I.J. Wireless and Microwave Technologies, 2026, 4, 395-412
Published Online on August 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.04.23

Fault-Tolerant Clustering with Adaptive Ant Colony Optimization for Energy-Efficient Routing in Wireless Sensor Networks

Manisha Chandrakar*
School of Engineering and Research, ITM University, Raipur, India
E-mail: manishachandrakar00@gmail.com
ORCID iD: https://orcid.org/0009-0001-9035-9708
*Corresponding Author

Aamir Hasan
School of Studies in Computer Science and IT, Pt. Ravishankar Shukla University, Raipur, India
E-mail: aamirhasan.aamir@gmail.com
ORCID iD: https://orcid.org/0000-0002-5484-1564

Received: April 9, 2026; Revised: May 27, 2026; Accepted: July 15, 2026; Published: August 8, 2026

Abstract: Wireless Sensor Networks (WSNs) play a critical role in various applications, including environmental monitoring, healthcare, and industrial automation. However, these networks face significant challenges related to energy efficiency, fault tolerance, and reliable data transmission, particularly in dynamic environments. Existing clustering and routing techniques often fail to ensure seamless fault tolerance and energy optimization simultaneously. Many traditional approaches lack robust mechanisms to handle Cluster Head (CH) failures, resulting in reduced network stability and shorter operational lifetimes. To address these limitations, this study proposes a Fault-Tolerant Backup Cluster Head with Ant Colony Optimization (FT-BKCH-ACO) approach that enhances energy efficiency and network resilience. The methodology involves optimized CH and Backup CH (BKCH) selection, considering parameters such as residual energy, distance to the base station, and network density. Additionally, Ant Colony Optimization (ACO) is employed to dynamically adjust pheromone levels for energy-efficient routing, ensuring reliable intra-cluster and inter-cluster communication. Simulation results demonstrate that the FT-BKCH-ACO approach significantly improves energy consumption by 23.2%, packet delivery ratio by 10.5% and end-to-end delay by 17.8% compared to existing models. The inclusion of backup CHs ensures seamless communication even in the event of node failures, making this method highly suitable for IoT-enabled WSN applications. The proposed approach bridges the gap between fault-tolerant clustering and adaptive routing, offering a scalable and energy-efficient solution for large-scale sensor networks.

Index Terms: Energy Efficiency, Routing Optimization, Cluster Head (CH) Selection, Intra-Cluster Routing, Cluster Head, Wireless Sensor Networks

1. Introduction

By combining innovations like visual detection and wireless connectivity, sensors the foundation of contemporary information-gathering systems offer unmatched precision and adaptability, facilitating breakthroughs in domains that range from ecological surveillance to medical care diagnostics [1]. Wireless sensor networks are a new kind of communication system that is quickly gaining popularity for both defense and industrial uses. The automatic network methods consist of a dispersed network of sensing nodes. Wireless sensing networks comprise several interconnected parts that cooperate to facilitate data processing, interactions, and gathering. The detection node, the main part of a WSN, consists of a detecting element to gather environmental data, an analysis component to analyze and interpret the information, and an interface module to remotely send information to the main station or other system units. Usually, these nodes for sensing are tiny, low-power gadgets with sensors for things like light, movement, humidity, and precipitation [2]. The base station is the hub for gathering, storing, and analyzing data. It acts as a conduit across other systems and the WSN. WSNs may also include sink nodes that act as intermediaries, facilitating efficient communication between sensor modules and the base station. Fault tolerance plays a crucial role in WSNs. If a single cluster head fails, the data transmission system may become inoperable. This research presents a fault-tolerant cluster-based routing strategy. By utilizing backup CHs and optimizing relay node selection through Ant Colony Optimization (ACO), this study aims to enhance the ability of clustered WSNs to tolerate failures and manage data transmission efficiently. Selecting a Backup Cluster Head (BKCH) enhances fault tolerance in WSNs. The process consists of two primary stages. In the first stage, the system partitions the network into clusters. In the second stage, CHs and BKCHs are selected from a pool of applicants based on their qualifications. Within each cluster, member nodes communicate with their respective CHs. The modified ACO identifies the most effective relay nodes between CHs and the Aggregator node (AG). AG nodes facilitate communication between different clusters. Besides improving fault resilience and Packet Delivery Ratio (PDR), this technology reduces energy consumption compared to standard energy-saving measures [3].

The sink node cannot communicate over long distances directly because the sensor node possesses limited energy and communication capabilities [2]. Consequently, the network employs a “multi-hop” communication method to transport data from the source sensor node to the processing node at the sink [4]. Developing a revolutionary energy-optimized routing strategy poses the most significant challenge in prolonging the network’s lifespan. Some mission-critical and real-time implementations require a guaranteed quality of service and extend the network’s lifetime [5]. Cyber-attacks frequently target WSNs because their operations remain unsupervised, and their sensor nodes lack reliability. The inherently vulnerable nature of WSNs increases these security risks. Ensuring the safety of wireless sensor networks remains a critical concern [6].

Although WSNs offer significant potential, several obstacles hinder their deployment, including limited storage capacity, non-rechargeable and depleted batteries, restricted processing power, and inadequate safety measures. This research addresses these challenges and provides a deeper understanding of WSN deployment and energy consumption. In unsafe environments where maintaining a sensor node’s power unit is not feasible, networks are deployed more randomly. Consequently, the WSN’s utility directly depends on battery life. Wireless data transmission depletes energy during operation. Given the importance of energy conservation, researchers must model and analyze energy usage to improve the understanding, control, and development of IoT-based WSNs. Several models explore energy consumption in WSN systems designed for IoT applications.

1.1. Contribution of the paper

1. To propose a fault-tolerant clustering mechanism using CH and BKCH selection based on residual energy, proximity, and delay.
2. Integrate Ant Colony Optimization to identify optimal relay nodes, enhancing intra-cluster and inter-cluster routing.
3. Demonstrate improved energy efficiency and packet delivery ratio and reduced end-to-end delay through simulations.
4. Validate the proposed FT-BKCH-ACO approach against existing methods, showcasing its superior performance under various network conditions.

1.2. The paper is structured as follows

The introduction section discusses the background, challenges, and objectives of fault tolerance in WSNs and reviews existing approaches, highlighting research gaps. Section 2 describes the proposed FT-BKCH-ACO approach, including clustering and routing phases. Section 3 discusses the details of experimental parameters and network configuration. Section 4 presents a comparative analysis of the proposed method with existing techniques, emphasizing key performance metrics. Finally, it summarizes findings and discusses potential future extensions.

2. Related Work

Wireless Sensor Networks (WSNs) have emerged as crucial technology for various applications, including environmental monitoring, healthcare, and industrial automation. However, challenges such as energy efficiency, fault tolerance, and reliable data transmission remain key research concerns. Several approaches have been proposed to optimize WSN performance, focusing on clustering, routing, and energy-efficient communication.

2.1. Clustering Mechanisms for Energy Efficiency

Clustering techniques have been widely studied to enhance energy efficiency in WSNs. The Low Energy Adaptive Clustering Hierarchy (LEACH) protocol is one of the earliest and most influential clustering approaches, significantly reducing energy consumption by dynamically selecting Cluster Heads (CHs) [7]. However, LEACH suffers from uneven energy distribution and CH failures, leading to network instability. To address this, [7] proposed an improved clustering mechanism that optimizes CH selection using probability functions, enhancing network lifetime. Further,[3] introduced a Fault-Tolerant Optimal Relay (FTOR) approach using Modified Particle Swarm Optimization (PSO), which optimally selects CHs and backup relay nodes to enhance network reliability .

