems, and real-time tracking systems. The integration of GPS modules with IoT devices enables location-aware applications where node mobility can be continuously monitored. For example, GPS-based vehicle tracking systems using ESP32 have demonstrated reliable performance with relatively low hardware cost, making them suitable for field deployments and experimental prototypes [24]. Despite these advantages, the computational limitations of microcontroller devices remain a challenge for complex network algorithms. Many graph-based optimization techniques require memory and processing power beyond what typical embedded devices can provide.

2.4. Centralized Routing Architecture for IoT Networks

To overcome hardware limitations in embedded devices, several studies propose a hybrid architecture in which computational tasks are centralized on a server while edge devices perform sensing and communication tasks. In such architectures, IoT nodes transmit sensor data or location information to a central processing unit that performs complex analysis or routing calculations. This approach has been successfully applied in distributed robotic systems and wireless sensor networks. Centralized computation enables the use of advanced graph algorithms and real-time topology optimization while maintaining lightweight firmware on individual nodes. Research indicates that combining centralized processing with lightweight client nodes can significantly reduce computational load on edge devices while maintaining low latency communication [25].

2.5. Research Gap and Contribution

Based on the literature discussed above, several research gaps remain. First, many MST-based routing approaches are validated only through simulation and lack real-world hardware implementations. Second, studies integrating IoT devices with graph-based routing algorithms rarely evaluate system performance under dynamic mobility conditions. Third, practical demonstrations of infrastructure-less communication systems using low-cost microcontrollers are still limited.

Therefore, this research proposes a real-time routing system that integrates Minimum Spanning Tree computation with GPS-enabled ESP32 nodes operating in an infrastructure-less environment. By combining graph-based topology optimization with a centralized server architecture, the proposed system aims to provide efficient and scalable routing for mobile multi-agent networks.

3. METHODOLOGY

The Methods section encompasses problem analysis and the architectural or design methods employed to address the problems. Problem analysis involves identifying the issues that exist and explaining how they are resolved in this study. The design aspect demonstrates the solution to the problem, which should be illustrated with diagrams and accompanied by thorough explanations. For instance, this could include data processing diagrams that show the flow from raw data to the final product, or hardware design diagrams.

3.1. Problem Analysis

Communication in mobile ad hoc environments is challenged by constantly shifting topologies, unstable wireless links, and limited computational resources on embedded devices [1]. Conventional routing protocols often struggle to maintain low overhead as node positions change unpredictably. To address these issues, the research evaluated the feasibility of implementing Minimum Spanning Tree routing, a graph-theoretic structure that minimizes total communication cost by connecting nodes with the smallest cumulative distance [3]. Real-time computation of MST requires efficient algorithms such as Kruskal's method [4] or Prim's approach [5], both of which remain computationally lightweight when executed on a dedicated server [16].

While dynamic MST routing has been explored in simulation studies [6], few implementations integrate real embedded hardware, GPS-based mobility, and continuous server-side computation. Moreover, low-cost microcontrollers such as the ESP32 have limited memory and processing capability, preventing them from executing complex graph algorithms locally [7]. These constraints motivate a hybrid design wherein computationally intensive tasks are delegated to a server, while mobile nodes focus on sensing, communication, and local visualization.

3.2. System Architecture

The proposed architecture adopts a centralized client-server model to balance computation and communication efficiency. The system consists of:

a. Mobile Nodes (Clients), five ESP32-WROOM devices equipped with NEO-6M GPS modules and LCD 16×2 displays. Each node transmits position updates over WiFi tethering and receives routing instructions generated by the server.
b. Server, a Python-based backend that aggregates GPS coordinates, constructs a weighted undirected graph, computes MST using Kruskal's algorithm, and distributes per-node routing information.
c. Communication Medium, all devices connect to a local WiFi hotspot, which requires no external internet connectivity. This design supports full offline operation, making the system suitable for field deployments.

