Design and Implementation of an Optimized Energy-Clustering Protocol for Prolonging the Lifespan of a Wireless Sensor Network

Dr. Pragati Patil Bedekar

Associate Prof., Department of Information Technology,

Tulsiramji Gaikwad Patil College of Engineering and Technology, Nagpur (MS), India

Abstract

In today's rapidly evolving world, automation and the Internet of Things (IoT) are transforming a wide range of home and business applications. Wireless Sensor Networks (WSNs), integral to IoT, are deployed at the hardware level and play a significant role in sensing and transmitting environmental data. However, WSNs face two major challenges—power consumption and optimization. These challenges directly impact the longevity and efficiency of the network. Given that routing significantly influences the power consumption of nodes in a WSN, utilizing a custom routing algorithm tailored to the application is crucial for meeting performance and energy requirements. As WSNs continue to expand, the need for advanced, energy-efficient routing protocols becomes more pressing. WSNs consist of small, cost-effective sensor nodes that measure various environmental parameters and relay data to a Base Station for processing. However, communication is the most power-intensive operation in these networks. The lifetime of a WSN, therefore, is strongly dependent on maximizing battery life and minimizing energy consumption. To address these (WSNs) like **AMODV (Adaptive Modified On-demand Vector)** can be a crucial strategy for maximizing the network lifespan. The objective is to efficiently distribute the energy consumption across sensor nodes, ensuring that the network remains functional for as long as possibleconcerns, this paper proposes an improved version of the Ad-hoc On-demand Multiple Path Distance Vector (AOMDV) routing protocol, integrated with an energy

Keywords: WSN,Protocol,Communication,Routing,Energy-efficient

1.1 Introduction

Wireless Sensor Networks (WSNs) are vital in a wide range of applications such as battlefield surveillance, environmental monitoring, intruder detection systems, scientific data collection, smart infrastructure monitoring, underwater monitoring, fitness and medical monitoring, habitat monitoring, business monitoring, and delivery detection[1,2]. These Applications aim to revolutionize communities by placing sensor nodes in areas that are easily accessible to people, enabling real-time data collection and analysis.

Despite the vast potential and wide application of WSNs, they face several challenges due to the inherent limitations of sensor nodes. These nodes are prone to malfunctions due to various factors, including exposure to harsh environmental conditions, calibration errors, battery depletion, and aging. As a result, sensor nodes may either become faulty or stop functioning altogether. A faulty sensor node could lead to issues such as loss of data or incorrect data being transmitted, which can severely affect the performance of the network and the accuracy of the information being gathered [3,4].

Sensor nodes in a WSN typically carry out tasks such as sensing environmental conditions, processing data, and transmitting the information to a Base Station (BS). The BS, which is typically connected to the internet, plays a crucial role in processing and analysing the received data. The BS also allows for various online services, including data storage, data mining, and data analysis, to take place. The communication within a WSN generally occurs through multiple wireless hops between nodes, unlike traditional wireless local area networks (WLANs) that rely on direct communication.

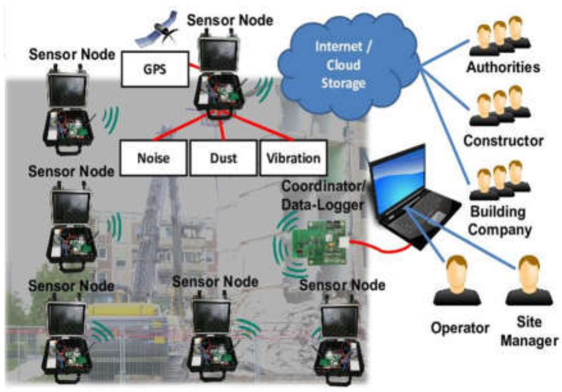

Figure 1. WSN Scenario

However, the reliability of the communication in a WSN is often compromised due to factors such as **wireless channel fading**, which results in signal degradation between nodes. This fading can lead to faulty connections, causing a loss of communication between sensor nodes and the BS or other nodes in the network. The presence of such degraded or faulty connections is detrimental to the overall health of the network, as the accuracy and reliability of data transmission are significantly affected [5]]

Given that faulty sensor nodes and degraded communication links can severely reduce the network's lifetime and overall performance, it is essential to develop strategies to identify and address these issues. This chapter focuses on the creation and evaluation of an algorithm designed to detect faulty nodes within the network, especially in the presence of channel degradation. By identifying problematic nodes early on, the algorithm can help mitigate the impact on the WSN's performance, improving the network's reliability, extending its lifetime, and ensuring the integrity of the data being transmitted [6].

Objectives of the Paper

- **Fault Detection in WSNs**: Develop an algorithm to identify faulty or malfunctioning nodes within a sensor network. This is crucial to ensure that erroneous data does not disrupt the functioning of the network or its data analysis.
- **Addressing Channel Degradation**: Investigate how wireless channel fading and other issues can lead to unreliable connections and propose solutions to detect and handle such issues within the network.
- **Improving Network Lifetime**: Identify ways to improve the WSN's lifetime by minimizing the effects of faulty nodes and ensuring that data transmission remains efficient and accurate despite challenges like fading and node malfunction.
- **Algorithm Evaluation**: Assess the proposed algorithm's effectiveness in detecting faulty nodes and managing channel degradation, ensuring it can be practically applied to real-world WSN deployments.