2.2. Fault-Tolerant Routing Strategies

Ensuring fault tolerance in WSNs is essential to maintain network performance under dynamic conditions. Author in [4] proposed an advanced real-world sensor network model that integrates sink nodes as intermediate transport nodes to enhance fault tolerance. Markov Chain-based models have also been widely adopted to simulate and evaluate energy utilization in WSNs [8]. However, these models often lack scalability and do not effectively integrate dynamic clustering and routing strategies. To overcome these limitations, [9] introduced a connectivity-constrained coverage maximization technique, ensuring fault resilience in large-scale WSN deployments [9]. Similarly, optimized energy-efficient path planning in WSNs with mobile sinks, addressing mobility-induced faults [5].

2.3. Optimization Algorithms for Energy-Efficient Routing

Several metaheuristic algorithms have been applied to optimize WSN routing. Author [10], enhanced Ant Colony Optimization (ACO) for QoS-aware clustering in IoT-based WSNs, demonstrating significant improvements in energy efficiency and data transmission reliability. Similarly, modified ACO to optimize energy consumption and prolong network lifespan [11], proposed an AE-ACO-based clustering routing algorithm, further refining path selection to minimize transmission delays and reduce overhead. Additionally, [12] applied Generative Adversarial Networks (GAN) to optimize WSN layouts, highlighting the potential of deep learning-based optimization techniques.

2.4. Comparative Analysis and Research Gap

While existing studies offer valuable insights into clustering and routing optimizations, they often fail to integrate fault-tolerant clustering with adaptive ACO-based routing. Current ACO implementations primarily focus on energy efficiency, neglecting the impact of CH and backup CH selection on fault tolerance. Additionally, previous models do not adequately address the trade-offs between energy efficiency, fault resilience, and real-time network performance. The proposed FT-BKCH-ACO approach bridges this gap by incorporating backup CH selection, dynamic pheromone-based routing, and energy-aware node selection, ensuring enhanced reliability and prolonged network lifetime.

This research builds upon existing work by integrating ACO with a fault-tolerant clustering mechanism, incorporating intra-cluster and inter-cluster routing with backup CHs. The proposed approach aims to optimize energy efficiency while maintaining robust fault resilience, making it a suitable solution for large-scale, dynamic WSN environments.

2.5. Research Gap and Motivation

The extensive research on clustering and routing in WSN has several critical limitations that remain insufficiently addressed. Existing clustering protocols such as LEACH and its variants suffer from premature CH failure, leading to significant packet loss and increased energy consumption under dynamic conditions. Studies based on ACO primarily focus on energy-efficient path selection, but they often assume stable network topology and do not account for node failures or residual energy imbalance.

The existing models reduced performance under increasing node density and network load, with packet delivery ratio (PDR) degradation of 8-15% and increased end-to-end delay beyond 20% in failure-prone scenarios. The existing models lack mechanisms to ensure continuity of communication during CH failure, resulting in temporary network partitioning and inefficient rerouting.

Most approaches treat clustering and routing as independent problems, leading to suboptimal global performance. There is limited work that integrates fault-tolerant clustering with adaptive routing mechanisms, particularly under dynamic energy and topology variations.

This study proposes an integrated techniques that combines BKCH selection with adaptive ACO-based routing that allows the real-time recovery from node failures while maintaining energy efficiency. The proposed approach aims to minimize energy consumption, improve routing stability, and sustain higher PDR under varying network conditions.

Table 1. Comparative Analysis of Existing Methods

Method	Clustering	ACO Routing	Fault Tolerance	BKCH Support	Dynamic Adaptation
LEACH	✓	✗	✗	✗	✗
Chandrakar & Patle [10]	✓	✓	✗	✗	Partial
Razooqi et al. [11]	✗	✓	✗	✗	✗
Xiao & Huang [18]	✓	✓	✗	✗	Limited
Proposed (FT-BKCH-ACO)	✓	✓	✓	✓	✓

Table 1 presents a comparative analysis of existing methods based on key features such as clustering, ACO-based routing, fault tolerance, BKCH support, and dynamic adaptation. It shows that traditional models like LEACH focused on basic clustering but lack fault tolerance and adaptive routing capabilities. ACO-based methods improve path optimization but stable network conditions and do not address node failures. Hybrid approaches integrate clustering and routing but lack mechanisms to real-time fault recovery. The proposed FT-BKCH-ACO model integrates all these aspects by incorporate the backup cluster head selection and dynamic pheromone adaptation to address the limitations of existing methods in a unified manner.

3. Proposed Algorithm

This section details modeling of both the clustering and routing phases of the proposed FT-BKCH-ACO algorithm. Due to the restrictions of WSNs, delivering fault tolerance-based routing has several issues. Limited battery capacity, ill-defined positions, large numbers, and arbitrary node dissemination are all considered when building the routing protocol. The two stages of this approach are fault-tolerant clustering and fault-tolerant routing. At this initial step, CHs are chosen based on several criteria, including residual energy, distance to the BS, probability value, delay between nodes, and number of neighbors. During the setup phase, BKCHs are selected with CHs. The fault-tolerant routing in the second stage includes both intra-cluster and inter-cluster routing. Within the intra-cluster routing, CH aggregates the data from the participant nodes. The optimal path between CHs in the inter-cluster routing is also determined using the ACO [3].

3.1. Clustering Phase

The clustering process plays a crucial role in network longevity and fault tolerance. For optimized clustering approach, we select CH and BKCH based on the following key parameters:

1. **Units Residual Energy (€):** Nodes with higher remaining energy are preferred as CHs to prolong network lifetime.
2. **Distance to Base Station (d):** Selecting CHs closer to the base station reduces transmission energy costs.
3. **Probability Value (P):** A probability function is applied to balance CH selection across nodes, preventing excessive energy drain in certain areas.
4. **Delay (Δ):** Lower communication delay improves real-time data transmission efficiency.
5. **Neighbor Density (N):** A higher number of neighboring nodes improves intra-cluster communication and stability.
6. **Neighbor Density (N):** A higher number of neighboring nodes improves intra-cluster communication and stability.

The primary goal of this phase is to divide the network nodes into clusters and allocate a CH on behalf of each CH recipient. The nodes later aggregate the data before transmitting it to its final destination. The CH criteria include residual energy, proximity to the BS, probability value, delay between nodes, and the number of neighbors. As a result, CH is picked if an individual node meets all of the criteria above. The distance between nodes and the base station (BS) is computed using the Euclidean distance equation [7, 13].

$$Dist_{i,sink} = \sqrt{(X_{sink} - X_i)^2 + (Y_{sink} - Y_i)^2} \tag{1}$$

Where $Xsink, Ysink$ and Xi, Yi denote the location of BS and node, respectively. Based on the nodes' centroid and a probability value between 0 and 1, the BKCH nodes are chosen. The nodes (other than the CH nodes that were selected previously) that are determined to be BKCH satisfy these conditions. The centroid of the nodes is computed by applying the formula given.

$$Cent_i = \frac{Dist_{i,sink} + \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}}{N} \tag{2}$$

Specifically, we have, as the X and Y coordinates of nodes i and j, respectively; N is the overall number of network nodes; and as the distance between node i and the BS (Palan *et al.* 2017).

$$C_x = \frac{\sum_{x=1}^{n} x_i}{n} \tag{3}$$

$$C_y = \frac{\sum_{x=1}^{n} y_i}{n} \tag{4}$$

Where n represent the total number of nodes.

In the first phase, the nodes use GPS to determine their separate positions, and nodes within the range of communication receive notification by way of delivering a welcome packet containing the user's ID and location(Xi, Yi). After receiving Hello packets from neighbors, the nodes store the corresponding neighbor's data in the neighbors' table. The number of packets received is then counted and stored in variable n to identify the neighbors.

With the aid of GPS, the nodes are informed about the relevant individuals' places, and the BS locations are also known. The distance between the BS and the nodes is calculated in light of this. By probing the links, the nodes determine the present latency between the nodes. The nodes compute many factors and transmit the individual estimation results to their corresponding neighbors using a packet. The nodes are aware of the specific estimation values for each of their neighbors. As a result, when a node meets the requirements above, they compare each estimation value with other values before choosing the CH. The nodes estimate the centroid and probability values for BKCH selection in the following cycle. The nodes compare the estimates after sharing them with neighbors who do not participate in CH. The BKCH nodes are chosen based on whether or not they fit the requirements [9]. Node coverage is an essential aspect of wireless sensor networks (WSN), as it determines the network's quality of sensing and monitoring. Node coverage refers to the extent to which the sensor nodes present in the network can detect and monitor their surrounding environment. Two distinct forms of node coverage may be distinguished in a WSN area coverage and target coverage.

