

Smart Fault Detection and Alert Mechanism for Energy Storage System

Diksha M M¹, Dhanapriya D², and Dr.P.Velmurugan³

¹Dept of EEE, St. Joseph's College of Engineering Chennai, India diksha534@gmail.com

²Dept of EEE, St. Joseph's College of Engineering Chennai, India dhanapriya0419@gmail.com

³Dept of EEE, St. Joseph's College of Engineering Chennai, India velmuruganp@stjosephs.ac.in

Abstract. Energy storage systems are really important for keeping the power supply stable. They can have a lot of problems that cause them to lose efficiency be unsafe and shut down without warning. The systems we have now need people to watch them or use methods, which are not very good and do not send warnings right away. We really need a system that can find faults and send alerts to make sure everything is reliable, safe and always working. The project we are talking about is the Smart Fault Detection and Alert Mechanism for Energy Storage Systems, which uses an Arduino Uno as the controller. This project uses sensors like a voltage sensor, current sensor and LM35 temperature sensor to keep an eye on the important battery parameters. The Arduino looks at the data. Shows it in real time on an I2C LCD display. If something is not right like if it goes beyond the limits the system uses a relay driver to protect it and a buzzer to warn people nearby. We also use a Node MCU Wi-Fi module to connect to the internet, which sends voltage, current and temperature readings to a platform so people can monitor and control it from far away. This means people can check the systems status at any time and get warnings if something goes wrong. The system we are proposing makes energy storage units more reliable, safer and more efficient which makes it really good for grids, renewable energy and industrial uses. It is, about making Energy Storage Systems better and Energy Storage Systems are very important. The Energy Storage Systems will be used in places and will make a big difference

INTRODUCTION

The use of Solar Energy and Wind Power is becoming more popular. This has also increased the need for Energy Storage System to be part of the Smart Grid. This is to make sure the grid is stable. Energy Storage System batteries, like the ones that use lithium-ion can get damaged easily because of much voltage, too much current and temperature changes. So it is very important to keep an eye on things, like voltage, current and temperature to make sure the Energy Storage System is working well to save money on maintenance and to prevent bad failures.

The old ways of watching over Energy Storage System batteries and keeping them safe are not very good. They usually involve someone checking on them by hand or a simple Battery Management System that just looks at limits. These systems do not try to predict what might go wrong. They are not watched from far away all the time. This has led to a lack of early fault detection and intervention, as faults are generally detected only after the occurrence of noticeable degradation. Therefore, there has been a recent trend in research towards developing intelligent and data-driven monitoring systems that are IoT-enabled.

Literature Survey:

The literature review is a summary of the previous studies related to intelligent battery management and fault detection through AI and IoT, the studies on SOC, SOH, and smart monitoring, and the gaps in the literature such as the absence of integrated IoT solutions for real-time fault response that the proposed system intends to fill.

1. Distributed Intelligent Battery Management System Using a Real-World Cloud Computing System – García et al. (2023)

Proposed a cloud-based distributed BMS using Petri Net modeling to improve real-time monitoring, scalability, synchronization, and fault tolerance in large-scale energy storage systems.

2. Li-ion Battery Management Systems Using Artificial Intelligence – Khawaja et al. (2023)

Showed that AI-powered machine learning models can improve State of Charge (SoC) and State of Health (SoH) estimation accuracy by a large margin compared to traditional approaches.

3. Towards an Intelligent Battery Management System for Electric Vehicle Applications: Dataset Considerations, Algorithmic Approaches, and Future Trends – Lyu et al. (2024)

Discussed SOH estimation methods and emphasized the need for high-quality datasets and hybrid data-driven algorithms for future intelligent BMS in EVs.

4. Intelligent Battery Management Strategy – K. V. R., D. B. et al. (2025)

Developed an adaptive intelligent BMS for HEVs that relies on real-time sensor inputs for predictive fault analysis, thus improving battery life and reliability.

5. An Artificially Intelligent Battery Management System – Suresh et al. (2024)

Developed a NodeMCU-based IoT BMS that incorporates voltage, current, and temperature sensors for real-time cloud monitoring of battery conditions.

6. An Intelligent Battery Management System (BMS) with EndEdge-Cloud Connectivity-A Perspective – Mulpuri et al. (2025)

Discussed a multi-layered edge-end-cloud BMS design that incorporates IoT, digital twin, and predictive models for intelligent battery management.