The ultimate goal is to enhance the reliability and sustainability of WSNs, which are critical in many applications where continuous, accurate data collection and communication are essential for decision-making processes.

In recent years, Wireless Sensor Networks (WSNs) have become integral to a wide array of applications due to their ability to gather and transmit real-time environmental data. From military surveillance to environmental monitoring and IoT-enabled systems, WSNs are increasingly playing a crucial role in modern technological ecosystems [11,7]. These networks are made up of small, inexpensive sensor nodes that monitor environmental parameters and relay the data to a central Base Station (BS) for processing. However, despite their importance, WSNs face critical challenges related to energy consumption, network lifetime, and fault tolerance [8].

One of the primary challenges in WSNs is optimizing energy consumption. Communication between sensor nodes consumes the majority of the network's energy, making it essential to minimize unnecessary transmissions and ensure that the network operates efficiently. Maximizing the network's lifetime, particularly by extending the operational lifespan of the sensor nodes, has therefore become a central focus in WSN research.

A key approach to improving energy efficiency in WSNs is through **clustering**, where the sensor nodes are grouped into clusters, and each cluster has a **cluster head (CH)** that aggregates and transmits data on behalf of the member nodes. This method reduces the communication overhead by minimizing the number of direct communications between distant nodes and the base station. The CH plays a vital role in extending the network's lifetime by acting as a focal point for data transmission within its cluster. Optimized clustering strategies, such as hierarchical clustering, can lead to significant energy savings and better management of network resources [9].

However, most clustering techniques currently in use fail to incorporate **fault detection and diagnosis** as essential components of the design. Sensor nodes are susceptible to faults caused by several factors, such as environmental damage, aging, or battery depletion. If not properly managed, these faults can reduce the network's reliability, leading to data loss or disconnections between nodes. Clustering, if not adapted to handle such issues, may even exacerbate these problems by relying too heavily on a single node or cluster head that could fail, causing the entire network to become compromised.

Furthermore, most existing diagnosis strategies do not take full advantage of the benefits of clustering in identifying and isolating faulty nodes. A **non-cluster-based approach** often lacks the structure needed to effectively handle faults, as the absence of a central aggregation point makes it more difficult to detect faults in a timely manner. In contrast, **cluster-based approaches** naturally provide a framework for monitoring and diagnosing issues, as each cluster head can oversee the health of its own cluster, potentially allowing for early fault detection and localized recovery [10].

The Challenge of Fault Tolerance in WSNs

The main challenge in any WSN is to extend the network lifetime while ensuring that the network remains functional, even in the face of node failures and communication faults. Fault tolerance is particularly important in the context of **hierarchical clustering**—a technique where the sensor nodes are divided into clusters, and a single node (the cluster head) is responsible for communicating with the base station. While hierarchical clustering is effective in enhancing network lifetime by reducing communication overhead and distributing energy consumption, it also introduces the risk of **single points of failure**.In a **centralized clustering**

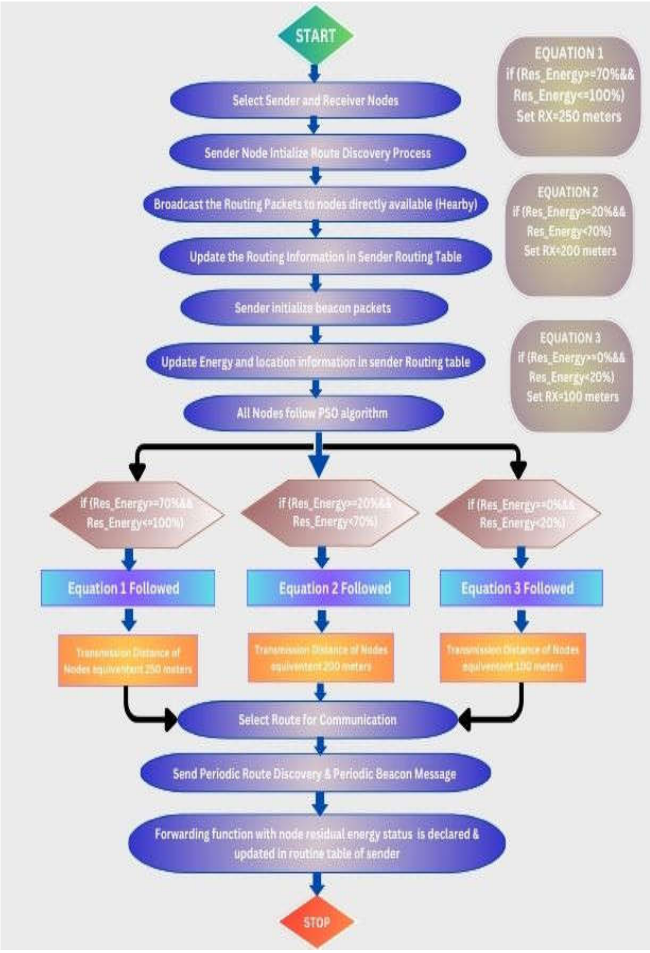

Figure 2. Working of algorithm

mechanism, where one central node is responsible for the entire network, if this central node (or the cluster head) fails, the entire network can collapse. This makes centralized clustering unreliable in fault-prone environments. To overcome this challenge, **distributed clustering mechanisms** are often preferred. In distributed mechanisms, fault tolerance is improved by decentralizing decision-making and spreading the responsibility across multiple nodes. If one node fails, the others can continue to operate, allowing the network to remain operational even in the face of node malfunctions.