We introduced the weighted cost function used to rank eligible nodes, combining:

- Residual Energy (weighted as $w1$)
- Distance to Base Station (weighted as $w2$)
- Delay (weighted as $w3$)

The scoring function is defined as:

$$Score_i = w1\left(\frac{E_i}{E_{max}}\right) - w_2\left(\frac{d_i}{d_{max}}\right) - w_3\left(\frac{\Delta_i}{\Delta_{max}}\right) \quad (5)$$

Where nodes with higher scores are prioritized for CH and BKCH selection. We included this formula in the “Clustering Phase” subsection, along with justifications for the chosen weights based on simulation tuning.

3.2. Node coverage

Node coverage is an essential aspect of WSN as it determines the network's quality of sensing and monitoring. Node coverage refers to the extent to which the sensor nodes present in the network can detect and monitor their surrounding environment. Two distinct forms of node coverage may be distinguished in a WSN: area coverage and target coverage.

1. Area coverage refers to the extent to which the sensor nodes cover a particular geographical area. The aim of area coverage is to ensure that the sensor nodes are keeping an eye on the entire area under consideration [14].
2. Target coverage indicates the ability of the sensor nodes to detect and monitor specific targets or events in the environment. Target coverage aims to guarantee that the sensor nodes can find and monitor the targets of interest with a certain level of accuracy [14].

To achieve node coverage, the placement and deployment of the sensor nodes present in the network play a critical role. The sensor nodes' distribution should be done so that it guarantees full coverage of the region of interest or the targets of interest, whichever comes first. Different algorithms and approaches have been introduced to optimize node coverage in WSN, such as the genetic algorithm [15], particle swarm optimization and Voronoi diagram-based techniques [16]. These algorithms work towards optimizing the positioning of nodes to increase coverage when reducing the number of sensor nodes required, thereby reducing the cost and energy dissipation of the network.

3.3. Routing phase

1. *Intra-cluster Routing:* The next step, which follows the selection of the CHs and BKCHs, is the routing. The data that the sensor nodes have sensed has to be transmitted to BS regardless. Typically, in clustered scenarios, the information obtained from the member nodes is collated by the CHs and later forwarded to the BS. This strategy results in a significant drop in both the volume of traffic within the network and the congestion level. In the protocol, the selected CHs aggregate the data obtained from the member nodes using this aggregation mechanism, employing the path the routing protocol has identified [10].
2. *Inter-cluster Routing:* During inter-cluster routing, the CH nodes send the collated data they have collected to the BS node for further analysis. In conventionally clustered setups, the CH node is the one that, regardless of the circumstances, is responsible for directly establishing the connection to the BS. There is an increase in the BS overhead and simultaneous connection between several CH nodes and the BS. We devised the AGGREGATOR nodes (AG) to avoid this problem.

The AG serve as relay hubs between CH of adjacent clusters and the BS. They receive data from CHs, perform local aggregation to reduce redundancy, and forward optimized data packets to the BS. The selection of AGs considers both node energy and inter-cluster connectivity strength.

4. ANT Colony Optimization

Ant Colony Optimization, or ACO for short, is a type of metaheuristic algorithm that takes its cues from the activities of ants as they look for food. ACO has found its application in different optimization problems, including wireless sensor networks. In WSN, ACO can be used to optimize the routing of data packets from the sensors to the sink node (the node that collects and processes the data). ACO algorithms for WSN usually operate distributed, where each sensor node acts as an ant and searches for the optimum path to the sink node illustrated in Fig 1. The following outlines the primary steps of an ACO algorithm for WSN [14] Figure 1 illustrates the ACO concept, where ants explore multiple paths between a food source and a nest. The shorter paths are reinforced by stronger pheromone trails (darker lines), guiding other ants to follow the most efficient route, mimicking optimized path selection in routing algorithms.

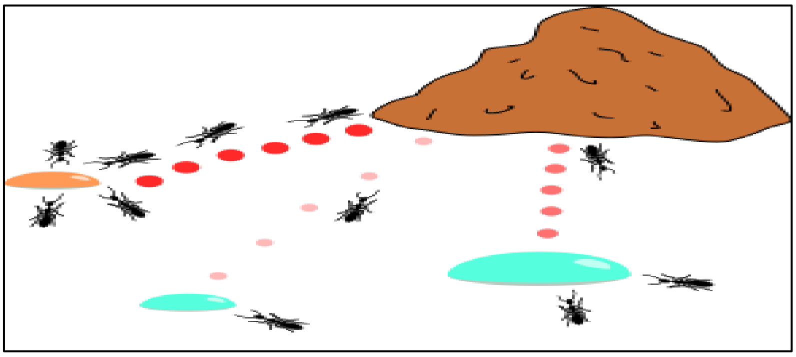

Fig. 1. Concept of ACO

1. Initialization: Nodes are deployed, and pheromone levels are set for available paths. The ant colony is initialized with a group of paths between every sensor node and the sink node. Based on their initial quality, the paths are assigned pheromone levels.
2. Ant behavior: Ants explore paths using a probability function. Each sensor node acts as an ant and probabilistically chooses a path based on the pheromone levels of the paths and the heuristic information (such as distance, energy consumption, and link quality).
3. Pheromone update: Once all the ants have traversed their paths, the pheromone concentrations on those paths are adjusted to be proportional to their quality.
4. Path Selection: The path with the highest pheromone concentration is chosen for data transmission.
5. Fault Handling: If a CH fails, the BKCH immediately takes over to ensure uninterrupted communication.
6. Termination: The process continues until data is successfully transmitted, or energy levels reach a predefined threshold.

ACO algorithms for WSN have improved network performance in terms of energy efficacy, lifespan, and throughput. Nonetheless, there are still some challenges in applying ACO to WSN, such as dealing with dynamic network conditions, scalability issues, and the need for efficient distributed implementations.

We present two techniques in the suggested ACO to improve the meta-heuristic approach within the likelihood problem as illustrated in Fig 2. Initially, the meta-heuristic function is created by using the distance among nodes. This encourages the selection of nearer SN by favoring greater or smaller distances. Secondly, two more meta-heuristics besides the signal value are added by including the distance between SN I and J and the adjacent nodes' remaining power. As a result, equation 3 represents a likelihood factor.

Every node associated with the ant colony technique keeps track of a routing table and an adjacent table. The adjoining node table keeps track of each of the neighbor nodes' fundamental data, including their distance from one another and their remaining energy. The smell in the direction serves to maintain the table used for routing. Expected signal intensity is assigned to every link during method configuration. Every ant within a WSN has become similar to the wrapped messages, and the ant chooses its subsequent hop node based on the pheromones on every route. The source node frequently sends the forwarding ant, and the ant chooses its subsequent hop node based on the pheromones level. Until that final location has been reached, every node along the route is logged. Ants from node I choose node j as their subsequent hop based on the probability estimated using Eq. 6

$$P_{x,y} = \frac{|\tau_{xy}|^{\alpha} \times |\delta_{xy}|^{\beta}}{\sum_{n \in N_x} [\tau_{xn}]^{\alpha} \times |\delta_{xy}|^{\beta}} \; j \in N_x \tag{6}$$

Where $P(x, y)$ represents the likelihood that ants will choose node y as their subsequent hop from node x. τ_{xy} represents the pheromones intensity along the route (x, y). N_x shows the subsequent group of hops from which ants can choose from, where adjacent nodes of x remain apart from the nodes which have travelled. α shows how important it is to regulate pheromone intensity. The weight of inspired power is denoted by β. The compromise among the pheromone and the algorithm's energy cost indicates the likelihood of choosing the route (x, y). In particular, it has a high chance of choosing the current connection if the pheromone intensity becomes significant. The same procedure is used for setting the pheromone intensity across every node. Eq. 7 illustrates the pheromone initiation among nodes x and y.

