International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 23s, 2025
https://www.theaspd.com/ijes.php

Machine Learning Based Fault Detection And Recovery Framework For Iot Environmental Monitoring Applications

[1]E. Nandhinipriya, [2]Dr. R. Manikandan, [3]Dr. K. Kamali,
[1]Research Scholar, Assistant Professor/Programmer, Dept of Computer science and Engineering, Annamalai University. Chidambaram, Tamilnadu, aksharashree2009@gmail.com.
[2]Associate Professor, Dept of Computer science and Engineering, Annamalai University. Chidambaram, Tamilnadu, rmkmanikandan1111@gmail.com.
[3]Assistant Professor/Programmer, Dept. of Computer and Information Science, Annamalai University. Chidambaram, Tamilnadu, kamaliaucse2006@gmail.com.

Abstract
In Internet of Things (IoT) enabled environmental monitoring sector, fault detection and recovery systems are crucial for guaranteeing continuous, accurate, and reliable service delivery. IoT devices often function in resource-constrained environments, making the implementation of complex fault-detection algorithms difficult. In this paper, Automatic Fault Detection and Recovery (AFDR) framework based on Artificial Neural Network Fuzzy Inference System (ANFIS) is proposed. In this framework, the device faults along a path are determined based on the ANFIS model. Fuzzy rules are provided based on the packet loss rate (PLR), Signal to Interference Noise Ratio (SINR) and round trip delay (RTD) metrics. In the fault recovery phase, the recovery agent at the primary path establishes an alternate fault-free route by excluding the faulty nodes. The evaluation metrics packet delivery ratio (PDR), number of packets dropped, average residual energy computational cost and end-to-end delay are measured by varying the number of fog nodes.Experimental results show that the proposed AFDR-ANFIS model attains higher fault detection accuracy with reduced packet drops and computational overhead. The proposed Machine Learning (ML) based technique plays a vital role in fault detection and recovery systems in IoT-based healthcare.

Keywords: *Internet of Things (IoT), Health-care applications, Fault detection, Recovery, Machine Learning, Artificial Neural Network Fuzzy Inference System (ANFIS),*

1. INTRODUCTION

The Internet of Things (IoT) is a network of interconnected physical objects, which are embedded with software, sensors, and communication technologies that enable them to gather and exchange data over the internet. These smart objects interrelate with other systems in real time, rendering them vital for modern applications such as cyber-physical systems (CPSIoT plays a major role in recognizing faults, monitoring machine performance, and ensuring timely communication of resource failures [1].

IoT environmental sensors are devices that use the Internet of Things (IoT) to monitor and measure various environmental conditions like temperature, humidity, air quality, and light levels [2]. In the IoT-enabled environment monitoring sector, fault detection and recovery systems are crucial for guaranteeing continuous, accurate, and reliable service delivery. These systems help recognize problems like communication failures, sensor degradation, and software errors. [3].

Fault detection ensures anomalies are rapidly identified, whereas recovery mechanisms ensure continuity of operations via redundancy or self-healing protocols. Intelligent systems using machine learning (ML) or AI can now forecast failures based on behavioural patterns of devices, improving response time. Since the number of IoT nodes and data volume grows, a robust fault detection and recovery infrastructure safeguards operational integrity and reinforces the trustworthiness of digital healthcare. It contributes to lower costs, reduced downtime, and higher-quality patient care [4].

1.1 Motivation and Objectives

In spite of their importance, fault detection and recovery systems in IoT environmental monitoring encounters several significant challenges. IoT devices often function in resource-constrained environments with restricted processing power, battery life, and memory, making the implementation of complex fault-detection algorithms difficult. Moreover, data imbalance is common in which failure events are rare when compared to normal operation, which hinders the training of ML models [5]. Additionally, IoT systems must deal with data packet losses, intermittent connectivity, and synchronization problems that may either mask or falsely signal faults. Since IoT healthcare systems become more complex and interconnected, developing fault detection and recovery systems that are both accurate and effective—without compromising data privacy or patient safety—remains a critical but evolving challenge [6].

Existing data collection approaches in IoT-WSN networks, did not ensure the reliability of devices and consistency of sensed data.

Machine Learning (ML) technique plays a vital role in fault detection and recovery systems in IoT-based healthcare. ML facilitates the continuous monitoring of medical devices and patient data for identifying abnormal patterns, detecting faults, and triggering alerts before failures occur. . ML models support recovery systems by recommending corrective actions or automatically adjusting device configurations [7]. Additionally, ML allows predictive maintenance of connected devices by analysing historical and real-time data, prediction when a device may fail, and preventing disruptions in healthcare services [8][9].