Although the recent intelligent BMS research using AI, IoT, and cloud monitoring is mainly focused on data analysis and prediction, the proposed Arduino-based smart fault detection and IoT alert system addresses the need for real-time sensing, immediate protection, and remote monitoring to ensure safe and reliable operation of Energy Storage System in the Smart grid environment.

Most of the IoT-based monitoring systems are primarily designed for data visualization and analysis purposes, but they may not necessarily provide automatic hardware-level protection, rapid local actions, and remote alerts. However, there is still a need for low-cost solutions that can be applied to distributed and industrial Energy Storage System applications. To address this issue, the proposed system is designed to use Arduino Uno and NodeMCU to monitor voltage, current, and temperature constantly, display real-time data on an LCD, turn on a relay and buzzer when there is a fault, and perform cloud-based remote monitoring.

EXISTING SYSTEM

The existing system proves that Proper thermal management of cylindrical portable battery packs is necessary for the safety and performance of electric vehicles and energy storage systems. The study emphasizes the use of hybrid nanofluids and phase change materials to enhance heat transfer and temperature management. These materials ultimately mitigate the weaknesses of traditional cooling methods by improving heat dissipation and stability. We need to work on cooling systems in the future. These hybrid cooling systems should have both passive methods. The hybrid cooling systems will be better if we make advancements in modeling, nanotechnology and materials science. This will help us make efficient thermal management solutions, for the hybrid cooling systems. The hybrid cooling systems will be very useful if we can make them work well.

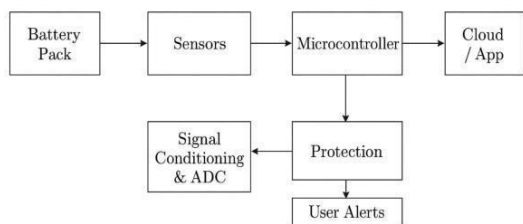


Fig.1. Block diagram of existing system

PROPOSED SYSTEM

This Smart Fault Detection and Alert Mechanism for Energy Storage Systems featured an Arduino Uno microcontroller that constantly monitors important battery data with a voltage sensor, a current sensor, and an LM35 temperature sensor. The Arduino processes the sensor input to determine whether there are irregularities, such as overvoltage, overcurrent, or overheating, and displays the live data on an I2C LCD screen so the user can easily check the data at each service. When the Arduino detects a fault, it engages the relay driver to disconnect the load and the buzzer to give a warning the user can hear. A NodeMCU Wi-Fi module provides IoT capabilities that retrieve the voltage, current, and temperature and send them to a cloud platform for remote monitoring of the energy storage system, as well as notify users about potential fault conditions. Overall, this all-digital relay and alert mechanism provides the ability to monitor, assess, and ensure reliable and safe operation of energy storage systems connected to a smart grid network and renewable energy systems.

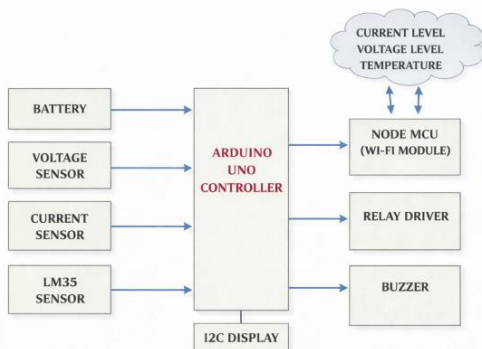


Fig.2. Block diagram of proposed system

WORKING PRINCIPLE

The Smart Fault Detection and Alert Mechanism for Energy Storage Systems works by keeping an eye on things in time. It can find faults. Send out alerts. The system uses sensors to check a things like voltage and current and temperature. These sensors are connected to the Arduino Uno microcontroller. The sensors send their data to the microcontroller. Then the microcontroller checks

this data against some limits that were set ahead of time. When the data is within these limits the Smart Fault Detection and Alert Mechanism for Energy Storage Systems works like it should. It also shows what is happening in time on the I2C LCD display. The Smart Fault Detection and Alert Mechanism, for Energy Storage Systems is always checking the data from the sensors. When the data is not within the range, for example, when the system is under over voltage, over current, and over temperature conditions, the Arduino microcontroller activates the relay driver to switch off the system and, at the same time, alerts the user using the buzzer. In the meantime, the system sends data on the current value to the IoT cloud monitoring system via the NodeMCU Wi-Fi module, which can alert the user when the user is off-site and cannot be reached. The system utilizes the advantages of fault detection to enhance safety and the reliability of the energy storage system.