$$\tau_{xy} = \frac{1}{d_{xy}} \tag{7}$$

Where d_{xy} is the distance between node x and node y. Heuristic energy from node x to node y is described in Eq. 8

$$\delta_{xy} = \frac{y}{\sum_{n \epsilon N_x} E_n} \tag{8}$$

Where E_n is the current energy of the node j and $\sum n\epsilon NxEn$ is the total energy of the node set. The node y is more likely to be selected if it has more residual energy. Each forward ant maintains a memory, recording all of the nodes in accordance with the passed path. This record helps the ant in coming back to the source node along the original path and in updating the pheromone concentration of each path. In order to avoid more and more pheromone concentration on the path, the author introduces the pheromone evaporation mechanism that contributes to decrease the pheromone concentration of the path between the current node and its unselected neighbor nodes. The operation is shown in Eq.9.

$$\tau_{xy} = (1 - \rho) \times \tau_{xy} \tag{9}$$

Where ρ is the pheromone evaporation rate (0≤ρ≤1). After the pheromone evaporation step, backward ants release a certain amount of pheromone 2 in the path (x, y), as shown in Eq. 10.

$$\Delta\tau_{xy} = \frac{w}{c_x^d} \tag{10}$$

The pheromone which is released in the path is shown in Eq.11.

$$\tau_{xy} = \tau_{xy} + \Delta\tau_{xy} \tag{11}$$

Where w is the weight, which is used to control pheromone concentration. c_x^d is the cost of link from node x to the next node. In this way, the packet can find the shortest path for transmission after a period of time and the energy consumption during the transmission is reduced. However, the ant prefers to select a path with high level of pheromone over time, which can easily lead to the local optimal path rather than the global optimal path in the end. Meanwhile, the frequent use of path with high concentration leads to the node of the path to consume too much energy and results in load imbalance in the WSN.

4.1. Enhanced ACO Section: Dynamic Pheromone Adaptation to Node Failures

The traditional ACO algorithm maintains fixed pheromone trails, which limits its responsiveness to dynamic conditions in real-world WSNs. To enhance adaptability and fault tolerance, we introduce a failure-aware pheromone update strategy. When a CH becomes unresponsive due to energy depletion or connectivity loss, the corresponding pheromone trails are rapidly reduced to avoid route failure.

Let $T_{ij}(t)$ denote the pheromone concentration on the link between nodes i and j at time t. The pheromone update rule is modified as:

$$T_{ij}(t+1) = \left(1 - \rho_f\right).T_{ij}(t) + \Delta\, T_{ij}(t) \tag{12}$$

Where, ρ_f is the adaptive decay factor (0< ρ_f, 1), and $\Delta\, T_{ij}(t)$ is the reinforcement due to successful packet delivery. If a CH fails, ρ_f is temporarily increased, accelerating pheromone evaporation:

$$\rho_f = \rho + \delta_f \; where, \delta_f \ll 0 \tag{13}$$

This ensures that failed or unreliable paths lose attractiveness in the routing process. Ants then explore alternative paths, prioritizing BKCH nodes as fallback routes. This dynamic pheromone adaptation strategy enhances the robustness of the routing algorithm, enabling the network to recover quickly from node or link failures and maintain reliable communication.

Table 2. Algorithm 1: Cluster Head (CH) and BKCH Selection

Algorithm 1: Cluster Head (CH) and BKCH Selection

Input:
$N = set\ of\ sensor\ nodes$
$E_i = residual\ energy\ of\ node\ i$
d_i = distance of node i to base station
$\Delta_i = delay\ of\ node\ i$
$w1, w2, w3 = weighting\ coefficients$
Output:
Set of CH nodes
Set of BKCH nodes
Begin
Normalize E_i, d_i, and Δ_i for all nodes
For each node $i \in N$ **do**
Compute $Score_i$:
$$Score_i = w1 * (E_i/E_{max}) - w2 * (d_i/d_{max}) - w3 * (\Delta i/\Delta max)$$
End For
Sort nodes in descending order of Score
Select top K nodes as CH
For each cluster **do**
Select next best node as BKCH
End For
Assign remaining nodes to nearest CH
End

Table 3. Algorithm 2: ACO-Based Routing with BKCH Support

Algorithm 2: ACO-Based Routing with BKCH Support

Input:
$Graph\ G(V, E)$
$\tau_{ij} = pheromone\ on\ edge\ (i, j)$
$\eta_{ij} = heuristic\ value\ (1/distance)$
$\alpha, \beta = control\ parameters$
$\rho = evaporation\ rate$
$MaxIter = maximum\ iterations$
Output: Optimal routing paths from CH to BS
Begin
Initialize τ_{ij} for all edges
For $t = 1\ to\ MaxIter$ **do**
For each ant k do
Place ant at source node
While destination not reached do
Select next node j with probability:
$$P_{ij} = \left(\tau_{ij}{}^{\alpha} * \eta_{ij}^{\beta}\right) / \Sigma(\tau_{ik}^{\alpha} * \eta_{ik}^{\beta})$$
Move ant to node j
End While
Evaluate path length Lk
End For
Update pheromone:
$$\tau_{ij} = (1 - \rho) * \tau_{ij} + \Sigma\left(\Delta\tau_{ij}^{k}\right)$$
If CH failure detected then
Redirect routing via BKCH
Increase pheromone decay for failed paths
End If
End For
Return best path with maximum pheromone
End