The main objective of this research work is to develop an automatic fault detection and recovery system using ML techniques for IoT environmental monitoring applications.

2. RELATED WORKS

A proactive methodology [10] is proposed to improve sensor fault prediction and classification within IoT systems. This two-stage solution starts with the training of a hybrid Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM) model, which is designed to forecast future sensor measurements using historical data. In the following stage, the predicted data are input into a hybrid CNN and Multi-Layer Perceptron (MLP) model, which is trained specifically for detecting and classifying various sensor faults. The system can accurately find fault types such as normal, bias, drift, random, and poly-drift, enabling faults to be anticipated before they actually occur, by utilizing the forecasted sensor values as inputs for classification. This approach improves the reliability and operational efficiency in IoT systems by facilitating preventive maintenance before system faults escalate.

In [11], a new technique for fault detection and classification in electrical systems is proposed through analysis of voltage and current behaviour across the transmission line phases. Multiple ML models are tested, using a rich dataset of varied fault scenarios. A combined ensemble model—RF-LSTM Tuned KNN—is proposed that yields exceptional performance. RF and KNN showed slightly lower performance levels when compared independently. The results offer significant advancements in the domain of grid fault detection, contributing to improved reliability and resilience of power infrastructure.

Fault detection and control is proposed in [12] within the manufacturing sector using a ML-based framework. Data is collected through IoT modules that observed historical manufacturing faults. Before analysis, the data experiences preprocessing steps including noise removal, normalization, and smoothing. Feature extraction is performed through Kernel Principal Vector Component Analysis. The control mechanism of these features is accomplished using a Gaussian Quadratic Kernelized Generative Adversarial Network. Performance indicators like Mean Average Precision (MAP), RMSE, Area Under Curve (AUC), recall, F1-score, accuracy, and precision are used for experimental validation. The Auto-Encoder Neural Network developed as the most effective model for identifying production line faults, whereas One-Class SVM was noted for delivering highly accurate results.

A new reliability assessment framework is proposed in [13] using Colored Resource-Oriented Petri Nets (CROPNs) integrated with IoT technology to support accurate analysis, prognosis, and self-repair mechanisms in complex CPSs. CROPNs are constructed for establishing the essential conditions for maintaining system liveness in the occurrence of resource failures and deadlocks. This method integrates IoT into the Petri Net structure, ensuring system reliability. Simulation and analysis are conducted through the GPenSIM tool. The results proved that the model is simpler in structure yet more effective in managing deadlock scenarios and modelling reliability in automated manufacturing systems when compared to prior literature.

The study in [14] focuses on improving power electronics reliability, mainly in hybrid Multi-Level Inverter (MLI) topologies. This configuration depends on a conventional two-level inverter structure to reduce the device count while generating a nine-level output voltage. A critical problem addressed is capacitor voltage imbalance, resolved through an optimized switching strategy. However, timely detection is essential since semiconductor switches are highly vulnerable to open-circuit faults. A method is proposed to detect these faults using load voltage waveform analysis. Feature extraction is accomplished using wavelet transformation, which is followed by fault classification using an Artificial Neural Network.

A method termed XAI-LCS is proposed in [15] for tackling some of the limitations of current AI-based fault detection systems, such as lack of interpretability and computational complexity. This approach uses the XGBoost algorithm for detecting early sensor faults in IoT environments. The model can identify a variety of fault types including drift, bias, complete failures, and precision degradation, while addressing data imbalance for avoiding skewed predictions. Achieving a validation accuracy of 98%, the solution also integrates explainable AI (XAI) components to improve user trust and model transparency. This method makes a significant contribution to consistent fault diagnosis in IoT systems by ensuring actionable intuitions and interpretability.

2.1 Research Gaps

In spite of the advancements in fault detection systems in IoT and CPS environments, several research gaps remain, especially in environmental monitoring applications in which system reliability and real-time responsiveness are critical. Existing studies have successfully applied ML and deep learning techniques for fault detection in industrial contexts including CNC machines and motor monitoring, often combining cyberattack resilience and MQTT protocols. Though, healthcare-focused IoT environments present unique challenges like unpredictable patient data patterns, heterogeneous sensor networks, privacy concerns, and the requirement for ultra-low-latency fault responses that are not adequately addressed in the existing literature. Most existing studies focus on industrial or mechanical tool condition monitoring and energy-efficient protocols, with limited generalizability to dynamic, patient-centric healthcare systems.