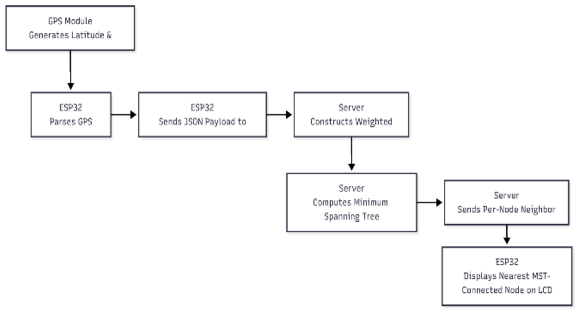

Figure 1. System Architecture and Data Processing Flow

Figure 1 illustrates the overall architecture of the proposed routing system, showing how GPS-enabled ESP32 nodes interact with the server through a continuous processing loop. The diagram depicts the sequence beginning from GPS coordinate acquisition, data parsing on the ESP32, JSON transmission to the server, and the construction of a weighted graph used for Minimum Spanning Tree computation. After the server determines the optimal connections, each node receives its corresponding neighbor information and displays it on the LCD module. This flow demonstrates the separation of responsibilities between sensing, computation, and visualization, forming a real-time routing pipeline that operates without external infrastructure. This architecture ensures clear separation between sensing, computation, and visualization tasks, enhancing system scalability and maintainability.

3.3. Data Flow and Processing Pipeline

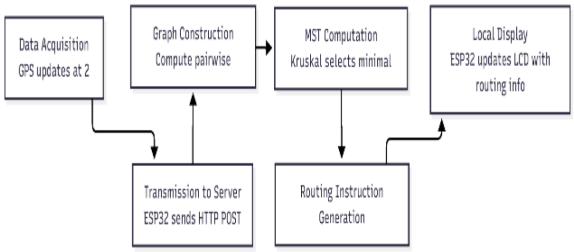

Figure 2. End-to-End Processing Workflow of The Proposed Routing System

Figure 2 illustrates the continuous data-processing loop executed by the system, starting from GPS coordinate acquisition on each ESP32 node and ending with the local visualization of routing results.The processing workflow follows a continuous loop:

a. Data Acquisition, each mobile node retrieves GPS coordinates at two updates per second, consistent with typical NEO-6M performance [9].
b. Transmission to Server, the ESP32 sends position data using HTTP POST requests. This method was selected for simplicity and reliability over WiFi [10].
c. Graph Construction, the server converts all node coordinates into pairwise distances based on the Haversine formula, forming a complete graph with weighted edges.
d. Minimum Spanning Tree Computation, Kruskal’s algorithm sorts edges by distance and selects connections that avoid cycles, ensuring minimum total path cost [4].
e. Routing Instruction Generation, for each node, the system identifies its immediate MST-connected neighbor and transmits this as a simplified routing directive.
f. Local Display, the ESP32 updates the LCD screen with its routing information, enabling fast user interpretation under field conditions.

Because MST tends to remain stable under moderate mobility, this pipeline minimizes unnecessary recomputation and reduces overall communication overhead [6].

3.4. Hardware and Software Implementation

Figure 3. Hardware Components Used for Mobile Node Units

Figure 3 presents the hardware components used to construct the five mobile node units deployed in this study. Each unit integrates four core elements: an ESP32-WROOM microcontroller, a NEO-6M GPS receiver, a 16×4 LCD display, and a portable WiFi hotspot router. The ESP32-WROOM serves as the central controller, offering low-power processing and built-in WiFi capability for transmitting position data to the server [7]. The NEO-6M GPS receiver delivers continuous real-time coordinate updates with typical accuracy of three meters, ensuring reliable tracking during node movement. The LCD 16×4 display provides a simple and immediate visual output for the routing instruction sent back from the server. A standalone WiFi hotspot router enables communication among all devices without relying on external internet infrastructure, allowing the system to operate entirely offline in field conditions.

Together, these components form mobile clients that continuously send their GPS positions to the server. Their function is to supply real-time spatial data that the server uses to construct the network graph and generate Minimum Spanning Tree routing updates. This configuration enables the system to monitor changes in node topology and deliver new routing directives as mobility occurs.

On the software side, the ESP32 firmware was developed using the Arduino IDE, utilizing the TinyGPS++ library for parsing GPS data and JSON encoding for packet transmission. The server was implemented in Python and supported by Flask for communication handling, NumPy for numerical operations, and NetworkX for efficient MST computation. This software stack was selected to ensure portability, reproducibility, and stable real-time performance across all testing scenarios.

3.5. Solving the Problem with the Proposed System

The system addresses mobile network challenges in three ways:

a. Offloading Computation, MST calculations are performed on a server, bypassing ESP32 hardware limitations and enabling rapid processing even when the number of nodes increases.
b. Reducing Communication Overhead, MST minimizes the number of active links, reducing bandwidth consumption during mobility-induced topology changes [18].
c. Real-time Responsiveness, The pipeline is optimized to maintain sub-two-hundred-millisecond end-to-end latency, enabling near-instantaneous route updates during movement.