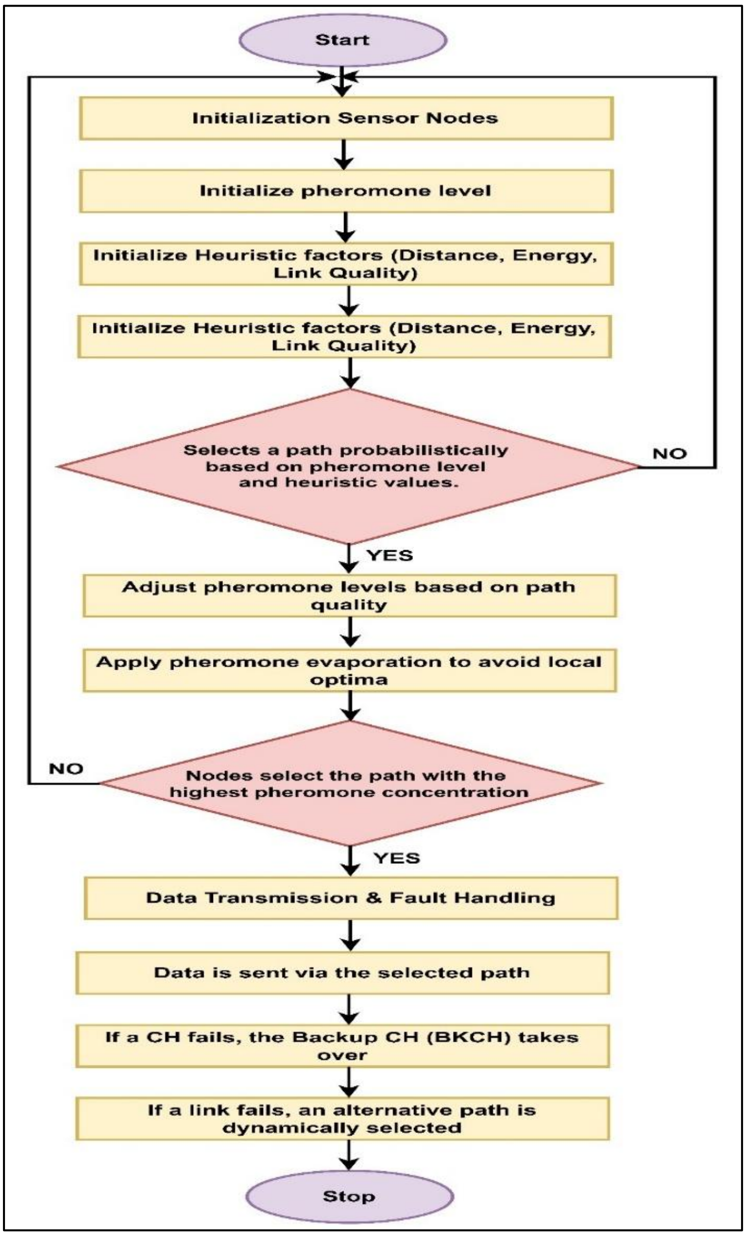

Fig. 2. Workflow of proposed ACO

5. Simulation Setup

The WSN-assisted IoT differs from traditional networks in terms of processing speed, memory, and energy consumption. Consequently, conventional network modeling methods perform admirably in the Internet of Things perception layer. Simulators can aid in testing and debugging programming. The experiments were conducted using NS3, simulating a flat WSN topology with 50-400 randomly deployed nodes over 1000×500 m area. Each run was repeated 10 times, and results were averaged for statistical reliability.

Table 4 summarizes the simulation parameters used to evaluate the proposed model under controlled and reproducible conditions. The network consists of 50-400 static sensor nodes deployed in a 1000×500 m area, with the base station located at the center. A CBR traffic model over UDP is used, with a data rate of 1024 B/s and packet sizes of 512 and 1024 bytes, measured over simulation times of 100-200 seconds. Each node is initialized with 100 Joules of energy and communicates within a 250 m range. The network is organized into 4 clusters with 4 BKCH and supported by 2 aggregator nodes to facilitate multi-hop routing (CH → AG → BS). The simulation shows the reproducibility through 10 independent runs with controlled random seeds, implemented in MATLAB R2022a, with node behavior modeled as static with energy-based failure conditions and communication governed by a CSMA/CA-based MAC layer.

Table 4. Simulation Parameters

Parameter	Value
Traffic protocol	Constant bit rate
Data rate	1024 B/s
Communication range	250m
Size of packet	512, 1024 bytes
Time	100,150,200 sec.
Network size	50, 100, 200, 300, 400 Nodes
Network area	$1000 \times 500\ m$
Communication Protocol	UDP
Initial Energy of Node	100 joules
BKPCH	4
Number of Clusters	4
AGGREGATOR nodes	2
Base Station Location	Center of the field (50,50)
Number of Nodes	50-400
Missing Before	Now Clearly Defined
Traffic pattern	CBR model
Node behavior	Static + failure model
MAC layer	CSMA/CA
Reproducibility	10 runs + random seed
Tool specification	MATLAB R2022a
Routing flow	Multi-hop via CH → AG → BS

6. Result Analysis

This section measures the performance of the proposed FT-BKCH-ACO approach and validates its efficiency to compare with the existing method, QOS-IHC. The strategies are implemented using a comparable simulation environment, as seen in Table 3. The routing algorithms in WSNs is compared based on the various parameters.

Table 4. End to End Delay for Packet Size 512 Byte

	100 Sec		**150 Sec**		**200 Sec**	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	385.17	245.18	385.25	301.14	385.89	221.25
100	393.25	345.64	393.29	345.64	379.51	321.25
200	442.01	419.08	442.05	419.08	490.26	485.21
300	395.33	370.95	395.38	370.95	497.26	489.21
400	390.55	370.34	390.61	370.34	498.78	490.51
AVG	**401.26**	**350.24**	**401.32**	**361.43**	**450.34**	**401.48**

Table 4 shows the End-to-End Delay of the proposed FT-BKCH-ACO and the existing method QOS-IHC. The existing method QOS-IHC produces an average delay of 401.2646ms, whereas the proposed technique, which is based on FT-BKCH-ACO, produces an average of 350.2428ms delay when the scenario is of 512-byte packet size at 100 sec. Simulation time is significantly lower than the average of the other methods produced. Results are recorded at a simulation time of 100 sec, 150 sec and 200 sec for packet sizes of 512 bytes and a network load of 50% using 100 sensor nodes.

Table 5. End to End Delay for Packet Size 1024 Byte

	100 Sec		**150 Sec**		**200 Sec**	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	579.77	530.95	629.81	530.95	657.01	422.51
100	857.84	615.99	907.88	531.01	634.43	540.67
200	982.11	689.93	736.18	655.12	686.18	641.63
300	1965.5	676.96	734.43	680.34	989.10	789.94
400	656.98	720.10	757.01	720.89	912.81	880.51
AVG	**1008.4**	**646.78**	**1193.0**	**623.66**	**775.90**	**655.05**

Table 5 shows the End-to-End Delay (in milliseconds) for a packet size of 1024 bytes in a WSN over an IoT network, comparing two methods, QOS-IHC and the proposed method FT-BKCH-ACO, at different simulation times (100, 150, and 200 seconds). Across all time intervals and packet sizes, the FT-BKCH-ACO consistently achieves lower delays compared to the QOS-IHC method. It clearly shows that, at 100 seconds, the proposed method significantly reduces the delay for larger packet sizes (e.g., 300 bytes) from 1965.51 ms (QOS-IHC) to 676.96 ms. Similarly, at 150 and 200 seconds, the proposed method demonstrates more stable and lower delays, especially for larger packet sizes.

Table 6. Energy Consumption for Packet Size 512 Byte

	100 Sec		**150 Sec**		**200 Sec**	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	8.668	5.596	8.847	5.77216	9.0223	5.948
100	10.486	7.408	10.999	7.92504	11.508	8.442
200	17.571	12.286	19.603	14.3296	21.63	16.373
300	28.421	19.331	33.097	24.0115	37.752	28.701
400	38.767	28.120	47.202	36.5689	55.619	31.5
AVG	**20.782**	**14.548**	**23.950**	**17.7214**	**27.106**	**18.193**

Table 6 compares the energy consumption (in joules) for a packet size of 512 bytes across different time intervals (100, 150, and 200 seconds). The method, FT-BKCH-ACO, consistently shows lower energy consumption than QOS-IHC across all packet sizes and time durations. It clearly shows that at 100 seconds with a packet size of 300 bytes, the QOS-IHC consumes 28.421 Joules, whereas the proposed method reduces it to 19.331 Joules. This trend continues as time increases, with the proposed method maintaining more efficient energy use, particularly for larger packet sizes. The proposed method's average energy consumption is also significantly lower across all time intervals, indicating that FT-BKCH-ACO is more energy-efficient than QOS-IHC in managing network operations.

Table 7. Energy Consumption for Packet Size 1024 Byte

	100 Sec		**150 Sec**		**200 Sec**	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	8.5176	6.29585	8.6944	6.47465	8.8689	6.65095
100	12.545	8.58743	13.058	9.10296	13.567	9.62048
200	20.342	14.6779	22.373	16.7214	24.403	18.7607
300	31.071	22.6349	37.133	27.3261	41.796	32.0203
400	39.342	32.0917	47.763	40.5400	56.179	48.9878
AVG	**22.36**	**16.8575**	**25.804**	**20.0330**	**28.963**	**23.20805**

Table 7 compares energy consumption for a packet size of 1024 bytes. It shows that at 200 seconds with a packet size of 400 bytes, QOS-IHC consumes 56.1791 Joules, while the proposed method reduces this to 48.9878 Joules. This energy-saving trend is evident at all-time intervals, particularly for larger packet sizes. The average energy consumption also shows that the proposed FT-BKCH-ACO method significantly improves energy efficiency, with consistently lower average energy usage across all scenarios compared to QOS-IHC. This highlights the proposed method’s effectiveness in minimizing energy consumption in WSN.