Furthermore, even though Colored Petri Nets (CPNs) and their variants such as CROPNs provide robust modelling capabilities, their incorporation with adaptive ML algorithms for real-time healthcare fault prediction and recovery remains underexplored. Additionally, sensor fusion methods have not been fully utilized for improved fault localization and prediction accuracy under noisy, complex clinical conditions. Therefore, there is a critical requirement for a fault detection and recovery framework that integrates secure IoT infrastructure, intelligent ML-driven decision-making, and dynamic reconfiguration mechanisms tailored particularly to the sensitive, life-critical domain of IoT-enabled healthcare applications.

3. PROPOSED METHODOLOGY

3.1 Overview

In this paper, an Artificial Neural Network Fuzzy Inference System (ANFIS) based automatic fault detection and recovery framework is proposed. In this framework, for estimating the node faults along a path, the ML based ANFIS is applied. The Fuzzy rules are provided based on the packet loss rate (PLR), Signal to Interference Noise Ratio (SINR) and round trip delay (RTD) metrics. In the fault recovery phase, the recovery agent at the primary path will establish an alternate fault-free route by excluding the faulty nodes.

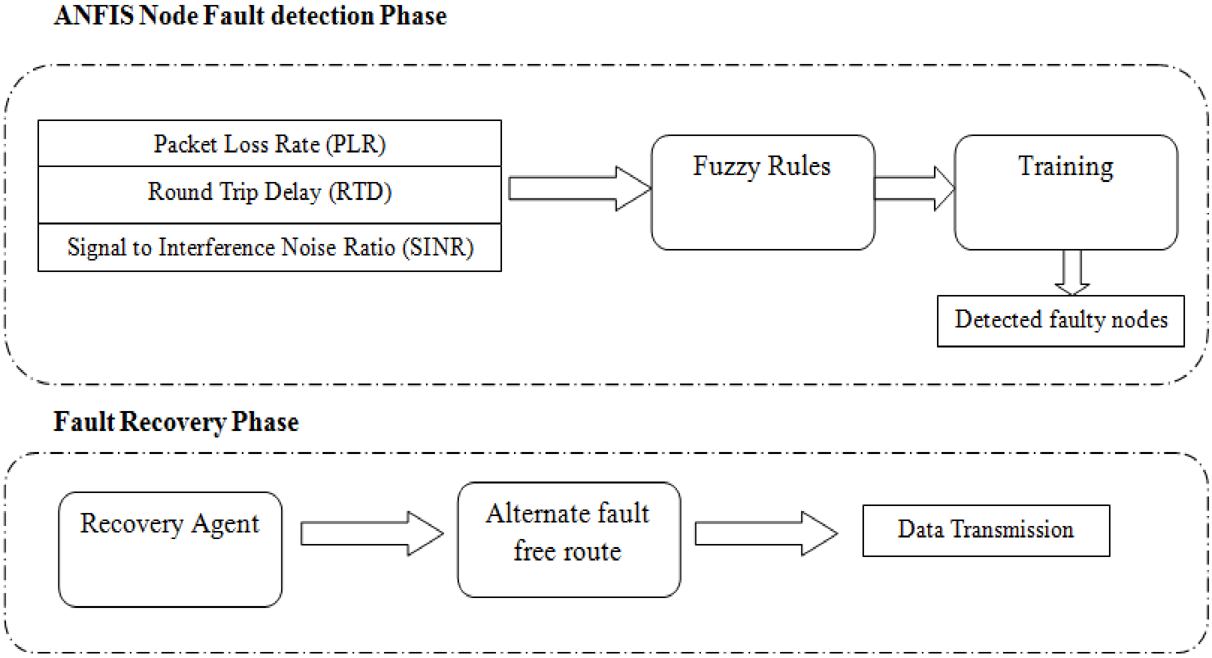

Figure 1 Block diagram of AFDR-ANFIS framework

3.2 Node fault detection Phase

3.2.1 ANFIS model

The fuzzy model of the modified network model is utilized by the machine learning model. Figure 2 illustrates the two fuzzy rules and two input parameters of the ANFIS model.There are five layers in the ANFIS and the working process of each layer is explained below:

