Precision Agriculture using rechargeable harvest batteries and wireless sensor nodes and Markov models (2025)

Soham Anand Kulkarni , Prof. Suparna Chakraborthy

***Abstract*—.**

In the face of rising global food insecurity and the increasing need for sustainable agricultural practices, precision agriculture (PA) powered by wireless sensor networks (WSNs) has emerged as a transformative solution. This project explores the deployment of WSNs for smart irrigation, combining both software simulations in MATLAB and hardware implementations using sensors and Raspberry Pi systems. Randomly deployed sensor nodes monitor critical parameters like soil moisture and temperature, transmitting data using optimized communication paths. Energy efficiency is a central focus, incorporating solar energy harvesting models and sustainable battery systems to prolong sensor node lifespans. Advanced simulation models, including the effects of log-normal shadowing due to environmental obstructions, are integrated to enhance realism. Wireless communication technologies such as LoRaWAN are evaluated for reliable and low-power data transmission. The project further investigates energy harvesting strategies—Harvest-Use, Harvest-Store-Use, and Harvest-Use-Store battery systems—optimizing sensor autonomy. The integration of IoT, renewable energy, and AI insights aims to bridge the gap between traditional farming methods and modern smart agriculture, contributing towards enhanced productivity, resource efficiency, and environmental sustainability.

Index Terms*—Enter keywords or phrases in alphabetical order, separated by commas. Using the *IEEE Thesaurus* can help you find the best standardized keywords to fit your article. Use the thesaurus access request form for free access to the *IEEE Thesaurus https://www.ieee.org/publications/services/thesaurus.html.

I. INTRODUCTION

Agriculture is fundamental to the economic and social development of populations worldwide since the food of millions of people depend on agriculture. According to the Food and Agriculture Organization (FAO) of the United Nations, in 2017, more than 100 million people were food insecure. In developing countries, where this situation is more pronounced, Agriculture is a family activity in which farming processes don't make use of technology. The use of wireless sensor networks (WSNs) to provide precision agriculture (PA) has demonstrated positive results related to crop yields and resource management, which raises the need to determine the progress of research on the impact of these technologies. The global food insecurity, highlighting factors such as climate change, government policies, and underdeveloped agriculture. Family farming, which contributes significantly to food production in India often fails to reach markets, hence technological Advancements are crucial for improving productivity.

Precision Agriculture (PA) using Wireless Sensor Networks (WSNs) is identified as a key solution for optimizing resource use, enhancing yields, and mitigating environmental impact. However, small farmers face barriers to adopting such Technologies due to high costs, lack of training, and insufficient government support. The study analyzes 86 research papers on WSN applications in PA, focusing on their technical characteristics, utility, and implementation. Findings indicate that WSNs improve decision-making, crop monitoring, and sustainability. Despite their potential, challenges remain in accessibility and integration into small-scale farming. The study emphasizes the need for affordable, adaptable technologies and supportive policies to bridge the gap between scientific advancements and rural farming practices.

Wireless Sensor Networks (WSNs) play a crucial role in modern agriculture by improving productivity, reducing costs, and enhancing crop quality. They help farmers optimize resource management, leading to eco-friendly and sustainable farming practices. By enabling precise monitoring of environmental factors, WSNs reduce pesticide usage and promote organic farming while ensuring compliance with agricultural regulations. Water management is another critical application, as WSNs help optimize irrigation and address water scarcity, which is expected to worsen due to climate change. These networks are cost-effective, adaptable, and easy to implement, making them suitable for both small-scale and industrial farming.WSNs primarily monitor three categories: soil (77.91%), environment (72.09%), and plants (16.28%) [1] . Soil moisture, temperature, and pH are key parameters while environmental factors like temperature, humidity, and solar radiation also significantly impacts crop health.

Despite some challenges, WSN technology is proving highly effective in improving agricultural efficiency, sustainability, and food security. Its integration can bridge the gap between traditional farming methods and modern technological advancements, ensuring better yields and resource conservation. However, it is in great demand in agriculture for different kinds of irrigation, from flooding to dripping. The application of WSNs optimizes water management and enables a more equitable distribution. Working with WSNs has many advantages over other technologies: low cost , adaptability, an easy learning curve,inclusive technology with open and proprietary solutions, a good cost–benefit relation ,and the possibility of nondestructive tests of the technology in crops. For instance, water/irrigation management can be monitored with a high degree of accuracy using two variables: soil moisture and temperature . These parameters are specific and are present in many investigations. For each acquisition location, further analysis is performed, identifying individual monitored variables.

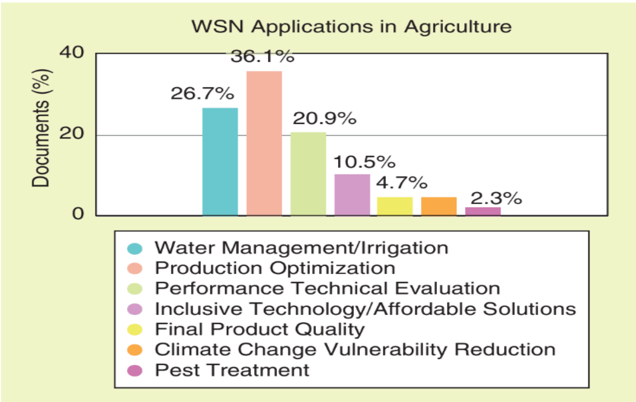


Fig. 1. WSN applications in Agriculture .

A . Project Overview

This project consists of both software and hardware components . This project is based on precision smart agriculture, which is a method of smart irrigation used in agriculture with the help of wireless sensor networks. The basic schematic of the project on precision agriculture will consist of sensors randomly deployed in the field following the **Flat network topology** .There will be a receiver at the boundary of the field to which all the data from the sensors will be transmitted using the most feasible paths based on the Euclidean distance between the sensors.[2] The data received by the receiver will be then transferred to the raspberry pi, which will be feeded with the threshold temperature and moisture levels required for optimal plant growth, and the water levels required for the same.The raspberry pi will then decide whether to start the irrigation process or not based on the sensor data received and the threshold values of water required for optimal plant growth. It can then control the water tank and irrigate the fields as per requirement without human intervention.

In this project, we will first start with the software component which involves MATLAB coding to deploy various sensors in a field and finding the area and range of communication for each sensor and using simulink to find the data transferred to the receiver and optimising their feasible paths. After that, we will use physical sensors available in the market to deploy the same code on actual plants, which makes the hardware component of the project. We will first start with listing all the available sensors which can be used in the project along with the types of communication methods available .We will mostly use temperature and soil moisture sensors to start with.
Energy model : We have used the following energy model . A sensor node consumes energy while sensing, transmitting, and receiving data. The energy spent while sensing is Esense , the energy spent in transmitting k bits of data over a distance d .

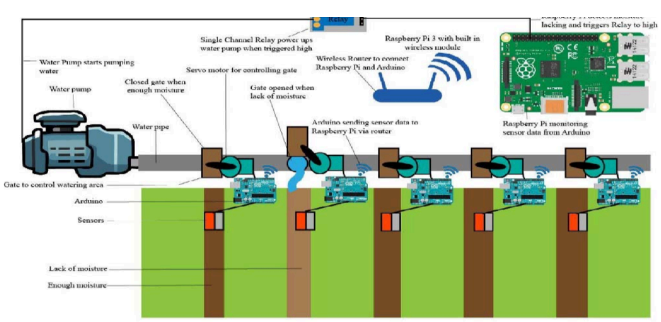


Fig. 2. Precision Agriculture diagram .

III. Literature Review and Previous work done

Wireless Sensor Networks (WSNs) are composed of spatially distributed sensor nodes used for monitoring environmental and physical parameters. Research on WSNs primarily focuses on improving energy efficiency, optimizing clustering methods, ensuring network reliability, and enhancing coverage strategies.

Energy Efficiency in WSNs
Energy consumption is a critical factor in WSNs, as sensor nodes are battery-powered with limited energy. Several approaches have been proposed to extend network lifetime, including clustering-based protocols such as Low-Energy Adaptive Clustering Hierarchy (LEACH) and its variants (LEACH-C, K-LEACH, DEEC, and SEP). The research highlights that clustering helps distribute energy consumption across nodes, enhancing longevity.

Reliability and Coverage in WSNs
Area Coverage Reliability (ACR) is a key metric for evaluating network performance. Research has introduced Monte Carlo simulation techniques to assess coverage reliability under multi-state nodes. Findings suggest that ensuring a balanced load distribution among cluster heads can significantly improve data transfer and network resilience.

Security Considerations in WSNs
Security remains a concern, especially in mission-critical applications such as military and healthcare. Studies suggest that clustering-based security frameworks can mitigate attacks while maintaining energy efficiency.

IV . MATLAB Models

A . Random sensors deployment :

First I started with deploying sensors randomly in a field of 10x10 using MATLAB code Then I used MATLAB to generate all possible paths between those sensors . The MATLAB code output for the same is given below :

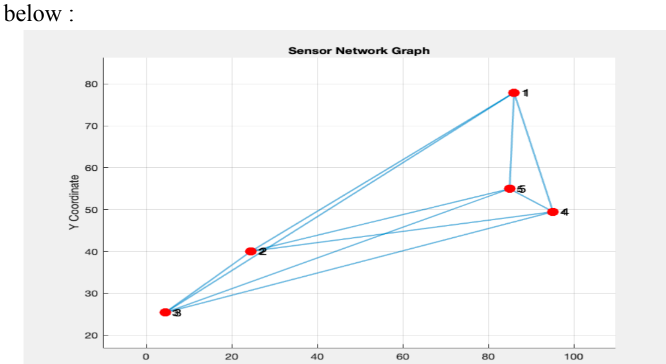


Fig. 3. MATLAB output of 5 sensor deploying code

B. Sensor diagram and field schematic

I have created a block in SIMULINK for a general temperature and soil moisture sensor . I have used MATLAB function blocks to create a sensor . For the field schematic 5 such sensors are attached to another function block which assigns random positions to these sensors in the field and shows their data transmission paths . Here the sensing range is taken as 40m , while initial energy of the sensor is taken as 2J .

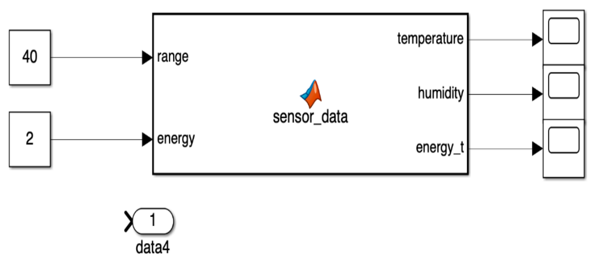


Fig. 4. Simulink sensor block diagram

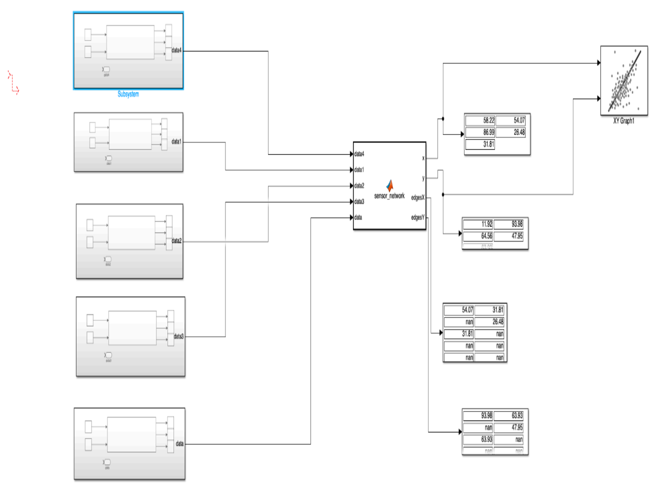


Fig. 5. Simulink field block diagram with 5 sensors

C. Area polygons for each sensor node

I used MATLAB Code to find the area sensing polygons for randomly deployed sensors in a field and also find the graph of the energy consumed and the area sensed. I have used the equations for energy as well as, K and euclidean distance d as per given in the reference research papers. This code helps us to analyse the transmitted energy as well as the consumed energy through the energy equations and euclidean distance of the randomly deployed sensors in a given field. The output shows a graph of randomly deployed sensors with their particular area sensing polygons with the help of Voronoi function and it shows the scatter graph of the sensors as red points .The output also shows a graph of the energy consumed Etx , Erx and gives us the final values of consumed energy and K in bits and d in meters . Using this, we can find the value of K in bits . The MATLAB code output for the same is given below:

