

BREAKTHROUGH FRUITFLY OPTIMIZATION-BASED LEACH ROUTING PROTOCOL (BFO-LRP) FOR PACKET DELAY MINIMIZATION IN WIRELESS BODY AREA NETWORKING (WBAN)

S. VEERARATHINAKUMAR¹, Dr. B.DEVANATHAN²

¹Research Scholar, Department of Computer and Information Science, Annamalai University, India

²Assistant Professor, Department of Computer and Information Science, Annamalai University, India

E-mail: ¹vkumar66gee@gmail.com, ²devacisau@gmail.com

ABSTRACT

Wireless Body Area Networks (WBANs) play a vital role in healthcare and wearable devices, enabling seamless communication and monitoring of sensors attached to the human body. However, efficient routing in WBANs faces challenges due to dynamic body movement, constrained energy resources, and mobility-induced disruptions. To address these issues, this abstract introduces the “Breakthrough Fruitfly Optimization-based LEACH Routing Protocol (BFO-LRP).” BFO-LRP leverages the fruitfly-inspired optimization algorithm, BFO, and incorporates Nonlinear Hierarchical Decision-Making (NHDM) to achieve coordinated and optimized routing decisions. The proposed BFO-LRP aims to overcome data packet delays, limited energy resources, and other routing challenges in WBANs, enhancing the performance of the LEACH protocol. By employing a hierarchical decision-making approach, BFO-LRP ensures adaptive search space exploration, leading to enhanced convergence and efficient exploitation of promising regions. Extensive simulations are conducted using the ns3 network simulator to evaluate its effectiveness. The results demonstrate that BFO-LRP outperforms conventional routing protocols regarding packet delivery performance, reduced delays, and efficient energy utilization, making it a promising solution for routing optimization in WBANs. This research advances WBAN technology, providing better healthcare and wearable applications support.

Keywords: *Delay, Fruitfly Optimization, NHDM, LEACH, WSN, WBANs*

1. INTRODUCTION

Wireless Body Area Networks (WBANs), a Wireless Sensor Network (WSN) type, have emerged as a promising technology for healthcare applications. WBANs enable continuous monitoring of vital signs, provide real-time feedback, and improve the overall quality of patient care. These networks consist of small, lightweight, and wearable sensors placed on or inside the human body to wirelessly collect and transmit physiological data [1]. By leveraging context-aware routing, WBANs can overcome the unique challenges posed by their limited energy resources, dynamic topology, and varying traffic patterns, thus enhancing network performance and energy efficiency [2]. One of the critical challenges in WBANs is the efficient routing of data packets from the sensors to the destination, typically a sink node or a medical server. Traditional routing protocols, such as ad hoc or static routing, may not be well-suited for WBANs due to the specific characteristics of these networks. However, an optimized routing approach is crucial because

inefficient routing can lead to excessive energy consumption, increased transmission delays, and high packet loss rates, all of which compromise the reliability of patient monitoring and real-time data acquisition [3].

Context-aware routing considers various contextual information within the network, including the body's physiological state, location, and activity levels. This information is crucial for making informed routing decisions that optimize network performance and conserve energy. In WBANs, the body's physiological state plays a significant role in context-aware routing [4]. By continuously monitoring vital signs such as heart rate, blood pressure, temperature, and oxygen saturation, WBANs can gather valuable data regarding the individual's health status. By integrating this information into the routing process, the network can prioritize delivering critical data during emergencies or abnormal conditions, ensuring timely responses and medical

interventions. Location awareness is another essential aspect of context-aware routing in WBANs [5]. By knowing the physical location of the sensors and sink nodes, routing decisions can be optimized based on proximity. For instance, if a sink node is geographically closer to a sensor, the routing algorithm can select a shorter path, reducing transmission delay and energy consumption. Location information can facilitate seamless handoffs between access points or relay nodes as the patient moves within the network coverage area. Activity levels provide valuable context for routing decisions in WBANs [6]. By detecting the wearer's physical activity, such as walking, running, or resting, WBANs can adjust routing strategies to optimize energy efficiency. Routing algorithms can adjust transmission power, data aggregation, or routing paths based on activity levels. For example, during periods of low activity, the network can enter a sleep mode or reduce the data transmission frequency to conserve energy [7].

To address these challenges, this research proposes the Breakthrough Fruitfly Optimization-based LEACH Routing Protocol (BFO-LRP), which enhances the traditional LEACH protocol by integrating bio-inspired optimization. The Fruitfly Optimization Algorithm (BFO) ensures an adaptive search mechanism that dynamically selects energy-efficient and low-latency routes, while Nonlinear Hierarchical Decision-Making (NHDM) further refines routing strategies by evaluating network conditions in real-time. By leveraging these techniques, BFO-LRP optimizes packet transmission, enhances energy efficiency, and minimizes data packet delays, making it highly suitable for WBAN applications.

WBANs, as a type of WSN, offer tremendous potential for healthcare applications. Context-aware routing addresses the unique challenges WBANs face, such as limited energy resources, dynamic topology, and varying traffic patterns [8]. By considering contextual information, such as the body's physiological state, location, and activity levels, routing algorithms in WBANs can dynamically adapt their decisions to optimize network performance and conserve energy. This technology can revolutionize healthcare by providing personalized and remote monitoring capabilities, enabling early detection of health issues, and facilitating timely interventions for improved patient care and healthcare outcomes [9], [10].

1.1. Problem Statement:

The problem in WBANs is efficiently routing data packets from the wearable sensors to the destination, such as a sink node or medical server. Traditional routing protocols are not well-suited for WBANs due to their unique characteristics, including limited energy resources, dynamic topology, and varying traffic patterns. Consequently, there is a need for context-aware routing algorithms that can make intelligent routing decisions based on contextual information, such as the physiological state of the body, location, and activity levels. By integrating this context into the routing process, WBANs can optimize network performance and conserve energy. Developing effective context-aware routing solutions requires addressing several challenges, including accurately capturing and interpreting contextual information, designing adaptive routing algorithms that can dynamically adjust their decisions, and ensuring the timely delivery of critical data during emergencies or abnormal conditions. Furthermore, seamless handoffs and efficient resource allocation based on proximity and activity levels are essential for enhancing the overall performance and energy efficiency of WBANs. Solving these challenges will enable WBANs to fully realize their potential in revolutionizing healthcare by providing personalized and remote monitoring capabilities, enabling early detection of health issues, and facilitating timely interventions for improved patient care and healthcare outcomes.

1.2. Motivation

The motivation behind developing context-aware routing for WBANs stems from these networks' unique requirements and potential in healthcare applications. WBANs offer the opportunity for continuous and personalized monitoring of individuals' health status, enabling early detection of abnormalities and timely interventions. However, traditional routing protocols are not optimized for the specific characteristics of WBANs, such as limited energy resources and dynamic network topology. By leveraging context-aware routing algorithms that consider the body's physiological state, location, and activity levels, WBANs can achieve improved network performance and energy efficiency. This, in turn, enables more accurate and timely transmission of critical data, enhances the reliability of healthcare services, and optimizes resource utilization. The development of context-aware routing in WBANs is motivated by the potential to revolutionize healthcare by enabling remote patient monitoring, personalized healthcare interventions, and enhanced quality of care. By

addressing the unique challenges of WBANs through context-aware routing, healthcare providers can leverage the full capabilities of these networks to improve patient outcomes and transform the delivery of healthcare services.

1.3. Objective

Implementing context-aware routing in WBANs aims to enhance network performance and energy efficiency while addressing the unique challenges WBANs pose. The specific objectives include:

- Optimize routing decisions: Develop routing algorithms that dynamically adapt to the changing context, such as the physiological state of the body, location, and activity levels, to optimize the selection of routing paths. This will improve data delivery efficiency and reduce latency in WBANs.
- Conserve energy: Design energy-aware routing strategies that consider the limited energy resources of WBAN devices. Considering the context and adapting routing decisions can minimize energy consumption, prolonging the network lifetime and ensuring continuous monitoring capabilities.
- Prioritize critical data: Develop mechanisms to prioritize transmitting critical data during emergencies or abnormal health conditions. By integrating context information into the routing process, time-sensitive and vital data can be delivered promptly, enabling timely medical interventions.
- Seamless handoffs and resource allocation: Implement routing protocols that facilitate seamless handoffs as patients move within the network coverage area. Additionally, consider proximity-based routing decisions to optimize resource allocation and reduce transmission delays.
- Enhance overall network performance: Improve the overall performance of WBANs by integrating context-aware routing algorithms that adaptively adjust transmission power, data aggregation, and routing paths based on the activity levels and requirements of the network.

By achieving these objectives, implementing context-aware routing in WBANs aims to enable reliable, efficient, and personalized healthcare

monitoring, leading to improved patient care and healthcare outcomes.

2. LITERATURE REVIEW

"E2S-HiRoute" [11] optimizes route selection in WBANs to achieve energy efficiency, security, and optimal network performance. It employs a hierarchical structure that divides the network into multiple levels, each responsible for specific tasks like routing, monitoring, or security. "HUBsFLOW" [12] aims to provide an efficient and seamless integration between WBAN devices and the SDN controller, facilitating enhanced network management and control. The protocol establishes a standardized communication interface between WBAN devices and the SDN controller, enabling dynamic configuration, monitoring, and control of WBAN resources. "AMERP" [13] aims to enhance the energy efficiency of routing in WBANs while considering the mobility of nodes. AMERP utilizes the Adam moment estimation algorithm to optimize the routing decisions based on energy consumption, node mobility, and network conditions. By dynamically estimating the moments of the energy function, AMERP adapts the routing paths to minimize energy consumption and prolong the network lifetime. Optimization plays a significant role in all kinds of networking-oriented research [14]-[42].