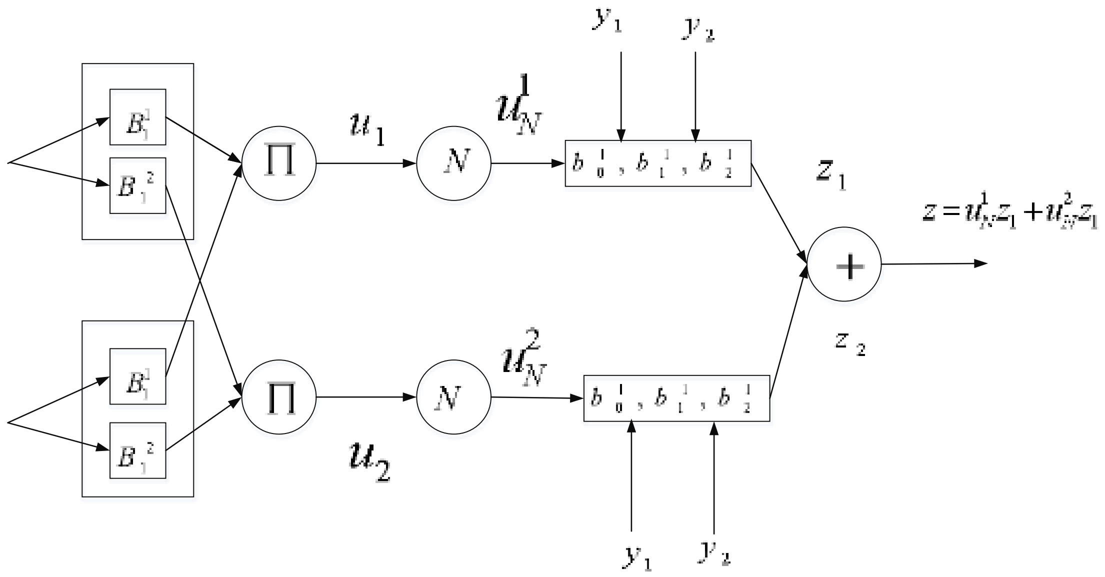

Figure 2 Structure of ANFIS

First layer: The fuzzification layer is the first layer, where each membership function's membership degrees are calculated. SC (subtractive clustering) provides the number of fuzzy rules in ANFIS, and the fuzzy C means (FCM) technique is utilized to determine the initial centres. The ANFIS's training phase, which optimizes regression elements and membership functions and lowers error, begins after the fuzzy rules are created as indicated in Table 1. Moving further, the regression components of fuzzy rules are optimized using the LSM (least square model). The membership function-related variables are optimized in reverse using the gradient descent methodology.

Second layer: The following expression is used to calculate the clustered value of the antecedent segment of each fuzzy rule after the membership degrees have been determined:

$$u^l = \prod_{j=1}^{n} B_j^l(y_j) \tag{1}$$

Third layer: The normalized weights for each rule are calculated using:

$$u_N^l = \frac{w^l}{\sum_l w^l} \tag{2}$$

Fourth layer: The sequential value of each rule is calculated for input variables using the regression elements of all the rules. It is then stated as follows:

$$z^l = b_0^l + \sum_{j=1}^{n} b_j^l y_j \tag{3}$$

Fifth layer: The ANFIS model's output for the input variables is computed as the weighting consequence values of every rule. It is given as:

$$z = \sum_l u_N^l z^k \tag{4}$$

where y_j and z^k are j^{th} fuzzy rule's input variable and output parameter. B_j^l and b_0^l are the antecedent segment and bias regression elements

3.2.2 Fuzzy Rules

Based on the PLR, SINR and RTD, Fuzzy rules are generated to derive the fault tolerant index (FTI). The packet loss rate at the receiver end is given by .

$$PLR = 1 - \left(1 - \sum_{i=t+1}^{n} \binom{n}{i} p_b^i (1 - p_b)^{n-i} \right)^{\left\lceil \frac{Lp}{k} \right\rceil} \tag{5}$$

Where p_b is the bit error rate and L_p is the payload's packet size in a single transmission, the receiver's packet loss rate is determined by

The following formula is used to estimate the SINR from the RSSI.

$$\text{SINR} = \frac{RSSI}{B_n} \tag{6}$$

where B_n is the background noise.

The Round Trip Delay (RTD) is calculated using the following equation for each path.

$$RTD_{est} = n\,T \tag{7}$$

Where n is the number of nodes and T is the total time interval.