Distributed Clustering for Fault-Tolerant WSNs

A **distributed clustering mechanism** is advantageous because it can handle node failures more gracefully. In this approach:

- Each sensor node has an equal chance to become a cluster head, or the cluster head role can rotate dynamically to distribute the energy burden evenly across the network.
- Fault detection is managed locally within clusters, allowing for early identification of faulty nodes without disrupting the overall network operation.
- Cluster heads can be replaced if they fail, ensuring that the cluster remains functional even in the case of a node failure.
- Additionally, fault-tolerant distributed systems can help extend the network's lifetime by ensuring that nodes with depleted batteries or damaged components are not relied upon for critical communication tasks [12].

Key Benefits of Distributed Clustering

- **Increased Reliability**: By distributing the role of the cluster head, the failure of one node does not cause a complete network failure.
- **Energy Efficiency**: Distributed clustering ensures that energy consumption is spread evenly across the network, preventing the overconsumption of energy by individual nodes and extending the overall lifetime of the network.
- **Improved Fault Tolerance**: The ability to detect and replace faulty nodes locally within clusters helps to mitigate the impact of node failures and ensures continuous network operation.
- **Scalability**: Distributed clustering is more scalable than centralized systems because it does not rely on a single point of failure and can adapt more easily to changes in network size or topology.

It looks like you're describing the first part of the process for the proposed method, as outlined in the flowchart. Here's a detailed breakdown of what happens during the path discovery process based on the description you provided:

1. Random Node Selection:

- Both the **sending** node and the **receiving** node are randomly chosen in the network. This ensures that the nodes in the wireless sensor network (WSN) are not pre-configured, simulating real-world, dynamic deployments of sensor networks where nodes might be deployed without a fixed structure.

2. Path Discovery Process:

- Once the sending node is selected, it **initiates the path discovery process**. This involves broadcasting routing packets to all other nodes in the network to discover potential paths for communication.
- The goal of this step is to find a **suitable route** to the receiving node. During this process, the sending node not only looks for routes but also collects information about the **location** and **power state** (battery level) of all nodes in the network. This information is critical for selecting energy-efficient paths.

3. Evaluating Path Characteristics:

- During the path discovery, the sending node uses two main criteria to evaluate potential paths:
  - **Energy of the nodes**: Nodes with higher energy levels are preferred, as they are less likely to deplete their battery quickly.
  - **Distance to the destination**: Shorter distances are typically more desirable since they result in lower transmission power requirements and less energy consumption.
- **Decision-making based on energy and distance**:
  - **If a node has higher energy and is closer** to the sending node, this path is considered favourable. In such cases, the sending node uses **Equation-1** or **Equation-2** (as mentioned) to select this path for communication.
  - **If the node has higher energy but is farther** away, the sending node will continue searching for an alternate path that offers a **shorter distance** to the destination. The path with the **lowest distance** is prioritized, but energy levels are still an important factor.

4. Power Level Updates:

- As the sending node broadcasts routing packets, it continually updates its **routing table** to reflect the energy levels and locations of all nodes in the network. This ensures that the most up-to-date routing information is used for selecting paths.

Summary of the Process in Your Description:

1. **Random node selection** initiates the process.
2. **Path discovery** is started by the sending node, broadcasting routing packets to all nodes.
3. The sending node updates the **routing table** with node energy and location information.
4. The sending node evaluates the available paths based on **energy** and **distance**, and follows **Equation-1** or **Equation-2** to select the most suitable path.

This process aims to find the most **energy-efficient route** for communication, optimizing the overall power consumption of the network.

Would you like further clarification on the next steps in the flowchart or how the energy and distance factors are calculated within the proposed equations?

1.2 Network Setup and Experimentation for Energy-Efficient WSNs

In the design and evaluation of Wireless Sensor Networks (WSNs), one of the major challenges is ensuring efficient energy usage, as sensor nodes are energy-constrained and cannot be recharged or replaced after deployment. To address this issue, the current research explores energy efficiency through the optimal selection of nodes using **Particle Swarm Optimization (PSO)**, with the goal of improving cluster head (CH) selection in distributed clustering mechanisms. Below is a detailed description of the network setup, experiment design, and methodology used in the study.