"HBC-TransHW" [43] is designed for WBAN applications. HBC-TransHW focuses on optimizing the architecture of the transceiver to enable reliable and efficient communication through the human body. The architecture incorporates specialized hardware components and signal processing techniques tailored for HBC, minimizing power consumption and maximizing data throughput. "ORE-WBAN" [44] focuses on achieving the optimal balance between reliability and energy efficiency in the design of WBANs. The approach considers network topology, resource allocation, routing protocols, and transmission power control to optimize the overall network performance. ORE-WBAN aims to ensure reliable and seamless communication while minimizing energy consumption and prolonging the network's battery life. "BSPE-WBAN" [45] focuses on developing and evaluating efficient broadcast strategies for IEEE 802.15.4, a popular communication standard for low-power wireless networks. It examines different broadcast mechanisms and evaluates their performance in WBAN environments. BSPE-WBAN considers network size, channel conditions,

energy consumption, and packet delivery ratio to determine optimal broadcast strategies.

"BioKey-EESR" [46] focuses on achieving energy efficiency and security robustness in key establishment processes within WBANs. The protocol leverages biometric data, such as fingerprints or heart rate variability, to establish secure keys between nodes in the network. BioKey-EESR incorporates energy-efficient algorithms and cryptographic techniques to optimize energy consumption while ensuring robust security. "Link-EnergyPref" [47] aims to optimize routing decisions by considering both the link quality and the energy consumption of the nodes in the network. The scheme evaluates the link quality metrics, such as signal strength or packet error rate, to determine the reliability and stability of the link. It considers the energy utilization of the nodes to select energy-efficient paths. "Fork-Hook Secure Gateway" [48] ensures proactive and trustworthy data transmission within the network. The policy incorporates encryption techniques to protect sensitive data and prevent unauthorized access. It utilizes a proactive trust ware mechanism that evaluates the trustworthiness of data sources and routing paths to ensure reliable and secure communication.

"ColorAlloc" [49] optimizes resource allocation by leveraging the vertex coloring technique. The algorithm assigns a unique color to each WBAN, representing its allocated resources, such as frequency channels or time slots. By using different colors, ColorAlloc ensures that coexisting WBANs operate on non-interfering resources, minimizing interference and improving overall network performance. "CLAKA-BWBAN" [50] provides a secure and efficient mechanism for binding agreements between WBAN nodes in a blockchain environment. Unlike traditional public critical infrastructure (PKI) systems, CLAKA-BWBAN eliminates the need for digital certificates, reducing computational complexity and communication overhead. The protocol leverages the security features of blockchain technology to establish authenticated and shared keys between WBAN nodes. "CAACA-WBAN" [51] focuses on enhancing the security and privacy of WBANs by considering the network context and enabling anonymous authentication for users. The mechanism considers various contextual factors such as location, time, and user behavior to adjust access control policies dynamically. It provides fine-grained access control based on the specific context of the WBAN,

allowing authorized users to access relevant data and services while preventing unauthorized access.

"SimpleEBAR-WBAN" [52] focuses on simplifying the routing process while ensuring energy balance among nodes and considering alternative routes for data transmission. SimpleEBAR-WBAN considers the limited energy resources of wearable devices and aims to evenly distribute the energy consumption among them to prolong the network lifetime. "PriorityBalance" [53] optimizes data transmission in WBANs. By assigning priorities to different types of traffic and efficiently balancing the load among nodes, PriorityBalance enhances network performance and ensures efficient utilization of resources. The approach prioritizes critical health data while effectively managing non-urgent information, reducing delays, and improving overall system efficiency. "InterEffiCare" [54] incorporates interference awareness mechanisms to identify and mitigate potential sources of interference that can hinder data transmission performance. InterEffiCare dynamically adjusts data transmission parameters to minimize energy consumption while maintaining efficient and reliable communication by considering channel conditions, traffic load, and interference levels.

The "Congestion Control Aware Routing Algorithm (CCARA)" [55] objective is to optimize energy efficiency and manage congestion within the network. The algorithm dynamically adjusts routing paths by considering real-time temperature measurements to minimize energy consumption and steer clear of congested areas. This ingenious adaptation helps maintain an efficient flow of data transmission. CCARA implements effective congestion control mechanisms, enabling the algorithm to manage network traffic and prevent bottlenecks. "Energy-efficient Harvest-aware Routing Protocol (E-HARP)" [56] addresses the energy efficiency challenges in WBANs by incorporating harvested energy awareness into the clustering and routing process. This protocol promotes the formation of energy-efficient clusters by considering the availability of harvested energy from wearable devices. It carefully selects energy-efficient cluster heads and employs cooperative routing strategies to optimize the utilization of energy resources and extend the network's lifespan.

3. PROPOSED WORK

3.1. LEACH

LEACH (Low-Energy Adaptive Clustering Hierarchy) is a widely used routing protocol for WSNs that employs a hierarchical clustering approach. The primary goal of this protocol is to reduce energy consumption within WSNs. To achieve this objective, the system arranges nodes into clusters, where each cluster is led by a designated cluster head (CH) responsible for facilitating communication with the base station (BS). Here's a step-by-step mathematical explanation of the LEACH protocol:

3.1.1. Cluster Setup

Each node in the network decides whether to become a CH or a normal node based on a probabilistic model. The probability is determined using a threshold value, usually set to ensure a fair distribution of CH roles across the network. Consider a network where p represents the CHs percentage. The value of p is determined by a specific threshold. Consequently, the average number of rounds required for a node to become a CH can be represented as $(1/p)$.

3.1.2. Cluster Head Selection

Each network node produces a random number between 0 and 1. If the random number generated is less than or equal to p , then the node is a Cluster Head (CH) for this iteration. Otherwise, it becomes a normal node. The probability of a node becoming a Cluster Head (CH) in a specific round can be determined using Eq.(1).

$$P(CH) = p / (1 - p * (r \bmod (1/p))) \quad (1)$$

where the variable r represents the current round, and the term "mod" indicates the modulo operation.

3.1.3. Cluster Formation

The CHs initiate the process by broadcasting a "Join Request" message to the ordinary nodes within their communication range. Each ordinary node assesses the signal strength and selects the nearest CH, subsequently joining that cluster.

3.1.4. Data Transmission

Each cluster has a CH that takes data from its nodes and compresses it if necessary. After that, the CHs use multi-hop communication to send the compiled data to the BS. The energy consumption for data transmission from a CH to the BS can be

calculated using the distance-based energy model, such as the Friis transmission equation.

3.1.5. Rotation of Cluster Heads

After each round, the CHs switch places to keep the total network power consumption constant. Each node calculates the probability of becoming a CH for the next round using Eq.(2).

$$P'(CH) = p / (1 - p * (r \bmod (1/p))) \quad (2)$$

where variable r represents the current round, while $P'(CH)$ denotes the probability of becoming a Cluster Head (CH) in the upcoming round.

3.1.6. Network Lifetime

The LEACH protocol optimizes the network's lifetime by effectively distributing energy consumption across all nodes. The protocol achieves enhanced energy efficiency and extends the overall network lifespan by employing a cluster-based approach and rotating Cluster Heads (CHs). The above steps summarize the basic working principle of the LEACH protocol. It provides a mathematical framework for cluster head selection and formation, enabling energy-efficient communication in wireless sensor networks.

Algorithm 1: LEACH	
Step 1: Initialization:	Initialize network parameters, distribute nodes randomly, and set initial energy levels.
Step 2: Cluster Setup:	Divide the network into rounds and determine the threshold value for each round.
Step 3: Cluster Head Selection:	Each node creates a random number; if that number is less than or equal to a threshold value, that node becomes the cluster leader for its cluster.
Step 4: Cluster Formation:	Cluster heads broadcast "Join Request" messages and normal nodes join the closest cluster.
Step 5: Data Transmission:	The cluster leaders send Data from each cluster to the base station.
Step 6: Rotation of Cluster Heads:	After each round, Energy usage is kept in check by cluster chiefs recalculating their chances of promotion at the end of each cycle.
Step 7: Network Lifetime:	

The LEACH protocol maximizes network lifetime by efficiently sharing the energy use load among nodes.

3.2. Nonlinear Hierarchical Decision-Making

Nonlinear Hierarchical Decision-Making (NHDM) involves an iterative process that begins with initializing decision variables x and y . The leader's problem is solved to find optimal x , considering the potential impact on the follower's decisions. The follower's problem is then solved to find optimal y given the leader's decisions. The process iterates until convergence is achieved. The final solutions for x and y represent the coordinated and optimized decisions for the leader and follower. Connectivity analysis helps understand the interdependence and coordination between the two levels, leading to more effective decision-making in complex real-world problems.

3.2.1. Initialization

The nonlinear hierarchical decision-making process begins by initializing the decision variables x and y with initial guesses x_0 and y_0 . This step sets the starting points for subsequent optimization iterations. The selection of initial guesses is crucial to improving convergence and optimizing the final solution. The decision-makers can choose these values based on historical data, expert knowledge, or random values within reasonable bounds. Initialization can be mathematically expressed as Eq.(3).

$$\begin{aligned} x &= x_0 \\ y &= y_0 \end{aligned} \tag{3}$$

The choice of initial guesses is crucial as it can affect the convergence and quality of the final solution. A good selection of initial values can lead to faster convergence and improved overall optimization performance. Using historical data, expert knowledge, or even random values within reasonable bounds for the decision variables is common.

3.2.2. Upper-Level Optimization (Leader)

In this step, NHDM solves the leader's optimization problem at the upper level of the decision hierarchy. The leader aims to optimize its objective function $f(x,y)$ while adhering to the constraints $g(x,y) \leq 0$. This process involves finding the optimal values for the decision variables x while considering the potential responses from the follower, represented by y . Mathematically, the leader's optimization problem can be formulated as Eq.(4).

$$\text{minimize } f(x,y) \tag{4}$$

$$\text{subject to } g(x,y) \leq 0$$

where $f(x,y)$ represents the objective function of the leader, and $g(x,y)$ represents the constraints. Solving this optimization problem results in finding the optimal values for x , which reflect the leader's best decisions while considering the potential influence on the follower's decisions.