The Smart Fault Detection and Alert Mechanism of the Energy Storage System works on the basis of continuous monitoring, real time analysis, and intelligent fault response. It uses an Arduino Uno as the central controller, which is interfaced with different sensors like a voltage sensor, current sensor, and LM35 temperature sensor to measure key battery parameters. These sensors are always checking the voltage, current and temperature of the energy storage unit. They send signals to the Arduino. The Arduino takes this information. Changes it into a form it can understand. Then it checks these signals to see if they are, within limits. The energy storage unit has to be working. If the voltage, current or temperature of the energy storage unit goes high. For example the voltage gets too high the current gets too strong or the temperature of the energy storage unit gets too hot. The Arduino sees this as a problem. As a result, the system activates a relay driver circuitry to disconnect the faulty part or turns off the battery to avoid further damage. In addition to this, a buzzer is turned on to produce an immediate local alarm, and an I2C LCD display displays real-time values and fault conditions.

Battery Temperature Monitoring System

The first purpose of the Battery Temperature Monitoring System is to monitor in real-time whether or not the temperature of the battery is within the guidelines. The Battery Temperature Monitoring System consists of an LM35 temperature sensor, Arduino Uno, I2C LCD display, and the buzzer. The LM35 temperature sensor is continually measuring the temperature of the battery surface and producing an output voltage directly proportional to the sensed temperature. The output voltage from the LM35 is sent to the Arduino Uno which is the main control unit for the Battery Temperature Monitoring System. The Arduino Uno converts the output voltage from the LM35 to a digital representation of the battery temperature using the built-in Analog-to-Digital Converter (ADC).

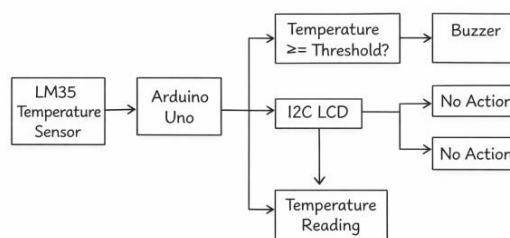


Fig.3. Working of battery temperature monitoring system

After reading the temperature from the LM35 sensor, the software running on the Arduino will compare the output to a predetermined threshold temperature. If the temperature reading from the LM35 exceeds the threshold temperature or equals the threshold, the Arduino will turn on the buzzer to alarm the user of a potential danger due to overheating. If the temperature stays below the threshold temperature, the battery temperature monitoring system will continue to monitor the LM35 output and not act. The LCD display is an I2C LCD that will allow the user to view the current real-time

temperature readings. This continuously monitored temperature helps users validate that the battery is not overheating and presents the battery while operating within the guidelines.

AI-Powered fault detection cycle

The cycle of AI-Powered Fault Detection is an intelligent and ongoing process that monitors a system, detects anomalies, and maintains safe operation. It begins with the reading of system parameters, such as temperature, voltage, and current, through the use of sensors connected to the system. The information is processed and used to update the AI baseline that defines the normal behavior of the system. The AI uses the baseline to continuously identify abnormal patterns and deviations from the baseline that could indicate system faults. If a fault is detected, it will automatically initiate safety responses, which in most cases involves isolating only the faulty parts or adjusting operation parameters to prevent them from being damaged. At the same time, the AI model will use its updates to reapply to the fault analysis process to ensure higher levels of diagnostic accuracy for future fault detections. Alerts or alarms will also be sent to a user or monitoring unit when behavior goes beyond the predetermined thresholds of acceptable operation for immediate user response. This process of learn and response leads to a more dependable system, reduces system downtime, and makes possible predictive maintenance through intelligent automation.

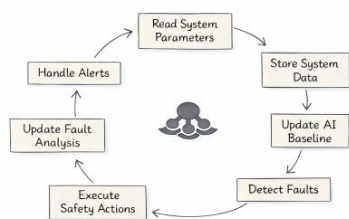


Fig.4. Working

ALGORITHM

Battery Temperature Monitoring and Alert System

Step 1: Initialize System

Start the LCD display and turn on the backlight. Set the buzzer pin as output and keep it off. Show the initialization message.

Step 2: Read Sensor Data

Read the analog value from the temperature sensor. Convert the ADC value to voltage. Convert the voltage to temperature in °C.

Step 3: Display Temperature

Show "Battery Temperature" on the LCD. Display the current temperature value.

Step 4: Check Over-Temperature Condition

If the temperature is greater than or equal to 45°C, turn the buzzer on (fault alert). Otherwise, keep the buzzer off (normal condition).