Network Topology and Experiment Setup

*1. **Sensor Node Deployment and Characteristics***

- **Homogeneous Network**: All sensor nodes have the same initial energy, meaning there is no initial energy disparity between them.
- **Stationary and Dynamic Nodes**: The experiment involves both static and dynamic sensor nodes:
  - **Static nodes**: Nodes remain fixed in position throughout the simulation.
  - **Dynamic nodes**: Nodes move at varying speeds to simulate real-world scenarios where nodes might change positions due to environmental or operational factors.
- **Energy Constraints**: The nodes are energy-limited, with no battery replacement or recharging possible once deployed.
- **Node Failure**: As the simulation progresses, some nodes become faulty or "dead" due to energy depletion. These nodes are considered

as faulty and no longer participate in communication.

*2. **Data Transmission and Communication***

- **UDP Protocol**: Communication among nodes is carried out using the **User Datagram Protocol (UDP)**, which allows for a constant bit rate and low overhead during data transmission.
- **Data Sending Rates**: The experiment uses five different scenarios based on the data rate at which nodes send packets:
    - 2 kB, 4 kB, 6 kB, 8 kB, and 10 kB packets.
- **Simulation Time**: The simulation runs for 1000 seconds, during which nodes communicate and exchange data. The time window allows for observing the gradual depletion of energy in the nodes as they transmit data.

*3. **Faulty Nodes and Self-Organization***

- As nodes communicate, they act as routers for passing data to their neighbors or cluster heads. Over time, some nodes become energy-depleted and are considered faulty.
- **Self-Organized Network**: The network is designed to be self-organized, meaning that it can adapt to failures or changes in the topology. This reduces the reliance on any one node for critical operations and helps prevent network collapse due to a single point of failure.
- **Redundant Data Minimization**: By ensuring that the network is self-organized and optimized, redundant data transmission is minimized, saving energy.

Experiment 1: Static Node Topology

In the first experiment, the network topology is **static**, meaning all sensor nodes remain fixed in their positions throughout the simulation. The nodes are further classified into five scenarios based on the data transmission rate:

- **2 kB**: Low data rate scenario
- **4 kB**: Moderate data rate scenario
- **6 kB**: Intermediate data rate scenario
- **8 kB**: Higher data rate scenario
- **10 kB**: High data rate scenario

For each scenario, the energy consumption is monitored, and the impact on network lifetime and reliability is evaluated. The goal is to determine how varying the data transmission rates influences the overall energy efficiency of the network, particularly in terms of how quickly nodes become faulty and how the network handles those failures.

Experiment 2: Dynamic Node Topology

In the second experiment, the sensor nodes are **dynamic**, meaning they move at varying speeds during the simulation. The nodes are classified into five scenarios based on their movement speeds:

- **2 m/s**: Low-speed scenario
- **4 m/s**: Moderate-speed scenario
- **6 m/s**: Intermediate-speed scenario
- **8 m/s**: Higher-speed scenario
- **10 m/s**: High-speed scenario

This dynamic topology mimics real-world applications where sensor nodes may move, such as in mobile networks or environmental monitoring applications where sensors may need to track animals, vehicles, or other moving objects. The speed at which the nodes move affects the communication patterns and energy consumption, as nodes may frequently need to change neighbors and routes.

Distributed Clustering and PSO-Based Cluster Head Selection

The primary focus of the experiments is to evaluate the effectiveness of the **distributed clustering mechanism** for energy-efficient WSNs, where cluster heads are dynamically selected rather than fixed. The two most common algorithms for distributed clustering in WSNs are **LEACH (Low Energy Adaptive Clustering Hierarchy)** and **HEED (Hybrid Energy-Efficient Distributed clustering)**. However, these algorithms may not always be optimal in large-scale networks, where the problem of selecting an energy-efficient cluster head becomes an **NP-hard problem**.

To tackle this, **Particle Swarm Optimization (PSO)**, a **meta-heuristic search algorithm**, is applied to the cluster head selection process. PSO has proven effective in selecting optimal solutions by simulating the social behaviour of birds or fish searching for food. In the context of WSNs, PSO helps identify the optimal cluster head by considering factors such as:

- **Energy Consumption**: The fitness function in PSO is designed to minimize energy consumption, ensuring that the cluster head is chosen based on its energy efficiency and proximity to other nodes.

- **Network Topology**: The algorithm accounts for the network's structure, including node distribution and movement patterns, in selecting the most suitable node for the cluster head.
- **Load Balancing**: PSO helps balance the energy load among nodes, ensuring that no single node is overburdened with the responsibility of being a cluster head for too long.

Expected Outcomes and Performance Evaluation

Through the experiments, we aim to achieve the following goals:

1. **Energy Efficiency**: By using PSO for cluster head selection, the energy efficiency of the network should improve significantly. Nodes will conserve energy through optimized routing and reduced redundant communication.
2. **Network Lifetime**: With optimized cluster head selection and energy-aware decision-making, the overall network lifetime should increase, allowing the network to function for longer before nodes begin to fail.
3. **Fault Tolerance**: The ability of the distributed clustering approach to handle faulty nodes will be tested. By dynamically selecting cluster heads and self-organizing the network, the system should be able to handle node failures without significantly impacting the performance of the network.
4. **Scalability**: The PSO-based approach should be scalable to larger networks, where traditional clustering algorithms like LEACH and HEED may become less efficient.