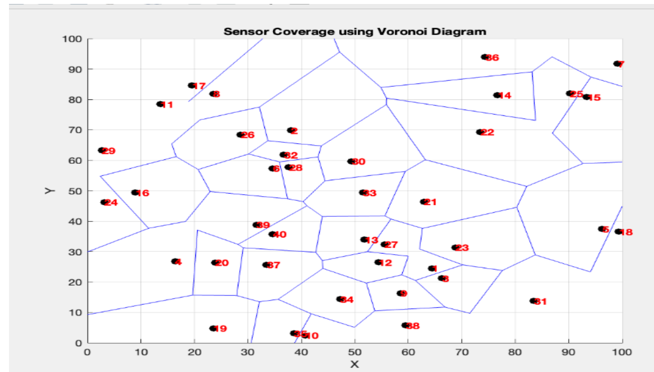


Fig. 6. MATLAB output of voronoi area polygons

D. Receiver and feasible paths

I have deployed sensors randomly in the field and in this MATLAB code the one of the sensors nearest to the boundary of the field will be chosen randomly as the receiver node . This is based on the assumption that the raspberry pie and controller will be outside the field and a sink node sensor nearest to the field boundary will be nearest to the controller and it will be within its sensing range . We will get the most feasible paths for data transmission based on the distance between the sensor nodes . We use flat network topology in this case .[3]

Till now whatever work was done , was done for only a single iteration But we need to keep in mind that the energy of the sensors will keep on reducing over time and few sensors will die out of energy after a given time interval if they are not recharged . Hence , this MATLAB code takes into account various iterations (1000 iterations in this case) and will show us the output continuously till 1000 iterations . It will show us the energy spent and energy levels of all the sensors at all times during every iteration and also show us the dead sensors if they run out of energy at any given period of time . The graph shows the sensors changing colour overtime as per their energy levels and it will show the area sensed by the sensors (K in bits) and their energy levels over various iterations. We also get to know the data received at the receiver node . The MATLAB code algorithm and the output for the same is given below:

```
Start
 ↓
Initialize parameters (field size, sensors, radius)
 ↓
Randomly generate sensor positions
 ↓
Compute Voronoi tessellation (V, C)
 ↓
For each sensor:
 └─> Check if Voronoi cell is bounded
     ├─ Yes:
     │   ├─ Calculate Voronoi area
     │   ├─ Calculate disjoint area (sensing area - Voronoi area)
     │   ├─ If disjoint area > 0:
     │   │   └─ Find intersection points (circle vs Voronoi)
     │   └─ Store data
     └─ No:
         └─ Skip or handle infinite cells
 ↓
```

Plot sensors and Voronoi polygons
↓
Create and display final results table
↓
End

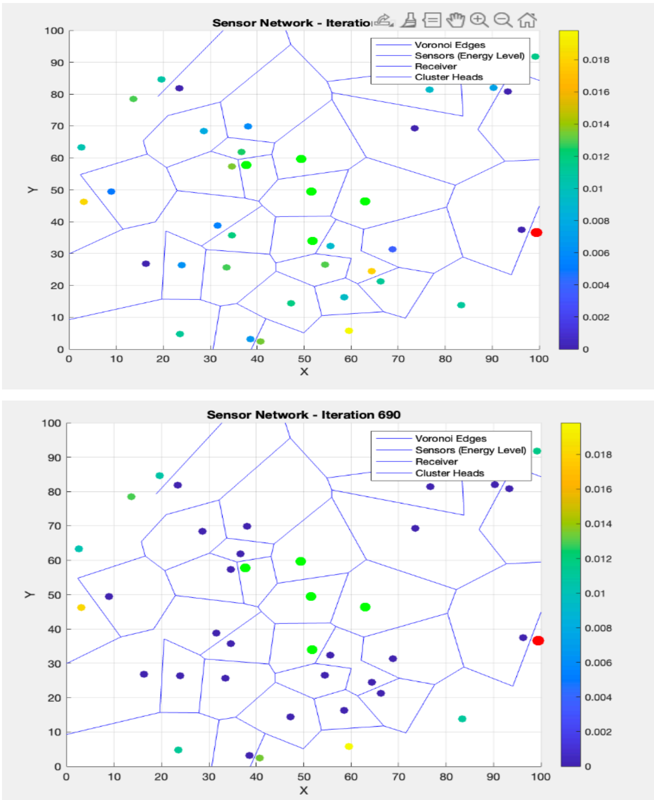


Fig. 7. MATLAB output of 1000 iterations

V. Log Normal Shadowing

In wireless sensor networks (WSNs), the log-normal shadowing model is used to represent the signal attenuation caused by obstacles, environmental factors, and multipath effects. Let's break it down and understand its effect on sensing range and power, along with key formulas.

Path Loss with Log-Normal Shadowing

The received power at a distance ddd from the transmitter can be modeled as:

$$Pr(d) = P\_t - PL(d) \qquad (1)$$

Where:

- Pr(d): Received power at distance ddd (in dBm or dB)
- PtP: Transmitted power (in dBm or dB)
- PL(d): Path loss at distance ddd (in dB)

The path loss with log-normal shadowing is given by:

$$PL(d) = PL(d0) + 10n\, log10\, (d/d0) + X\sigma \qquad (2)$$

Where:

- PL(d0); Path loss at a reference distance d0d_0d0 (usually 1 meter) , n: Path loss exponent (depends on the environment, e.g., 2 for free space, 3–5 for urban areas)
- d: Distance from the transmitter to the receiver
- Xσ: Zero-mean Gaussian random variable with standard deviation σ\sigmaσ (in dB) representing shadowing effects

A sensor's sensing range is typically the maximum distance d max where the received power is still above the receiver's sensitivity Pmin. Using the path loss model:

$$Pr(d) = Pt - PL(d) \qquad (3)$$

Setting Pr(d)=Pmin gives:

$$Pmin = Pt - PL(d\, max) \qquad (4)$$

$$dmax = d0 \cdot 1010nPt - Pmin - PL(d0) \qquad (5)$$

The shadowing term Xσ introduces randomness, causing sensors to have varying effective ranges. In a probabilistic sense, you'd treat d max as a random variable influenced by Xσ.

I have tried to add the log normal shadowing effect in the MATLAB Model code we used earlier . Hence , I have made a new function which implements a log normal shadowing model in the field . For this the function deploys several sensors randomly in the field and also applies a few obstructions, such as high-rise buildings or trees in the field as well . Now the sensors which are in the range of the high-rise buildings or trees like obstruction will get affected and show a log normal shadow effect . We can then find out the change, sensing radius, changed energy, and effect on the sensing range of the sensor in an around obstruction in the field . The MATLAB code algorithm and the output for the same is given below.

Start
↓
Set field, sensor, sensing parameters
↓
Change working directory
↓
Randomly place sensors
↓
Compute Voronoi diagram (V, C)
↓
Initialize area arrays
↓
For each sensor:
└─> If Voronoi cell bounded:
    └─> Compute Voronoi area and disjoint area
    └─> Find intersection points if needed
↓
Set parameters for Precision Agriculture simulation
↓

```
Call simulatePrecisionAgriculture function
 ↓
Plotting:
 └─> Voronoi polygons and sensor nodes
 └─> Random buildings (red squares)
 └─> Random sink (blue circle)
 └─> Sensing areas (gray circles)
 ↓
Add plot title, labels, and legend
 ↓
Print "Simulation Complete!"
 ↓
Create and display final result table
 ↓
End
```

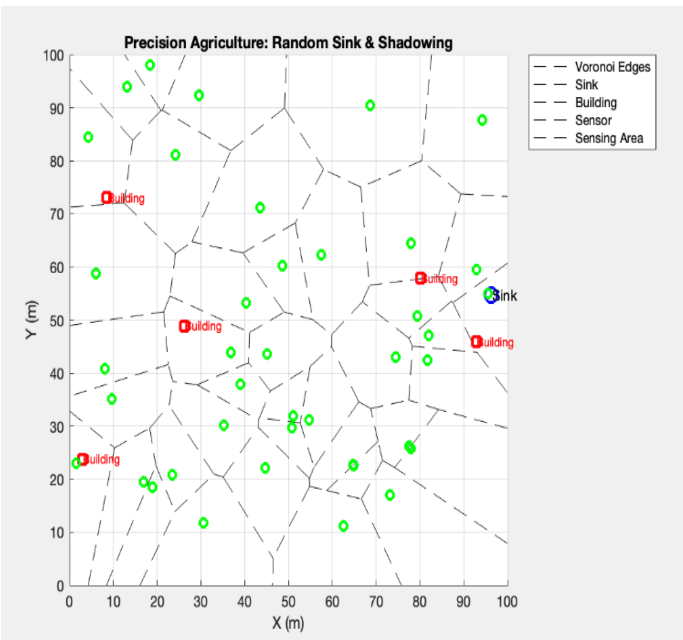


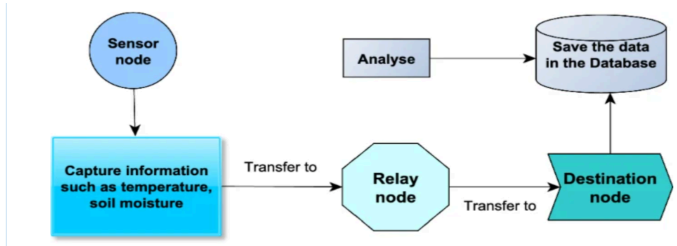
Fig. 8. MATLAB output of log normal shadowing

B. Work done in advanced technologies

The increasing demand for sustainable agricultural practices has driven innovation at the intersection of renewable energy, the Internet of Things (IoT), and Artificial Intelligence (AI). As global food security challenges grow, integrating these technologies has become pivotal in enhancing productivity, minimizing resource consumption, and ensuring environmental sustainability. This literature review synthesizes recent studies that explore the confluence of renewable energy, IoT, and AI in transforming traditional agricultural systems into smart agriculture frameworks.

Khernane et al. (2024) present a study focused on enhancing the lifespan of Wireless Sensor Networks (WSNs) through renewable energy harvesting. The primary objective was to extend sensor network lifetimes by harnessing ambient energy sources to power agricultural monitoring systems. The study highlights the importance of replacing conventional battery-powered sensors with self-sustaining ones through solar energy, vibration, thermoelectric, and ambient radio frequency energy.

Solar energy emerged as the most efficient method, offering 15 mW/cm³, compared to piezoelectric (330 µW/cm³), vibration (116 µW/cm³), thermoelectric (40 µW/cm³), and acoustic noise (960 µW/cm³) sources. The proposed system integrates a solar panel, regulator circuit (TCTL 431), triggering circuit (IC555), and a NiMH battery. The regulator circuit stabilizes the voltage at 5V, which is then used to charge the battery. Over time, the battery voltage increased from 0.8V after 10 minutes to 1.8V after 30 minutes.



Block diagram showing the functioning of the prototype work

In comparing wireless technologies like WiFi, Zigbee, and LoRaWAN, the study concluded that LoRaWAN achieved superior energy efficiency, extending the sensor network's lifespan significantly. The findings suggest that energy harvesting not only prolongs sensor lifetimes but also reduces maintenance costs associated with battery replacement. Hussein et al. (2024) explored the transformative potential of AI and IoT in agriculture by enabling precision farming practices. The study focuses on optimizing crop management through the deployment of IoT sensors, drones, and satellites, which provide real-time data on soil moisture, temperature, and crop health. AI algorithms analyze this data to inform farmers about optimal irrigation schedules, disease detection, and yield predictions.[4]

Rehman et al. (2024) proposed a comprehensive smart agriculture framework that integrates renewable energy resources (RERs), IoT-based environmental monitoring, and precision robotics. The study aimed to improve operational efficiency and reduce environmental impacts by combining solar power with battery energy storage systems (BESS) to provide a sustainable energy supply.

The framework consists of three key components:

Renewable Energy Integration (REI): Solar panels provide primary power, while BESS stores excess energy to ensure continuous power supply during low sunlight conditions. The energy optimization model is defined as:
The model balances energy generation, storage, and consumption, ensuring optimal resource allocation.

IoT-based Monitoring System: IoT sensors track real-time data such as soil moisture, temperature, and humidity, transmitting this data to a centralized platform via the Blynk IoT platform. The collected data informs dynamic load management strategies to optimize irrigation practices.

Future research should focus on scaling these solutions, enhancing accessibility for small-scale farmers, and addressing socio-economic barriers to adoption. These papers collectively demonstrate that the future of agriculture lies in smart, data-driven, and sustainable systems. As these technologies continue to evolve, they promise not only increased productivity but also a more sustainable agricultural landscape for future generations. Now that we have found out, how can we use Matlab code to implement precision agriculture by deploying sensors in the field virtually and also using the log. Normal shadow effect to consider a very realistic model of a farm having several obstructions in it as well. It is now time to find out how we can sustain this model. The sensors used in precision agriculture might be rechargeable or non-rechargeable. In order for us to make this model sustainable and energy efficient, we will have to use rechargeable batteries so that this model can run throughout the day and we don't have to change the sensors every time they die out of energy, once they are done with their full quota of battery level. Hence, we rely on rechargeable batteries and batteries, which use renewable forms of energy, which can be harvested through the environment using cost-effective methods . There can be many ways of harvesting energies, such as mechanical energy, solar energy, or even wind energy for that matter. We use solar energy because it is experimentally proven to be the most efficient and most feasible way of harvesting energy through the environment.

