Received: July 10, 2021. Revised: September 4, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Fuzzy Logic and Modified Butterfly Optimization with Efficient Fault Detection and Recovery Mechanisms for Secured Fault-Tolerant Routing in Wireless Sensor Networks

Hasib Daowd Esmail Al-ariki[1]* **Mohammed Hamdi[2]**

[1]Department of Computer Networks and Distributed Systems,
Al Saeed faculty for Engineering and Information Technology,
Taiz University, Taiz, Yemen
[2]Department of Computer Science, College of Computer Science and Information Systems,
Najran University, Saudi Arabia
* Corresponding author's Email: hasibalariki@gmail.com

Abstract: Energy efficiency and security are the most vital requirements in Wireless Sensor Networks (WSN). Therefore, modern routing protocols have been forced to consider these issues as the primary challenges. This study proposes the develop meat of an Energy Efficient Secured Fault-Tolerant Routing (EESFTR) protocol that improves routing efficiency and also increasing network reliability. The proposed EESFTR protocol consists of cluster head (CH) selection, optimal route selection, fault detection and recovery process. First, the clustered network utilizes Fuzzy logic to select optimal energy surplus CH. The routes are then discovered and the optimal routes are selected based on energy, link quality and network lifetime designing a Modified Butterfly Optimization Algorithm (MBOA). To ensure security and fault-tolerance, the efficient Elliptical Curve Digital Signature Algorithm is enhanced through unknown parameter signature verification. Finally, the faults which are detected to be malicious to the network are either repaired or eliminated based on a disruptive process. The proposed EESFTR protocol has been evaluated and compared with the existing routing protocols to highlight its efficiency. The results showed that the proposed EESFTR protocol has better a routing performance with 13% less time consumption, 8.3% less energy consumption, 20% increased lifetime, and 12% reduced packet drops to ensure higher fault tolerance and security when compared to the performance of the efficient existing routing protocols.

Keywords: Wireless sensor networks, Energy efficient secured fault-tolerant routing, Fault tolerance, Fuzzy logic, Modified butterfly optimization algorithm, Enhanced elliptical curve digital signature algorithm.

1. Introduction

Wireless Sensor Networks (WSNs) have seen greater advancements in recent years owing to the manufacturing of multi-functional sensors and low-power circuits with digital processing and better communication entities. These developments have created a situation where WSNs are primarily deployed in large scale physical environments for fine-grain monitoring in various applications. However, the limited characteristics of the WSNs pose greater challenges [1]. Battery constraint WSNs often faces the problem of power constraints in addition to energy hotspot problems near their base stations. Apart from power constraints, the nodes and links of WSN are prone to failures due to their deployment in tough environments. Such failures in the links and nodes will result in loss of data packets and their subsequent retransmissions, which in turn tend to increase both power and time consumption. Further, these failures also negatively impact the data delivery ratio, accuracy and overall reliability of the network. The current network equipment also gets affected eventually through the degraded reliability and stability of the network, thus making it unsuitable for transmission [2].

DOI: 10.22266/ijies2021.1231.36

Reliability and security in data transmission are two major factors that are commonly monitored to evaluate WSN performance [3]. Hence, the degradation of such metrics due to faults becomes a very serious issue. Stable network topology can ensure reliable data transmission and it depends on the following three factors namely, energy consumption, delay and link quality. Therefore, routing protocols must be designed with a stable structure so that they minimise both energy consumption and time consumption during data transmissions [4]. One of the prominent strategies for reducing energy consumption is clustering in which the network is divided into a cluster of nodes and a CH acts as the primary point of transmission. This process considerably minimises the energy consumed by sensor nodes but at the cost of higher energy consumption and subsequent early death of the CH. Hence, it becomes essential to select these nodes with high residual energy as that of the CH. Further, the nodes of CH should also be switched to balance energy exploitation. Secondly, fault detection and recovery process are mainly performed through external devices and even the in-network strategies are dependent on base stations, thus increasing the cost of implementing the system. Finally, an effective and secured routing protocol must be attained and the authentication schemes are quite vulnerable to intruder attacks [5]. Therefore, a reliable and stable routing algorithm must be energy efficient, secure and fault-tolerant.