Step 5: Repeat Monitoring

Wait 1 second. Continuously repeat the process for real-time monitoring.

AI-Powered fault detection cycle

Step 1: Start

Power ON the controller.

Step 2: Initialize System

Initialize sensors, relays, buzzer, LED, Wi-Fi server, and load stored baseline values.

Step 3: Acquire Data

Read voltage, current, and temperature; store in history buffer.

Step 4: Learn Normal Condition

Compute average readings during the first cycles to form baseline.

Step 5: Detect Faults

Check threshold limits; compare with baseline to identify anomalies and patterns.

Step 6: Take Action

Critical → Disconnect supply, enable cooling, continuous alarm.

Warning → Cooling and intermittent alert. Normal →

Safe operation.

Step 7: Health & Alerts

Update LED status, log faults, estimate system health, and update web dashboard.

Step 8: Repeat

Continue monitoring every second.

METHODOLOGY

The proposed system has been developed by observing the battery voltage, current, and temperature through respective sensors connected to an Arduino UNO board, which will be able to acquire and process the data in real-time. The acquired data is displayed on an I2C display and transmitted in real-time to the cloud through a NodeMCU Wi-Fi module. Additionally, the system has been extended to offer battery protection in the event that any of the parameters go beyond the predetermined limit values.

A. System Architecture

The proposed method is an IoT-based battery monitoring and protection system that continuously tracks the voltage, current, and temperature parameters of the battery. The system architecture is designed with sensing units that have an Arduino UNO microcontroller as the processing unit, a NodeMCU Wi-Fi module for cloud communication, a relay driver circuit for protection, a buzzer for alerting, and an I2C LCD display. The system will acquire real-time sensor data, process that in the microcontroller, display the values locally, forward the information to a cloud platform, and trigger protective actions if abnormal conditions were detected.

B. Voltage Sensing Unit

The voltage sensing unit makes use of a resistive voltage divider circuit that will be connected across the battery terminals. Since the battery voltage is expected to be much higher than the allowed input voltage of the Arduino Analog Pins, it will be reduced to a level which will be considered adequate to measure (0 to 5V). After that, it will be connected to the Analog Input Pins of Arduino, where it will undergo Analog to Digital Conversion using its ADC. Calibration will take place to get the battery voltage in volts.

C. Current Sensing Unit

The present current sensing mechanism would make use of a Hall-effect-based sensor, like that of an ACS712, connected in series with the battery and load. This sensor provides an analog output voltage that is proportional to the current flowing through the conductor. The resulting signal is fed to the Arduino's analog input for digitization. The microcontroller operates on the obtained ADC values and converts the data into current values in amperes, so that the conditions can actually be monitored in real time for charging or discharging.

D. Temperature Sensing Unit

Battery temperature is monitored using an LM35 temperature sensor, which provides a linear output voltage proportional to the temperature in degrees Celsius (10 mV/°C). The sensor is placed near the battery surface to accurately detect thermal variations. The analog output from the LM35 is read by the Arduino and converted into temperature values using appropriate conversion formulas. This ensures effective detection of overheating conditions.

E. Data Acquisition and Processing

In this system, Arduino UNO is used as the central processing unit. It is always in a state of reading analog signals from the voltage, current, and temperature sensors by using its 10-bit ADC. The read voltage is then put into engineering units through a set of calibration equations. The data is then compared to a set of defined limits. The state of the system is defined as either normal or faulty, depending on the comparison.

F. Local Display Module

The Arduino interfaces the battery parameter display on a real-time basis with an I2C-based LCD display. Because it is I2C communication, wiring complexity has been reduced by utilizing only two lines for communication, SDA and SCL. It displays voltage, current, temperature, and system status, providing local visualization of battery parameters to the user without necessarily depending on remote communication.

G. Mechanism of Protection

The protection is provided with a relay driver circuit controlled via Arduino. When the sensed parameters exceed the predefined limits of safety, like over-voltage, over-current, or over-temperature, the microcontroller will trigger the relay to disconnect the battery from either the load or the charging source. At the same time, a buzzer triggers an audible warning. This will prevent battery damage and enhance operational safety.

H. IoT Communication Module

The remote monitoring can be done by using a NodeMCU (ESP 8266) Wi-Fi module. It sends the battery parameters sent by Arduino via a serial communication protocol. The NodeMCU module then joins a network and sends the values received from Arduino to a cloud server or an IoT platform. It provides remote monitoring with a websheet or a mobile app.