VI Harvest Batteries

1. HARVEST USE BATTERIES

These batteries immediately consume harvested energy without storing it. They are designed for systems that only need power when the energy source is actively generating electricity. These batteries are used to immediately utilize harvested energy without storing excess power. Energy is harvested from sources like solar panels, kinetic energy, or regenerative braking. Instead of storing energy for later use, it is directly supplied to a device or system. If more energy is needed than what is harvested, an external power source might be required. [5]

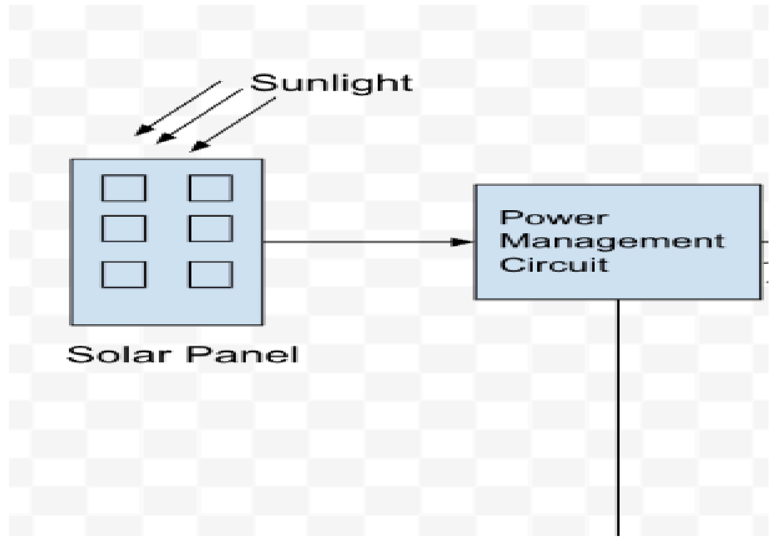


Fig 9 : Harvest use battery without storage

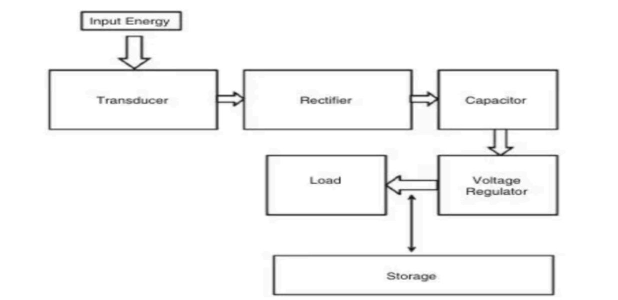
Technologies Used in Harvest Use Batteries

(a) Piezoelectric Energy Harvesting , how it works: converts mechanical stress (e.g., vibration, pressure, movement) into electrical energy. Example: Piezoelectric tiles in train stations generate electricity only when people walk on them. Self-powered RFID tags and IoT sensors use this energy instantly to transmit data. (b) Photovoltaic (PV) Direct Use : how it works: Solar panels convert sunlight into electricity, which is used directly without battery storage. Example:Solar calculators that only work when exposed to light. Solar-powered water pumps that only operate when the sun is shining. (c) Regenerative Braking with Direct Use : Converts kinetic energy into electrical energy and uses it immediately within a vehicle's system. Example:Formula 1 kinetic energy recovery systems (KERS) use braking energy to give an immediate power boost to the car.

Limitations of harvest use batteries : If energy input fluctuates (e.g., solar panels on a cloudy day), performance may be inconsistent. No backup power if the energy source is unavailable. Examples : Solar-Powered Devices: Solar calculators, solar-powered street lights, and solar irrigation pumps use energy as it is harvested , regenerative Braking in Electric Vehicles (EVs): Kinetic energy is harvested and immediately used to assist acceleration or recharge the battery , Wearable Tech & Smart Sensors: Smartwatches, fitness trackers, and IoT sensors use motion or body heat energy without needing battery storage , Piezoelectric Systems: Sensors in roadways or floors generate power when people walk or cars drive over them, and the energy is used instantly.

2. HARVEST STORE USE BATTERIES

These batteries first store harvested energy before supplying it for later use. This ensures energy availability even when the source is inactive (e.g., at night for solar power). These batteries first store harvested energy before supplying it for use. This system ensures that energy is available even when harvesting conditions are poor. Harvesting: Energy is collected from solar, wind, or other renewable sources. Storing: Excess energy is stored in batteries, capacitors, or other energy storage devices. Using: The stored energy is later used when needed. Fig 10 : harvest store batteries .



Technologies Used in Harvest Store Use Batteries

(a) Lithium-Ion (Li-ion) Batteries : Stores harvested energy in chemical form and releases it as needed.

Ex: Tesla Powerwall stores solar energy for home use at night. Smartphones & laptops store energy from the grid or solar panels for later use. Flow Batteries (Redox Batteries) : Stores energy in liquid electrolytes and releases it on demand. Example: Large-scale solar and wind farms use vanadium flow batteries to store excess energy. Off-grid microgrids rely on flow batteries for backup power. Supercapacitors for Short-Term Storage : Stores energy electrostatically and releases it quickly. Example: Regenerative braking in hybrid buses stores and later releases braking energy. High-powered flashlights use supercapacitors to store harvested solar energy.[6] .

Table 1 : Advantages & Disadvantages of harvest store use batteries

Feature	Pros	Cons
Energy Availability	Provides stored energy even when harvesting stops	Battery degradation over time
Efficiency	Can optimize energy use bas on demand	Some energy lost during storage
Scalability	Suitable for large and small applications	More complex and expensive than direct-use systems

Examples : Home Battery Storage Systems: Tesla Powerwall, LG Chem, and Bluetti battery systems store solar energy for later use, like at night. Off-Grid and Microgrid Energy Storage: Remote villages and farms store solar or wind energy in large batteries for continuous power. Hybrid Electric Vehicles (HEVs): Regenerative braking stores energy in the battery before using it later for acceleration. Military and Disaster Relief Energy Storage: Portable solar and wind batteries store energy for emergency use in war zones or disaster-hit areas.

4 Advantages and limitations of harvest store use batteries :

Energy is available even when harvesting stops. Provides stable and reliable power. Reduces dependency on the grid and lowers electricity costs.

Requires battery storage, which can increase cost and complexity. Batteries degrade over time, reducing efficiency.

3. HARVEST USE STORE BATTERIES

These batteries immediately use some of the harvested energy while storing the rest for later use. This approach provides both real-time power and backup storage.These batteries first use a portion of the harvested energy immediately and store the rest for future use. Harvesting: Energy is captured from a renewable source. Using: A portion of the energy is immediately consumed by the system or device. Storing: Any excess energy is stored in the battery for later use.

Technologies Used in Harvest Use Store Batteries

(a) Hybrid Energy Management Systems (HEMS) : Uses AI to determine how much harvested energy should be used immediately and how much should be stored. Example: Smart grids balance solar energy use and storage to reduce peak demand.

(b) Bidirectional Inverters (DC-DC & DC-AC Converters) : Converts electricity between direct use and stored battery power efficiently. Example: Vehicle-to-grid (V2G) systems in EVs let users consume stored energy and return excess to the grid.

(c) Hybrid Battery-Supercapacitor Systems : Uses a combination of lithium-ion batteries and supercapacitors to balance short-term and long-term storage. Example: Electric buses in China use supercapacitors for rapid energy use and batteries for long-term storage.

Advantages & Disadvantages of harvest use store batteries

Maximizes energy efficiency by ensuring no energy is wasted. Balances real-time energy use with storage for peak demand. Provides continuous power even if harvesting stops.

Limitations: Requires both real-time energy distribution and battery storage management, increasing system complexity. Energy conversion losses can occur when switching

Example : Electric Vehicles with Regenerative Braking: Some of the energy from braking is immediately reused to power auxiliary systems, while the rest is stored in the battery. Smart Grid & Energy Management Systems: Homes and businesses use solar energy in real-time, while storing excess power for later (e.g., during peak hours). Industrial & Commercial Battery Storage: Factories use solar power during the day and store extra energy in batteries to power operations at night. Hybrid Renewable Energy Systems: Wind turbines can directly power a facility, while any excess energy is stored in a battery for future use.

Feature	Harvest-Use Batteries	Harvest-Store Use Batteries	Harvest Use Store Batteries
Energy Harvesting	Direct use, no storage ,	Harvest first, then store	Harvest, use, then store
Energy Storage	No storage , Small, low-power devices	Stored in a battery for later use, M-L appl	Partial storage for dynamic energy management
Real-Time Energy Use	Yes, always	No, only stored energy is used	Yes, some energy is used immediately
Power Availability	Only when energy is harvested	Continuous, even if no energy is harvested	Continuous, balancing real-time use & storage
Examples	Solar calculators, kinetic energy devices	Home battery systems, off-grid storage	EV regenerative braking, hybrid energy systems
Best For	Devices that need simple, real-time power	Systems that require consistent power	Systems balancing real-time use & backup storage

Table 2 : Comparison of Harvest Battery Types :

WHICH SYSTEM IS BEST FOR YOU?

1. Choose HARVEST USE BATTERIES if:

You only need power when the energy source is active. You want a simple, maintenance-free system. Example: Solar garden lights, piezoelectric sensors.

2. Choose HARVEST STORE USE BATTERIES if:

You need power even when the source is unavailable. You want to store energy for later use (e.g., night-time solar power).Example: Tesla Powerwall, home battery storage, microgrids.

3. Choose HARVEST USE STORE BATTERIES if:

You want a balance between real-time energy use and long-term storage. You need a smart, flexible energy system for varying energy conditions. Example: Electric vehicles (EVs), smart grids, hybrid energy systems.

Now once we know which batteries are we going to use with the WSN notes now the question is about which technology should the sensors be using for communicating the sense data to each other and to the sink node. Usually wireless communication in wireless sensors is done through methods and technologies like DSRC, Bluetooth, LoRaWAN technologies. Let us look into it in detail.

VII Wireless Communication Technologies: DSRC, Bluetooth, and LoRaWAN

Wireless communication technologies have become integral to the modern world, enabling seamless connectivity across various devices and systems. Three important wireless technologies are Dedicated Short Range Communications (DSRC), Bluetooth, and LoRaWAN. Each of these technologies serves specific use-cases and is optimized for certain environments and requirements. This document provides an in-depth analysis of these technologies, comparing their functionalities, applications, strengths, and limitations.

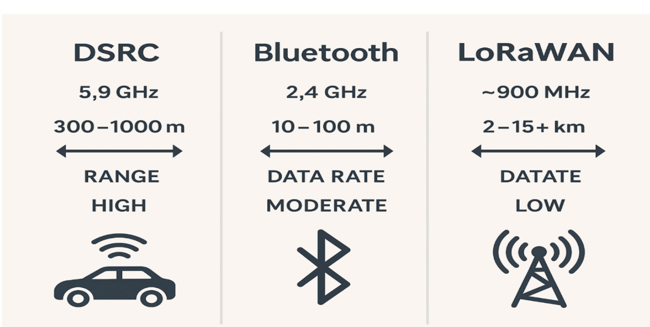


Fig. 11. Comparison of DSRC , Bluetooth , LoRaWAN

A . DSRC (Dedicated Short Range Communications)

Dedicated Short Range Communications (DSRC) is a two-way short- to medium-range wireless communication capability specifically designed for automotive use. It allows vehicles and infrastructure to communicate, forming the basis for intelligent transportation systems (ITS).

