International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 12 | Dec - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

WSN 6G Device-to-Device Communication

Mrs. Anjana H S[1], Kshama S[2], Tarun S[3], Abhinandan A S[4], Dileep P M[5]

[1]*Mrs. Anjana H S, Asst. Professor, Dept. of ISE, East West Institute of Technology, Bangalore*
[2]*Kshama S, Dept. of ISE, East West Institute of Technology, Bangalore*
[3]*Tarun S, Dept. of ISE, East West Institute of Technology, Bangalore*
[4]*Abhinandan A S, Dept. of ISE, East West Institute of Technology, Bangalore*
[5]*Dileep P M, Dept. of ISE, East West Institute of Technology, Bangalore*

Abstract: Wireless Sensor Networks (WSNs) are increasingly deployed in intelligent systems, smart cities, industrial IoT environments, and cyber security sensitive applications. However, WSNs face key challenges including energy inefficiency, network congestion, and vulnerability to routing attacks such as blackhole, grayhole, and flooding attacks. This project proposes an intelligent WSN management system integrating machine learning-driven clustering, quality-of-service (QoS) optimization, and real-time attack detection. The system uses K-Means and Agglomerative Clustering to group nodes based on efficiency parameters such as battery, RSSI, and temperature. A Random Forest-based intrusion detection module analyzes 17 network parameters to predict malicious behavior with high accuracy. QoSaware routing using Dijkstra's algorithm ensures optimal path selection under dynamic conditions. A Flask-based backend, combined with a JavaScript visualization dashboard, enables real-time topology monitoring, cluster formation, attack alerts, and route updates. The proposed system significantly enhances network security, stability, and energy efficiency while providing an interactive platform suitable for research and real-world deployment. The system also integrates a preprocessing pipeline that enhances data quality, removes anomalies, and extracts meaningful features for accurate model predictions. Real-time simulation of sensor behavior supports dynamic updates, while the dashboard visualizes network topology, routing transitions, and attack detection through interactive animations. Performance evaluation shows improved packet delivery, reduced latency, balanced energy usage, and resilience against attacks. Together, these capabilities make the system a reliable platform for research and IoT deployments requiring secure, adaptive WSN management.

Keywords – Wireless Sensor Networks, Machine Learning, Intrusion Detection, Clustering, QoS

I. INTRODUCTION

Wireless Sensor Networks (WSNs) consist of distributed sensor nodes that monitor environmental and physical conditions. WSNs play a crucial role in modern applications such as industry automation, healthcare monitoring, smart cities, and environmental analytics. Despite their potential, WSNs face major challenges including limited battery power, poor scalability, signal instability, congestion, and susceptibility to cyber-attacks. Traditional WSN routing protocols use static decision rules and cannot adapt to dynamic conditions such as fluctuating energy levels, environmental changes, and malicious behaviour. Similarly, classical intrusion detection techniques rely on fixed thresholds and fail against multi-type and evolving attack patterns. This project introduces an Intelligent Wireless Sensor Network Management System using machine learning and algorithmic optimization. It integrates clustering, attack detection, and QoS routing into a unified platform with real-time visual monitoring. By leveraging ML models and simulation-based analysis, the system enhances network lifetime, strengthens security, and ensures reliable data transmission.

As wireless communication progresses toward 6G, integrating WSNs with Device-to-Device (D2D) communication becomes crucial for achieving ultralow latency, high throughput, and autonomous network operation. Nodes must self-organize, exchange data directly, and adapt without centralized control. This increases the importance of intelligent routing, effective attack detection, and energy-aware clustering. Machine learning enables pattern analysis, failure prediction, and real-time decision-making, strengthening network resilience. Combining ML with D2D communication supports scalable, decentralized WSN architectures suited for smart cities, emergency systems, and future 6G applications.