Then the generated Fuzzy rules table is given in Table 1

S.No	PLR	SINR	RTD	RI
1	High	Low	High	Low

S.No	PLR	SINR	RTD	RI
2	Medium	Low	High	Low
3	Low	Low	High	Medium
4	High	Medium	High	Low
5	High	High	High	Low
6	Medium	Medium	High	Medium
7	Medium	High	High	Medium
8	High	Low	Medium	Low
9	High	Low	Low	Medium
10	High	Medium	Medium	Medium
11	High	Medium	Low	Medium
12	High	High	Low	Medium
13	Low	Medium	Medium	Medium
14	Low	Medium	Low	High
15	Low	High	High	Medium
16	Low	High	Low	High
17	Low	Low	Low	High
18	Medium	High	Low	High

Table 1 Fuzzy Rules

3.3 Fault Recovery Phase

The network recovers from the problematic nodes and safeguards the remaining functional nodes once the faulty node has been verified.

The following lists the steps that are part of the fault recovery module:

1. The RA at the PC broadcasts the Fault Recovery Request message (FR_REQ) to SCs, which contains the PC ID, faulty node ID, and detection time, if the PC receives the CONFIRM message from the initiators.
2. Every SC will look for the problematic node ID in its routing table after getting FR_REQ. The fault recovery response message FR_RES, which contains the PC ID, SC ID, and its path information towards the sink, is returned if it is present.
3. The PC will attempt to create a new route, excluding the problematic node, to replace the damaged route after receiving the FR_REP message from every SC.
4. The PC will send a MOBILITY information packet to the relevant SCs in order to create a new path towards the sink if no such route can be created.
5. On receiving the MOBILITY information packet, the corresponding SC move towards the position as specified by the PC.
6. PC will then resend the stored packets to the sink via the newly established route.

4. EXPERIMENTAL RESULTS

4.1 Simulation Setup

The proposed AFDR-ANFIS technique has been simulated in NS3. The simulation results demonstrated the effectiveness and originality of the used model. Here, the evaluation metrics packet delivery ratio (PDR), number of packets dropped, average residual energy computational cost and end-to-end delay are measured by varying the number of fog nodes. Besides, the performance of the proposed technique is compared with the ML based Fault Detection and Diagnostics (ML-FDD) [7] technique and CNN-LSTM-MLP based prediction of IoT sensor faults [10].

Number of nodes	20 to 100

Number of Faulty nodes	10% of the total nodes
Topology size	500m X 500m
MAC Protocol	IEEE 802.15.4
Traffic type	Constant Bit Rate
Traffic rate	50Kbps
Propagation model	Two Ray Ground
Antenna model	Omni Antenna
Initial Energy	12 Joules
Transmission Power	0.660 watts
Receiving Power	0.395 watts

Table 2 Simulation parameters

4.2 Results & Discussion

The performances of the techniques are evaluated by varying the number of devices from 20 to 100.

Number of devices	AFDR-ANFIS	CNN-LSTM-MLP	ML-AFDD
20	0.9582	0.9350	0.9220
40	0.9436	0.9261	0.9111
60	0.9402	0.9174	0.9071
80	0.9375	0.9040	0.8892
100	0.9289	0.8911	0.8705

Table 3 Results of PDR

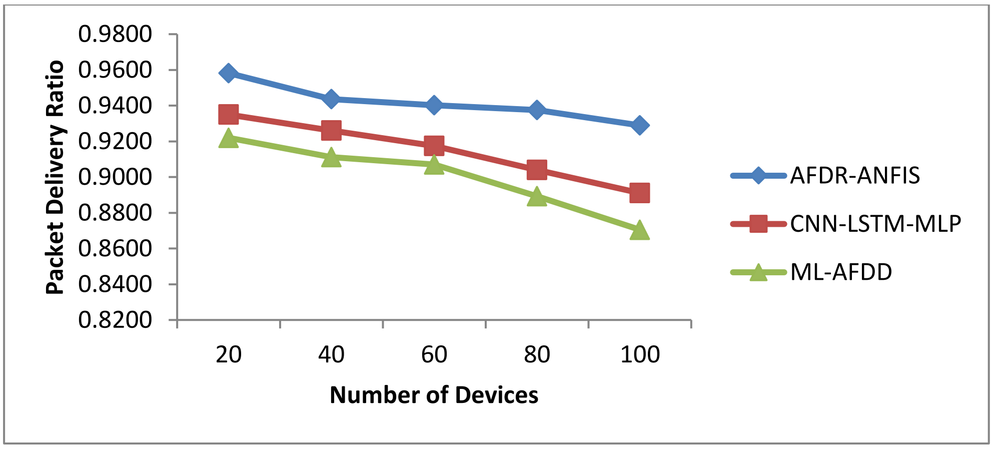

Figure 4 PDR for varying the devices

Table 3 and Figure 4 show the results of PDR for varying the devices from 20 to 100. As it can be seen from the figure, the AFDR-ANFIS attains 2.8% higher PDR than CNN-LSTM-MLP and 4.4% higher PDR than ML-AFDD.