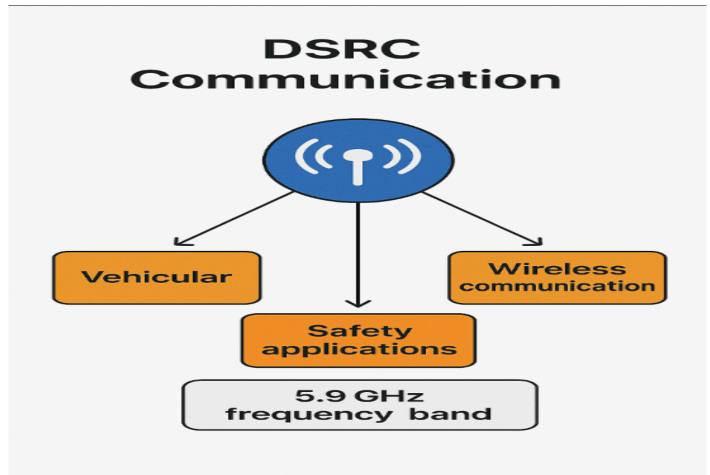


Fig. 12. DSRC Communication

Frequency Range: 5.850-5.925 GHz (USA) . Standards: IEEE 802.11p . Channel Bandwidth: 10 MHz per channel Data. Rate: 6 Mbps to 27 Mbps . Communication Range: Up to 1,000 meters . Latency: Extremely low, 10 milliseconds. Security: Strong encryption and authentication mechanisms

Key Features

Reliable communication under high-speed mobility . Rapid establishment of communication links. Low latency essential for safety-critical applications. Decentralized communication; no need for cellular networks

Applications : Vehicle-to-Vehicle (V2V) communication for collision avoidance , Vehicle-to-Infrastructure (V2I) communication for traffic management , Emergency vehicle preemption , Electronic toll collection

Advantages and Limitations of DSRC :

Advantages: High reliability and low latency . Strong focus on vehicular safety applications Limitations: Limited to automotive and infrastructure scenarios . High deployment costs for infrastructure.

B . Bluetooth

Bluetooth is a universal short-range wireless communication technology designed for exchanging data over short distances. Developed initially by Ericsson in 1994, Bluetooth has evolved through numerous versions and is now managed by the Bluetooth Special Interest Group (SIG).

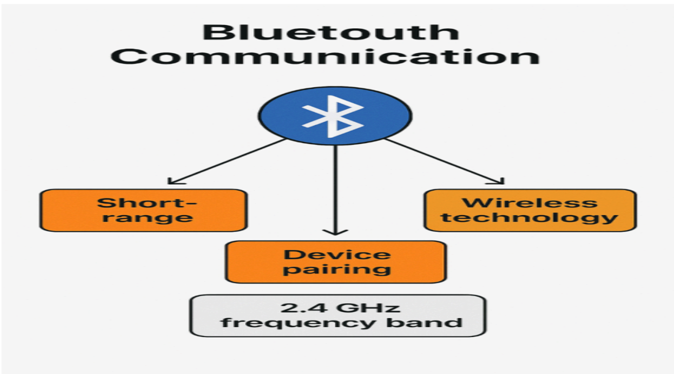


Fig. 13. Bluetooth communication features

Frequency Range: 2.400-2.4835 GHz (ISM band) . Standards: IEEE 802.15.1 . Data Rate: Classic Bluetooth: up to 3 Mbps , Bluetooth Low Energy (BLE): up to 2 Mbps .Communication Range: 1-100 meters depending on device class Latency: Moderate, lower with BLE .
Security: AES-128 encryption

Key Features:

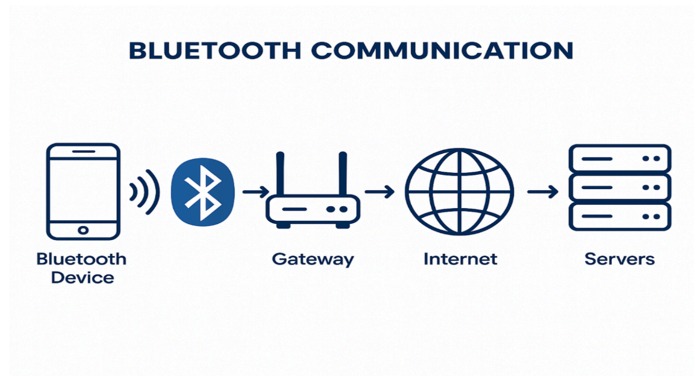
Low power consumption, especially with BLE . Universal adoption in consumer electronics . Support for point-to-point, broadcast, and mesh networking . Profiles for various use cases such as audio streaming (A2DP), health devices (HDP), and file transfer (FTP)

Applications : Wireless headphones and speakers . Wearable devices (e.g., fitness trackers, smartwatches) . Smart home devices (e.g., smart locks, thermostats) . Industrial applications (e.g., asset tracking).

Advantages and Limitations of bluetooth :

Advantages: Ubiquity and wide adoption . Low power operation with BLE , Interoperability across many devices and platforms. Limitations: Limited range compared to other technologies . Susceptible to interference in crowded 2.4 GHz environments

Fig. 14. Bluetooth communication



C. LoRaWAN (Long Range Wide Area Network)

LoRaWAN is a protocol for wide-area networking designed specifically for low-power, long-range communication. It is intended for IoT applications where sensors and devices need to send small amounts of data over large distances with minimal power usage.

Frequency Range: Sub-GHz ISM bands (e.g., 433 MHz, 868 MHz, 915 MHz). Standards: LoRaWAN specifications managed by the LoRa Alliance . Data Rate: 0.3 kbps to 50 kbps . Communication Range: 2 to 15 kilometers depending on the environment . Latency: Higher compared to DSRC and Bluetooth . Security: AES-128 encryption at network and application layers

Key Features : Extremely long communication range . Very low power consumption allowing years of battery life . Star-of-stars topology where devices communicate through gateways . Adaptive Data Rate (ADR) for optimizing battery life and network capacity

Applications : Smart agriculture (soil moisture sensors, livestock tracking) .Smart cities (parking sensors, street lighting control) . Industrial monitoring (pipeline monitoring, environmental sensors) . Remote healthcare monitoring

Advantages and Limitations of LoRaWAN

Advantages: Ideal for remote and rural applications . Long battery life for devices
Limitations: Low data throughput . Higher latency unsuitable for real-time applications

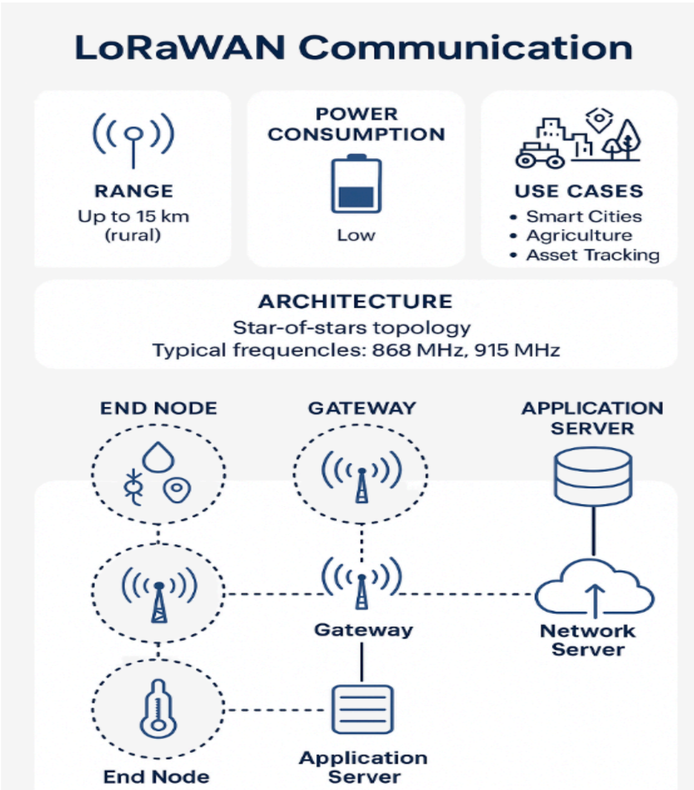


Fig. 15. LoRaWAN communication

Table 3 : Comparative Analysis between DSRC, Bluetooth, and LoRaWAN

Feature	DSRC	Bluetooth	LoRaWAN
Primary Use	Vehicle communication	Short-range device	IoT network communication
Frequency Band	0..9 GHz	2.4 GHz	Sub-GHz ISM Bands
Range	300-1000 m	10–100 m	2–15+ kilometers
Date Rate	High(6-27 mbps)	Moderate (up to 2 Mbps)	Low (up to 50 kbps)
Power Usage	Moderate	Low	Very low
Latency	Very low(~10 ms)	low to moderate	High

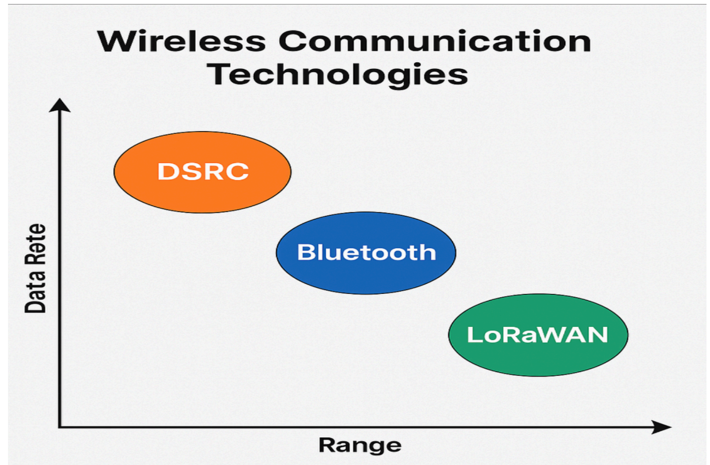


Fig16 : Technical Comparison

Cost and Deployment : Bluetooth is the cheapest to implement. LoRaWAN requires investment in gateways but is scalable. DSRC requires expensive infrastructure like RSUs.

Understanding the particular strengths of each technology is essential for deploying the right solution for specific use cases. The future will see these technologies either coexist or integrate with emerging innovations such as 5G and edge computing to meet the growing demand for wireless communication.

After this we will look into how we can integrate the harvest batteries which we studied before this in the report with the WSN sensor nodes using any of the above wireless communication technologies in the precision agriculture field deploying each sensor with a particular type of harvest energy battery. We will first start with harvest store use batteries following it with harvest use store batteries and finally harvest store batteries integrated with sense nodes. [7]

VIII Integrating Batteries with Wireless Sensor Nodes Setup -

We will be focusing on batteries harvesting solar energy as it is the most optimal and readily available energy harvesting option . Solar harvest storage systems are designed to capture solar energy and store it for later use.

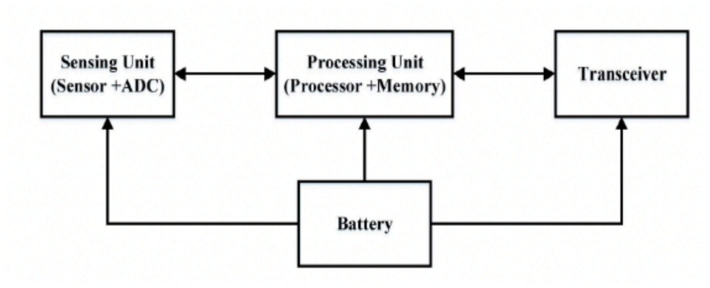


Fig 17 : Basic architecture of Sensor node

A . Energy-Harvesting Sensor Node (Solar + Battery) { Harvest store use }

System Components and Architecture

a. Solar Energy Harvesting Unit : Solar Panel (Photovoltaic Cell): Converts sunlight into electrical energy.
Maximum Power Point Tracking (MPPT): Ensures the solar panel operates at its maximum efficiency.
Charge Controller: Regulates the voltage/current from the solar panel to prevent battery overcharging.

b. Energy Storage : Battery (Li-ion / Li-Po): Stores excess energy harvested from the solar panel to power the system during low-light or night times. **Supercapacitor :** May be used for fast charge/discharge cycles.

c. Power Management Circuit (PMC) : Manages power flow from solar to battery and then to the sensor node. Includes DC-DC converters (buck/boost) to step up/down voltage levels to match sensor node requirements (usually 3.3V or 5V).

d. Wireless Sensor Node : Microcontroller (MCU): Controls sensor data acquisition, processing, and wireless communication.
Sensors: Soil moisture, temperature, humidity, etc. **Wireless Transceiver:** Sends data wirelessly (LoRa, ZigBee, Wi-Fi, etc.).

e. Charge Controller : A **charge controller** is an electronic circuit that **regulates the voltage and current** coming from a **solar panel** to **safely charge a battery**. **Input:** Solar Panel , **Output 1:** Connects to **Battery** for charging , **Output 2:** Connects to **Power Management Circuit** (usually a regulated output, e.g., 5V or 3.3V.

Here, the **battery is connected to the charge controller**, and the charge controller manages both: Charging the battery from solar, And outputting power to the PMC from either **solar or battery** (based on availability).It is needed because Solar panels can produce variable voltage. Batteries (like Li-ion) have **strict voltage ranges** (e.g., 3.0V–4.2V). Overcharging = fire risk or permanent damage , under-discharging = battery degradation.

Function: The **charge controller has 3 sets of terminals**:

- **Solar input (PV+ / PV–)**
- **Battery (BATT+ / BATT–)**
- **Load output (LOAD+ / LOAD–)**

When **solar is available**, power flows to: Charge the battery , and the power load When **solar is unavailable**, the charge controller **automatically switches** to drawing from the battery to power the load (the PMC in this case).

f. Power Management Circuit (PMC) : A **Power Management Circuit** is a circuit that takes power from the battery (or solar source) and **delivers stable, regulated voltage** to the wireless sensor node.