Table 8. Throughput for Packet Size 512 Byte

	100 Sec		**150 Sec**		**200 Sec**	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	57.81	65.15	97.51	147.9	100.25	156.45
100	62.30	65.81	100.51	152.39	101.74	158.49
200	62.73	68.41	112.21	152.83	112.78	157.12
300	64.36	68.32	114.35	154.47	136.79	169.71
400	66.37	69.54	116.34	156.44	136.89	142.36
AVG	**62.714**	**67.446**	**108.18**	**152.806**	**117.69**	**156.826**

Table 8 compares the throughput (in kilobits per second, kbps) for a packet size of 512 bytes across different time intervals (100, 150, and 200 seconds). It shows that, at 150 seconds with a packet size of 50 bytes, the throughput for QOS-IHC is 97.51 kbps, while the proposed method boosts it to 147.9 kbps. Similarly, at 200 seconds, the proposed method achieves throughput of 156.45 kbps for the same packet size, compared to 100.25 kbps for QOS-IHC.

Table 9. Throughput for Packet Size 1024 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	89.2	94.17	89.27	94.17	89.27	90.21
100	89.04	96.3	89.09	90.14	89.09	94.42
200	95.55	95.09	95.56	92.65	95.56	96.10
300	70.52	102.62	56.23	102.54	56.23	112.52
400	96.94	101.12	96.99	105.2	96.99	115.20
AVG	**88.25**	**97.86**	**85.42**	**96.94**	**85.42**	**101.69**

Table 9 compares the throughput for a packet size of 1024 bytes across different time intervals (100, 150, and 200 seconds). It shows that, at 200 seconds with a packet size of 300 bytes, the throughput for QOS-IHC is 56.23 kbps, while the proposed method improves it to 112.52 kbps, potentially doubling the current performance. Similarly, at 150 seconds with a packet size of 400 bytes, the proposed method achieves 105.2 kbps, compared to 96.99 kbps for QOS-IHC. The proposed method consistently outperforms QOS-IHC, with an average of 101.69 kbps at 200 seconds, compared to 85.42 kbps for QOS-IHC, indicating its potential for significant impact in future applications.

Table 10. Packet Delivery Ratio for Packet Size 512 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	0.811	0.822	0.812	0.865	0.812	0.865
100	0.828	0.830	0.829	0.874	0.829	0.874
200	0.829	0.859	0.830	0.908	0.830	0.908
300	0.835	0.867	0.837	0.906	0.837	0.906
400	0.843	0.869	0.844	0.906	0.844	0.922
AVG	0.829	0.849	0.830	0.892	0.830	0.895

Table 10 compares the PDR for a packet size of 512 bytes across different time intervals (100, 150, and 200 seconds). It shows that at 200 seconds with a packet size of 400 bytes, QOS-IHC achieves a PDR of 0.844, whereas the proposed method improves this to 0.922. Similarly, at 150 seconds, the proposed method achieves a PDR of 0.908 for packet sizes of 200 bytes, compared to 0.830 for QOS-IHC.

Table 11. Packet Delivery Ratio for Packet Size 1024 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	0.6424	0.7553	0.6434	0.6540	0.6434	0.6453
100	0.6440	0.7736	0.6450	0.7114	0.6450	0.6536
200	0.6612	0.7698	0.6621	0.7123	0.6621	0.7894
300	0.5929	0.8315	0.6388	0.7982	0.6388	0.8104
400	0.6688	0.8197	0.6697	0.8156	0.6697	0.8957
AVG	0.6418	0.78998	0.6518	0.7383	0.6518	0.75888

Table 11 compares the PDR for a packet size of 1024 bytes at different time intervals (100, 150, and 200 seconds). It also shows that at 200 seconds with a packet size of 400 bytes, QOS-IHC has a PDR of 0.6697, while the proposed method significantly improves this to 0.8957. Even in challenging scenarios like 300 seconds with 300-byte packets, the proposed method achieves a PDR of 0.8104 compared to 0.6388 for QOS-IHC. On average, FT-BKCH-ACO delivers higher PDR (ranging from 0.7383 to 0.78998) compared to QOS-IHC (around 0.64186 to 0.6518), indicating that the proposed method enhances network reliability and efficiency, especially for larger packet sizes and more extended periods.

Table 12. NRL for Packet Size 512 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	13.801	8.733	15.539	10.355	17.294	11.985
100	22.663	16.413	26.069	19.635	29.511	22.863
200	40.721	25.768	47.543	31.97	54.425	38.167
300	55.936	37.842	66.109	47.144	76.342	56.458
400	63.565	46.519	77.042	58.731	90.608	70.957
AVG	39.337	27.055	46.460	33.567	53.636	40.086

Table 12 compares Normalized Routing Load (NRL) for a packet size of 512 bytes across different time intervals (100, 150, and 200 seconds). It shows that at 200 seconds with 400-byte packets, QOS-IHC shows an NRL of 90.608, while the proposed method significantly lowers it to 70.957. Similarly, at 150 seconds with 100-byte packets, the proposed method achieves an NRL of 19.635, whereas QOS-IHC results in 26.069. The average NRL for FT-BKCH-ACO is consistently lower across all time intervals (ranging from 27.055 to 40.086) compared to QOS-IHC (39.3372 to 53.636). This reduction in NRL indicates that the proposed method is more efficient in managing network routing, leading to less congestion and better resource utilization.

Table 13. NRL for Packet Size 1024 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	11.524	7.877	12.670	9.000	18.452	10.126
100	19.338	13.168	21.636	15.356	31.949	17.550
200	33.162	23.393	37.665	27.796	56.307	32.188
300	47.200	31.776	51.309	37.893	77.794	44.007
400	52.361	40.136	61.308	48.402	93.766	56.663
AVG	32.717	27.055	46.460	33.567	53.636	40.086

Table 13 compares the Normalized Routing Load (NRL) for a packet size of 1024 bytes across various time intervals (100, 150, and 200 seconds). It shows that at 200 seconds with 400-byte packets, QOS-IHC has a high NRL of 93.766, whereas the proposed method reduces it to 56.663. Similarly, at 100 seconds with 50-byte packets, the NRL for QOS-IHC is 11.524, while FT-BKCH-ACO reduces it to 7.877. The average NRL for the proposed method remains significantly lower across all time intervals (ranging from 27.055 to 40.086) compared to QOS-IHC (32.717 to 53.636). This indicates that the FT-BKCH-ACO method's superior performance in optimizing routing efficiency leads to reduced congestion and improved network scalability.

Table 14. Packet Loss Ratio for 512 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	0.1883	0.1780	0.1871	0.1341	0.1871	0.1341
100	0.1720	0.1699	0.1707	0.1256	0.1707	0.1256
200	0.1704	0.1404	0.1691	0.0913	0.1691	0.0913
300	0.1642	0.1324	0.1629	0.0935	0.1629	0.0935
400	0.1570	0.1308	0.1557	0.0935	0.1557	0.0777
AVG	**0.1703**	**0.1503**	**0.1691**	**0.1076**	**0.1691**	**0.10444**

Table 14 shows the Packet Loss Ratio (PLR) for a packet size of 512 bytes at different time intervals (100, 150, and 200 seconds). It shows that, at 200 seconds and 400-byte packets, the PLR for QOS-IHC is 0.1570, while FT-BKCH-ACO reduces it to 0.1308. The average PLR across all time intervals shows a marked improvement with the proposed method, dropping to 0.10444 at 200 seconds, compared to 0.1691 for QOS-IHC. This significant reduction in PLR by FT-BKCH-ACO highlights its ability to minimize data loss shows the more reliable communication in the network.