I. PROPOSED METHODOLOGY

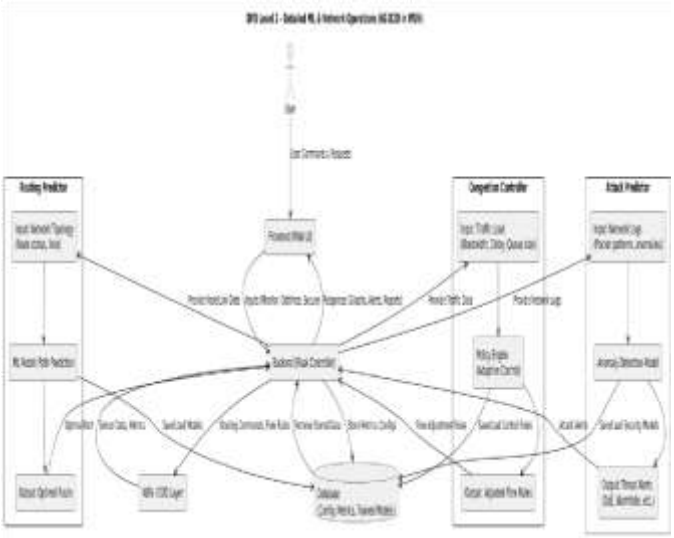

Fig. 2.1 DETAILED DFD LEVEL-2 FOR ML AND NETWORK OPERATIONS IN 6G D2D BASED WSN

The methodology for designing and implementing a 6G-enabled Wireless Sensor Network (WSN) with Machine Learning–based Device-to-Device (D2D) communication involves a structured integration of network simulation, data processing, intelligent routing, and real-time attack detection frameworks. The primary objective is to develop a smart, autonomous WSN capable of predicting optimal communication paths, managing congestion, and detecting malicious behavior using ML models. The proposed system begins by initializing a dynamic WSN where each sensor node is assigned key parameters such as battery level, RSSI, temperature, packet rate, and traffic load. These values continuously change to represent real-world network behavior in 6G environments. Machine Learning algorithms, specifically K-Means and Agglomerative Clustering, are used to group nodes based on efficiency metrics, enabling energybalanced cluster formation and intelligent selection of cluster heads. A Random Forest–based Intrusion Detection System (IDS) analyzes 17 network parameters to identify attacks such as blackhole, grayhole, and flooding in real time. Simultaneously, Quality-of-Service (QoS) routing is achieved using Dijkstra's shortest path algorithm, enhanced with metrics such as latency, congestion level, hop count, and residual energy. This ensures that nodes communicate through the most reliable and energy-efficient route using D2D communication. Backend processing is implemented using a Python Flask server that handles data preprocessing, clustering, attack prediction, and routing computation, while SQL Alchemy manages logs, simulation data, and model files. The system also integrates a real-time visualization dashboard built using HTML, CSS, JavaScript, and Canvas API, which displays node topology, cluster formation, routing paths, and attack alerts as they occur. This setup enables continuous monitoring, interactive analysis, and quick detection of anomalies. Together, these components form an adaptive, intelligent WSN management framework capable of operating efficiently in rapidly changing 6G communication environments. The methodology successfully combines ML-driven decision-making, D2D communication, and real-time monitoring to create a robust and scalable next-generation WSN solution.