Table 4 : Main Functions of PMC

Function	Purpose
Voltage Regulation	Converts battery voltage (e.g., 3–4.2V) to a stable 3.3V or 5V
Load Current Management	Delivers clean power to microcontroller, sensors, and radio module
Switching/Logic Control	In advanced PMCs, it might include power routing, enable/disable signals for modules
Boost/Buck Conversion	Boosts or reduces voltage if needed (e.g., MT3608 for 3.3V boost from 1-cell Li-ion)

The PMC takes **raw battery voltage** (e.g., 3.0V–4.2V for Li-ion) and steps it down to a **regulated 3.3V** or 5V for the sensor node. This works well if the **battery voltage range** is within the **input range of the DC-DC converter** in the PMC. Inputs and outputs : Input (vin) : connects to battery or Charge controller , GND , Vout : connects to WSN node

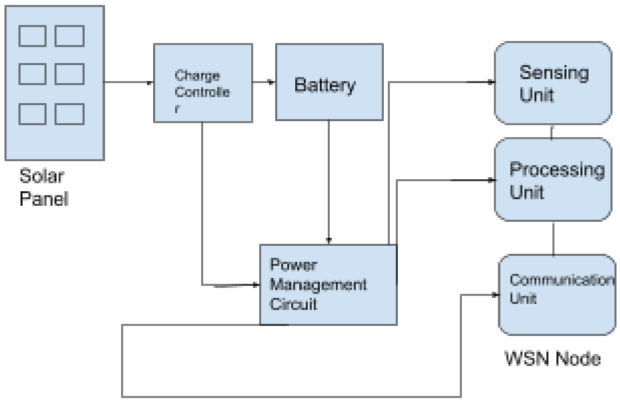


Fig 18 : Typical Architecture of Harvest store use batteries with WSN

Energy Equations

Instantaneous Power from Solar Panel:

$$Psolar = Eirr{\cdot}Apanel{\cdot}\eta panel \quad (6)$$

Where:

- Eirr: Solar irradiance (W/m²) — typically ~1000 W/m² in full sunlight
- Apanel: Area of the solar panel (m²)
- ηpanel: Efficiency of the solar panel (e.g., 0.18 for 18%)

Total Energy Harvested Over Time (t in seconds)

$$E\ harvested = P\ solar{\cdot}t\ sunlight \quad (7)$$

Where: tsunlight: Duration of sunlight exposure (hours or seconds) . This gives energy in Watt-hours (Wh) or Joules if converted 1 Wh=3600 J

Energy Stored in Battery

Battery Energy Capacity

$$E\ battery = V\ batt{\cdot}Cbatt \quad (8)$$

Where: V batt = Battery voltage (e.g., 3.7V) , Cbatt = Battery capacity (mAh)

Total Usable Energy

Every component has efficiency losses:

ηcharge = Charge controller efficiency (e.g., 90%)

ηPMC = Power management circuit efficiency (e.g., 85%)

Overall System Efficiency:

η total = η panel × η charge × η PMC (9)

E usable = E harvested × ηtotal (10)

Example Calculation

Say: Panel area = 0.01 m², efficiency = 0.18 , Irradiance = 1000 W/m² , Sunlight time = 5 hours .

Battery = 3.7V, 2000 mAh Node power: 60 mW avg (e.g., 30 µA sleep, 30 mA active for 1s every 60s)

Energy Harvesting

Energy is harvested using a solar panel. The power generated by the solar panel can be calculated using the equation:

Psolar = Eirr × Apanel × ηpanel

For example, if Eirr = 1000 W/m², Apanel = 0.01 m², and ηpanel = 0.18, then:

Psolar = 1000 × 0.01 × 0.18 = 1.8 W

To find the energy harvested over a period of time (say 5 hours of sunlight), we multiply the power by the duration:

E harvested = Psolar × t = 1.8 × 5 = 9 Wh

This is the raw energy collected from the environment.

Energy Storage

This harvested energy is stored in a rechargeable battery. The energy capacity of the battery in watt-hours is given by:

E battery = V batt × Cbatt / 1000

where V batt is the voltage of the battery, and Cbatt is the capacity in mAh. For example, if the battery has a voltage of 3.7 V and a capacity of 2000 mAh:

E battery = 3.7 × 2000 / 1000 = 7.4 Wh

To convert this to joules (since 1 Wh = 3600 J):

Ebattery_J = 7.4 × 3600 = 26640 J

This is the total energy available for the node's operation.

c. Energy Consumption by the WSN Node

Let’s assume the following power values:

Power in active mode, $P_{active} = 30\ mA \times 3.3\ V = 99\ mW$

Power in sleep mode, $P_{sleep} = 30\ \mu A \times 3.3\ V = 0.099\ mW$

(we are assuming standard values used in the reference papers mentioned in the end) If in a 60-second cycle, the node spends 1 second in active mode and 59 seconds in sleep mode, then:

$E_{cycle} = (P_{active} \times t_{active}) + (P_{sleep} \times t_{sleep}) = (99\ mW \times 1\ s) + (0.099\ mW \times 59\ s) = 99\ mWs + 5.841\ mWs = 104.841\ mWs = 0.0291\ mWh$

The average power consumption per hour would be: $P_{avg} = 0.0291$ mWh per minute = 1.746 mWh/hour

Battery Runtime Estimation : To estimate how long the battery will last, we divide the total energy in the battery by the average hourly power consumption:

$t_{runtime} = E_{battery} / P_{avg} = 7.4\ Wh / 0.001746\ Wh/h \approx 4238$ hours ≈ 176 days

Efficiency Consideration : The total usable energy depends on the efficiencies of the components: the solar panel, the charge controller, and the power management circuit. If*:*

$\eta_{panel} = 0.9$ (solar panel efficiency) , $\eta_{charge} = 0.95$ (charge controller efficiency) , $\eta_{PMC} = 0.85$ (power management circuit efficiency)
Then the total efficiency is:

$\eta_{total} = \eta_{panel} \times \eta_{charge} \times \eta_{PMC} = 0.9 \times 0.95 \times 0.85 \approx 0.72675$

So, the usable energy from the solar panel becomes:

$E_{usable} = E_{harvested} \times \eta_{total} = 9\ Wh \times 0.72675 \approx 6.54\ Wh$

This is the realistic energy that will be available for storage and use after conversion losses.

Working of Harvest store use batteries with WSN node

The **solar panel** captures sunlight and converts it into DC electrical power. Output voltage from the solar panel is variable (based on light intensity). This power is fed into the **charge controller**.

The **charge controller** regulates the voltage and current coming from the solar panel to: Prevent **overcharging** the battery , prevent **over-discharging** the battery when power is low. It ensures only safe and usable power reaches the battery. Simultaneously, it can **supply power to the system** if sunlight is sufficient.

Energy Storage (Battery) : The **battery** (usually Li-ion or Li-Po) stores excess power for **non-sunlight hours** (night/cloudy).
It acts as a **backup energy reservoir**.

Power Distribution (PMC) : The **Power Management Circuit**: Takes power from the **battery or solar panel**. **Regulates and steps down** the voltage (e.g., 3.3V or 5V) as needed. Ensures **stable supply** to the WSN node. **Power Consumption (WSN Node) .** The **Wireless Sensor Node** is powered by the PMC.

Modes of operation of WSN : A WSN node has **three main operating modes**:

1. **Active Mode** – The node is awake, sensing data, processing it, and transmitting it.
2. **Relay Mode** – The node forwards data from other nodes. It receives and then transmits.
3. **Sleep Mode** – The node shuts down most of its functions to conserve power.

The choice of mode at any given time depends on: **Energy availability** (harvested energy vs. stored energy), **Time of day** (solar intensity and harvesting), **Duty cycle** (sensing/reporting frequency),**Network role** (whether it acts as a sensor only or also a relay). In **Active Mode**, all units are used together. In **Relay Mode**, only the **communication unit** is needed—sensing and processing units are mostly idle. In **Sleep Mode**, all units are either off or operate in a deep sleep state (ultra-low power). Depending on the time of the day when the node is operating we can say :

Morning (6 AM – 10 AM)
Solar harvesting begins but is not yet at peak. **Energy availability is moderate**. Node enters **Active Mode** for short durations: **Sensing Unit** briefly turns on to collect environmental data. **Processing Unit** processes and compresses data. **Communication Unit** transmits data to the next node or sink.After its task, the node enters **Sleep Mode** to conserve energy. Behavior is **conservative**; energy usage is minimized to build up charge in the battery.

Midday (10 AM – 3 PM)
Solar intensity is highest, leading to **maximum energy harvesting**. **Battery is either charging or fully charged**. Node operates at **full duty cycle**. **Active Mode** is frequent and longer. **Relay Mode** is also active—nodes help forward others' packets.
Sensing Unit, **Processing Unit**, and **Communication Unit** are all used: Sensing and processing local data . Receiving and transmitting neighbor data. If required, data rates or sensing frequency may be increased due to surplus energy. Node behaves **aggressively**, utilizing the available energy for higher network throughput.

Evening (3 PM – 6 PM)

Solar intensity decreases, energy harvesting declines. Node assesses **battery level**: If the battery is still full → continues operating in **Active** and **Relay** modes. If the battery is getting low → prioritizes **Active Mode**, limits **Relay Mode**. **Communication Unit** may limit long-range transmissions. **Sleep Mode** is used more frequently.
Behavior becomes **adaptive**, depending on residual battery charge.

Night (6 PM – 6 AM)

No solar energy harvested. Node runs **entirely on a stored battery**. Enters **low-power duty cycle**: Occasional **Active Mode** for critical sensing tasks. May **skip Relay Mode** to save energy. Long periods of **Sleep Mode** dominate. Only **essential units** are powered:**Sensing Unit** may sample at longer intervals . **The Processing Unit** works minimally. **Communication Unit** transmits only when necessary. Node behaves **conservatively**, prioritizing essential tasks and extending battery life through the night. Table shows 3 modes and their behaviour.

Table 5 : Sensor nodes and behaviour

Mode	SU	PU	CU	Energy demand	Behavior
Active	ON	ON	ON	HIGH	Own data collection and transmission
Relay	OFF	MIN	ON	MEDIUM	Forwarding other nodes' data
Sleep	OFF	OFF	MIN	LOW	Conserves battery

Time of Day	Energy Source Status	Battery Level	Likely Operating Mode(s)	Behavior Summary
6 AM – 10 AM	Solar begins harvesting	Low → Moderate	Mostly Sleep + brief Active	Node wakes briefly for sensing. Relay mode avoided. Prioritizes storing energy.
10 AM – 3 PM	Peak solar harvesting	Moderate → Full	Active + Relay + brief Sleep	Node fully operational. High sensing, processing, and relay activities.
3 PM – 6 PM	Decreasing solar input	High → Moderate	Active + some Relay + Sleep	Continues sensing. Relay activity depends on remaining charge. Conserves energy.
6 PM – 10 PM	No solar input	Moderate → Low	Mostly Sleep + limited Active	Sensing done periodically. Relay mode usually skipped.
10 PM – 6 AM	No solar input	Low	Predominantly Sleep	Ultra low-power mode. Occasional sensing only if needed.

Fig 18 : Time of the day v/s sensor behaviour

B . Energy-Harvesting Sensor Node (Solar + Battery) { Harvest-use-store } :

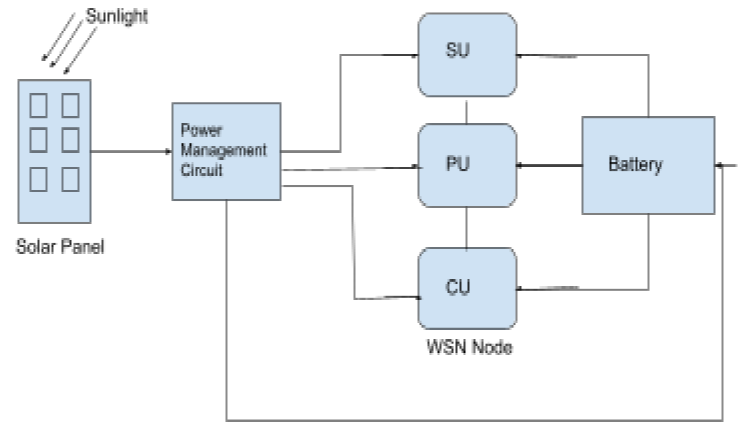


Fig 19 : Architecture of Harvest use battery with WSN node

Working

Solar Panel Operation: The **solar panel** (photovoltaic module) captures sunlight and converts it into **direct current (DC)** electricity.
This electricity is generally **low voltage** (typically 5V or 6V) and **variable**, depending on sunlight conditions (clouds, time of day, etc.).
The **amount of power** generated is proportional to: **Intensity of sunlight** (measured in watts per square meter) and the **efficiency** of the solar panel (how much sunlight it can convert into usable electricity

Power Management Circuit : The **power management circuit** (including **MPPT** and **DC-DC converters**) ensures that the energy harvested is appropriately regulated and directed to where it's needed. **MPPT (Maximum Power Point Tracking)**: The **MPPT controller** monitors the output of the solar panel to ensure that the panel operates at its **maximum efficiency**—that is, it finds the optimal voltage and current for power extraction. This is crucial since solar panels have varying voltage and current outputs throughout the day.