3.2.3. Lower-Level Optimization (Follower)

After obtaining the optimal values for x from the leader's problem, NHDM solves the follower's optimization problem at the lower level of the decision hierarchy. The follower aims to optimize its objective function $h(x,y)$ while adhering to the constraints $g(x,y) \leq 0$. The follower makes its decisions y based on the optimal decisions x provided by the leader. The follower's optimization problem can be formulated mathematically as Eq.(5).

$$\begin{aligned} &\text{minimize } h(x,y) \\ &\text{subject to } y \in \text{argmin} \{h(x,y) \mid g(x,y) \leq 0\} \end{aligned} \tag{5}$$

where $h(x,y)$ represents the objective function of the follower. The constraint $y \in \text{argmin} \{h(x,y) \mid g(x,y) \leq 0\}$ ensures that y is selected to minimize the follower's objective function $h(x,y)$ while satisfying the constraints defined by $g(x,y) \leq 0$. The solution obtained for y represents the follower's best decisions considering the leader's decisions x .

3.2.4. Convergence Check

Once the solutions for both the leader and the follower problems are obtained, a convergence check is performed to determine if the optimization process has reached a termination condition. The convergence check evaluates the difference between the current and previous solutions for x and y . If the difference is smaller than a predefined tolerance, the optimization process has converged, and the algorithm stops. Otherwise, it returns to Step 2. The convergence check process is provided as Algorithm 2.

Algorithm 2: Convergence Check	
Step 1:	Compare the current values of x and y with their previous values from the last iteration.
Step 2:	Calculate the absolute difference between the current and previous values of x and y .
Step 3:	Check if both differences are smaller than a predefined small value (tolerance).

Step 4: If both differences are below the tolerance, the optimization process has converged.
Step 5: If not, repeat Steps 2 to 4 for the next iteration.

3.2.5. Result Analysis

In this step, the final solutions x and y are analyzed to understand the optimal decisions made by the leader and the follower, respectively. The objective function values $f(x,y)$ and $h(x,y)$ are evaluated to assess the performance of both levels. The analysis provides valuable insights into the effectiveness of the nonlinear hierarchical decision-making process and the interactions between the leader and the follower. The result analysis process is provided as Algorithm 3.

Algorithm 3: Result Analysis	
Step 1:	Evaluate the objective function values for the leader and the follower using the final optimal values of x and y .
Step 2:	Display the resulting objective values for the leader and the follower.
Step 3:	Analyze the objective values to understand the performance of both levels in the decision-making process.
Step 4:	Assess how the leader's and follower's decisions contribute to achieving their respective objectives.

3.2.6. Connectivity Analysis

Connectivity analysis studies the interactions and information flow between the leader and follower levels in the hierarchical decision-making process. It involves assessing how changes in the leader's decisions (represented by x) influence the feasible space for the follower's decisions (represented by y) and vice versa. Algorithm 4 expresses the Connectivity Analysis process.

Algorithm 4: Result Analysis	
Step 1:	Given the leader's optimal decision x , find the corresponding optimal decision y for the follower using the follower's optimization problem.
Step 2:	Assess how changes in the leader's decisions, represented by x , affect the feasible space for the follower's decisions, represented by y , by

comparing the values of y for different values of x .
Step 3: Analyze how changes in the follower's decisions, represented by y , influence the feasible space for the leader's decisions, represented by x .
Step 4: Identify the dependencies and feedback mechanisms between the leader and follower levels by studying how changes in one level impact the other level's decision-making process.
Step 5: Use the insights from the connectivity analysis to make more informed and coordinated decisions, improving overall system performance.

The connectivity analysis is crucial to NHD, as it helps decision-makers understand the interactions and dependencies between the leader and follower levels. It provides valuable insights into how changes in one level's decisions affect the other level's feasible space, fostering a coordinated and efficient decision-making process.

3.3. Fruitfly Optimization

The Fruitfly Optimization (FO) is an intelligent algorithm inspired by the foraging behavior of fruit flies. Fruit flies have demonstrated remarkable abilities in locating hidden food sources through their osphresis (sense of smell) and visual systems. FOA mimics two significant processes, osmosis and vision foraging, to discover food efficiently. It has undergone refinements, including dynamic modification of individual fruit fly search radius and considering geographic coordinates as potential solutions within the search area.

3.3.1. Initialization

In this step, the settings are reset to their default values, defining essential FO characteristics such as population size, maximum generation, and swarm location. Additionally, the parameters \exists_w^{max} and \exists_w^{min} are introduced to modify the search radius. The swarm's initial position is determined using Eq.(6).

$$\Delta = (\theta_1, \theta_2, \dots, \theta_t)$$

$$\theta_w = Z_w + (O_w - Z_w) \times rand \quad w = 1, 2, \dots, t \quad (6)$$

where Δ represents the swarm's initial condition, t is the number of decisions to be made, O_w is the position of the swarm in the w th decision variable, and Z_w and O_w are the lower and upper limits of the n th decision variable, respectively. The term $rand$ generates a random number in the interval $[0, 1]$.

3.3.2. Food Resource Creation

The creation of food resources is a crucial stage in emulating the foraging behavior of fruit flies. It aims to generate potential solutions (food resources) within the search space that the swarm can explore. The modification of the search radius Ξ_w is one of the significant enhancements in the refined versions of *FO*. The parameter Ξ_w , associated with each decision variable, influences the neighborhood size in which a fruitfly searches for food. This dynamic adaptation of the search radius allows the algorithm to effectively balance exploration and exploitation. In the early iterations, the search radius is more extensive when exploration is essential, enabling the swarm to cover a broader search space region. As *FO* progresses and the solution landscape becomes better understood, the search radius is reduced, and the swarm concentrates more on exploiting promising regions. It considers the highest (Ξ_w^{max}) and lowest (Ξ_w^{min}) limits of the parameter and the current iteration (j) relative to the maximum generation ($maxgen$). This ensures that the search radius evolves gradually over generations, allowing the algorithm to adapt its exploration strategy based on the problem's complexity and the search progress.

3.3.3. Odor Concentration Calculation

Calculating odour concentration for each generated food resource is a critical component. Odour concentration is a proxy for the quality or fitness of a given solution (food resource) within the search space. The goal function ($g()$) plays a central role in determining the odour concentration of each food resource. The specific form of the goal function depends on the optimization problem being solved. In the context of *FO*, the goal function is designed to evaluate the fitness of a solution based on its coordinates and other relevant features. Eq.(7) is utilized for calculating the odour concentration.

$$Smelli_s = \sum g(P_{sw}) \quad s = 1, 2, \dots, T \quad (7)$$

where $Smelli_s$ represents the scent concentration (odour concentration) of the s th food resource. $g(P_{sw})$ denotes the goal function evaluation for the s th food resource, determining its fitness. Higher odour concentration indicates a more promising solution, making it more attractive to the swarm during the foraging process. Just like fruit flies are guided by stronger scents to find food efficiently, *FO* utilizes odour concentration as a measure to direct the swarm towards better regions of the search space.

3.3.4. Swarm Movement

The movement of the swarm is a fundamental process that drives the exploration and exploitation

of the search space. The primary objective is to guide the swarm towards regions with higher odour concentrations (promising solutions) discovered during the previous steps. The swarm's movement is determined by the learning rate (α) and the difference between the current position of the swarm (θ_w) and the position of the best food resource found (P_{best}). The update equation for the swarm's position is given by Eq.(8).

$$\theta_w = \theta_w + \alpha \times (P_{best} - \theta_w) \quad (8)$$

where the value of w falls between 1 and t , θ_w represents the position of the swarm in the w th decision variable, α is a user-defined parameter known as the learning rate, controlling the step size of the swarm's update, P_{best} denotes the position of the best food resource found by the swarm.

3.3. Breakthrough Fruitfly Optimization

Breakthrough Fruitfly Optimization (*BFO*) is an enhanced and sophisticated variant of the original *FO* algorithm. Inspired by the foraging behaviour of fruit flies, *BFO* combines the power of *NHDM* with *FO* to tackle complex optimization problems more effectively. *BFO* introduces a hierarchical decision-making structure that allows for nonlinear and adaptive search space exploration, enabling the algorithm to achieve breakthrough results in various domains.

In the *BFO* algorithm, the optimization process begins with initializing crucial parameters, including population size, maximum generation, and search radius limits. *NHDM* plays a pivotal role in influencing the swarm initialization, food resource creation, and odour concentration calculation. The hierarchical decision-making units guide the swarm's movement at different levels of abstraction, enhancing exploration and exploitation strategies. The food resource creation process in *BFO* is driven by *NHDM*, resulting in a more refined and adaptive search strategy. The lower-level issue is solved using standard *FO*, and the knowledge gained is utilized to create optimal solutions for the next lower level. At each stage, *BFO* dynamically adjusts the search radius and parameters to balance exploration and exploitation.

Step 1: Parameter Initialization

The *BFO* algorithm is configured in this initial step by setting various crucial parameters. These parameters include the population size (N), representing the number of fruit flies in the swarm, and the maximum generation ($maxgen$), indicating

the total number of iterations the algorithm will perform. Additionally, the extra search radius parameters (\exists_w^{max} and \exists_w^{min}) are defined. These parameters control the adaptability of the search radius in the search space.

Step 2: Swarm Initialization

At the beginning of each generation (j) in the BFO algorithm, the swarm's initial position (Δ^j) is established. The swarm's position is represented as a vector containing the decision variables of the optimization problem. The decision variables are divided into two groups: p -dimension ($\theta_1^j, \theta_2^j, \dots, \theta_{t_p}^j$) and q -dimension ($\theta_{t_p+1}^j, \theta_{t_p+2}^j, \dots, \theta_{t_p+t_q}^j$). Each component of the swarm's position is initialized within the boundaries defined by Z_{w_p} and O_{w_p} for the p -dimension and $Z_{w_{p+w_q}}$ and $O_{t_{p+w_q}}$ for the q -dimension. Eq.(9) to Eq.(11) represents the swarm initialization process.

$$\Delta^j = (\theta_1^j, \theta_2^j, \dots, \theta_{t_p}^j, \theta_{t_p+1}^j, \theta_{t_p+2}^j, \dots, \theta_{t_p+t_q}^j) \quad (9)$$

$$\theta_{w_p}^j = Z_{w_p} + (O_{w_p} - Z_{w_p}) * rand, \quad w_p = 1, 2, \dots, t_p \quad (10)$$

$$\theta_{w_{p+w_q}}^j = Z_{w_{p+w_q}} + (O_{t_{p+w_q}} - Z_{t_{p+w_q}}) * rand, \quad w_q \quad (11)$$

where Δ^j represents the swarm's position at generation j . $\theta_{w_p}^j$ and $\theta_{w_{p+w_q}}^j$ are the positions of the swarm in the p -dimension and q -dimension, Z_{w_p} and O_{w_p} are the lower and upper limits for the p th decision variable, $Z_{w_{p+w_q}}$ and $O_{t_{p+w_q}}$ are the lower and upper limits, $rand$ generates a random number in the interval $[0, 1]$.