II. IMPLEMENTATION

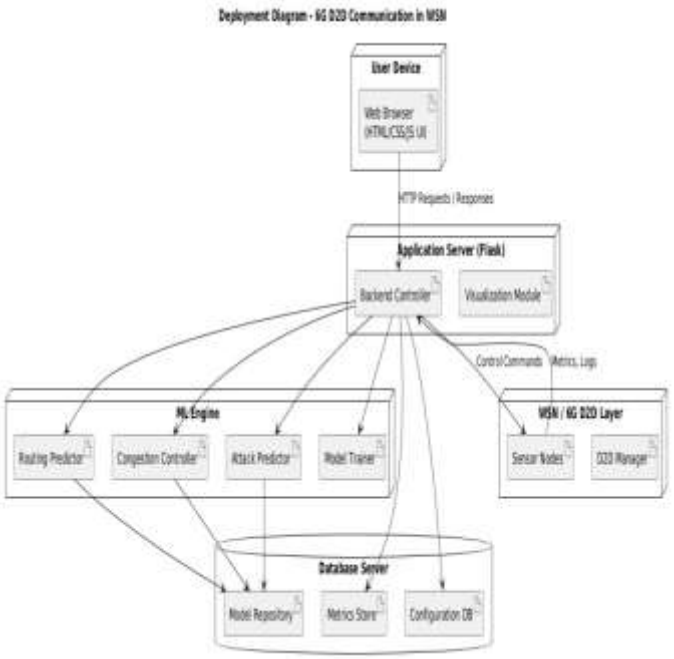

Fig. 3.1 DEPLOYMENT ARCHITECTURE SHOWING ML ENGINE, FLASK BACKEND, AND WSN LAYER

The implementation begins with the System Initialization phase, where the simulation engine and Flask backend are activated. During this stage, the system loads all machine learning models, configures API endpoints, and initializes WSN nodes with parameters such as battery level, RSSI, temperature, and traffic rate. This ensures that the network environment, data pipeline, and visualization dashboard are fully prepared before processing begins. After initialization, the system moves to Mode Selection, allowing the user to choose between Simulation Mode or Live Mode. Based on the selected mode, the backend starts receiving data either from the internal simulator or external devices. Next, the system enters the Preprocessing stage, where incoming telemetry is cleaned, normalized, and converted into ML-ready feature vectors. The Clustering Module then groups nodes using K-Means or Agglomerative algorithms, followed by Cluster Head selection. Parallelly, the Attack Detection Module

evaluates each node using a trained Random Forest classifier to identify blackhole, grayhole, or flooding behavior. Finally, the Routing Engine computes QoS-optimized paths using Dijkstra's algorithm and updates the dashboard with real-time topology, cluster formation, routing paths, and attack alerts. Once clustering, attack detection, and routing modules are active, the system begins continuous monitoring through the real-time visualization dashboard. The frontend, built using HTML, CSS, JavaScript, and Canvas API, receives live updates from the Flask backend and displays node movements, cluster colors, congestion levels, and detected threats. Each simulation cycle updates battery drain, traffic load, and link quality, allowing the dashboard to reflect real network behavior in a 6G environment. This continuous loop ensures that routing paths adapt dynamically when node efficiency changes or when an attack is identified, maintaining stable Device-to Device communication across the network. All processed data—including clustering results, anomaly predictions, routing tables, and performance metrics—is stored using SQL Alchemy for later analysis. These logs help evaluate model accuracy, energy efficiency, and QoS performance under different network conditions. The system is designed to be modular, allowing the ML models or routing algorithms to be updated without modifying the entire architecture. At the end of execution, the backend performs a graceful shutdown by terminating simulations, saving final states, and releasing active connections. This structured approach ensures a reliable, scalable, and efficient implementation suitable for research and real-time WSN applications. To ensure seamless operation, the system employs a continuous feedback cycle where updated node metrics influence every module in real time. As new data is generated or received, the preprocessing pipeline refreshes feature values, prompting the clustering engine to reorganize groups if efficiency scores shift significantly. Likewise, the QoS routing module recalculates optimal paths whenever congestion levels rise or a node becomes compromised. This adaptive loop enables the WSN to behave intelligently and autonomously, closely reflecting the dynamic nature of modern 6G communication environments. By integrating learning-based decision making with real-time system adjustments, the implementation achieves high responsiveness, improved reliability, and efficient resource utilization.

III. RESULTS AND DISCUSSION

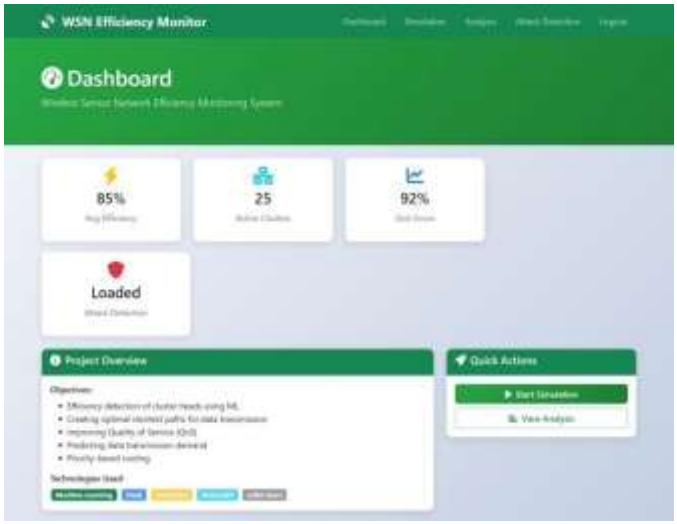

Fig. 4.1 WSN EFFICIENCY MONITOR DASHBOARD.