Many studies [6-17] Rare failed to consider the authentication schemes and fault recovery strategies due to the fear of increased cost. To address this issue, this study aims at designing and developing an energy-efficient, secured fault-tolerant routing (EESFTR) protocol that improves the applications of WSN through low consumption of energy, reduced delay and higher reliability. The proposed EESFTR protocol has been built using fuzzy logic and MBOA for CH selection and route selection, respectively. Additionally, the Enhanced Elliptical Curve Digital Signature Algorithm has been used for secured fault detection. The faults are recovered through Disruptive fault repair and elimination. The experimental results have been compared with state-of-the-art methods to estimate the efficiency of the proposed EESFTR. The article is organized as follows: Related works in section 2, System model in section 3, explanation of the proposed methodology in section 4, experimental results in section 5 and conclusions in section 6.

2. Related works

Cluster-based multichannel routing protocols in WSNs can be designed based on different initial energy levels. Many studies developed algorithms for fault-tolerant routing and energy-aware routing in clustered WSNs. Low energy adaptive clustering hierarchy (LEACH) [6] is one of the well-known, clustering-based routing models to attain dynamicity in routing. Yet, LEACH selects CH randomly without taking into account energy consumption. Further, CH is also not uniformly distributed. Min and Zaw [7] designed an improved model of LEACH, namely Energy Efficient Fault-tolerant LEACH (EF-LEACH) protocol, which overcomes LEACH by improving network connectivity when failures occur. However, this protocol has a limitation in that it does not have a method for automatically detecting and recovering the faults. Shelke et al. [8] developed a fuzzy-based, fault-tolerant, low-energy, adaptive clustering hierarchy-routing protocol (FTLEACH) considering both power and node density to enhance the routing performance of the LEACH algorithm. However, this approach also has limitations in clustering in that it is not uniform and it also fails to cover the entire network.

Apart from LEACH, Hybrid Energy-Efficient Distributed (HEED) routing [9] and Power-Efficient Gathering in Sensor Information Systems (PEGASIS) [10] were also extensively utilized. Zhou et al. [11] presented HEED clustering fault-tolerant (HEED-FT) routing that increases both reliability and the lifetime of the network. But, this approach lacks the automatic management of faults. Abba and Lee [12] proposed an autonomous self-aware and adaptive fault-tolerant routing technique (ASAART) to overcome the limitations of the self-healing routing and self-selective routing by the slotted priority technique. Although this efficient than algorithm was more traditional algorithms, it provides a less reliable cost table and lesser knowledge of the properties of the global node. Xu et al. [13] introduced Byzantine fault-tolerant routing (BFTR) using Fast Elliptic Curve Digital Signature Algorithm to counterattack both timing and energy attacks with reduced verification proportion. However, this approach suffers from overhead due to the complexities of digital signatures. Pairwise Directional Geographical Routing (PWDGR) [14] and its enhanced versions namely Energy Enhanced PWDGR (EE-PWDGR) [15], Energy Enhanced Load Balancing PWDGR (EELB-PWDGR) [16] and Enhanced Artificial Bee Colony (EABC) based EELB-PWDGR [17] were designed to provide reliable and energy-efficient routing. However, these algorithms considered only the energy and load balancing, while overlooking fault tolerance.

Azharuddin and Jana [18] designed Particle Swarm optimization-based fault-tolerant routing (PSO-FTR) that maximizes lifetime and reduces overload conditions. Yet again, the PSO-FTR failed to consider energy efficiency. Further, it fails to tackle gateway failures. Ye et al. [19] proposed a security fault-tolerant routing approach in which the fuzzy logic selects the best CH and ensures priority-trust values routing. However, this approach does not have an effective approach for fault recovery. Nor does it seem to offer a better solution for CH faults. Yue et al. [20] developed a swarm intelligence algorithm of artificial bee colony optimized particle swarm optimization algorithm (ABCPSO) for fault-tolerant alternate routing path selection. However, the message capacity of the network is significantly reduced in this approach due to the handling of the mobile sink. Khabiri and Ghaffari [21] designed an energy-aware cuckoo optimization algorithm based routing protocol (COARP) to minimize the energy and delay to improve the lifetime of the network. However, this model has limitations in terms of handling different types of failures in the network.