Number of devices	AFDR-ANFIS	CNN-LSTM-MLP	ML-AFDD
20	2503	4007	4507
40	2851	4434	4834
60	3036	4889	5219
80	3511	5030	5230
100	3911	5302	5705

Table 4 Results of Packets drop

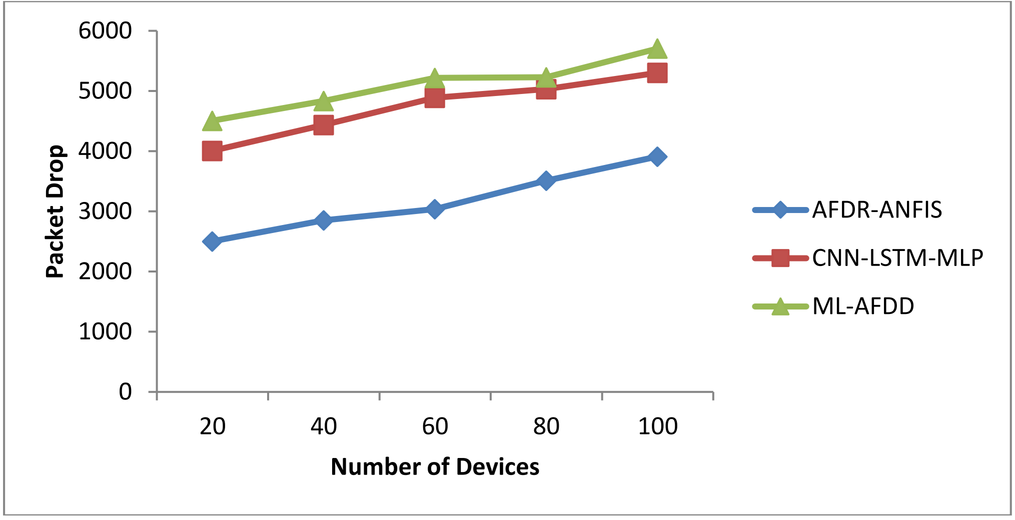

Figure 5 Packets dropped for varying the devices

Table 4 and Figure 5 show the results of packet dropped for varying the devices from 20 to 100. As it can be seen from the figure, the AFDR-ANFIS attains 33% lesser packet drops than CNN-LSTM-MLP and 38% lesser packet drops than ML-AFDD.

Number of devices	AFDR-ANFIS (Joules)	CNN-LSTM-MLP (Joules)	ML-AFDD (Joules)
20	7.65	4.98	5.68
40	7.44	4.76	5.76
60	8.19	5.19	6.36
80	8.70	5.49	6.46
100	8.72	5.71	6.73

Table 5 Results of Residual energy

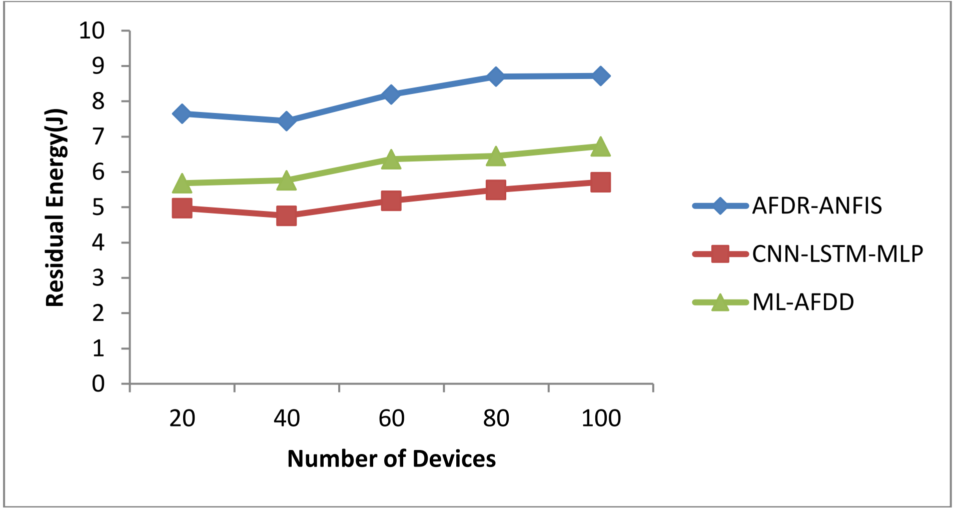

Figure 6 Residual Energy for varying the devices

Table 5 and Figure 6 show the results of average residual energy for varying the devices from 20 to 100. As it can be seen from the figure, the AFDR-ANFIS attains 35% higher residual energy than CNN-LSTM-MLP and 24% higher residual energy than ML-AFDD.