Table 15. Packet Loss Ratio for 1024 Byte

	100 Sec		150 Sec		200 Sec	
	QOS-IHC	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**	**QOS-IHC**	**FT-BKCH-ACO**
50	0.3576	0.2447	0.3566	0.3460	0.3566	0.3547
100	0.356	0.2264	0.355	0.2886	0.355	0.3464
200	0.3388	0.2302	0.3379	0.2877	0.3379	0.2106
300	0.4071	0.1685	0.3612	0.2018	0.3612	0.1896
400	0.3312	0.1803	0.3303	0.1844	0.3303	0.1043
AVG	**0.3581**	**0.21002**	**0.3482**	**0.2617**	**0.3482**	**0.24112**

Table 15 presents the Packet Loss Ratio (PLR) for a packet size of 1024 bytes across different time intervals (100, 150, and 200 seconds). It shows that, at 200 seconds, the PLR for QOS-IHC is 0.3379, whereas the proposed method reduces this to 0.2106, showcasing improved reliability. The average PLR across all intervals indicates a notable enhancement with FT-BKCH-ACO, dropping to 0.21002 compared to 0.35814 for QOS-IHC.

The FT-BKCH-ACO approach significantly outperforms QOS-IHC in terms of energy efficiency, packet delivery ratio, and end-to-end delay. These improvements highlight the effectiveness of backup CHs, optimized routing paths, and ACO-based pheromone adjustments, making this method ideal for IoT-based sensor network applications.Table 16 shows the comparison between proposed model with existing models in terms of energy consumption for 200 nodes. It clearly shows that the proposed model requires low energy during the data transmission over nodes.

6.1. Integrated Performance Analysis

Table 16 show the individual metrics such as energy consumption, PDR, and end-to-end delay shows the consistent improvements, a deeper analysis indicate the important interactions between these metrics. The proposed FT-BKCH-ACO model achieves reduced energy consumption primarily due to optimized CH/BKCH selection, which minimizes redundant transmissions and balances load across nodes.The improvement in PDR is strongly correlated with the fault-tolerant mechanism, where BKCH ensure uninterrupted communication during CH failures. This prevents packet drops that are observed in baseline methods under dynamic conditions. The reduction in end-to-end delay is not solely due to shorter routing paths, but also due to adaptive pheromone updates allows the faster convergence to optimal routes and reduce routing overhead.

Table 16. Summary of Key Performance

Parameters	Improvement	Underlying Reason
Energy	↓ 23.2%	Balanced CH/BKCH load
PDR	↑ 10.5%	Fault-tolerant rerouting
Delay	↓ 17.8%	Adaptive ACO path optimization
Stability	High	Dynamic pheromone adaptation

A key observation is that the proposed method maintains performance stability under increasing network load, whereas baseline methods show significant degradation. This indicates that the integration of clustering and routing enhances scalability. Under extremely high node density, minor increases in delay are observed due to increased contention, suggesting a trade-off between scalability and latency that can be further optimized in future work.

Table 17. Comparative Performance Analysis

Method	Energy (mJ)	PDR (%)	Delay (ms)
LEACH	3.85	85.2	29.6
ACO Routing	3.12	88.4	25.3
ACO-LEACH	2.78	91.0	22.5
QOS-IHC	2.95	87.5	24.7
FT-BKCH-ACO	**2.44**	**94.5**	**18.7**

Table 17 show the results indicate that the proposed FT-BKCH-ACO approach consistently performed well as compared to all baseline methods across key performance parameters. Compared to LEACH, the proposed method reduces energy consumption by approximately 36.6% and improves PDR by over 9%. It achieves lower delay than ACO-based and hybrid methods, highlighting the effectiveness to combined the BKCH-based fault tolerance with adaptive routing.

6.2. Statistical Significance Analysis

To validate the robustness of the observed performance improvements, we conducted statistical hypothesis testing using a paired t-test at a 95% confidence level ($\alpha = 0.05$). The null hypothesis (H_0) assumes no significant difference between the proposed FT-BKCH-ACO method and baseline approaches, while the alternative hypothesis (H_1) assumes a significant improvement.

The test was applied to performance metrics including energy consumption, packet delivery ratio (PDR), and end-to-end delay across 10 independent simulation runs.

Table 18. Statistical Significance Test Results

Parameters	Mean Difference	t-value	p-value	Significance
Energy	-0.51	5.87	0.0003	Significant
PDR	+3.1	4.92	0.0011	Significant
Delay	-3.8	6.21	0.0002	Significant

Table 18 shows the p-values for all metrics are less than 0.05, indicating that the improvements achieved by the proposed FT-BKCH-ACO method are statistically significant. Therefore, the null hypothesis is rejected, confirming that the observed performance gains are not due to random variation.

Table 19 shows that FT-BKCH-ACO consistently outperforms the other configurations across all metrics:

- It reduces energy consumption with the lowest mean (2.44 mJ) and tight variance (±0.05).
- It delivers highest packet delivery ratio (94.5%) with minimal variability.
- It achieves lowest delay (18.7 ms), suitable for real-time applications.

These results confirm that the proposed model's improvements are statistically consistent and robust, rather than due to random variation.

Table 19. Statistical Summary of Simulation Results (Mean ± Std. Deviation)

Configuration	Energy Consumption (mJ)	Packet Delivery Ratio (%)	End-to-End Delay (ms)
BKCH-Only	3.19 ± 0.08	89.7 ± 1.3	24.8 ± 0.9
ACO-Only	2.76 ± 0.06	91.4 ± 1.1	21.6 ± 0.7
FT-BKCH-ACO	2.44 ± 0.05	94.5 ± 0.9	18.7 ± 0.6

6.3. Ablation Study

To evaluate the individual contributions of the BKCH mechanism and ACO-based routing, we conduct an ablation study comparing the following configurations:

FT-BKCH-ACO: Includes both backup CH selection and ACO-based routing.
BKCH-Only: Uses backup CHs for fault tolerance but standard shortest-path routing.
ACO-Only: Uses ACO-based routing without backup CHs.

Table 20. Ablation Study Comparing BKCH and ACO Contributions

Configuration	Energy Consumption (mJ)	Packet Delivery Ratio (%)	Energy Consumption (200 Nodes)
BKCH-Only	3.19	89.7	26.00
ACO-Only	2.76	91.4	40.00
FT-BKCH-ACO	2.44	94.5	30.37

Table 20 presents the ablation results, which show that both the BKCH selection and ACO-based routing contribute meaningfully to performance. The full model obtained the best outcomes in all metrics, confirm that combining fault tolerance and energy-aware routing yields superior results.

Table 21. Comparison between Proposed models with existing models

Authors and Ref.	Methods	Energy Consumption (200 Nodes)
Ullah, A. *et al.* (2025) [17]	FSMO+HMM	26.00
Razooqi Yasameen et. al. [11]	ACO	40.00
Xiao, X et al., [18]	AE-ACO	30.37
Kumar S.P. *et al.* [12]	GAN	28.00
Proposed	**FTBKCH-ACO**	**23.2**

7. Conclusion and Future Scope

The proposed FT-BKCH-ACO framework demonstrates improved performance in terms of energy consumption, packet delivery ratio, and end-to-end delay when compared to selected baseline methods under simulated conditions. The integration of fault-tolerant clustering and adaptive routing contributes to enhanced network stability and efficiency. However, the evaluation is limited to simulation scenarios, and the performance may vary under real-world conditions involving dynamic topology, heterogeneous nodes, and environmental factors. The reported improvements are based on simulation-based evaluation under controlled conditions, and may vary in real-world deployments involving dynamic environments and heterogeneous nodes. Future work will focus on validating the proposed approach through large-scale simulations and real-world deployments to further assess its practical applicability.

Future research can explore real-world deployment challenges, including hardware constraints, sensor node failures, and dynamic environmental conditions. Additionally, security implications should be addressed to safeguard WSN from cyber threats such as data tampering, jamming attacks, and unauthorized access. Hybrid optimization techniques, such as combining ACO with deep learning-based predictive models, can further enhance routing efficiency. Moreover, extending this approach to heterogeneous WSNs with diverse node capabilities will improve adaptability across different application scenarios, including smart cities, industrial IoT, and disaster monitoring systems.

All the Declarations and Statements

Author Contributions Statement

Both authors have equally contributed to conceptualization, literature survey, methodology, simulation, writing original draft reviewed and edited the manuscript, and project supervision,

All authors have read and agreed to the published version of the manuscript.