DC-DC Converter: The **DC-DC converter** then regulates the voltage to the required level (e.g., 3.3V or 5V) suitable for charging the **battery** or powering the **sensor node**.It either steps up or steps down the voltage from the panel, depending on the need.

Battery Charging:If there's **extra energy** (i.e., more power than the sensor node needs), it is directed to the **battery** for storage.The **charge controller** ensures the **battery is not overcharged** or **damage**

Power Flow to the Sensor Node: When the **solar panel** is active and producing more power than the node needs: The power is used **directly** to run the **sensor node**. Any **excess power** is stored in the **battery** for later use. If the **solar panel** is not generating enough energy (e.g., cloudy day or night), the **battery** automatically **supplies power** to the sensor node.

Battery Backup: Operation During Low Sunlight :

When the sunlight is insufficient (e.g., nighttime or cloudy weather), the battery plays a critical role: The battery that was charged during the day supplies power to the sensor node, ensuring that data collection and communication can continue even without direct sunlight.
The power management circuit ensures the battery is used efficiently:
If the battery voltage drops below a safe threshold, the system may enter a low-power mode, reducing sensor or communication activity to extend operation time.

Automatic Switching Between Power Sources :

When solar energy is available and sufficient, the sensor node is powered by the solar panel. If solar energy is insufficient (e.g., during the night or cloudy weather), the system automatically switches to draw power from the battery. This switching is typically handled by a diode or power path controller, which ensures that the battery is always ready to supply power without risk of reverse current flow (which could damage components).

Example : During the Day (Sunlight): The solar panel generates power. The power management circuit feeds power to the sensor At Night (No Sunlight): The battery supplies power to the sensor node. The sensor node continues to operate by drawing energy from the battery.

Energy Harvesting Equation

Solar-Panel Power Output

$PPV(t) = \eta PV \times APV \times G(t)$ (11)

PPV(t): instantaneous DC power from the panel (W)

ηPV: panel efficiency (unitless)
APV: panel area (m²)

G(t): solar irradiance at time ttt (W/m²)

Over one day, total energy harvested is

$E\ harv, day = \int t\ dawnt\ dusk\ PPV(t)dt.$ (12)

Energy Consumption by WSN Node with 3 Modes

Each node cycles through:

- Active mode: sensing + processing + transmission
- Relay mode: forwarding others' data
- Sleep mode: low-power idle stateG. Color Space

T cycle = t act + t sleep + t relay (13)

E node,day = 24.h.P load,avg (14)

E node,day = (24h/T cycle) (Pact.tact+ prel.tral + psleep.tsleep)

C . Solar Harvest-and-Use Batteries with WSN Nodes

The **harvest-and-use** battery without energy storage is also known as a **battery-less or capacitor-less system**, sometimes called an "energy-neutral operation" system if the power budget is precisely matched. In a **harvest-and-use** system, the solar panel powers the wireless sensor node **directly**. There is **no intermediate energy storage** like a battery or supercapacitor. This means: The node only operates **when there is sufficient solar energy** available. No energy is saved for nighttime or cloudy periods. The system must be highly energy-efficient and designed to work under **intermittent power**.

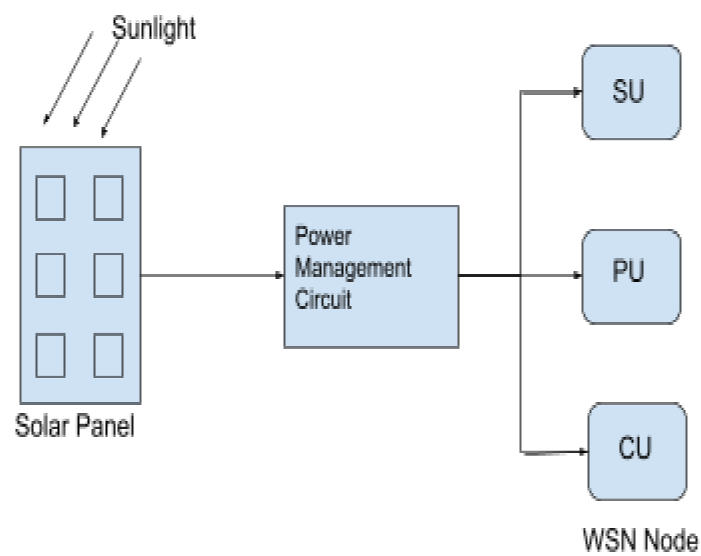


Fig 20 . Typical Architecture of Harvest use batteries with WSN node

Power Management Circuit (PMC) :
The **PMC** is the **brain of the power delivery system** in a solar-powered node. It's responsible for:

Function	**Why it's Important in Harvest-and-Use System**
Regulating voltage	Solar panel voltage varies with light; sensor nodes need stable voltage (e.g., 3.3V)
Protecting the node	Prevents under-voltage or over-voltage damage

Power-up sequencing	Ensures node turns on only when enough energ available
Voltage detection	Detects if minimum voltage threshold is met b allowing the node to operate
Optional cold-start logic	Keeps node OFF until voltage crosses startup threshold
Efficiency maximization	Optional DC-DC converters reduce loss, impro use of limited energy

Components of a PMC

a. **Diode or Ideal Diode :** Prevents **reverse current** from the node back to the solar panel. Protects against energy leakage when solar power is weak.

b. **Voltage Regulator :** LDO (Low Dropout Regulator) for minimal power systems (low efficiency, simple) . Provides stable voltage (e.g., 3.3V) regardless of solar fluctuation . DC-DC Buck/Boost Converter for higher efficiency . Buck: steps voltage down
Boost: steps voltage up.

c. **Undervoltage Lockout (UVLO) :** Prevents the system from turning ON until voltage exceeds a set threshold. Protects from brown-out behavior (e.g., failed transmission, memory corruption).

d. **Voltage Detector or Comparator :** Monitors solar panel output. When voltage crosses a threshold, triggers a switch to power the sensor node.

e. **Power Gating Switch (e.g., MOSFET) :** Acts like an **electronic switch**. Turns ON the sensor node only when the conditions are right (voltage, energy level). Prevents wasteful energy usage when the system can't fully operate.

Component	Purpose	Example Part
D1: Input Diode	Prevents reverse current from flowing into solar panel	1N5819 (Schottky Diode)
U1: Voltage Regulator	Regulates panel output to stable 3.3 V	AMS1117, TLV73333, or TPS62177
U2: Voltage Detector	Detects if voltage is above threshold before enabling power	TLV803S, MCP100, or LM8365
Q1: MOSFET Switch	Acts as a power switch controlled by the voltage detector	IRLML6344 or 2N7002
C1/C2: Capacitors	For filtering and stabilization of input/output voltages	0.1 µF, 10 µF

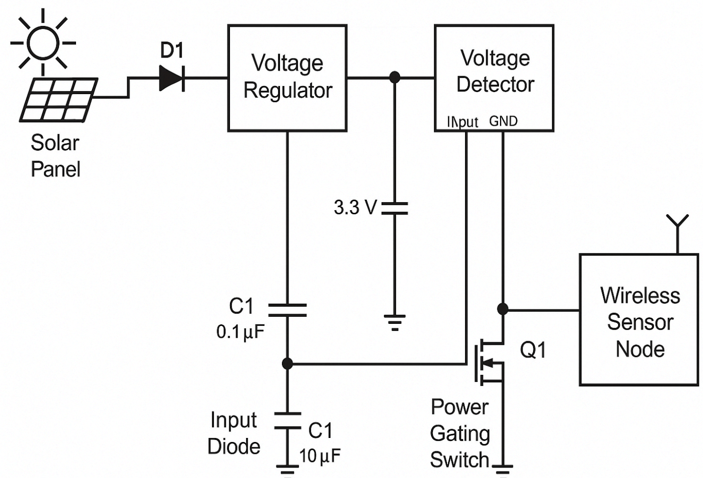


Fig 21 PMC circuit diagram

D1 protects against backflow of current when the panel isn't generating. **U1 (Vol Reg)** regulates the solar panel's variable output (say, 4–6 V) to a fixed 3.3 V . **U2**(Vol Det) only allows operation when voltage is above a safe level. **Q1** (MOSFET) acts like a gate—only allows current through when U2 says it's safe. **C1/C2 (Caps)** stabilize the voltage and filter out noise.

PMC Working

When Sunlight is Present: Solar panel outputs voltage (say 4–6 V depending on irradiance).
This goes to the **PMC**.
PMC checks the voltage level using UVLO or voltage detector. If voltage > threshold (say 3.3 V):
Power switch **closes**. Voltage regulator ensures a clean 3.3 V.
MCU, sensors, and radio turn on and run sensing + transmission. If voltage < threshold:
Node stays OFF and Prevents unstable behavior or data loss. U1 can be replaced by a DC-DC Buck converter if needed for >90% efficiency.

When Cloudy or Night:

Solar output is too low. PMC keeps the switch **open**.
Node is OFF. Waiting for light to return.

With respect to the components

1. **Sunlight available** → solar panel outputs voltage.
2. Voltage flows through **D1** (diode) to protect the circuit.
3. **Voltage Regulator (U1)** brings the voltage down to 3.3 V.
4. **Voltage Detector (U2)** checks: "Is input voltage high enough?"
   - If **yes**: it pulls the gate of **Q1 MOSFET** HIGH → turns ON the power to t
   - If **no**: Q1 remains OFF → node stays off.
5. **C1/C2** smooth the voltage to avoid dips/spikes during operation.

Why Not Just Power the Node Directly? Because solar voltage is **not stable**:

At noon: you might get 5.5 V. On cloudy days: drops to 2 V. Sensors and MCU need stable 3.3 V.

Without a PMC, you risk:
Over-voltage damage , Brown-outs , Unpredictable resets. PMC **acts as a voltage translator and gatekeeper**.

Working

1. Start of the Day: Sunrise — Solar Panel Begins Generating Power : As sunlight hits the **solar panel**, it begins converting light energy into **DC electrical energy**. The output voltage and current of the solar panel depend on : **Irradiance** (amount of sunlight) , **Temperature , Panel characteristics** (like open-circuit voltage, V_oc)
Example: If your panel is rated 6 V at peak sunlight, early morning might produce only 2–3 V.

2. Solar Output Reaches PMC: Flow into Input Diode (D1) : The first component that the panel connects to is the **Schottky diode (D1)**. To **prevent backflow** of current — if later in the day sunlight reduces, current from the system shouldn't leak back into the panel.
D1 is typically a **low forward voltage drop diode** (~0.2V–0.4V loss) to minimize power loss.

3. Voltage Stabilization: Voltage Regulator (U1) : The panel's output is **variable and unstable** without regulation. The **Voltage Regulator (U1)** (could be an LDO or Buck Converter): Accepts variable input (say 4 V to 6 V from the solar panel). Outputs a **fixed voltage**, e.g., **3.3 V**, required by: The **MCU** (microcontroller), **Sensors** (soil moisture, temperature, etc.), **Radio module** (LoRa, ZigBee, etc.).
Without this stage, the wireless node would behave **unpredictably** (brownouts, resets, memory corruption*)*.

4. Energy Condition Monitoring: Voltage Detector (U2): Next, the output from the voltage regulator is **monitored** by a **Voltage Detector** (U2). U2 continuously checks: **"Is the regulated voltage high enough to safely run the node?"**
If voltage > threshold (say 3.0 V): Voltage detector **asserts a control signal** (HIGH).

If voltage < threshold: Control signal remains **LOW**

5. Controlling the Power Switch: MOSFET (Q1) : The output of the **Voltage Detector** controls the **MOSFET switch (Q1)**. Q1 is normally OFF. When detector output goes HIGH (enough voltage available):
Gate of the MOSFET is driven HIGH.
MOSFET turns ON → **Connects 3.3 V output to the wireless node**.
Now the **Wireless Sensor Node** powers ON.