I. Operational procedure

The process is repeated indefinitely in a circular loop wherein we initialize all the hardware components and communication interfaces. We capture sensor readings, convert them to useful form, and display the readings on the LCD screen. We process the readings and compare them to safety limits to initiate protective actions if abnormal conditions exist, all while sending the readings to the cloud using NodeMCU for online monitoring.

RESULT AND DISCUSSION

Hardware:

The hardware prototype designed with Arduino Uno, ESP32 NodeMCU, and voltage, current, and temperature sensors successfully implemented the real-time monitoring of battery parameters in the Energy Storage System. The system immediately identified any abnormal situations like overvoltage and overheating, instantly triggering the relay and buzzer. In contrast to most existing IoT systems that are only concerned with data visualization on cloud platforms, the proposed hardware system offers immediate local protection, which is more efficient and cost-effective.

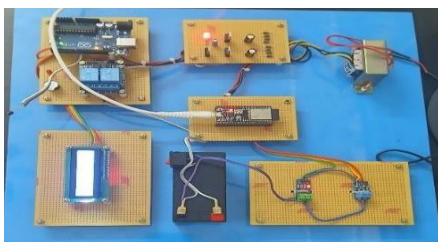


Fig.5. Hardware

Software:

The AI Fault Detection Dashboard successfully implemented real-time data visualization and analysis of voltage, current, and temperature parameters using IoT technology. By employing neural network and machine learning algorithms, the system was able to identify trends and predict potential fault situations in advance. Compared to most existing AI systems that are only dependent on cloud computing, the proposed software system integrates with the hardware protection system, which is more efficient in terms of early prediction and immediate action.

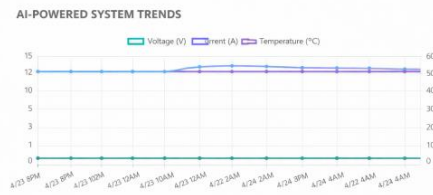


Fig.6. Software

CONCLUSION

This paper has demonstrated a smart fault detection and notification system for Energy Storage System using Arduino Uno and ESP32 NodeMCU with IoT connectivity, which allows for the continuous real-time monitoring of voltage, current, vibration, and temperature parameters, as well as instant hardware-level protection through the relay and buzzer notification system during faulty conditions. In contrast to most existing IoT and AI-powered battery management systems that are primarily designed for cloud-based data analysis, the proposed system provides a comprehensive solution for real-time sensing, instant local fault handling, and remote monitoring in a single, low-cost, and scalable solution. The proposed system has been shown to provide enhanced safety, reliability, and efficiency and is thus applicable for smart grid, renewable energy, and industrial energy storage systems, and can be further improved for predictive analytics and large-scale implementation.

FUTURE SCOPE

The proposed smart fault detection system can also be improved further with the inclusion of advanced battery management techniques like State of Charge (SoC) and State of Health (SoH) calculations using machine learning techniques. The system can be improved further to work with multiple battery modules used in systems like a microgrid or an EV charging station, as well as other sensors like gas, humidity, and vibration sensors for more effective and improved system safety and prevention of thermal runaway conditions. Apart from that, during the project, the fixed threshold method can also be replaced with more effective and adaptable protection algorithms. The proposed cloud monitoring system can be improved further with the inclusion of a specific mobile application with real-time alert systems and effective communication protocols. All of the above improvements will definitely ensure the creation of a smart and effective system, making it more suitable for industrial use cases or smart gridsystems.

REFERENCE

- [1] P. S. Hou, R. Chen, Q. Shi, Q. Huang, and Y. Wang, "Energy harvest of multiple smart sensors with real-time fault detection," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 2, pp. 1592–1606, 2023.
- [2] P. Wu, N. Lyu, Y. Song, X. Jiang, and Y. Jin, "Li-ion battery failure warning methods for energy-storage systems," *Chinese Journal of Electrical Engineering*, vol. 10, no. 1, pp. 86–100, 2023.
- [3] V. S. R. Kosuru and A. K. Venkitaraman, "A smart battery management system for electric vehicles using deep learning-based sensor fault detection," *World Electric Vehicle Journal*, vol. 14, no. 4, p. 101, 2023.
- [4] I. Rojek, D. Mikołajewski, A. Mroziński, and M. Macko, "Machine learning- and artificial intelligence-derived prediction for home smart energy systems with PV installation and battery energy storage," *Energies*, vol. 16, no. 18, p. 6613, 2023.
- [5] N. Rathika *et al.*, "IoT-based automatic braking control system for electric vehicles and monitoring system."
- [6] A. Rath and S. K. Lenka, "Smart speed control for EVs," *Journal of Technical Education*, p. 82.