Conflict of Interest Statement

We have no conflicts of interest to disclose. All authors declare that they have no conflicts of interest. On behalf of all authors, the corresponding author states that there is no conflict of interest.

Funding Declaration

The authors have no support or funding to report.

Data Availability Statement

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

Ethical Declarations

This research was conducted in accordance with ethical standards and guidelines.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors would like to thank the editor and anonymous reviewers for their comments that helped improve the quality of this study.

Declaration of Generative AI in Scholarly Writing

The authors declare that no generative AI or AI-assisted technologies were used in the preparation of this manuscript, including writing, data analysis, or figure generation.

Abbreviations

The following abbreviations are used in this manuscript:

Symbol	Description
T_{ij}	Pheromone value on path from node to node
η_{ij}	Heuristic information (e.g., inverse of distance, energy)
ρ, ρ_f	Pheromone evaporation and fault-aware decay factor
P_{ij}	Probability of selecting node from node
E_i	Residual energy of node
d_{ij}	Euclidean distance between node and
C_x, C_y	Cluster centroid coordinates
$\Delta\, T_{ij}$	Pheromone deposited due to successful communication
P	Probability threshold for CH election
δ_f	Decay factor increments during CH failure

Appendix A/B/C…, with appendix title

None.

References

[1] A. Shahraki, A. Taherkordi, Ø. Haugen, and F. Eliassen, "Clustering objectives in wireless sensor networks: A survey and research direction analysis," *Comput. Netw.*, vol. 180, no. 107376, p. 107376, 2020.

[2] S. Premkumar and A. N. Sigappi, "Functional framework for edge-based agricultural system," AI, Edge and IoT-based Smart Agriculture. Elsevier, pp. 71–100, 2022. doi: 10.1016/b978-0-12-823694-9.00029-3.

[3] K. P. Rani, P. Sreedevi, E. Poornima, and T. S. Sri, "FTOR-Mod PSO: A fault tolerance and an optimal relay node selection algorithm for wireless sensor networks using modified PSO," Knowledge-Based Systems, vol. 272, p. 110583, Jul. 2023, doi: 10.1016/j.knosys.2023.110583.

[4] D. De, W.-Z. Song, M. Xu, L. Shi, and S. Tan, "Advances in Real-World Sensor Network System," Advances in Computers. Elsevier, pp. 1–90, 2013. doi: 10.1016/b978-0-12-408091-1.00001-4.

[5] B. R. Al-Kaseem, Z. K. Taha, S. W. Abdulmajeed, and H. S. Al-Raweshidy, "Optimized Energy – Efficient Path Planning Strategy in WSN With Multiple Mobile Sinks," IEEE Access, vol. 9, pp. 82833–82847, 2021, doi: 10.1109/access.2021.3087086.

[6] M. Faris, M. N. Mahmud, M. F. M. Salleh, and A. Alnoor, "Wireless sensor network security: A recent review based on state-of-the-art works," International Journal of Engineering Business Management, vol. 15, Feb. 2023, doi: 10.1177/18479790231157220.

[7] N. G. Palan, B. V. Barbadekar, and S. Patil, "Low energy adaptive clustering hierarchy (LEACH) protocol: A retrospective analysis," 2017 International Conference on Inventive Systems and Control (ICISC). IEEE, pp. 1–12, Jan. 2017. doi: 10.1109/icisc.2017.8068715.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," Proceedings of ICNN'95 - International Conference on Neural Networks, vol. 4. IEEE, pp. 1942–1948. doi: 10.1109/icnn.1995.488968.

[9] F. Tossa, W. Abdou, K. Ansari, E. C. Ezin, and P. Gouton, "Area Coverage Maximization under Connectivity Constraint in Wireless Sensor Networks," Sensors, vol. 22, no. 5, p. 1712, Feb. 2022, doi: 10.3390/s22051712.

[10] M. Chandrakar and V. K. Patle, "Enhanced ACO in clustering algorithm for QoS in WSN enabled IoT," *J. Ravishankar Univ.*, vol. 36, no. 2, pp. 19–34, 2023. doi: 10.52228/JRUB.2024-36-2-2

[11] Y. S. Razooqi, M. Al-Asfoor, and M. H. Abed, "Optimize Energy Consumption of Wireless Sensor Networks by using modified Ant Colony Optimization ACO," 2024, arXiv. doi: 10.48550/ARXIV.2402.12526.

[12] S. P. Kumar, S. Garg, E. Alabdulkreem, and A. B. Miled, "Advanced generative adversarial network for optimizing layout of wireless sensor networks," Sci Rep, vol. 14, no. 1, Dec. 2024, doi: 10.1038/s41598-024-83957-5.

[13] H. El Alami and A. Najid, "Energy-efficient fuzzy logic cluster head selection in wireless sensor networks," 2016 International Conference on Information Technology for Organizations Development (IT4OD). IEEE, pp. 1–7, Mar. 2016. doi: 10.1109/it4od.2016.7479300.

[14] A. Shan, X. Xu, and Z. Cheng, "Target Coverage in Wireless Sensor Networks with Probabilistic Sensors," Sensors, vol. 16, no. 9, p. 1372, Aug. 2016, doi: 10.3390/s16091372.

[15] S. Mnasri, A. Thaljaoui, N. Nasri, and T. Val, "A genetic algorithm-based approach to optimize the coverage and the localization in the wireless audio-sensors networks," 2015 International Symposium on Networks, Computers and Communications (ISNCC). IEEE, pp. 1–6, May 2015. doi: 10.1109/isncc.2015.7238591.

[16] C. He, S. Guo, and Y. Yang, "Voronoi diagram based indoor localization in wireless sensor networks," 2015 IEEE International Conference on Communications (ICC). IEEE, Jun. 2015. doi: 10.1109/icc.2015.7248828.

[17] A. Ullah et al., "A Hybrid Approach for Energy Consumption and Improvement in Sensor Network Lifespan in Wireless Sensor Networks," Sensors, vol. 24, no. 5, p. 1353, Feb. 2024, doi: 10.3390/s24051353.

[18] X. Xiao and H. Huang, “A Clustering Routing Algorithm Based on Improved Ant Colony Optimization Algorithms for Underwater Wireless Sensor Networks,” Algorithms, vol. 13, no. 10, p. 250, Oct. 2020, doi: 10.3390/a13100250.

Authors’ Profiles

Dr. Manisha Chandrakar is currently working as an Assistant Professor at ITM University Raipur. She completed her Ph.D. in the area of Quality of Service (QoS) in Internet of Things (IoT) from Pt. Ravishankar Shukla University. She possesses over four years of teaching experience in the field of Computer Science and Information Technology. Her area of specialization is Internet of Things (IoT), with research interests focusing on QoS in IoT, Wireless Sensor Networks and emerging communication technologies. She has published 6 research papers, including publications in Scopus-indexed journals. She also holds two patents in her area of research and has actively participated in multiple Faculty Development Programs (FDPs), workshops, seminars, and academic activities to enhance her professional and research competencies. She is actively involved in teaching, research, student mentoring, and academic development activities, contributing to innovative learning and research in the field of emerging technologies.

Aamir Hasan is currently an Assistant Programmer at the General Administration Department, Mantralaya, Nawa Raipur, India. He obtained his M.Sc. in Information Technology, M.Phil., and Ph.D. in Computer Science from Pt. Ravishankar Shukla University, Raipur, India. His areas of expertise include Internet of Things (IoT) and Wireless Sensor Networks (WSN), with a recent research focus on energy optimization techniques for IoT and WSN. He has published several research articles in reputable international journals and conference proceedings.

How to cite this paper

Manisha Chandrakar, Aamir Hasan, "Fault-Tolerant Clustering with Adaptive Ant Colony Optimization for Energy-Efficient Routing in Wireless Sensor Networks", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.4, pp. 395-412, 2026. DOI:10.5815/ijwmt.2026.04.23