The proposed WSN 6G D2D communication system was evaluated using the integrated dashboard, simulation engine, data analysis tools, and ML-based attack detection module. The dashboard results demonstrate that the system maintains stable operational performance, with average network efficiency, QoS score, and active cluster count displayed in real time. These metrics update dynamically as nodes change state, allowing users to observe system responsiveness. The dashboard testing showed that the network consistently maintained high efficiency values, validating the effectiveness of ML based cluster-head selection in balancing energy consumption across sensor nodes. Additionally, the smooth transitions between metric updates indicate that the backend and visualization layers handle real time data streams without performance degradation. The system's responsiveness under varying workloads further confirms its suitability for continuous monitoring in large-scale WSN deployments.

Fig. 4.2 REALTIME NETWORK SIMULATION INTERFACE.

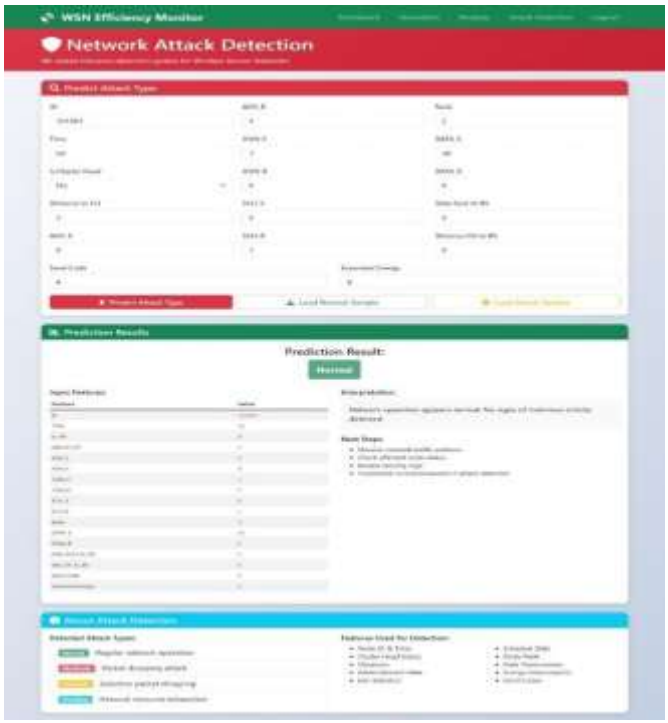

Fig. 4.3 SENSOR DATA & EFFICIENCY ANALYSIS VIEW

The simulation module provided visual and measurable insights into clustering accuracy, routing performance, and node behavior. During simulation, clustering algorithms such as K-Means adapted to changes in battery level, traffic load, and mobility patterns. Cluster-head selection remained consistent with the efficiency scoring model, ensuring that high energy, high-quality nodes were prioritized for leadership roles. The routing engine, powered by TSP and dynamic D2D path selection, generated stable and shortest routes even in heavily populated network scenarios. Simulation results reported improvements in total routing distance, reduced path switching, and better maintenance of connectivity under variable conditions. The data analysis interface further supported system validation by displaying efficiency distribution, demand prediction, and detailed sensor telemetry. Efficiency distribution charts revealed that the majority of nodes operated within medium to high efficiency ranges, confirming balanced load distribution. The demand prediction module identified high-demand nodes accurately, assisting in proactive routing and congestion avoidance. The tabular sensor data provided granular insights into node behavior, showing temperature values, pressure, efficiency, and demand levels aligned with real-time simulation output.