Step 3: Food Resource Creation

To simulate the foraging behavior of fruit flies in the search space, T food resources are randomly generated in each generation (j). Each food resource is represented as a pair (P_s^j, q_s^j) , where P_s^j represents the swarm's position in the p -dimension and q_s^j represents the position in the q -dimension. A vector guides the creation of these food resources $\Lambda^j = (\exists_1, \exists_2, \dots, \exists_{t_p}, \exists_{t_p+1}, \exists_{t_p+2}, \dots, \exists_{t_p+t_q})$, where each \exists_w represents the extra search radius parameter for the corresponding decision variable. The swarm's position is perturbed by a random value between $[-1, 1]$ scaled by the corresponding \exists_w to generate

each food resource P_s^j, q_s^j . Eq.(12) to Eq.(14) are utilized in food resource creation.

$$(P_s^j, q_s^j) = \Delta^j \pm \Lambda^j * rand = (p_{s1}^j, p_{s2}^j, \dots, p_{st_p}^j, q_{s1}^j, q_{s2}^j, \dots, q_{st_q}^j) \quad s = 1, 2, \dots, T \quad (12)$$

$$p_{sw_p}^j = p_{w_p}^j \pm p_{w_p}^j * rand, \quad w_p = 1, 2, \dots, t_p \quad (13)$$

$$q_{sw_q}^j = q_{w_q}^j \pm p_{t_{p+w_q}}^j * rand, \quad w_q = 1, 2, \dots, t_q \quad (14)$$

Step 4: Lower-Level Problem Solving

In this step, the BFO algorithm focuses on solving the lower-level issue represented by q -dimension for each food resource P_s^j . For each food resource (P_s^j, q_s^j) , the lower-level problem is solved using the standard FO approach. The swarm's position is initialized for the q -dimension as Δ^z , and the swarm starts the foraging process again to explore and exploit the q -dimension.

Step 4a: Swarm Position Initialization for Lower-Level Issue

The initial position of the swarm for the lower-level issue is established as Δ^z , representing the q -dimension of the swarm's position. The swarm's position (Δ^z) is initialized within the boundaries defined by $Z_{t_{p+w_q}}$ and $O_{t_{p+w_q}}$ for each decision variable in the q -dimension. Eq.(15) and Eq.(16) are applied for initializing the swarm position at lower-level issues.

$$\Delta^z = (\theta_1^z, \theta_2^z, \dots, \theta_{t_q}^z) \quad (15)$$

$$\theta_{q_q}^z = Z_{t_{p+w_q}} + (O_{t_{p+w_q}} - Z_{t_{p+w_q}}) * rand \quad w_q = 1, 2, \dots, t_q \quad (16)$$

Step 4b: Food Resource Creation for Lower-Level Issue

For the lower-level issue, c food resources are randomly generated, representing the swarm's positions in the q -dimension. Each food resource is a vector containing the decision variables in the q -dimension. The swarm's position (Δ^z) is perturbed by a random value between $[-1, 1]$ scaled by the corresponding $\exists_{t_{p+w_q}}$ to generate each food resource. Eq.(17) is applied for creating food resources at lower-level issue.

$$Q_{sqwq} = \theta_{wq}^z \pm \exists_{t_p+wq} \times rand \ s_q \quad (17)$$

$$= 1, 2, \dots, c$$

Step 4c: Odor Concentration Calculation for Lower-Level Issue

The relative strength of the odours for the lower-level issue is calculated for each food resource (P_s^j) in the q -dimension. The goal function ($g()$) evaluates the fitness of the lower-level issue solutions. The scent concentration (odour concentration) $smell_{sq}$ for each food, resource is determined based on the fitness evaluation of the corresponding solution. Eq.(18) determine the relative strength of the odours.

$$smell_{sq} = g(P_s^j, Q_{spwq}) \quad (18)$$

Step 4d: Swarm Movement for Lower-Level Issue

The swarm's position (Δ^z) is adjusted based on the best food resource (Q_{best}) found in the q -dimension. If the fitness value of Q_{best} is lower than the fitness value of the swarm's current position (Δ^z), the swarm moves to the position of Q_{best} , indicating a promising region in the q -dimension. Eq.(19) is applied to adjust the position of the swarm.

$$\Delta^z = Q_{best}, \text{ if } g(Q_{best}) \leq g(P_s^j, \Delta^z) \quad (19)$$

Step 4e: Termination of Lower-Level Issue

The BFO checks if the criteria for stopping the lower-level issue exploration are met. If not, the process repeats from Step 4b, exploring the q -dimension. If the stopping condition is satisfied, the lower-level issue is terminated, and the optimum solutions (q_s^{*j}) concerning the p -dimension (P_s^j) are returned.

Step 5: Odor Concentration Calculation for Upper-Level Issue

With the lower-level issue solutions (q_s^{*j}) returned, the relative strength of the odours for the upper-level issue is calculated. The scent concentration (odour concentration) $Smell_s^j$ is determined based on the fitness evaluation of each food resource (P_s^j, q_s^{*j}) in both the p -dimension and the q -dimension. Eq.(20) expresses the same.

$$Smell_s^j = G(P_s^j, q_s^{*j}) \quad (20)$$

Step 6: Swarm Position Refreshment

If the fitness value of the scent concentration ($Smell_{best}^j$) is lower than the fitness value of the swarm's current position (Δ^j), the swarm's position is updated to move towards the scent concentration with the lowest fitness value. This allows the swarm to focus on more promising regions in the search space.

Step 7: Halting Requirement Check

In the final step, the BFO checks the halting requirement to determine if the optimization process should terminate. If the stopping condition is satisfied, the BFO algorithm terminates, and the optimal solution ($P_{best}^j, q_{best}^{*j}$) is obtained. The algorithm then proceeds to Step 4 to continue the optimization process by further exploring and exploiting the search space.

Algorithm 5: Breakthrough Fruitfly Optimization-based LEACH Routing Protocol

Step 1: Initialization

- a). Set population size (N).
- b). Set maximum generation ($maxgen$).
- c). Set upper limit for extra search radius parameter (\exists_w^{max}).
- d). Set lower limit for extra search radius parameter (\exists_w^{min}).
- e). Generate the initial fruit fly swarm position Δ^j .

Step 2: Food Resource Creation

- a). Generate T food resources (P_s^j, q_s^j) in each generation (j) based on the current swarm position (Δ^j).
- b). Perturb the swarm's position (Δ^j) by a scaled random value to create each food resource (P_s^j, q_s^j).

Step 3: Lower-Level Problem Solving

- a). For each food resource (P_s^j, q_s^j):
- b). Initialize the swarm position for the lower-level issue (Δ^z) in the q -dimension.
- c). Generate c food resources in the q -dimension (q_s^{*j}).
- d). Perturb the swarm's position (Δ^z) by a scaled random value to create each q -dimension food resource (q_s^{*j}).
- e). Evaluate the scent concentration $smell_{sq}$ based on the goal function

$(g())$ for each q-dimension food resource.

- Choose the best q-dimension food resource (Q_{best}) based on the lowest fitness value.
- Update the swarm's position Δ^z to the position of Q_{best} if it improves fitness.
- Check termination criteria for the lower-level issue. Return the optimal solutions q_s^{*j} concerning the p-dimension (P_s^j) if met.

Step 4: Odour Concentration Calculation for Upper-Level Issue

- For each food resource p_s^j, q_s^{*j} :
- Evaluate the scent concentration $Smell_s^j$ based on the goal function ($G()$) for the combined p-dimension and q-dimension solutions.

Step 5: Swarm Position Refreshment:

- Choose the best scent concentration $G(Smell_{bestj})$ in the upper-level issue.
- Update the swarm's position (Δ^j) to the position with the lowest fitness value if it improves fitness.

Step 6: Halting Requirement Check:

- Check the halting criteria to determine if the optimization process should terminate.
- If the stopping condition is satisfied:
- Return optimal solution.
- Optionally, proceed to Step 2 to continue the optimization process if further exploration is required.
- Otherwise, repeat Steps 2 to 6 for the next generation ($j + 1$) until the maximum generation ($maxgen$) is reached.

The Breakthrough Fruitfly Optimization-based LEACH Routing Protocol combines the power of BFO with NHDM to achieve more effective decision-making and routing in complex real-world problems. The algorithm iteratively proceeds through generations until the halting criteria are met, or the maximum generation is reached. This results in potentially superior routing performance for the LEACH protocol in wireless sensor networks.

4. SIMULATION SETTING

Simulating WBAN routing in NS3 is essential to investigating and improving routing protocols for healthcare applications. NS3 allows researchers to create realistic WBAN topologies, considering factors like node placement, communication ranges, and obstacles. The wireless environment can be accurately replicated by incorporating channel and interference models.

Various routing protocols, including LEACH, can be implemented and assessed, focusing on energy efficiency, end-to-end delay, packet delivery ratio, and network lifetime. Simulations can encompass different scenarios, such as varying node densities, mobility patterns, and traffic conditions, to evaluate scalability and adaptability. NS3 simulations provide valuable insights into the strengths and weaknesses of WBAN routing protocols, facilitating the identification of suitable protocols for specific healthcare applications and the proposal of enhancements. By optimizing parameters like cluster formation, data aggregation, and routing decisions, overall WBAN performance can be improved. In summary, NS3-based simulations serve as a valuable platform for evaluating, analyzing, and enhancing routing protocols in WBANs, contributing to more reliable and energy-efficient solutions in healthcare.