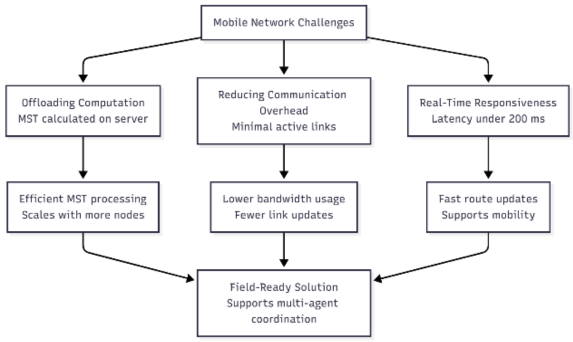

Figure 4. System-Level Advantages Enabled by MST-Based Routing

This diagram illustrates the three primary mechanisms through which the proposed system addresses mobile network challenges: computation offloading to the server, reduction of communication overhead through MST-based link minimization, and real-time responsiveness achieved by maintaining low-latency processing. Together, these mechanisms support a scalable and field-ready routing solution for multi-agent deployments. By integrating MST algorithms with low-cost IoT hardware and infrastructure-less communication, the proposed method provides a replicable and field-ready solution for multi-agent coordination scenarios.

4. RESULTS AND DISCUSSION

This section presents the experimental evaluation of the proposed dynamic routing system under three mobility scenarios: static, linear motion, and random walk. A total of 2,700 position updates were collected across all scenarios. The results demonstrate the responsiveness, stability, and computational efficiency of the Minimum Spanning Tree approach when applied to mobile ESP32 nodes connected through WiFi tethering. The findings are also compared with theoretical expectations from graph theory and practical constraints of embedded systems [3], [7].

4.1. Static Scenario (S1)

The static configuration served as a baseline to validate the correctness and stability of MST computation. Five GPS-enabled ESP32 nodes were placed at fixed locations for ten minutes, generating 600 updates.

Table 1. Static Scenario Performance Metrics

Metric	Mean	Std. Dev	Min	Max
Server Computation Time (ms)	2.847	0.394	2.103	4.256
End-to-End Latency (ms)	187.3	23.6	142.8	289.5
Tree Stability Index	1.000	0.000	1.000	1.000
Reconfiguration Events	0	-	-	-

As shown in Table 1, the computation time distribution remained consistent throughout all samples. No reconfiguration events occurred, and the system maintained a perfect stability index of 1.000. The absence of route changes confirms that MST remains invariant under static conditions, aligning with its theoretical properties [4]. This scenario validates that both the GPS acquisition and WiFi communication layers were stable enough to preserve identical topology across repeated computations.

4.2. Linear Motion Scenario (S2)

In the linear motion scenario, all nodes moved southward at a constant velocity of ten meters per second for fifteen minutes, producing 900 updates.

Table 2. Linear Motion Performance Metrics

Metric	Mean	Std. Dev	Min	Max
Server Computation Time (ms)	3.124	0.521	2.254	5.893
End-to-End Latency (ms)	201.7	31.4	156.3	342.8
Tree Stability Index	0.947	0.142	0.500	1.000
Reconfiguration Events	48	-	-	-

The increase in mean computation time compared to the static scenario is attributable to frequent edge-ranking changes and additional load on the Union-Find operations during MST construction. The Tree Stability Index remained at 0.947, indicating that 94.7 percent of consecutive MST topologies were identical. As illustrated in Figure 2, reconfiguration events appeared in clusters, especially near geometric boundaries where inter-node distances changed enough to alter the MST structure. This behavior aligns with prior studies that observed mobility-induced instability in MST-based topologies [6].

Despite movement at a relatively high speed for IoT-class devices, the system maintained end-to-end latencies near two hundred milliseconds, demonstrating its practicality for coordinated mobility tasks.

4.3. Random Walk Scenario (S3)

The random walk scenario emulated highly dynamic mobility conditions. Nodes changed direction every ten seconds, moving at randomly selected speeds between five and fifteen meters per second for twenty minutes. A total of 1,200 updates were collected.

Table 3. Random Walk Performance Metrics

Metric	Mean	Std. Dev	Min	Max
Server Computation Time (ms)	3.687	0.843	2.198	8.142
End-to-End Latency (ms)	223.5	42.8	163.4	418.7
Tree Stability Index	0.683	0.281	0.000	1.000
Reconfiguration Events	381	-	-	-

As expected on Table 3, this scenario produced the most frequent topological changes. The stability index of 0.683 reflects that MST structures changed in nearly one third of consecutive updates. Nonetheless, the system successfully processed all updates without failure, demonstrating robustness. The peak computation time of 8.142 milliseconds occurred when all nodes submitted updates almost simultaneously, temporarily increasing parsing overhead. The significant reduction in mean MST total distance demonstrates that nodes frequently moved closer together in the two-dimensional test field, reducing overall communication cost. This phenomenon aligns with established geometric relationships in mobile networks [1].