Number of devices	AFDR-ANFIS (Kb)	CNN-LSTM-MLP (Kb)	ML-AFDD (Kb)
20	228.37	501.85	341.85
40	246.54	547.91	377.91
60	264.7	552.05	422.05
80	270.94	565.26	445.26
100	273.32	595.32	475.32

Table 6 Results of Computational cost

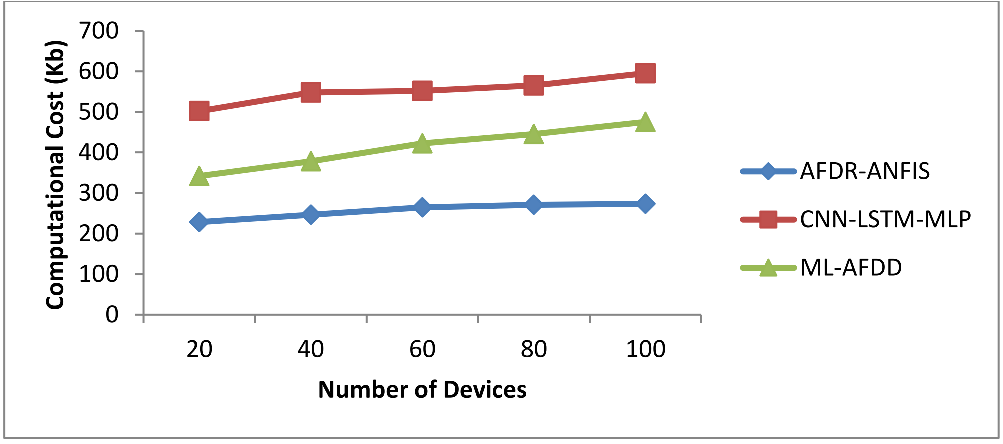

Figure 7 Computational cost for varying the devices

Table 6 and Figure 7 show the results of computational cost for varying the devices from 20 to 100. As it can be seen from the figure, the AFDR-ANFIS attains 53% lesser cost than CNN-LSTM-MLP and 37% lesser cost than ML-AFDD.

Number of devices	AFDR-ANFIS (%)	CNN-LSTM-MLP (%)	ML-AFDD (%)
20	98.37	95.5	94.85
40	96.54	94.71	94.12
60	95.71	93.45	92.15
80	95.14	92.66	91.58
100	95.03	91.72	90.52

Table 7 Results of detection accuracy

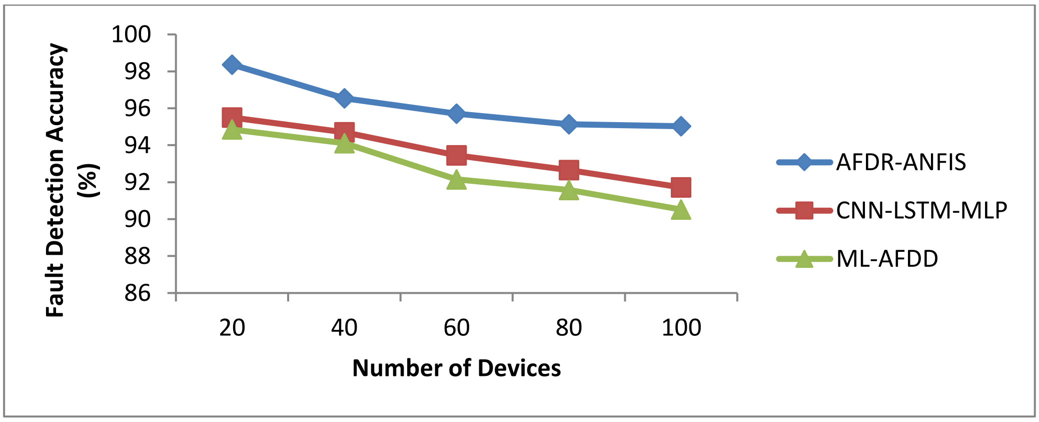

Figure 8 Fault detection accuracy for varying the devices

Table 7 and Figure 8 show the results of fault detection accuracy for varying the devices from 20 to 100. As it can be seen from the figure, the AFDR-ANFIS attains 2.6% higher accuracy than CNN-LSTM-MLP and 3.6% higher accuracy than ML-AFDD.