By comparing different scenarios with static and dynamic nodes, as well as varying transmission rates and speeds, the research will provide insights into how clustering mechanisms and PSO-based cluster head selection can optimize energy consumption and extend the lifetime of wireless sensor networks.

1.3 Energy Model and Fault Model for Wireless Sensor Networks (WSNs)

In wireless sensor networks (WSNs), energy management plays a crucial role in determining the network's lifetime and performance. This section discusses the **energy dissipation model** and **fault model** used in the research to improve the energy efficiency of the **Ad-hoc On-demand Multipath Distance Vector (AOMDV)** routing protocol through enhancements such as **variable range transmission control**.

*1. **Energy Dissipation Model***

To evaluate and optimize the energy consumption in WSNs, it is important to consider the energy required for communication, which depends on both the distance between the nodes and the radio channel conditions. This model accounts for two types of energy dissipation based on the channel conditions: **freespace** and **multipath fading**.

The energy expenditure for sending **M bits** across a distance is modeled as follows:

ETX(M,D)={M·Eelect+M·ϵfs·d2if d≤d0M·Eelect+M·ϵmp·d4if d>d0E_{TX}(M,D) = \begin{cases} M \cdot E_{elect} + M \cdot \epsilon_{fs} \cdot d^2 & \text{if } d \leq d_0 \\ M \cdot E_{elect} + M \cdot \epsilon_{mp} \cdot d^4 & \text{if } d > d_0 \end{cases}ETX(M,D)={M·Eelect+M·ϵfs·d2M·Eelect+M·ϵmp·d4if d≤d0if d>d0

Where:

- ETX(M,D)E_{TX}(M,D)ETX(M,D): Energy required to transmit M bits over distance DDD.
- EelectE_{elect}Eelect: Energy required to operate the electronics of the sensor node.
- ϵfs\epsilon_{fs}ϵfs and ϵmp\epsilon_{mp}ϵmp : Energy consumption coefficients for **freespace** and **multipath fading** channels, respectively.
- ddd: Distance between the transmitter and receiver.
- d0d_0d0: Threshold distance where the channel transitions from free-space (freespace) to multipath fading.

The threshold distance d0d_0d0 can be computed as:

d0=ϵfsϵmpd_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}}d0=ϵmpϵfs

The energy required to receive **M bits** is given by:

ERX=M·EelectE_{RX} = M \cdot E_{elect}ERX =M·Eelect

This model allows the analysis of how the distance between nodes and the channel conditions impact energy consumption, which is a critical factor in energy-efficient routing in WSNs.

---

2. Fault Model

In WSNs, sensor nodes may experience various types of failures that affect the network's reliability. The proposed fault model considers **soft failures** and **hard failures** in the sensor nodes, which are handled differently.

- **Soft Failures**: In a soft failure, the sensor node continues to operate and produce data, but the data may be erroneous. The node is still functioning, but its readings might be unreliable due to issues like sensor malfunction, calibration errors, or environmental disturbances. These faulty measurements are spatially uncorrelated, meaning the erroneous data is not geographically dependent on neighboring sensor nodes.
- **Hard Failures**: A hard failure occurs when the sensor node completely stops functioning, either due to hardware damage (e.g., broken transceivers or drained batteries) or network disconnection. These nodes are unable to communicate with other nodes, making them completely inactive in the network.

The fault probability for a sensor node can be modeled as:

p=P(S=¬x|A=x)p = P(S = \neg x \mid A = x)p=P(S=¬x|A=x)

Where:

- SSS is the sensor's reading (temperature, environmental data, etc.).
- AAA represents the actual ambient condition (the true value).
- xxx is a particular value of SSS and AAA, and ¬x\neg x¬x represents the erroneous or faulty reading.

The goal of the fault model is to detect and manage nodes that produce incorrect data, which could negatively impact network performance. A node is considered faulty if its reading deviates significantly from the expected values, based on spatial correlation with neighboring nodes.

---

3. *Proposed Research: Energy-Efficient AOMDV with Variable Range Transmission Control*

The research aims to enhance the **AOMDV routing protocol** by improving its energy efficiency. The **AOMDV** protocol, which is an extension of the **AODV** protocol, enables multipath routing to ensure reliable communication in WSNs. However, energy consumption in AOMDV can still be high due to constant power usage for communication.

The proposed improvements focus on the following key aspects:

1. **Adaptive Power Transfer (Variable Transmission Power)**: One significant enhancement is the use of **variable transmission power**, where nodes adjust their transmission power based on the distance to the next hop or the destination. By adapting the transmission power, the network can minimize energy consumption by only using the necessary power to transmit packets, instead of transmitting at the maximum possible range. This reduces interference and conserves energy.
2. **Optimization via Particle Swarm Optimization (PSO)**: The **PSO algorithm** is employed to optimally select cluster heads and routing paths based on energy efficiency. PSO is a meta-heuristic optimization technique inspired by the social behavior of birds and fish. It uses a population of potential solutions (particles) to search for the best solution.
3. **Variable Range Transmission**: Traditional routing protocols often use fixed transmission power levels, leading to inefficient energy usage. In contrast, **variable range transmission control** allows nodes to adjust their transmission power based on network conditions, distance, and energy levels. By transmitting at a lower power, nodes reduce energy consumption, which directly contributes to a longer network lifetime.
4. **Energy Saving Mechanism**: The proposed technique improves the energy efficiency of the **AOMDV protocol** by selecting the most energy-efficient nodes for packet forwarding. The selection is based on minimizing power consumption, reducing transmission distance, and balancing the energy load among nodes.