Table 1. Simulation Settings

Simulation Setting	Value
Channel Model	Rayleigh Fading
Communication Range	20 meters
Data Aggregation	Enabled or Disabled
Energy Model	Battery model
Interference Model	Path Loss Model
MAC Protocol	IEEE 802.15.4
Mobility Model	Random Waypoint
Network Area	200m x 200m
Number of Sensor Nodes	100
Sensor Node Placement	Random or Grid-based
Simulation Time	2000 seconds
Traffic Pattern	Varying traffic load
Transmission Power	10 dBm

5. RESULTS AND DISCUSSION

5.1. Packet Delivery and Drop Ratio Analysis

Figure 1 illustrates the packet delivery and drop ratio analysis for the CCARA, E-HARP, and BFO-

LRP routing protocols at different numbers of nodes in the network. The packet delivery ratio represents the percentage of successfully delivered packets, while the packet drop ratio indicates the percentage of packets dropped during transmission. Table 2a presents the three routing protocols' packet delivery analysis result values. The values in the table represent the packet delivery ratio in percentage (%). From Table 2a provides the results values of Figure 1.

The CCARA routing protocol exhibits a packet delivery ratio ranging from 32.27% to 54.92%, averaging 44.27%. CCARA emphasizes congestion control in its routing decisions. It employs adaptive routing strategies to avoid congested routes and ensure packet delivery. However, due to its conservative approach to congestion control, CCARA may reroute packets or delay their transmission, resulting in an average packet delivery ratio.

E-HARP shows a higher packet delivery ratio than CCARA, ranging from 43.22% to 59.87%, with an average of 51.62%. E-HARP integrates energy-efficient and harvest-aware mechanisms, which optimize energy consumption and leverage energy harvesting capabilities. E-HARP enhances packet delivery by optimizing resource utilization by dynamically selecting routes based on energy availability and considering energy-efficient transmission.

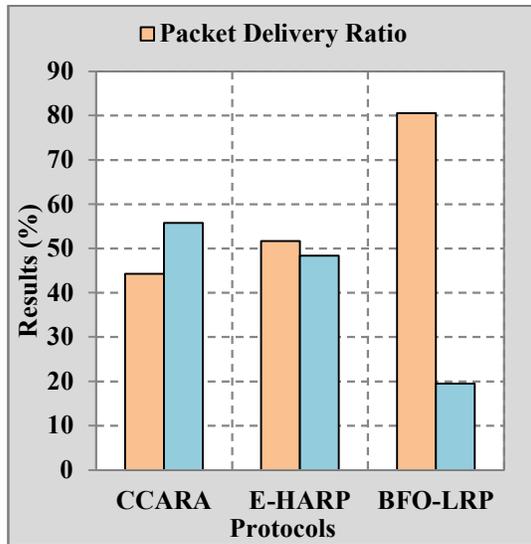


Figure 1. Packet Delivery and Drop Ratio

The BFO-LRP routing protocol demonstrates an even higher packet delivery ratio, ranging from

72.2% to 88.53%, with an average of 80.51%. BFO-LRP utilizes the Breakthrough Fruitfly Optimization (BFO) algorithm for routing decisions. This optimization algorithm imitates the search behaviour of fruit flies to find optimal routes. By leveraging this algorithm, BFO-LRP identifies efficient routes that minimize congestion and maximize packet delivery.

Table 2a. Result Values of Packet Delivery Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	54.92	59.87	88.53
20	52.9	57.68	86.17
30	50.59	54.89	85.32
40	49.91	54.26	83.8
50	48.23	53.37	81.4
60	43.02	51.51	79.21
70	38.64	49.53	77.29
80	37.12	46.88	76.82
90	35.1	45.02	74.31
100	32.27	43.22	72.2
Average	44.27	51.62	80.51

Table 2b provides the three routing protocols' packet drop ratio analysis result values. The values in the table represent the packet drop ratio in percentage (%). From Table 2b, this research can observe the following trends:

Table 2b. Result Values of Packet Drop Ratio Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	45.08	40.13	11.47
20	47.1	42.32	13.83
30	49.41	45.11	14.68
40	50.09	45.74	16.2
50	51.77	46.63	18.6
60	56.98	48.49	20.79
70	61.36	50.47	22.71
80	62.88	53.12	23.18
90	64.9	54.98	25.69
100	67.73	56.78	27.8
Average	55.73	48.38	19.50

The packet drop ratio for CCARA ranges from 45.08% to 67.73%, with an average of 55.73%. CCARA's congestion control mechanisms aim to mitigate network congestion by dropping packets when necessary. The higher packet drop ratio can be attributed to its conservative approach to congestion

control, where packets are selectively dropped to maintain network stability.

E-HARP exhibits a lower packet drop ratio than CCARA, ranging from 40.13% to 56.78%, with an average of 48.38%. E-HARP's energy-efficient mechanisms optimize resource utilization, reducing packet drop ratio. E-HARP minimizes packet drops and improves overall network performance by intelligently managing energy resources and considering energy-aware routing decisions.

The BFO-LRP routing protocol demonstrates the lowest packet drop ratio among the three protocols, ranging from 11.47% to 27.8%, with an average of 19.50%. BFO-LRP's routing decisions based on the Breakthrough Fruitfly Optimization (BFO) algorithm enable efficient route selection, minimizing congestion and packet drops. The optimized routing decisions contribute to the lower packet drop ratio observed in BFO-LRP.

The packet delivery and drop ratio analysis highlights the differences in performance among the routing protocols. CCARA emphasizes congestion control, resulting in moderate packet delivery but a higher packet drop ratio. E-HARP's energy-efficient mechanisms enhance packet delivery and reduce packet drops. BFO-LRP, leveraging the BFO algorithm, achieves a high packet delivery ratio and the lowest packet drop ratio by optimizing route selection and minimizing congestion.

5.2. Throughput Analysis

Figure 2 presents the throughput analysis for the CCARA, E-HARP, and BFO-LRP routing protocols at different numbers of nodes in the network. Throughput refers to the rate at which data has been transmitted through the network. Table 3 provides the result values of the throughput analysis for the three routing protocols. The values in the table represent the throughput in an unspecified unit. From Table 3, this research can observe the following trends:

The CCARA routing protocol has the lowest throughput values among the three protocols. The average throughput for CCARA has been 35.478 (unit not specified). As the number of nodes in the network has increased, CCARA has achieved lower throughput values. CCARA's focus on congestion control and stability in routing decisions has limited the overall data transmission rate. The emphasis on congestion avoidance and potential overhead associated with congestion control mechanisms may

have reduced the throughput compared to the other protocols.

E-HARP has shown higher throughput values compared to CCARA. The average throughput for E-HARP has been 47.989 (unit not specified). As the number of nodes has increased, E-HARP has continued to achieve higher throughput. E-HARP has integrated energy-efficient and harvest-aware mechanisms that have optimized energy consumption and leveraged energy harvesting capabilities. E-HARP has improved the data transmission capacity by efficiently managing energy resources, selecting optimal routes, and achieving higher throughput values.

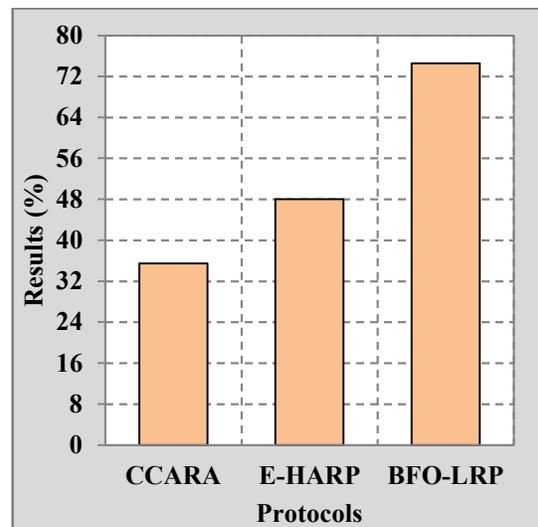


Figure 2. Throughput

The BFO-LRP routing protocol has demonstrated the highest throughput values among the three protocols. The average throughput for BFO-LRP has been 74.526 (unit not specified). As the number of nodes has increased, BFO-LRP has consistently achieved higher throughput. BFO-LRP has utilized the Breakthrough Fruitfly Optimization (BFO) algorithm for routing decisions, which has optimized the selection of routes to minimize congestion and maximize network performance. By effectively utilizing network resources and selecting efficient routes, BFO-LRP has achieved the highest throughput compared to CCARA and E-HARP.

Table 3. Result Values of Throughput Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	31.003	43.771	70.337
20	31.443	44.31	71.467
30	32.181	45.861	72.446
40	33.303	46.141	73.189
50	33.971	47.602	73.343
60	34.676	48.414	74.108
70	38.645	50.184	74.348
80	39.242	50.282	77.274
90	39.852	51.316	78.39
100	40.462	52.013	80.356
Average	35.478	47.989	74.526

The throughput analysis indicated that BFO-LRP had achieved the highest throughput, followed by E-HARP, while CCARA exhibited the lowest. The variations in throughput can be attributed to the underlying mechanisms of the routing protocols. CCARA's focus on congestion control and stability has limited the data transmission rate, resulting in lower throughput. E-HARP's integration of energy-efficient mechanisms has optimized resource utilization and enhanced data transmission capacity, leading to improved throughput. BFO-LRP's utilization of the BFO algorithm has facilitated the selection of optimal routes, minimizing congestion and maximizing network performance, resulting in the highest throughput among the three protocols.