Haseeb et al. [22] presented Dynamic Energy-aware Fault-Tolerant Routing (DEFTR) protocol to perform network partitioning and multi-facet routing with improved lifetime and throughput and reduced delay and cost. Although efficient, this approach has limitations in terms of path maintenance. Lin et al. [23] developed the bipartite-flow graph model for fault-tolerant routing and achieved minimum energy consumption and increased lifespan. However, it considers only the loaded CH faults, while ignoring sink node failures. Maratha et al. [24] developed an improved Fault-Tolerant Optimal Route Reconstruction (IFTORR) approach for increasing the lifetime of WSN through optimized routing. However, this method considered only the route failures, while ignoring both node and link failures. Talmale et al. [25] developed an energy-aware Distributed Pre Fault Detection Routing Mechanism (DPFDRM) for WSN in which the Kuartz graph was used for detection, while the actuator nodes were used for path selection based on energy. However, this approach is complex and also degrades the overall QoS. Effah and Thiare [26] presented Realistic Cluster-Based Energy-Efficient and Fault-Tolerant (RCEEFT) Routing Protocol using effective spatial correlation and Mass Measurement. However, in large scale WSNs, this model lacks effective distant-hop communication.

The routing protocols discussed in the preceding section suffer from one or more of the following issues: re-clustering overhead, lack of in-network verification of attacks, inability to handle different types of faults, higher energy consumption, is the CH selection is not based on priorities. These problems further compounded by computation complexities that hinder effective data transmission, eventually degrades the quality of service. Considering these limitations, an efficient routing protocol named EESFTR has been developed in this study to ensure energy-efficient, stable and secured routing.

3. System model

The following sections present details about the EESFTR protocol with the following network and energy model along and their underlying assumptions. The AODV routing protocol is the fundamental routing process used in EESFTR to discover and maintain routes. Hence, the network and energy models have been designed along those lines. Table 1 shows the list of notations used in this paper.

3.1 Network model

The proposed EESFTR protocol is utilized in a network setup where the nodes are deployed randomly in a two-dimensional space. Once deployed, all nodes are fixed so that these nodes can neither be removed nor included in the network. It is assumed that all the nodes deployed have similar computing abilities, transmission, storage memory, initial energy and transmission range. The transmission and reception properties of a node are limited by utilizing wireless strategies, following which the interference range d_0 is equal to the nodes' transmission range. The transmission between two nodes will begin only when they are in the coincidence of both their transmission range. Each sensor node maintains a table to store all the details of the neighbour nodes such as their unique ID, exact location, transmission details, list of their neighbor nodes, node priorities, trust values and the list of malicious nodes. The nodes themselves assume that all the transmission links support both forward and backward data transmission (bidirectional) and that all the routing channels are secure. Since all nodes exhibit similar properties, each node is capable of acting as both the CH and the normal node. It is also noted that each node is capable of completing the proposed algorithm with sufficient power for computation.

3.2 Energy model

The energy consumption both in free space and in the multipath fading channels is computed based on