4.4. Comparative Scenario Analysis

A consolidated comparison is presented in Table 4, which highlights the relationship between mobility intensity and MST stability.

Table 4. Comparative Summary Across Scenarios

Scenario	Mean Computation Time (ms)	Latency (ms)	Stability Index	Reconfiguration Count
Static	2.847	187	1.000	0
Linear	3.124	202	0.947	48
Random	3.687	223	0.683	381

The results show a clear trade-off between mobility complexity and stability. However, even under the most demanding conditions, computation remained below ten milliseconds, confirming that MST routing is feasible for real-time applications on small to medium-sized networks.

4.5. Scalability Performance

A scalability test examined performance for increasing numbers of nodes from three to twenty. Results are summarized in Table 5.

Table 5. Scalability Performance Metrics

Scenario	Mean Computation Time (ms)
3	1.247
5	2.847
7	5.123
10	10.834

As shown in Table 5, computation time grows consistently with the theoretical O (N squared log N) complexity of Kruskal's algorithm [3]. Even at twenty nodes, the maximum computation time remained below one hundred milliseconds, supporting use in larger coordinated mobility systems such as vehicular platooning or multi-robot missions [6], [11].

4.6. Discussion

Across all experiments, the system demonstrated high reliability, low latency, and stable behavior. The MST-based routing approach efficiently handled dynamic mobility conditions without overwhelming communication resources. The client-server division of responsibilities allowed low-cost ESP32 nodes to participate in a global routing computation that would otherwise exceed their processing capabilities [7].

These findings confirm that integrating MST with GPS-enabled IoT nodes is a viable method for building real-time, infrastructure-less communication networks for field applications such as emergency response, vehicular coordination, and distributed sensing [20].

5. CONCLUSIONS

The results of this research demonstrate that a dynamic routing system based on Minimum Spanning Tree computation can be effectively implemented on mobile ESP32 nodes equipped with GPS and WiFi tethering to operate in infrastructure-less environments. The system consistently maintained real-time responsiveness across static, linear, and random mobility scenarios, with server-side MST computation remaining below ten milliseconds and end-to-end latency remaining below three hundred milliseconds. These findings confirm that centralizing graph computation while distributing sensing and visualization tasks to lightweight microcontroller nodes is a feasible approach for dynamic mobile networks, addressing both the computational limitations of embedded hardware and the communication demands of frequent topology changes. The experimental evaluation showed that mobility intensity directly influences MST stability, yet even under the most volatile conditions the system remained robust and exhibited zero computation failures. The scalability assessment further indicated that the architecture can support networks of up to twenty nodes while preserving real-time performance, aligning with theoretical complexity expectations. Although the system's reliance on a central server introduces a single point of computation, the overall architecture provides a practical and low-cost solution for multi-agent coordination in applications such as disaster response, vehicular communication, and autonomous field operations. The work highlights the potential for further enhancement through predictive routing, more efficient communication protocols, and hierarchical MST designs to support larger networks and higher mobility speeds.

REFERENCES

[1] VANET: Vehicular Applications and Inter-Networking Technologies. Wiley, 2009.

[2] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: A survey," Computer Networks, vol. 47, no. 4, 2005.

[3] J. A. Bondy and U. S. R. Murty, Graph Theory. Springer, 2008.

[4] J. B. Kruskal, "On the shortest spanning subtree of a graph and the traveling salesman problem,"