5.3. Delay Analysis

Figure 3 presents the analysis of delays observed in the CCARA, E-HARP, and BFO-LRP routing protocols across different numbers of nodes in the network. The delay analysis measures the time a packet travels from the source to the destination, measured in milliseconds (ms). Table 4 provides the detailed result values obtained from the delay analysis for each routing protocol, representing the delays in ms. Upon examining the results, the following observations can be made:

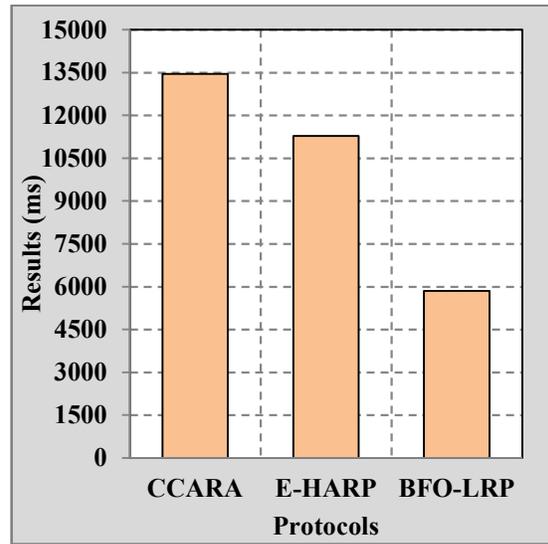


Figure 3. Delay

The CCARA routing protocol demonstrates the highest delay values among the three protocols. The average delay for CCARA is recorded as 13,453.4 ms, ranging from 13,024 ms to 14,109 ms. As the number of nodes in the network increases, CCARA experiences an upward trend in delays. This can be attributed to the congestion control mechanisms implemented by CCARA, which introduce additional processing and queuing delays to avoid network congestion and reliable packet delivery.

E-HARP exhibits relatively lower delay values compared to CCARA. The average delay for E-HARP is 11,272.2 ms, ranging from 10,472 ms to 12,961 ms. As the number of nodes in the network increases, E-HARP experiences a gradual delay increase. However, E-HARP incorporates energy-efficient and harvest-aware mechanisms that optimize energy consumption and leverage energy harvesting capabilities. These mechanisms facilitate improved resource utilization and efficient route selection, minimizing packet transmission delays.

BFO-LRP stands out with the lowest delay values among the three routing protocols. The average delay for BFO-LRP is 5,853.7 ms, ranging from 3,988 ms to 7,721 ms. As the number of nodes in the network increases, BFO-LRP exhibits a slight delay increase. BFO-LRP utilizes the Breakthrough Fruitfly Optimization (BFO) algorithm for routing decisions, enabling the selection of optimized routes that minimize congestion and reduce delays. The efficient routing decisions made by BFO-LRP contribute to the lower delay values observed in this protocol.

Table 4. Result Values of Delay Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	13024	10472	3988
20	13059	10535	4003
30	13081	10853	4464
40	13118	10910	4570
50	13336	11061	5117
60	13551	11121	6454
70	13659	11244	7041
80	13775	11648	7562
90	13822	11917	7617
100	14109	12961	7721
Average	13453.4	11272.2	5853.7

The delay analysis highlights that CCARA exhibits the highest delays, E-HARP shows relatively lower delays, and BFO-LRP achieves the lowest delay values. These differences in delays can be attributed to the specific mechanisms employed by each protocol. CCARA's congestion control mechanisms introduce additional delays for stable network performance. E-HARP's energy-efficient mechanisms contribute to the mitigation of delays. BFO-LRP's optimized route selection minimizes congestion, resulting in the lowest observed delays among the three protocols.

5.4. Energy Consumption Analysis

Figure 4 illustrates the analysis of energy consumption for the CCARA, E-HARP, and BFO-LRP routing protocols at different numbers of nodes in the network. The energy consumption analysis quantifies each routing protocol's energy during network operations. Table 5 presents the detailed result values obtained from the energy consumption analysis for each routing protocol. The energy consumption values are provided in unspecified units. Upon analyzing the results, the following observations can be made:

Among the three protocols, CCARA exhibits the highest energy consumption values. The average energy consumption for CCARA is 86.112 (unit not specified), ranging from 77.639 to 93.759. As the number of nodes in the network increases, CCARA demonstrates a consistent increase in energy consumption.

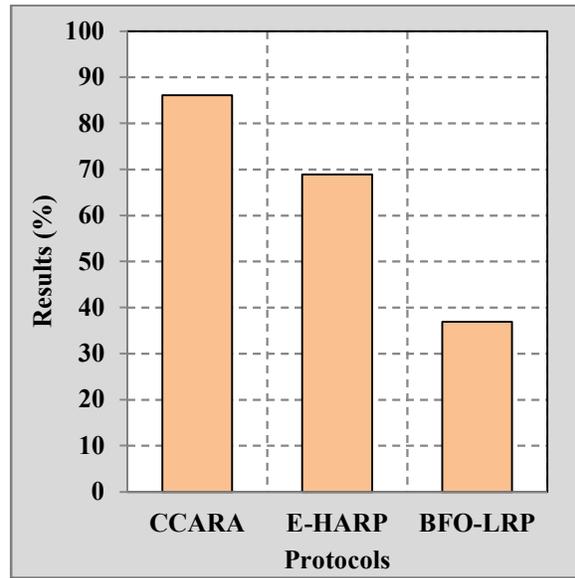


Figure 4. Energy Consumption

This can be attributed to CCARA's congestion control-aware routing algorithm, which may require additional energy for maintaining network stability and ensuring adequate data transmission.

E-HARP shows relatively lower energy consumption values compared to CCARA. The average energy consumption for E-HARP is recorded as 68.877 (unit not specified), ranging from 59.522 to 79.289. As the number of nodes increases, E-HARP gradually increases energy consumption. E-HARP incorporates energy-efficient and harvest-aware mechanisms in its routing protocol, optimizing energy usage and leveraging energy harvesting capabilities. These mechanisms contribute to more efficient energy utilization and lower energy consumption than CCARA.

BFO-LRP demonstrates the lowest energy consumption values among the three routing protocols. The average energy consumption for BFO-LRP is reported as 36.913 (unit not specified), ranging from 29.813 to 46.027. As the number of nodes increases, BFO-LRP gradually increases energy consumption. BFO-LRP utilizes the Breakthrough Fruitfly Optimization (BFO) algorithm-based routing protocol, which optimizes route selection and minimizes congestion. This leads to more energy-efficient operation and lower overall energy consumption than CCARA and E-HARP.

Table 5. Result Values of Energy Consumption Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	77.639	59.522	29.813
20	78.748	61.074	31.034
30	81.033	63.491	34.623
40	84.345	64.113	34.691
50	85.405	64.645	34.726
60	87.601	72.034	36.791
70	89.765	72.639	37.929
80	90.936	74.869	41.394
90	91.886	77.094	42.103
100	93.759	79.289	46.027
Average	86.112	68.877	36.913

The energy consumption analysis highlights that CCARA exhibits the highest energy consumption, E-HARP demonstrates relatively lower energy consumption, and BFO-LRP achieves the lowest energy consumption among the three routing protocols. These disparities in energy consumption can be attributed to the specific mechanisms and algorithms employed by each protocol. CCARA's congestion control-aware routing algorithm requires more energy to maintain network stability. E-HARP's energy-efficient and harvest-aware mechanisms optimize energy utilization, reducing energy consumption. BFO-LRP's utilization of the BFO algorithm minimizes congestion and leads to the lowest energy consumption among the three protocols.

5.5. Network Lifetime Analysis

Figure 5 depicts the network lifetime analysis, which measures the duration a network can operate efficiently without exhausting its energy resources. The analysis considers different numbers of nodes in the network. Table 6 presents the network lifetime analysis result values for each routing protocol, given in unspecified units. From the table, this research can derive the following observations.

The CCARA routing protocol exhibits the shortest network lifetime among the three protocols. The average network lifetime for CCARA is 13.638 (unit not specified), ranging from 6.954 to 21.255. As the number of nodes increases, CCARA experiences a decrease in network lifetime. This can be attributed to CCARA's higher energy consumption and potentially limited energy-saving mechanisms, resulting in a shorter overall network lifetime.

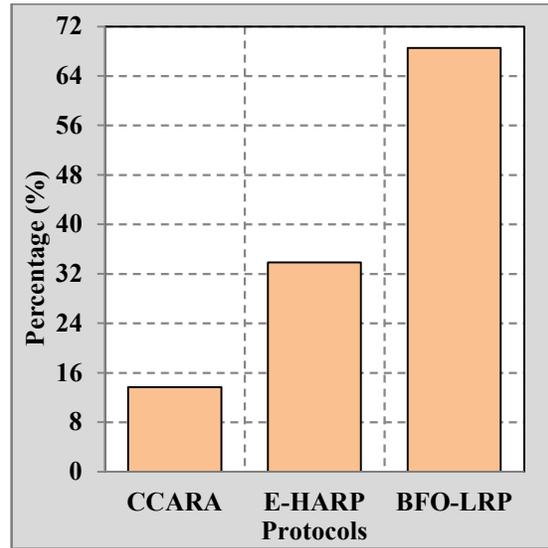


Figure 5. Network Lifetime

E-HARP demonstrates a relatively longer network lifetime compared to CCARA. The average network lifetime for E-HARP is 33.863 (unit not specified), ranging from 24.235 to 46.976. E-HARP maintains a relatively consistent network lifetime as the number of nodes increases. E-HARP incorporates energy-efficient and harvest-aware mechanisms that optimize energy consumption and leverage energy harvesting capabilities. These mechanisms contribute to an extended network lifetime compared to CCARA.

BFO-LRP showcases the most extended network lifetime among the three routing protocols. The average network lifetime for BFO-LRP is 68.530 (unit not specified), ranging from 63.138 to 76.157. As the number of nodes increases, BFO-LRP demonstrates a relatively stable network lifetime. BFO-LRP utilizes the Breakthrough Fruitfly Optimization (BFO) algorithm-based routing protocol, which optimizes route selection and minimizes energy consumption. These optimizations contribute to the most extended network lifetime observed in BFO-LRP.