6. Cloudy Conditions or Sunset : As sunlight decreases: Solar panel output voltage drops. Regulated output voltage also starts to drop.
The Voltage Detector (U2) senses when voltage falls below threshold. It immediately turns OFF the MOSFET (Q1). This disconnects power from the sensor node safely.
Thus avoiding half-failed transmissions, corrupted data, unstable behavior. Since this is a harvest-and-use design:

There's no battery or large storage capacitor. Real-time solar energy directly powers the node.
When there's no sunlight → Node shuts OFF immediately.
This saves system complexity and cost but **limits operation time to sunnyhours**.

Energy Flow & Equations

Solar Power Output:

$$Psolar = G \cdot A \cdot \eta \tag{15}$$

G: solar irradiance (W/m²)

A: panel area (m²)

η: panel efficiency (typically 10–20%)

Voltage and Current:

$$Vout = Voc - I \cdot Rs \tag{16}$$

Voc: open-circuit voltage

Rs: series resistance of the panel

I: current drawn by node

Power Consumption of Node:
$Pnode = Vnode{\cdot}Inode$ (17)

MCU active ~5 mA @ 3.3 V → 16.5 mW

Radio Tx ~30–50 mA @ 3.3 V → 100–165 mW (brief)

Sensors ~1–10 mW

Energy Balance Condition: $Psolar \geq Pnode$

Harvested Energy Equation

The total energy harvested over a period [t0, t1]is the integral of power from the solar panel:

$Esolar(t0, t1) = \int t0t1Psolar(t)dt$ (18)

If you sample power every few seconds (like in simulation), you can approximate this as:

$Esolar \approx i = 1\sum NPsolar(ti){\cdot}\Delta t$ (19)

Where: Δt is your sampling interval (e.g., 10s)
N is the number of intervals between t0 and t1 .

Now energy per mode (active , sleep , relay) = P x t

At any time, the **energy harvested must equal or exceed energy consumed**:

$Esolar(t) \geq Eactive(t) + Erelay(t) + Esleep(t)$ (20)

Since this is a **harvest-use system (no battery)**, the equation is **instantaneous** — not cumulative :

$Psolar(t) \geq Pmode(t)$ (21)

This is the **core condition** for selecting the operating mode.

Using **thresholds**, the node selects a mode based on real-time power availability:

$$P_{solar}(t) = \begin{cases} < P_{sleep} & \Rightarrow \text{Node OFF (Deep Sleep)} \\ \geq P_{sleep} \text{ and } < P_{relay} & \Rightarrow \text{Sleep Mode} \\ \geq P_{relay} \text{ and } < P_{active} & \Rightarrow \text{Relay Mode} \\ \geq P_{active} & \Rightarrow \text{Active Mode} \end{cases}$$

fig 22 Sensor mode conditions

Mode	Power Required (Typical)	Activities
Sleep	< 0.1 mW (100 µW)	RTC, watchdog, memory retention
Relay	5–10 mW	MCU @ low freq, radio RX+TX only
Active	30–60 mW	Sensors, ADCs, processing, radio TX

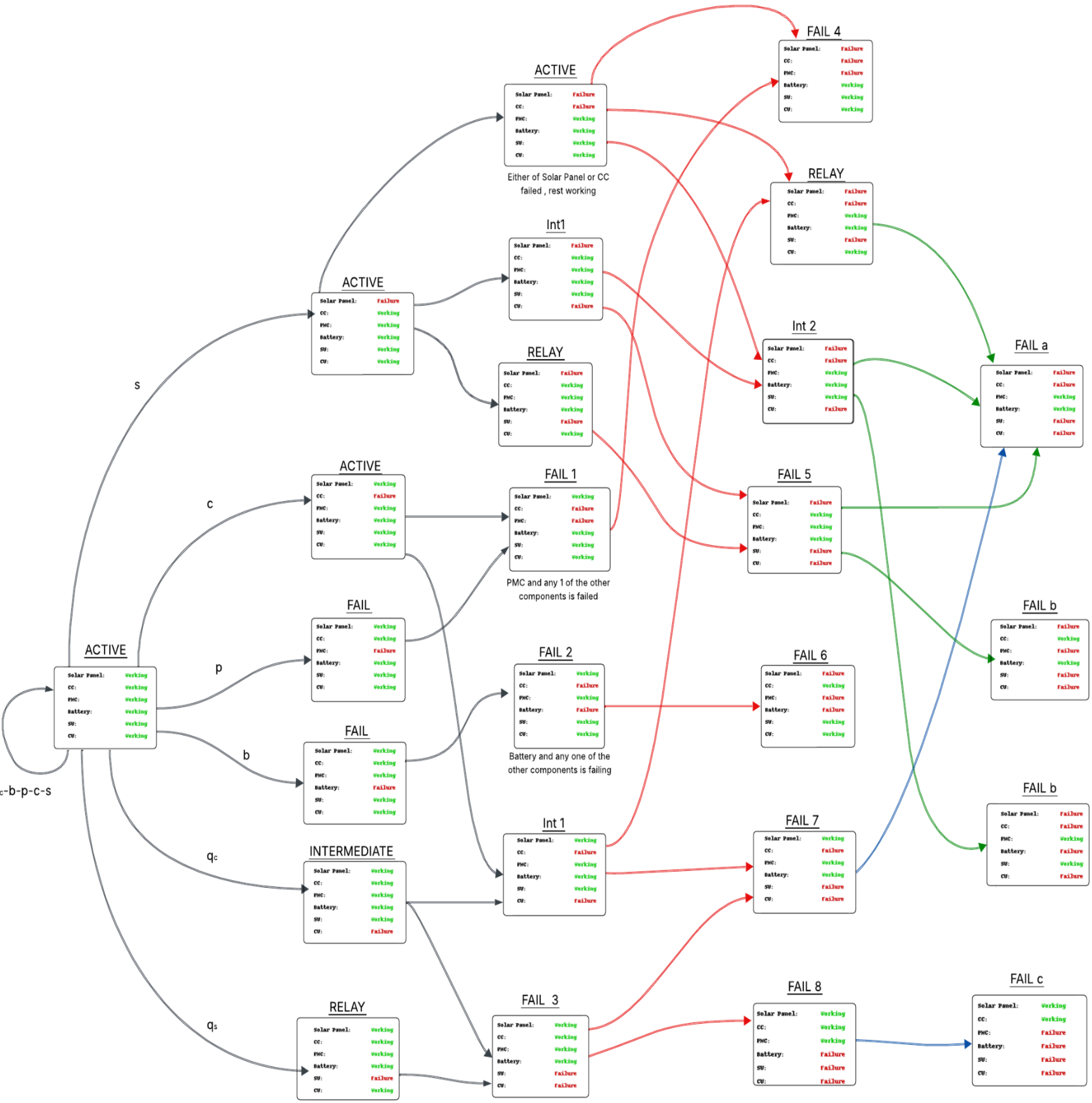
The node (or the PMC in hybrid designs) uses **voltage levels** or an internal **energy prediction model** to determine:

- If voltage ≥ threshold1 → enter Active (peak mornings , noon)
- Else if voltage ≥ threshold2 → enter Relay (early mornings and late evenings)
  Else → enter Sleep (dawn and before , dusk and after, late night)

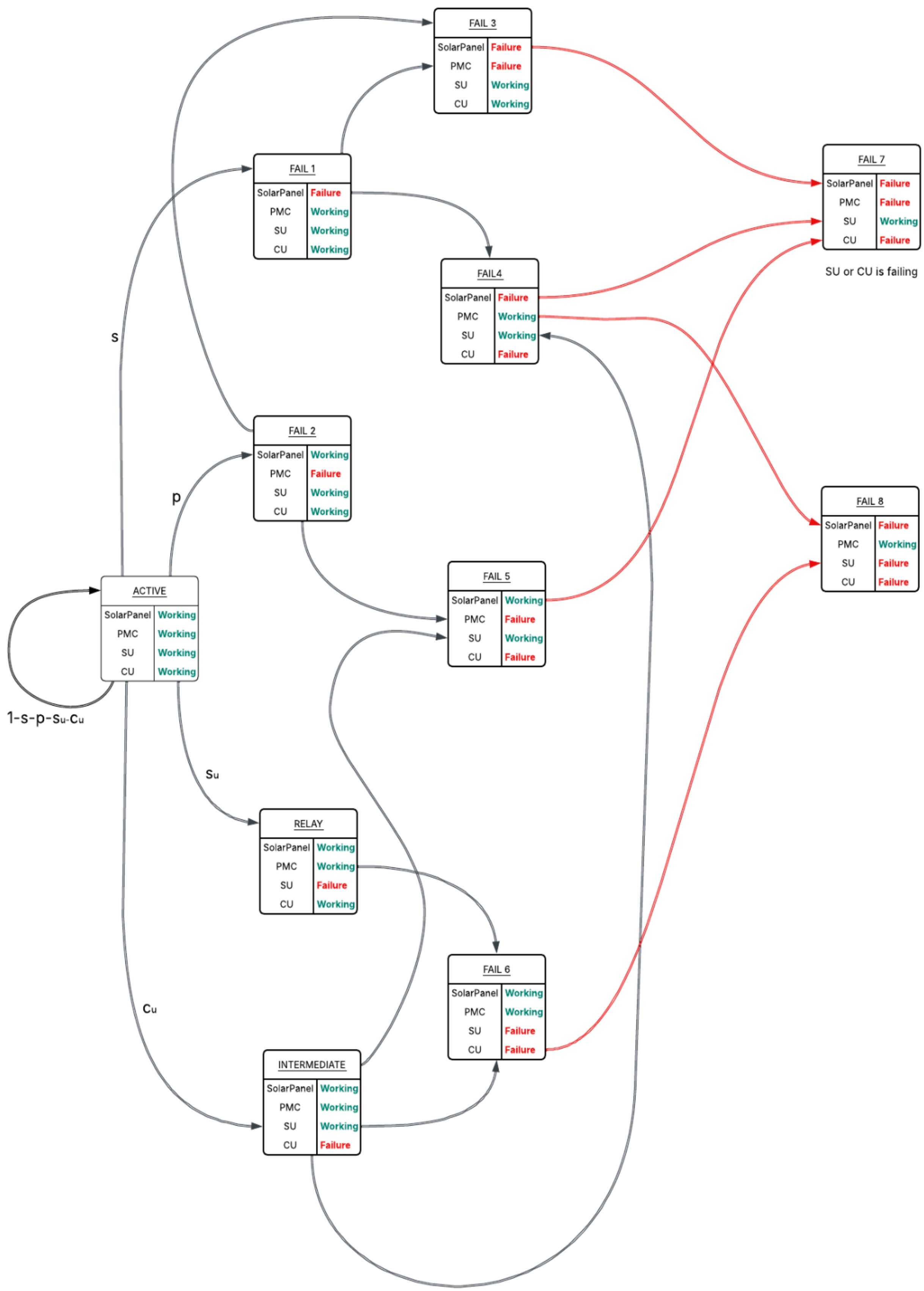
IX Generating Markov Model Diagrams for all three types of batteries

The construction of the Markov models for the above battery setups involves the following steps: first, discrete energy levels are defined as states, representing the charge level in the battery (for HSU and HUS) or instantaneous energy availability (for HU). Second, probabilities are assigned to transitions between states based on energy harvesting patterns (e.g., modeled as a stochastic or Poisson process), energy consumption rates of the sensor node, and battery storage behavior and efficiency (where applicable). Third, model formulation is customized for each configuration: in HSU, transition probabilities account for harvesting followed by energy storage and eventual usage; in HUS, immediate usage is prioritized and remaining energy is stored based on battery capacity; and in HU, transitions depend solely on whether the harvested energy meets current consumption requirements. Fourth, a transition probability matrix is constructed for each setup to model the dynamics of energy flow across states. Finally, steady-state or time-dependent probabilities may be derived to evaluate key metrics such as the probability of energy outage, battery depletion, and average operational time. We eliminate similar failed states caused due to similar failing conditions.