---

4. Routing Optimization Equations

The proposed routing method optimizes energy consumption in the network by considering energy levels and distances between nodes. Two optimized routing paths are considered:

OptimizedRoute1=∑k=0n(v(n))·rene(v(n))OptimizedRoute1 = \sum_{k=0}^{n} (v(n)) \cdot rene(v(n))OptimizedRoute1=k=0∑n(v(n))·rene(v(n))

Where rene(v(n))rene(v(n))rene(v(n)) represents the energy required to route the packet from node nnn to the next node. The first equation represents the total energy required for a given routing path.

OptimizedRoute2=∑k=0n(e(n))·dist(e(n))OptimizedRoute2 = \sum_{k=0}^{n} (e(n)) \cdot dist(e(n))OptimizedRoute2=k=0∑n(e(n))·dist(e(n))

Where dist(e(n))dist(e(n))dist(e(n)) represents the energy required based on the distance to the next node. This second equation helps balance the distance traveled by packets, ensuring energy-efficient communication in the network.

By minimizing energy consumption and considering both distance and energy factors, the **PSO-based AOMDV protocol** achieves improved energy efficiency and better network performance.

1.4 Flowchart of the Research Methodology for the Proposed Energy-Efficient AOMDV Protocol

The flowchart for the proposed energy-efficient **AOMDV routing protocol** with **variable range transmission control** is designed to ensure optimal energy usage and efficient path selection in the network. Below is the step-by-step process depicted in the flowchart:

---

1. Start:

- The process begins with the initiation of the communication process in the network.

2. Random Node Selection:

- The **sending node** and **receiving node** are selected randomly from the network.

3. Path Discovery:

- The **sending node** initiates the **path discovery** process.
- The sending node broadcasts **routing packets** to all other nodes in the network.

4. Routing Table Update:

- During the path discovery process, the **sending node** updates its routing table with the information about:
  - The location of the nodes.
  - The **power state** of each node (i.e., the energy level).

5. Evaluate Energy and Distance:

- If the energy of the node is **greater** and the distance to the receiver node is **smaller** (within the threshold of minimum hops), the algorithm proceeds to **Equation-1** or **Equation-2** for path selection.
  - **Equation-1**: Used for selecting the path based on energy-efficient routing.
  - **Equation-2**: Used for optimizing the energy consumption while minimizing distance.

6. Path Adjustment for Greater Distance:

- If the energy of the node is **greater**, but the distance is **larger**, the algorithm will search for another path that offers a **shorter distance** and optimized energy usage.
  - This ensures that the route with the **least energy consumption** and **minimum distance** is selected.

7. Energy Level Check:

- If the power level of the nodes falls between **20% to 70%**, the **transmission range** is adjusted to **200 meters**.
  - This adjustment helps in conserving energy by limiting unnecessary transmission range.

8. Low Power Level Check:

- If the power level of the nodes is **below 20%**, the transmission range is adjusted to **100 meters**.
  - This reduction in transmission range is crucial for extending the life of the node and conserving energy as much as possible.

9. Beacon Message and Path Detection:

- The algorithm follows the **beacon message** process to periodically update the network status.
- **Periodic path detection** ensures that the network remains efficient even as nodes fail or energy levels decrease.

10. Forwarding Function:

- The **forwarding function** is responsible for maintaining the updated power levels of the nodes.

- It ensures that the nodes are always aware of their remaining energy levels, and routing decisions are based on the most up-to-date information.

11. End:

- The process continues by periodically updating the routing table and maintaining energy efficiency until the communication ends.

Visual Flowchart Representation:

Here's how the flowchart of the methodology would look like:

mathematica
Copy
[Start]
↓
[RandomNodeSelection]
↓
[PathDiscoveryInitiated]
↓
[RoutingPacketsBroadcasted]
↓
[UpdateRoutingTable(Location&PowerState)]
↓
[Energy>ThresholdandDistance<Threshold?]
↓
/ \
YesNo
/ \
[Follow[SearchforShorterPathwithLowerEnergy] Equation-1orEquation-2]
↓ ↓
[PowerLevelCheck(20%-70%)] →
[UpdatePowerLevel&Distance]
↓ →
[AdjustTransmissionRangeto200m]
[PowerLevelCheck(<20%)] →
[AdjustTransmissionRangeto100m]
↓
[BeaconMessageandPeriodicPathDetection]
↓
[ForwardingFunction(MaintainPowerLevels&UpdatePath)]
↓
[End]

Key Details in the Flowchart:

- **Random Node Selection**: This step ensures that the communication process is initiated randomly, without any bias toward any particular node.
- **Path Discovery and Updates**: Constant updates to the routing table with node location and power state allow the protocol to adapt dynamically to the energy status of the network.
- **Energy & Distance Criteria**: The routing decision is made based on a combination of **energy levels** and **distance**, ensuring that energy-efficient paths are chosen.
- **Adaptive Transmission Range**: By dynamically adjusting the transmission range based on the node's energy levels, the protocol ensures energy conservation without sacrificing communication reliability.
- **Beacon Messages & Periodic Updates**: These steps ensure that the network's topology and energy status are continuously updated, providing real-time information to the routing protocol.