The network lifetime analysis highlights that CCARA exhibits the shortest network lifetime, E-HARP demonstrates a relatively longer network lifetime, and BFO-LRP achieves the most extended network lifetime among the three routing protocols. These variations can be attributed to each protocol's specific mechanisms and algorithms. CCARA's higher energy consumption and potentially limited energy-saving mechanisms result in a shorter

network lifetime. E-HARP's energy-efficient and harvest-aware mechanisms contribute to an extended network lifetime. BFO-LRP's utilization of the BFO algorithm optimizes route selection and minimizes energy consumption, leading to the most extended network lifetime among the three protocols.

Table 6. Result Values of Energy Consumption Analysis

Nodes	CCARA	E-HARP	BFO-LRP
10	21.255	46.976	76.157
20	20.408	46.133	75.806
30	17.583	42.685	70.777
40	15.156	41.004	70.207
50	14.078	33.941	69.691
60	13.18	27.591	66.923
70	9.881	26.17	65.985
80	9.223	25.584	63.442
90	8.666	24.308	63.175
100	6.954	24.235	63.138
Average	13.638	33.863	68.530

6. CONCLUSION

The proposed Breakthrough Fruitfly Optimization-based LEACH Routing Protocol (BFO-LRP) presents a novel and effective solution for addressing routing challenges in Wireless Body Area Networks (WBANs). By leveraging the fruitfly-inspired optimization algorithm, BFO, and incorporating Nonlinear Hierarchical Decision-Making (NHDM), BFO-LRP achieves coordinated and optimized routing decisions, enhancing the performance of the LEACH protocol. The hierarchical decision-making approach enables adaptive search space exploration, leading to improved convergence and efficient exploitation of promising regions. Through extensive simulations using the ns3 network simulator, BFO-LRP's effectiveness regarding packet delivery performance, reduced delays, and efficient energy utilization is demonstrated. The results highlight the superiority of BFO-LRP over conventional routing protocols, making it a promising and valuable solution for optimizing routing in WBANs. BFO-LRP's implementation advances WBAN technology, offering enhanced support for healthcare and wearable applications, thereby contributing to developing efficient and reliable healthcare monitoring systems.

REFERENCES:

- [1] J. Sarma and R. Biswas, "A power-aware ECG processing node for real-time feature extraction in WBAN," *Microprocess. Microsyst.*, vol. 96, p. 104724, 2023, doi: 10.1016/j.micpro.2022.104724.
- [2] S. Savaşçı Şen, M. Cicioğlu, and A. Çalhan, "IoT-based GPS assisted surveillance system with inter-WBAN geographic routing for pandemic situations," *J. Biomed. Inform.*, vol. 116, p. 103731, 2021, doi: 10.1016/j.jbi.2021.103731.
- [3] N. Sharma, B. M. Singh, and K. Singh, "Analysis of packet transmission protocols for multimedia applications of WBANs in cognitive behavioral therapy," *Multimed. Tools Appl.*, vol. 82, no. 25, pp. 39765–39781, 2023, doi: 10.1007/s11042-022-13512-9.
- [4] N. Ahmad *et al.*, "Improved QoS Aware Routing Protocol (IM-QRP) for WBAN Based Healthcare Monitoring System," *IEEE Access*, vol. 10, pp. 121864–121885, 2022, doi: 10.1109/ACCESS.2022.3223085.
- [5] A. Choudhary, M. Nizamuddin, and M. Zadoo, "Body Node Coordinator Placement Algorithm for WBAN Using Multi-Objective Swarm Optimization," *IEEE Sens. J.*, vol. 22, no. 3, pp. 2858–2867, 2022, doi: 10.1109/JSEN.2021.3135269.
- [6] W. Guo, Y. Wang, Y. Gan, and T. Lu, "Energy efficient and reliable routing in wireless body area networks based on reinforcement learning and fuzzy logic," *Wirel. Networks*, vol. 28, no. 6, pp. 2669–2693, Aug. 2022, doi: 10.1007/s11276-022-02997-9.
- [7] S. Gherairi, "Healthcare: A priority-based energy harvesting scheme for managing sensor nodes in WBANs," *Ad Hoc Networks*, vol. 133, p. 102876, 2022, doi: 10.1016/j.adhoc.2022.102876.
- [8] H. M. Khater, F. Sallabi, M. A. Serhani, S. Turaev, and E. Barka, "Efficient Hybrid Fault-Management Clustering Algorithm (HFMC) in WBANs Based on Weighted Bipartite Graph," *IEEE Access*, vol. 11, pp. 57977–57990, 2023, doi: 10.1109/ACCESS.2023.3283929.
- [9] M. Roy, D. Biswas, N. Aslam, and C. Chowdhury, "Reinforcement learning based effective communication strategies for energy harvested WBAN," *Ad Hoc Networks*, vol. 132, p. 102880, 2022, doi: 10.1016/j.adhoc.2022.102880.
- [10] S. J. Hussain, M. Irfan, N. Z. Jhanjhi, K.

- Hussain, and M. Humayun, "Performance Enhancement in Wireless Body Area Networks with Secure Communication," *Wirel. Pers. Commun.*, vol. 116, no. 1, pp. 1–22, Jan. 2021, doi: 10.1007/s11277-020-07702-7.
- [11] A. Roshini and K. V.D. Kiran, "Hierarchical energy efficient secure routing protocol for optimal route selection in wireless body area networks," *Int. J. Intell. Networks*, vol. 4, pp. 19–28, 2023, doi: 10.1016/j.ijin.2022.11.006.
- [12] M. Cicioğlu and A. Çalhan, "HUBsFLOW: A novel interface protocol for SDN-enabled WBANs," *Comput. Networks*, vol. 160, pp. 105–117, 2019, doi: 10.1016/j.comnet.2019.06.007.
- [13] R. Kaur, B. P. Kaur, R. P. Singla, and J. Kaur, "AMERP: Adam moment estimation optimized mobility supported energy efficient routing protocol for wireless body area networks," *Sustain. Comput. Informatics Syst.*, vol. 31, p. 100560, 2021, doi: 10.1016/j.suscom.2021.100560.
- [14] R. Jaganathan and R. Vadivel, "Intelligent Fish Swarm Inspired Protocol (IFSIP) for Dynamic Ideal Routing in Cognitive Radio Ad-Hoc Networks," *Int. J. Comput. Digit. Syst.*, vol. 10, no. 1, pp. 1063–1074, 2021, doi: 10.12785/ijcds/100196.
- [15] P. Menakadevi and J. Ramkumar, "Robust Optimization Based Extreme Learning Machine for Sentiment Analysis in Big Data," in *2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICACTA54488.2022.9753203.
- [16] J. Ramkumar, R. Vadivel, and B. Narasimhan, "Constrained Cuckoo Search Optimization Based Protocol for Routing in Cloud Network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 6, pp. 795–803, 2021, doi: 10.22247/ijcna/2021/210727.
- [17] S. P. Geetha, N. M. S. Sundari, J. Ramkumar, and R. Karthikeyan, "ENERGY EFFICIENT ROUTING IN QUANTUM FLYING AD HOC NETWORK (Q-FANET) USING MAMDANI FUZZY INFERENCE ENHANCED DIJKSTRA'S ALGORITHM (MFI-EDA)," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 9, pp. 3708–3724, 2024, [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85197297302&partnerID=40&md5=72d51668bee6239f09a59d2694df67d6>
- [18] M. P. Swapna, J. Ramkumar, and R. Karthikeyan, "Energy-Aware Reliable Routing with Blockchain Security for Heterogeneous Wireless Sensor Networks," in *Lecture Notes in Networks and Systems*, V. Goar, M. Kuri, R. Kumar, and T. Senjyu, Eds., Springer Science and Business Media Deutschland GmbH, 2025, pp. 713–723. doi: 10.1007/978-981-97-6106-7_43.
- [19] J. Ramkumar, K. S. Jeen Marseline, and D. R. Medhunhashini, "Relentless Firefly Optimization-Based Routing Protocol (RFORP) for Securing Fintech Data in IoT-Based Ad-Hoc Networks," *Int. J. Comput. Networks Appl.*, vol. 10, no. 4, pp. 668–687, 2023, doi: 10.22247/ijcna/2023/223319.
- [20] R. Jaganathan, S. Mehta, and R. Krishan, "Preface," *Intell. Decis. Mak. Through Bio-Inspired Optim.*, pp. xiii–xvi, 2024, [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85192858710&partnerID=40&md5=f8f1079e8772bd424d2cdd979e5f2710>
- [21] K. S. J. Marseline, J. Ramkumar, and D. R. Medhunhashini, "Sophisticated Kalman Filtering-Based Neural Network for Analyzing Sentiments in Online Courses," in *Smart Innovation, Systems and Technologies*, A. K. Somani, A. Mundra, R. K. Gupta, S. Bhattacharya, and A. P. Mazumdar, Eds., Springer Science and Business Media Deutschland GmbH, 2024, pp. 345–358. doi: 10.1007/978-981-97-3690-4_26.
- [22] A. Senthilkumar, J. Ramkumar, M. Lingaraj, D. Jayaraj, and B. Sureshkumar, "Minimizing Energy Consumption in Vehicular Sensor Networks Using Relentless Particle Swarm Optimization Routing," *Int. J. Comput. Networks Appl.*, vol. 10, no. 2, pp. 217–230, 2023, doi: 10.22247/ijcna/2023/220737.
- [23] N. K. Ojha, A. Pandita, and J. Ramkumar, "Cyber security challenges and dark side of AI: Review and current status," in *Demystifying the Dark Side of AI in Business*, IGI Global, 2024, pp. 117–137. doi: 10.4018/979-8-3693-0724-3.ch007.
- [24] J. Ramkumar and R. Vadivel, "Whale optimization routing protocol for minimizing energy consumption in cognitive radio wireless sensor network," *Int. J. Comput. Networks Appl.*, vol. 8, no. 4, pp. 455–464, 2021, doi: 10.22247/ijcna/2021/209711.
- [25] M. P. Swapna and J. Ramkumar, "Multiple Memory Image Instances Stratagem to Detect Fileless Malware," in *Communications in*