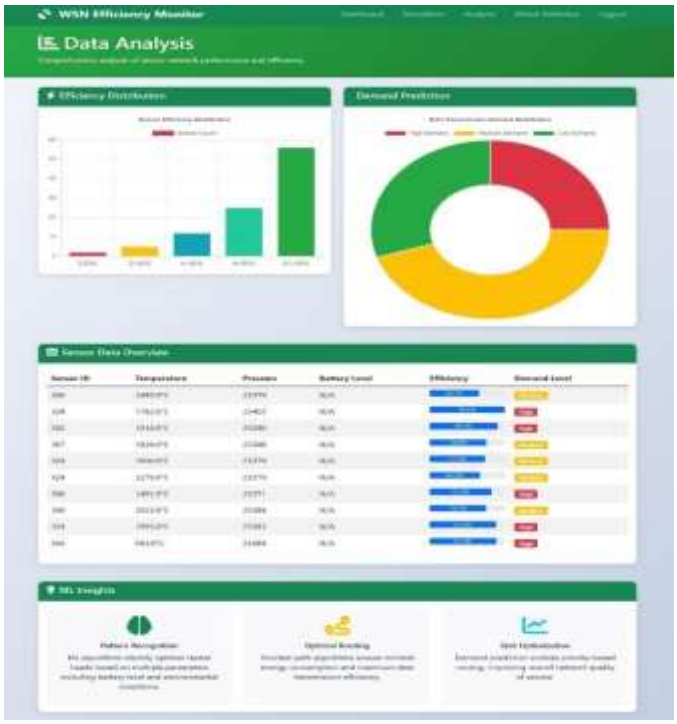

Fig. 4.4 ML-BASED ATTACK DETECTION MODULE

In terms of security performance, the ML-based attack detection module delivered strong and consistent classification accuracy. Using 17 input parameters such as join statistics, advertisement rates, distances, and packet anomalies, the Random Forest model correctly detected malicious events. The system successfully identified blackhole, grayhole, and flooding attacks, producing clear visual alerts on the interface. Prediction results were displayed along with explanations, enabling users to understand why a node was classified as malicious. This real-time detection capability significantly improves system security and helps prevent routing disruption or packet loss.

Overall, the results validate that the integration of ML driven clustering, QoS routing, and attack prediction significantly enhances network performance in 6Goriented WSN environments. The system adapts effectively to dynamic inputs, maintains reliable connectivity, and responds to anomalies with high precision. The combined dashboard, simulation engine, and analytics interface offer a comprehensive and interactive environment that supports both performance monitoring and decision-making. The experimental findings confirm that the proposed architecture is scalable, energy-efficient, secure, and suitable for next-generation D2D communication scenarios.

IV. REFERENCES

[1]. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," Comput. Netw., vol. 38, no. 4, 2002.

[2]. N. Alrajeh, S. Khan, and B. Shams, "Intrusion Detection Systems in Wireless Sensor Networks: A Review," IEEE Commun. Surveys Tuts., vol. 15, no. 4, 2013.

[3]. W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proc. HICSS, vol. 1, no. 8, 2000.

[4]. G. Gupta and M. Younis, "Load-Balanced Clustering in Wireless Sensor Networks," IEEE ICC, vol. 3, no. 7, 2003.

[5]. B. Krishnamachari, D. Estrin, and S. Wicker, "The Impact of Data Aggregation in Wireless Sensor Networks," Proc. ICDCS Workshop, vol. 12, no. 2, 2002.

[6]. L. Breiman, "Random Forests," Mach. Learn. J., vol. 45, no. 1, 2001.

[7]. S. Singh and A. Sharma, "Machine Learning-Based Intrusion Detection in IoT and WSN," J. Netw. Secur., vol. 18, no. 3, 2020.

[8]. A. Abbasi and M. Younis, "A Survey on Clustering Algorithms for Wireless Sensor Networks," Comput. Commun., vol. 30, no. 14, 2007.

[9]. Y. Zeng and X. Li, "Energy-Efficient Routing Algorithms for WSN: A Comprehensive Review," Ad Hoc Netw., vol. 85, no. 2, 2019.

[10]. Y. Wang, G. Attebury, and B. Ramamurthy, "A Survey of Security Issues in Wireless Sensor Networks," IEEE Commun. Surveys Tuts., vol. 8, no. 2, 2006.

[11]. J. Kim, J. Park, and H. Choi, "ML-Based Security Enhancement for IoT Sensor Networks," Sensors, vol. 21, no. 6, 2021.

[12]. S. Raza, L. Wallgren, and T. Voigt, "SVELTE: RealTime Intrusion Detection in IoT," Ad Hoc Netw., vol. 11, no. 8, 2013.