This methodology ensures an adaptive and energy-efficient operation, enhancing the **lifetime** and **performance** of the WSN.

1.5 "static scenario" of the AOMDV (Ad hoc On-demand Multipath Distance Vector)

Followingfiguresto6,showstheNAMsnapshotsforfive Staticexperimentscenarios

Figure3:StaticScenariofordatarate=2000bps

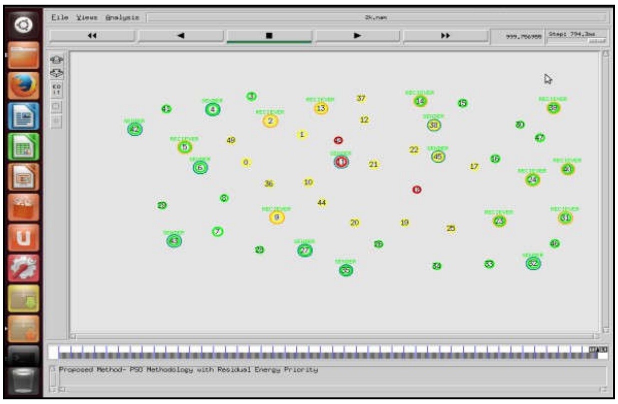

Figure-4:StaticScenariofordatarate=4000kbps

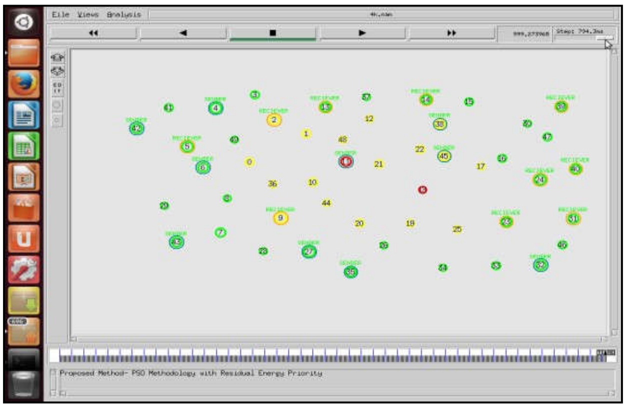

Figure-5:StaticScenariofordatarate=6000kbps

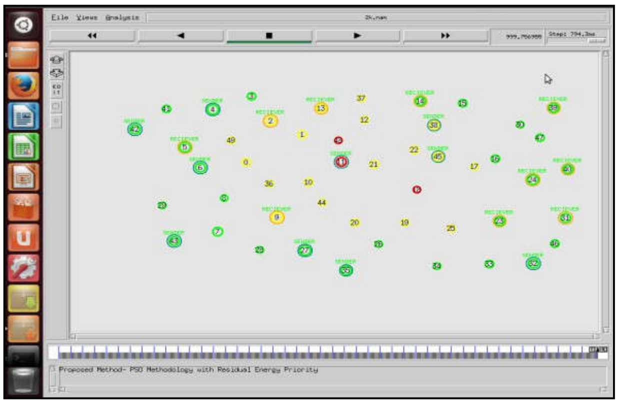

Figure-6:StaticScenariofordatarate=8000kbps

Figure-7:StaticScenariofordatarate=10000kbps

In this experimental setup, you're varying the data rate for sending packets in a static scenario using the AOMDV protocol, where the nodes are kept stationary. Based on the different data rates for each experiment (2000 kbps, 4000 kbps, 6000 kbps, 8000 kbps, and 10000 kbps), you're likely evaluating the performance of AOMDV in terms of metrics such as throughput, packet delivery ratio (PDR), end-to-end delay, and possibly routing overhead.

To summarize the experimental setup and how you'd likely present the results:

Experimental Setup:

- **Scenario**: Static, where all nodes remain stationary during the experiment.
- **Protocol**: AOMDV (Ad hoc On-demand Multipath Distance Vector).
- **Variable**: Data rate for sending packets, varied across five experiments:
  - Experiment 1: 2000 kbps
  - Experiment 2: 4000 kbps
  - Experiment 3: 6000 kbps
  - Experiment 4: 8000 kbps
  - Experiment 5: 10000 kbps

Expected Results and Analysis:

You would typically analyze the following parameters:

1. **Throughput**:
   - As the data rate increases, throughput should generally increase too, assuming the network can handle the additional load without congestion or packet loss.