- [7] P. Mane *et al.*, “Adaptive braking system for e-bikes using electromagnetic brakes,” *Computer Integrated Manufacturing Systems*, vol. 29, no. 5, pp. 237–253, 2023.
- [8] C. V. S. Babu, N. S. Akshayah, and R. Janapriyan, “IoT-based smart accident detection and alert system,” in *Handbook of Research on Deep Learning Techniques for Cloud-Based Industrial IoT*. Hershey, PA, USA: IGI Global, 2023, pp. 322–337.
- [9] I. Khaleque *et al.*, “IoT-based smart battery management and monitoring system for electric vehicles,” *AIUB Journal of Science and Engineering*, vol. 22, no. 2, pp. 181–188, 2023.
- [10] A. Jadhav, H. Rathod, and V. Kanade, “Smart battery parameter monitoring system for electric vehicles.”
- [11] R. Hemalatha, P. BharathiRajan, and R. Sathwik, “IoT-based automatic braking control system for EV vehicle and monitoring system,” in *Proceedings of the 2nd International Conference on Computer, Communication and Control (IC4)*, 2024.
- [12] S. Bergies *et al.*, “An IoT-based deep-learning architecture to secure automated electric vehicles against cyber-attacks and data loss,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2024.
- [13] H. Jamil *et al.*, “Analysis on the driving and braking control logic algorithm for mobility energy efficiency in electric vehicle,” *Smart Grids and Sustainable Energy*, vol. 9, no. 1, p. 12, 2024.
- [14] M. Sunil *et al.*, “Automatic braking system and driver safety using Arduino Uno Atmega 328p,” in *Proceedings of the 15th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, 2024.
- [15] M. Wagle *et al.*, “Proactive electric vehicle braking system,” in *Proceedings of the 4th International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET)*, 2024.
- [16] Y. He *et al.*, “Personalized automated braking for one-pedal driving in electric vehicles using model predictive control,” *IEEE Transactions on Transportation Electrification*, 2024.
- [17] G. Shini, J. L. Febin Daya, and P. Balamurugan, “IoT-based real-time monitoring of supercapacitors used in electric vehicles,” *Journal of Applied Research and Technology*, vol. 22, no. 1, pp. 42–51, 2024.
- [18] B. Dey *et al.*, “Design, simulation, building, and testing of a microcontroller-based automatic drowsiness detection, vehicle braking, and alert system,” *Journal of Engineering Research and Reports*, vol. 26, no. 3, pp. 208–223, 2024.
- [19] S. D. Bensam *et al.*, “An enhanced electric vehicle monitoring system using IoT,” in *Proceedings of the Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, 2024.
- [20] H. Dui *et al.*, “IoT-enabled real-time traffic monitoring and control management for intelligent transportation systems,” *IEEE Internet of Things Journal*, 2024.
- [21] P. Wu, N. Lyu, Y. Song, X. Jiang, and Y. Jin, “Li-ion battery failure warning methods for energy-storage systems,” *Chinese Journal of Electrical Engineering*, vol. 10, no. 1, pp. 86–100, 2023.
- [22] V. S. R. Kosuru and A. K. Venkitaraman, “A smart battery management system for electric vehicles using deep learning-based sensor fault detection,” *World Electric Vehicle Journal*, vol. 14, no. 4, p. 101, 2023.
- [23] I. Khaleque *et al.*, “IoT-based smart battery management and monitoring system for electric vehicles,” *AIUB Journal of Science and Engineering*, vol. 22, no. 2, pp. 181–188, 2023.
- [24] G. Shini, J. L. Febin Daya, and P. Balamurugan, “IoT-based real-time monitoring of supercapacitors used in electric vehicles,” *Journal of Applied Research and Technology*, vol. 22, no. 1, pp. 42–51, 2024.
- [25] I. Rojek, D. Mikołajewski, A. Mroziński, and M. Macko, “Machine learning- and artificial intelligence-derived prediction for home smart energy systems with PV installation and battery energy storage,” *Energies*, vol. 16, no. 18, p. 6613, 2023.
- [26] P. S. Hou *et al.*, “Energy harvest of multiple smart sensors with real-time fault detection,” *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 2, pp. 1592–1606, 2023.