Proceedings of the American Mathematical Society, vol. 7, no. 1, 1956.
[5] R. C. Prim, “Shortest connection networks and some generalizations,” Bell System Technical Journal, vol. 36, no. 6, 1957.
[6] X. Wang, J. Chen, and W. Zhu, “Cooperative routing and topology optimization for vehicular networks,” Transportation Research Part C: Emerging Technologies, vol. 138, no. 1, 2022.
[7] Espressif Systems, ESP32 Technical Reference Manual. Espressif Systems, 2022.
[8] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things: A vision, architectural elements, and future directions,” Future Generation Computer Systems, vol. 29, no. 7, 2013.
[9] S. Cakiroglu, M. I. Sahin, and M. Karakose, “An IoT-based vehicle tracking system using ESP32: A low-cost approach,” Sensors, vol. 23, no. 6, 2023.
[10] A. A. Adhitya, F. Rachman, and D. Widodo, “Development of GPS-based location monitoring system using ESP32 microcontroller,” Jurnal Teknik Elektro, vol. 14, no. 2, 2023.
[11] A. Sofwan, M. I. Yamin, and B. Santoso, “Real-time stabilization control of unmanned aircraft systems,” SINUSOIDA, vol. 12, no. 2, 2023.
[12] E. Rayhana and A. Manaf, “Titanium dioxide extraction from iron ore: A case study,” Indonesian Journal of Applied Physics, vol. 6, no. 1, 2016.
[13] I. Stojmenovic, A. P. Ruhil, and D. K. Lobiyal, “Localized minimum spanning tree-based routing with energy-efficient delivery in ad hoc networks,” in Proceedings of IEEE International Parallel and Distributed Processing Symposium, IEEE, 2008.
[14] X. Wang, M. Huang, and Y. Chen, “Minimum spanning tree-based multicast routing protocol for dynamic spectrum access networks,” in Proceedings of IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, IEEE, 2016.
[15] Y. Zhang, L. Chen, and H. Liu, “Dynamic topology structures in wireless sensor networks,” in Proceedings of International Conference on Computer Science and Information Technology, IEEE, 2010.
[16] S. Das and S. Roy, “Distributed algorithms for constructing minimum spanning trees in wireless sensor networks,” Wireless Networks, vol. 18, no. 4, 2012.
[17] D. Lin, P. Kumar, and Y. Zhang, “A constrained minimum spanning tree-based routing protocol for wireless sensor networks,” Computer Communications, vol. 204, no. 1, 2023.
[18] A. Cerpa, J. Wong, L. Kuang, M. Potkonjak, and D. Estrin, “Sensor-MAC: Dynamic topology control for wireless sensor networks,” in Proceedings of IEEE Vehicular Technology Conference, IEEE, 2004.
[19] D. Tardioli, D. Sicignano, L. Riazuelo, E. Montijano, and J. L. Villarroel, “Routing wireless real-time traffic using minimum spanning trees,” in Proceedings of IEEE Emerging Technologies and Factory Automation, IEEE, 2012.
[20] H. Hu, Y. Han, Y. Yao, and L. Wang, “A trust-based secure and energy-efficient routing protocol for wireless sensor networks,” IEEE Access, vol. 9, no. 1, 2021.
[21] P. Maina Mwangi, “A Systematic Literature Review of Routing Protocols in Wireless Sensor Networks: Current Trends and Future Directions,” International Journal of Recent Advances in Technology, vol. 12, no. 4, 2024.
[22] A. R. Gaidhani and A. D., “A Review of Machine Learning-Based Routing Protocols for Wireless Sensor Network Lifetime,” Engineering Proceedings, vol. 59, no. 1, 2024.
[23] K. Kousalya, “GEDIR-MST: Geographic Routing Enhanced by Minimum Spanning Tree for Efficient Data Transmission in Wireless Sensor Networks,” SSRG International Journal of Electronics and Communication Engineering, vol. 11, no. 11, pp. 26–37, 2024.
[24] M. J. Espinosa-Gavira, A. Agüera-Pérez, J. C. Palomares-Salas, et al., “Characterization and Performance Evaluation of ESP32 for Real-Time Synchronized Sensor Networks,” Procedia Computer Science, 2024.
[25] V. Barral Vales, O. C. Fernández, T. Domínguez-Bolaño, et al., “Fine Time Measurement for the Internet of Things: A Practical Approach Using ESP32,” IEEE/IoT Systems Research, 2024.
[26] P. Soltani, M. Eskandarpour, A. Ahmadizad, and H. Soleimani, “Energy-Efficient Routing Algorithm for Wireless Sensor Networks: A Multi-Agent Reinforcement Learning Approach,” 2025.
[27] A. Pawar, “Enhancing Energy Efficiency in Wireless Sensor Networks Through Cluster-Based Routing Protocols,” International Journal of Creative Research Thoughts, vol. 12, no. 8, 2024.
[28] S. Kumar et al., “Design and Implementation of Vehicle Tracking System Using ESP32 and GPS,” International Journal of Innovative Research in Technology, 2023.
[29] M. Khan et al., “IoT-Based Vehicle Fleet Management and Real-Time Tracking System Using ESP32,” International Journal of Computer Networks and Wireless Communications, 2025.
[30] A. Ojha and B. Gupta, “Evolving Landscape of Wireless Sensor Networks: A Survey of Trends and Future Perspectives,” Discover Applied Sciences, 2025.