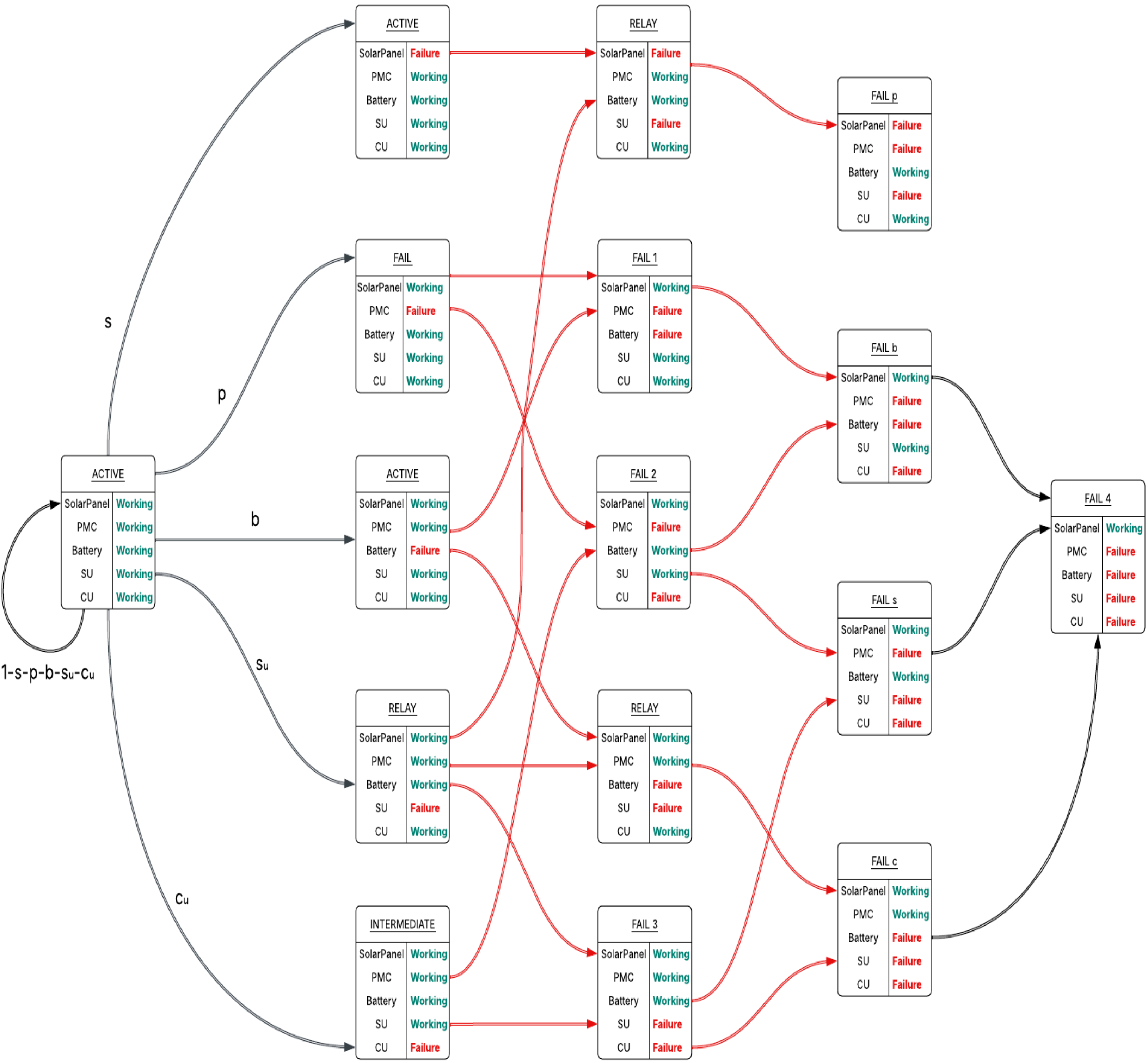
A) Markov Model for Harvest Store Use Battery



B) Markov Model For Harvest Use Battery



C) Markov Model For Harvest Use Store Battery



X Conclusion

This report presented a comprehensive framework for implementing precision agriculture using wireless sensor networks (WSNs) powered by solar energy and supported by various rechargeable battery configurations. The project encompassed both simulation-based and theoretical components, focusing on energy efficiency, sustainable power management, and robust system behavior under real-world conditions.The software simulations, developed in MATLAB and Simulink, enabled the virtual deployment of sensor nodes in a field and provided detailed insights into communication range, energy consumption, and sensor lifespan. By incorporating concepts like Voronoi tessellation and log-normal shadowing, the simulations modeled realistic agricultural environments, accounting for irregular sensing zones and environmental obstructions such as buildings and trees.

A major contribution of this work lies in the development and comparative analysis of discrete-time Markov models for three energy management strategies: Harvest–Use (HU), Harvest–Store–Use (HSU), and Harvest–Use–Store (HUS). These models provided a structured way to predict the probabilistic behavior of sensor nodes with respect to energy harvesting, usage, and depletion. By constructing the state transition probability matrices and analyzing energy flows, the models offered valuable insights into the sustainability and operational efficiency of each configuration.The integration of solar harvesting systems with sensor nodes was also explored in depth. Detailed architectures were provided for each battery setup, including power management circuits, charge controllers, and their role in ensuring reliable sensor operation. Energy equations and runtime estimations based on real solar irradiance values and component specifications helped quantify energy availability and consumption throughout different times of the day.

Overall, this work highlights the importance of choosing the right energy harvesting strategy depending on application-specific requirements. While Harvest–Use systems offer simplicity and real-time responsiveness, they suffer from discontinuous operation. Harvest–Store–Use systems provide reliability at the cost of complexity and storage losses, whereas Harvest–Use–Store systems strike a balance between immediate power needs and energy buffering. The analysis presented here can guide future deployments of smart irrigation systems in precision agriculture, particularly in resource-constrained rural areas where sustainable and autonomous operation is critical.

XI Future Work

Future enhancements of this study may include field-level deployment of the sensor nodes integrated with the respective battery systems to validate simulation results. Additional modeling of multi-node communication behaviors, sensor failures, and real-time data collection via cloud-based IoT platforms could improve the scalability and robustness of the proposed framework. Moreover, integrating adaptive machine learning algorithms for dynamic energy management and predictive maintenance could further enhance system efficiency. Exploring other renewable energy sources such as wind or hybrid systems may also offer new dimensions to optimize energy availability across various climatic conditions

XII. Acknowledgment

I would like to express sincere gratitude to Prof. Suparna Chakraborthy for her valuable guidance and mentorship throughout the duration of this project. Special thanks are also extended to the department faculty and lab technicians for providing the necessary resources and support. This work would not have been possible without the technical assistance and collaborative environment fostered at WILP , BITS Hyderabad campus .

References

[1] Adu-Manu, Kofi Sarpong, et al. “Energy-Harvesting Wireless Sensor Networks (EH-WSNs): A Review.” *ACM Transactions on Sensor Networks*, vol. 14, no. 2, May 2018, pp. 1–50. *DOI.org (Crossref)*, https://doi.org/10.1145/3183338.

[2] Bennani, Mohamed Taj, et al. “Comparing Leach Protocol and Its Descendants on Transferring Scalar Data.” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 35, no. 1, July 2024, p.255.*DOI.org (Crossref)*, https://doi.org/10.11591/ijeecs.v35.i1.pp255-262.

[3] Boban, Marija, and Vedran Uroš. “Energy Efficiency Improvement of Wireless Sensor Networks in the Digital Age.” *2024 47th MIPRO ICT and Electronics Convention (MIPRO)*, IEEE, 2024, pp. 1076–80. *DOI.org (Crossref)*, https://doi.org/10.1109/MIPRO60963.2024.10569533.

[4] Chakraborty, Suparna, N.K. Goyal, et al. “A Monte-Carlo Markov Chain Approach for Coverage-Area Reliability of Mobile Wireless Sensor Networks with Multistate Nodes.” *Reliability Engineering & System Safety*, vol. 193, Jan. 2020,p.106662. *DOI.org (Crossref)*, https://doi.org/10.1016/j.ress.2019.106662

[5] Chakraborty, Suparna, Neeraj Kumar Goyal, et al. “On Area Coverage Reliability of Mobile Wireless Sensor Networks With Multistate Nodes.” *IEEE Sensors Journal*, vol. 20, no. 9, May 2020, pp. 4992–5003*DOI.org\\(Crossref)*, https://doi.org/10.1109/JSEN.2020.2965592.

[6] Filios, Gabriel, et al. “A Smart Energy Harvesting Platform for Wireless Sensor Network Applications.” *Information*, vol. 10, no. 11, Nov. 2019, p. 345. *DOI.org (Crossref)*, https://doi.org/10.3390/info10110345.

[7] Hussein, Abbas Hameed Abdul, et al. “Harvesting the Future: AI and IoT in Agriculture.” *E3S Web of Conferences*, edited by K. Slimani et al., vol. \477, 2024,p.00090.*DOI.org(Crossref)*, https://doi.org/10.1051/e3sconf/202447700090.

[8] Juwaied, Abdulla, et al. “Enhancing Clustering Efficiency in Heterogeneous Wireless Sensor Network Protocols Using the K-Nearest Neighbours Algorithm.” *Sensors*, vol. 25, no. 4, Feb. 2025, p. 1029. *DOI.org (Crossref)*, https://doi.org/10.3390/s25041029.

[9] Kar, K., et al. “Dynamic Node Activation in Networks of Rechargeable Sensors.” *IEEE/ACM Transactions on Networking*, vol. 14, no. 1, Feb. 2006, pp. 15–26. *DOI.org (Crossref)*, https://doi.org/10.1109/TNET.2005.863710.

[10] Ku, Meng-Lin, et al. “Advances in Energy Harvesting Communications: Past, Present, and Future Challenges.” *IEEE Communications Surveys & Tutorials*, vol.18,no.2,]2016,pp.1384–412. *DOI.org (Crossref)*, https://doi.org/10.1109/COMST.2015.2497324.

[11] Li, Hao, et al. “Design and Experimental Evaluation on an Advanced Multi Source Energy Harvesting System for Wireless Sensor Nodes.” *The Scientific World Journal*, vol. 2014, 2014, pp. 1–13. *DOI.org (Crossref)*, https://doi.org/10.1155/2014/671280.

[12] Li, Tao, et al. *Optimum Transmission Policies for Energy Harvesting Sensor Networks Powered By a Mobile Control Center*. arXiv:1601.00721, arXiv, 5 Jan. 2016. *arXiv.org*, https://doi.org/10.48550/arXiv.1601.00721.

[13] *Optimum Transmission Policies for Energy Harvesting Sensor Networks Powered By a Mobile Control Center*. 2016. *DOI.org (Datacite)*, https://doi.org/10.48550/ARXIV.1601.0072

[14] Müller, G., et al. “A Wireless Sensor Network Using Energy Harvesting for Agricultural Machinery.” *E & i Elektrotechnik Und Informationstechnik*, vol. 127, no. 3, Mar. 2010, pp. 39–46. *DOI.org (Crossref)*, https://doi.org/10.1007/s00502-010-0719-0.

[15] Padmavathy, N., and Sanjay K. Chaturvedi. “Reliability Evaluation of Mobile Ad Hoc Network: With and without Mobility Considerations.” *Procedia Computer Science*, vol. 46, 2015, pp. 1126–39. *DOI.org (Crossref)*, https://doi.org/10.1016/j.procs.2015.01.025.

[16] Prauzek, Michal, et al. “Energy Harvesting Sources, Storage Devices and System Topologies for Environmental Wireless Sensor Networks: A Review.” *Sensors*, vol. 18, no. 8, July 2018, p. 2446. *DOI.org (Crossref)*, https://doi.org/10.3390/s18082446.

[17] Sharma, Himanshu, et al. "Maximization of Wireless Sensor Network Lifetime Using Solar Energy Harvesting for Smart Agriculture Monitoring." *Ad Hoc Networks*, vol. 94, Nov. 2019, p. 101966. *DOI.org (Crossref)*, https://doi.org/10.1016/j.adhoc.2019.101966.

[18] Wang, Cong, et al. "A Mobile Data Gathering Framework for Wireless Rechargeable Sensor Networks with Vehicle Movement Costs and Capacity Constraints." *IEEE Transactions on Computers*, vol. 65, no. 8, Aug. 2016, pp. 2411–27. *DOI.org (Crossref)*, https://doi.org/10.1109/TC.2015.2490060.

[19] Yuan, Fangchao, et al. "Optimal Harvest-Use-Store Strategy for Energy Harvesting Wireless Systems." *IEEE Transactions on Wireless Communications*, vol. 14, no. 2, Feb. 2015, pp. 698–710. *DOI.org (Crossref)*, https://doi.org/10.1109/TWC.2014.2358215.

SOHAM ANAND KULKARNI
is currently pursuing a Bachelor of Technology degree in Electronics and Instrumentation Engineering at BITS Pilani Goa campus. His academic interests lie at the intersection of wireless communication, microelectronics, VLSI design, embedded systems, and sustainable technologies. He is particularly passionate about applying engineering principles to solve real-world problems in agriculture, energy, and the Internet of Things (IoT).

As part of his undergraduate research, he has worked extensively on integrating wireless sensor networks with energy harvesting systems to improve the efficiency and sustainability of smart agricultural practices. His recent work involves MATLAB-based modeling and hardware prototyping of energy-aware sensor nodes, with a focus on using Markov models to evaluate and optimize energy management strategies.

He is also an enthusiast in renewable energy systems and embedded hardware design, and aims to pursue future research in intelligent cyber-physical systems and sustainable IoT frameworks.