5. CONCLUSION

In this paper, AFDR-ANFIS framework is proposed for IoT environmental monitoring applications. In this framework, the device faults along a path are determined based on the ANFIS model. Fuzzy rules are provided based on the PLR, SINR and RTD metrics. In the fault recovery phase, the recovery agent at the primary path establishes an alternate fault-free route by excluding the faulty nodes. The performance of the proposed framework is compared with the ML-FDD) and CNN-LSTM-MLP techniques. Experimental results show that the proposed AFDR-ANFIS model attains higher packet delivery ratio, fault detection accuracy with reduced packet drops and computational overhead.

REFERENCES

1. G. Wang, Y. Liu, X. Chen, Q. Yan, H. Sui, C. Ma, and J. Zhang, "Power transformer fault diagnosis system based on Internet of Things," *J. Wireless Com. Netw.*, vol. 2021, no. 21, 2021.
2. S. Kumar, Kanchan, A. Kumar, P. Agarwal, and H. Maurya, "Internet of Things (IoT) applications and challenges: A review," *Int. J. Eng. Sci. Emerg. Technol.*, vol. 11, no. 2, pp. 359–367, Oct. 2023.
3. S. Sharma, K. Gupta, D. Gupta, S. Rani, and G. Dhiman, "An insight survey on sensor errors and fault detection techniques in smart spaces," *CMES*, vol. 138, no. 3, 2024, doi: 10.32604/cmes.2023.029997.
4. G. Ciaburro, "Machine fault detection methods based on machine learning algorithms: A review," *Math. Biosci. Eng.*, vol. 19, no. 11, pp. 11453–11490.
5. Manju and V. K. Srivastav, "Review of self-healing IoT networks based AI-driven fault detection and recovery," *Int. J. Appl. Behav. Sci. (IJABS)*, vol. 2, no. 1, 2025.
6. A. Maged, S. Haridy, and H. Shen, "Explainable artificial intelligence techniques for accurate fault detection and diagnosis – A review," *arXiv*, 2024.
7. W. Nelson and C. Dieckert, "Machine learning-based automated fault detection and diagnostics in building systems," *Energies*, vol. 17, no. 2, p. 529, 2024, doi: 10.3390/en17020529.
8. S. A. Y. Ahmed, A. A. Abubakar, A. F. M. Arif, and F. A. Al-Badour, "Advances in fault detection techniques for automated manufacturing systems in industry 4.0," *Front. Mech. Eng.*, vol. 11, p. 1564846, 2025, doi: 10.3389/fmech.2025.1564846.
9. A. Saeed, M. A. Khan, U. Akram, W. J. Obidallah, S. Jawed, and A. Ahmad, "Deep learning based approaches for intelligent industrial machinery health management and fault diagnosis in resource-constrained environments," *Sci. Rep.*, vol. 15, no. 1114, 2025, doi: 10.1038/s41598-024-79151-2.
10. A. M. Seba, K. A. Gemeda, and P. J. Ramulu, "Prediction and classification of IoT sensor faults using hybrid deep learning model," *Discov. Appl. Sci.*, vol. 6, p. 9, 2024, doi: 10.1007/s42452-024-05633-7.
11. T. Anwar et al., "Robust fault detection and classification in power transmission lines via ensemble machine learning models," *Sci. Rep.*, vol. 15, p. 2549, 2025, doi: 10.1038/s41598-025-86554-2.
12. E. H. Abualsauod, "Machine learning based fault detection approach to enhance quality control in smart manufacturing," *Prod. Plan. Control*, 2023, doi: 10.1080/09537287.2023.2175736.
13. H. Kaid, A. Al-Ahmari, and K. N. Alqahtani, "Fault detection, diagnostics, and treatment in automated manufacturing systems using Internet of Things and colored Petri nets," *Machines*, vol. 11, no. 2, p. 173, 2023, doi: 10.3390/machines11020173.
14. A. Chappa, K. D. Rao, M. Dhananjaya, S. Dawn, A. Al Mansur, and T. S. Ustun, "Machine learning based fault detection technique for hybrid multilevel inverter topology," *J. Eng.*, 2024.
15. D. G. Takale et al., "Fault detection in IoT sensor networks with XAI-LCS: Explainable AI-driven diagnosis for low-cost sensor," *J. Electr. Syst.*, vol. 20, no. 4s, pp. 46–54, 2024.