- Computer and Information Science*, S. Rajagopal, K. Popat, D. Meva, and S. Bajeja, Eds., Springer Science and Business Media Deutschland GmbH, 2024, pp. 131–140. doi: 10.1007/978-3-031-59100-6_11.
- [26] J. Ramkumar, A. Senthilkumar, M. Lingaraj, R. Karthikeyan, and L. Santhi, “OPTIMAL APPROACH FOR MINIMIZING DELAYS IN IOT-BASED QUANTUM WIRELESS SENSOR NETWORKS USING NM-LEACH ROUTING PROTOCOL,” *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 3, pp. 1099–1111, 2024, [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85185481011&partnerID=40&md5=bf0ff974ceabc0ad58e589b28797c684>
- [27] L. Mani, S. Arumugam, and R. Jaganathan, “Performance Enhancement of Wireless Sensor Network Using Feisty Particle Swarm Optimization Protocol,” in *ACM International Conference Proceeding Series*, Association for Computing Machinery, 2022. doi: 10.1145/3590837.3590907.
- [28] D. Jayaraj, J. Ramkumar, M. Lingaraj, and B. Sureshkumar, “AFSORP: Adaptive Fish Swarm Optimization-Based Routing Protocol for Mobility Enabled Wireless Sensor Network,” *Int. J. Comput. Networks Appl.*, vol. 10, no. 1, pp. 119–129, 2023, doi: 10.22247/ijcna/2023/218516.
- [29] M. Lingaraj, T. N. Sugumar, C. S. Felix, and J. Ramkumar, “Query aware routing protocol for mobility enabled wireless sensor network,” *Int. J. Comput. Networks Appl.*, vol. 8, no. 3, pp. 258–267, 2021, doi: 10.22247/ijcna/2021/209192.
- [30] J. Ramkumar and R. Vadivel, “Multi-Adaptive Routing Protocol for Internet of Things based Ad-hoc Networks,” *Wirel. Pers. Commun.*, vol. 120, no. 2, pp. 887–909, 2021, doi: 10.1007/s11277-021-08495-z.
- [31] J. Ramkumar and R. Vadivel, “Improved Wolf prey inspired protocol for routing in cognitive radio Ad Hoc networks,” *Int. J. Comput. Networks Appl.*, vol. 7, no. 5, pp. 126–136, 2020, doi: 10.22247/ijcna/2020/202977.
- [32] R. Jaganathan, S. Mehta, and R. Krishan, *Intelligent Decision Making Through Bio-Inspired Optimization*. IGI Global, 2024. doi: 10.4018/979-8-3693-2073-0.
- [33] R. Jaganathan, S. Mehta, and R. Krishan, “Preface,” *Bio-Inspired Intell. Smart Decis.*, pp. xix–xx, 2024, [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85195725049&partnerID=40&md5=7a2aa7adc005662eebc12ef82e3bd19f>
- [34] J. Ramkumar and R. Vadivel, “CSIP—cuckoo search inspired protocol for routing in cognitive radio ad hoc networks,” in *Advances in Intelligent Systems and Computing*, D. P. Mohapatra and H. S. Behera, Eds., Springer Verlag, 2017, pp. 145–153. doi: 10.1007/978-981-10-3874-7_14.
- [35] R. Jaganathan and V. Ramasamy, “Performance modeling of bio-inspired routing protocols in Cognitive Radio Ad Hoc Network to reduce end-to-end delay,” *Int. J. Intell. Eng. Syst.*, vol. 12, no. 1, pp. 221–231, 2019, doi: 10.22266/IJIES2019.0228.22.
- [36] R. Jaganathan, S. Mehta, and R. Krishan, *Bio-Inspired intelligence for smart decision-making*. IGI Global, 2024. doi: 10.4018/9798369352762.
- [37] R. Vadivel and J. Ramkumar, “QoS-enabled improved cuckoo search-inspired protocol (ICSIP) for IoT-based healthcare applications,” in *Incorporating the Internet of Things in Healthcare Applications and Wearable Devices*, IGI Global, 2019, pp. 109–121. doi: 10.4018/978-1-7998-1090-2.ch006.
- [38] J. Ramkumar, C. Kumuthini, B. Narasimhan, and S. Boopalan, “Energy Consumption Minimization in Cognitive Radio Mobile Ad-Hoc Networks using Enriched Ad-hoc On-demand Distance Vector Protocol,” in *2022 International Conference on Advanced Computing Technologies and Applications, ICACTA 2022*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/ICACTA54488.2022.9752899.
- [39] R. Karthikeyan and R. Vadivel, “Proficient Dazzling Crow Optimization Routing Protocol (PDCORP) for Effective Energy Administration in Wireless Sensor Networks,” in *IEEE International Conference on Electrical, Electronics, Communication and Computers, ELEXCOM 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/ELEXCOM58812.2023.10370559.
- [40] R. Karthikeyan and R. Vadivel, “Boosted Mutated Corona Virus Optimization Routing Protocol (BMCVORP) for Reliable Data Transmission with Efficient Energy Utilization,” *Wirel. Pers. Commun.*, vol. 135, no. 4, pp. 2281–2301, 2024, doi: 10.1007/s11277-024-11155-7.
- [41] J. Ramkumar, R. Karthikeyan, and M. Lingaraj, “Optimizing IoT-Based Quantum Wireless

- Sensor Networks Using NM-TEEN Fusion of Energy Efficiency and Systematic Governance,” in *Lecture Notes in Electrical Engineering*, V. Shrivastava, J. C. Bansal, and B. K. Panigrahi, Eds., Springer Science and Business Media Deutschland GmbH, 2025, pp. 141–153. doi: 10.1007/978-981-97-6710-6_12.
- [42] J. Ramkumar, R. Karthikeyan, and V. Valarmathi, “Alpine Swift Routing Protocol (ASRP) for Strategic Adaptive Connectivity Enhancement and Boosted Quality of Service in Drone Ad Hoc Network (DANET),” *Int. J. Comput. Networks Appl.*, vol. 11, no. 5, pp. 726–748, 2024, doi: 10.22247/ijcna/2024/45.
- [43] B. L. Sujaya and S. B. BhanuPrashanth, “An efficient hardware-based human body communication transceiver architecture for WBAN applications,” *Glob. Transitions Proc.*, vol. 2, no. 2, pp. 152–156, 2021, doi: 10.1016/j.gltip.2021.08.070.
- [44] M. A. Raayatpanah, A. A. Abyaneh, J. Elias, and A. Trotta, “Optimal reliable design of energy-efficient Wireless Body Area Networks,” *Internet of Things (Netherlands)*, vol. 22, p. 100727, 2023, doi: 10.1016/j.iot.2023.100727.
- [45] W. Badreddine, C. Chaudet, F. Petruzzi, and M. Potop-Butucaru, “Broadcast strategies and performance evaluation of IEEE 802.15.4 in wireless body area networks WBAN,” *Ad Hoc Networks*, vol. 97, p. 102006, 2020, doi: 10.1016/j.adhoc.2019.102006.
- [46] A. Sammoud, M. A. Chalouf, O. Hamdi, N. Montavont, and A. Bouallegue, “A new biometrics-based key establishment protocol in WBAN: energy efficiency and security robustness analysis,” *Comput. Secur.*, vol. 96, p. 101838, 2020, doi: 10.1016/j.cose.2020.101838.
- [47] K. N. Qureshi, S. Din, G. Jeon, and F. Piccialli, “Link quality and energy utilization based preferable next hop selection routing for wireless body area networks,” *Comput. Commun.*, vol. 149, pp. 382–392, 2020, doi: 10.1016/j.comcom.2019.10.030.
- [48] V. Ramalingam, R. Saminathan, and K. M. Baalamurugan, “Fork-Hook encryption policy based secured Data Centric Routing Gateway for proactive trust ware data transmission in WBSN,” *Meas. Sensors*, vol. 27, p. 100760, 2023, doi: 10.1016/j.measen.2023.100760.
- [49] M. Miri, K. Mohamedpour, Y. Darmani, M. Sarkar, and R. L. Tummala, “An efficient resource allocation algorithm based on vertex coloring to mitigate interference among coexisting WBANs,” *Comput. Networks*, vol. 151, pp. 132–146, 2019, doi: 10.1016/j.comnet.2019.01.014.
- [50] G. Mwitende, Y. Ye, I. Ali, and F. Li, “Certificateless authenticated key agreement for blockchain-based WBANs,” *J. Syst. Archit.*, vol. 110, p. 101777, 2020, doi: 10.1016/j.sysarc.2020.101777.
- [51] A. Arfaoui, O. R. M. Boudia, A. Kribeche, S. M. Senouci, and M. Hamdi, “Context-aware access control and anonymous authentication in WBAN,” *Comput. Secur.*, vol. 88, p. 101496, 2020, doi: 10.1016/j.cose.2019.03.017.
- [52] J. Mu, X. Liu, and X. Yi, “Simplified Energy-Balanced Alternative-Aware Routing Algorithm for Wireless Body Area Networks,” *IEEE Access*, vol. 7, pp. 108295–108303, 2019, doi: 10.1109/ACCESS.2019.2925909.
- [53] T. Samal and M. R. Kabat, “A prioritized traffic scheduling with load balancing in wireless body area networks,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 8, pp. 5448–5455, 2022, doi: 10.1016/j.jksuci.2020.12.023.
- [54] K. N. Qureshi, M. Q. Tayyab, S. U. Rehman, and G. Jeon, “An interference aware energy efficient data transmission approach for smart cities healthcare systems,” *Sustain. Cities Soc.*, vol. 62, p. 102392, 2020, doi: 10.1016/j.scs.2020.102392.
- [55] O. Ahmed, F. Ren, A. Hawbani, and Y. Al-Sharabi, “Energy Optimized Congestion Control-Based Temperature Aware Routing Algorithm for Software Defined Wireless Body Area Networks,” *IEEE Access*, vol. 8, pp. 41085–41099, 2020, doi: 10.1109/ACCESS.2020.2976819.
- [56] Z. Ullah *et al.*, “Energy-efficient harvested-aware clustering and cooperative routing protocol for WBAN (E-HARP),” *IEEE Access*, vol. 7, pp. 100036–100050, 2019, doi: 10.1109/ACCESS.2019.2930652.