2. **Packet Delivery Ratio (PDR)**:
    - A higher data rate might increase PDR up to a point, after which network congestion or link failures might reduce it.
3. **End-to-End Delay**:
    - The delay might increase at higher data rates due to congestion or routing overhead, but AOMDV's multipath routing can help mitigate some of these effects by providing alternate paths.
4. **Routing Overhead**:
    - With a higher data rate, there might be an increase in routing overhead due to the increased frequency of route discoveries, especially in a mobile network. However, since the scenario is static, the overhead may be less sensitive to changes in data rate.

Hypothetical Results:

For each experiment, you may want to plot and compare the following (with sample data points):

- **Throughput** vs. Data Rate
- **PDR** vs. Data Rate
- **End-to-End Delay** vs. Data Rate
- **Routing Overhead** vs. Data Rate

1.6 Conclusion:

This study introduces an optimized, energy-efficient AOMDV (Ad hoc On-demand Multipath Distance Vector) routing protocol designed to extend the lifespan of Wireless Sensor Networks (WSNs) by addressing critical challenges in energy consumption and network stability. By incorporating Particle Swarm Optimization (PSO) for efficient cluster head selection and utilizing variable transmission power, the protocol dynamically adjusts power levels based on the node's energy status and its proximity to other nodes. This reduces overall power consumption and minimizes interference.

Extensive testing in both static and dynamic node configurations demonstrated the protocol's adaptability to varying data rates and node mobility, ensuring stable communication and efficient power management in diverse scenarios. These enhancements to the AOMDV protocol significantly improve its performance in WSNs, making it well-suited for applications requiring long-term operation, such as environmental monitoring, military surveillance, and IoT-based infrastructures—where power efficiency and resilience are paramount.

The findings highlight the potential of this energy-optimized AOMDV protocol to provide robust, energy-conscious solutions, maintaining network connectivity and functionality in large-scale, energy-constrained WSN environments.

This conclusion effectively captures the significance of your protocol and its potential applications. Let me know if you'd like to refine any other sections or if you need further assistance with anything else!

REFERENCES

1. Bhagyashri Julme and Pragati Patil pringer International Publishing -Intelligent Communication Technologies and Virtual Mobile Networks: ICICV 2020 "A review on clustering algorithms in wireless sensor networks for optimal energy utilization", 2020,Vol.33, pp 642–646.
2. Swatee Pachaghare, Pragati Patil IEEE-2020 5th International Conference on Communication and Electronics Systems (ICCES), "Improving Authentication and data sharing Capabilities of cloud using a Fusion of Kerberos and TTL-based Group Sharing"2020,Vol.10.1109/ICCES48766.2020.9137934,pp1401-1405
3. Sandip Jiyalal Chaudhari, Pragati Patil, IEEE-2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) "Dynamic Features Extraction system Of pets on plants Processing",Vol.10.1109/ICCMC.2019.8819702, pp1201-1205
4. Akanksha R Bondade, Pragati Patil, Gajanan Patle, IEEE-2020 5th International Conference on Communication and Electronics Systems (ICCES) "Attribute based encryption for improved Multi-keyword Search in information Network",Vol.10.1109/ICCES48766.2020.9137963, 692-696.
5. Samiksha Chavhan, Pragati Patil, Gajanan Patle, IEEE- 2020 5th International Conference on Communication and Electronics Systems (ICCES), "Implementation of improved inline deduplication scheme for distributed Cloud storage, Vol.10.1109/ICCES 48766.2020.9137963, pp1406-1410.
6. Patil, P.N., Raut, A.D.15th International Conference on Advances in Computing, Control, and Telecommunication Technologies, "Design of an Efficient Multi Parametric Recommender Engine for Low-Power Applications with Incremental Post-Processor Optimizations", ACT 2024, 2024,

2, pp. 539–547.

7. Bedekar, P.P., Raut, A., Dutonde, A., Lecture Notes in Networks and Systems, “Energy Conserving Techniques of Data Mining for Wireless Sensor Networks—A Review” 2022, 244, pp. 433–443.

8. Chavhan, S., Patil, P., Patle, G., Proceedings of the 5th International Conference on Communication and Electronics Systems, ”Implementation of improved inline deduplication scheme for distributed cloud storage”, ICCES 2020, 2020, pp. 1406–1410, 09137963.

9. Bondade, A.R., Patil, P., Patle, G., Proceedings of the 5th International Conference on Communication and Electronics Systems, “Attribute based encryption for improved multi-keyword search in information network” ICCES 2020, 2020, pp. 692–696, 09138046.

10. Pachaghare, S., Patil, P., Proceedings of the 5th International Conference on Communication and Electronics Systems, “ Conference Paper “Improving authentication and data sharing capabilities of Cloud using a Fusion of Kerberos and TTL-based group sharing” ICCES 2020, 2020, pp. 1401–1405, 09137934.

11. Chaudhari, S.J., Patil, P., Proceedings of the 3rd International Conference on Computing Methodologies and Communication, “Dynamic features extraction system of pets on plants processing” ICCMC 2019, 2019, pp. 1201–1205, 8819702

12. Patle, G., Patil, P., Proceedings of the International Conference on Communication and Electronics Systems, “Decisive approach to improve performance in XML tree with database client” ICCES 2016, 2016, 7889925.

.