Table 1. Notations and definitions

Notation	Definition
d_0	interference range
t	time
k, k' and k''	number of packets transmitted, received and monitored
E_{tx}	transmission energy
E_{rx}	reception energy
E_m	monitoring energy
k	packets
d	distance between two nodes
E_{elec}	electronic circuit energy
E_{amp1} and E_{amp2}	amplifier energy in free space and multi-path
M	number of layers
$d(s,i)$	distance between a node *i* and the sink node *s*
M_i	i-th layer
$CH(i)$	number of CH of i-th layer
N_i	number of nodes in i-th layer
H	adjustable parameter
$RE(i)$	relative energy of a node
E_{max}	maximum residual energy of cluster
E_i	residual energy of i-th node
x_i and y_i	horizontal and vertical coordinates of i-th node
x_j and y_j	horizontal and vertical coordinates of the j-th node
C_{min}	ratio of the minimum centrality
$RC(i)$	relative centrality of i-th node
$\mu_{NP}(x)$	fuzzy membership function
$E_{res,i}(t)$	residual energy at time *t*
$E_{0,i}$	initial energy available at *i*
E_t	total energy consumption at time *t*
$T_{sk,CH}$	sending time of CH for k-packets
$T_{rk,CH}$	receiving time of CH for k-packets
$T_{sk,i}$	sending time of k packets
$T_{rk,sink}$	receiving time of k packets by the sink node
$\mathbb{E}[L]$	expected lifetime
P	constant continuous power depletion
ε_0	non-rechargeable initial energy
λ	average wavelength
$\mathbb{E}[E_w$	expected wasted energy
$\mathbb{E}[E_r]$	expected reporting energy
RSS_{iz}	received signal strength of a node *i* at a distance *z*
T_j	threshold
G_a	average gain
G_{tx}	transmitting gain
G_{rx}	receiving gain
$P_{t,max}$	maximum transmission power of the antenna
R	maximum range of the antenna
ρ	density of node deployment
w_1, w_2, w_3, w_4, w_5	weight values
F_i	objective function
g^*	best current position
b_i^q	current position of the i-th butterfly
b_i^{q+1}	next positions of i-th butterfly
q	current iteration
β_n	chaotic parameter
α	controlling parameter
S	Step size
b	present location of the butterfly
$f(b)$	fitness function value
Ep	elliptical points
$EC\ (a,b)$	elliptic curve
PG	point generator
HF	hash function
pn	order of PG
a_1, a_2	random integers

the radio model of energy consumption. This computation depends on the communication between the sender and receiver nodes. The total energy at time t is given as the sum of the transmission energy, reception energy and monitoring energy and is computing as follows:

$$E_t = E_{tx}(k,i) + E_{rx}(k',i) + E_m(k'',j) \quad (1)$$

Where k, k' and k'' refer to the number of packets transmitted, received and monitored, respectively by their nodes while transmitting data from node *i* to node *j* and E_{tx}, E_{rx} and E_m refer to the transmission energy, reception energy and monitoring energy respectively.

The transmission energy consumed to transmit k packets between two nodes at distance d will be computed as follows:

$$E_{tx} = \begin{cases} \left(E_{elec} + d^2 \times E_{amp1}\right) \times k & d < d_0 \\ \left(E_{elec} + d^4 \times E_{amp2}\right) \times k & d \geq d_0 \end{cases} \quad (2)$$

Here, the interference range d_0 is used as the threshold value to determine whether to initiate the free-space transmission or multi-path transmission. This equation is the determining factor so that when the distance d is lesser than the threshold d_0, free space model is used and in other cases, the multi-path model is used for energy computation. E_{elec} represents the electronic circuit energy while E_{amp1} and E_{amp2} refer to the amplifier energy in free space and multi-path, respectively. Likewise, the energy consumed by a radio node to receive the k packets is computed as follows:

$$E_{rx} = E_{elec} \times k' \tag{3}$$

3.3 Multi-layer model

To reliable and secure multi-layer clustered network topology, the multi-layer model has been proposed which manages the overall network energy and also enhances the network performance. The network is clustered into different layers based on the distance between the nodes and the sink node. When the interference range d_0 is chosen as the clustering metric, the network is divided into M layers. Let $d(s,i)$ be the distance between a node i and the sink node s. When clustering, the i-th node may belong to the M_i layer and it can be represented as follows:

$$M_i = 2 \times \frac{d(s,i)}{d_0} \tag{4}$$

The value of M can be obtained by maximizing M_i i.e. $M = max\, M_i\,, i = 1,2,3, \dots n$. Considering the real-time scenarios, the nodes around the sink tend to experience high pressure and end up losing energy much faster than the other node. Hence, the clusters can't be uniform. Therefore, the different cluster layers with different densities and quantities are presented. Automatically, the number of CHs in different layers becomes different and it is determined by the number of sensor nodes in the layer and the measurable distance between the layers to the sink node. The number of cluster heads of the i-th layer is represented as $CH(i)$, which is dependent on the number of layers and the number of nodes in the layers. It is computed as follows:

$$CH(i) = H \times \frac{N_i}{M_i} \tag{5}$$

Where N_i denotes the number of nodes in the i-th layer and H is the adjustable parameter.

4. Energy-efficient and fault-tolerant routing protocol

The proposed EESFTR protocol consists of three main processes: CH selection, optimal route selection and fault detection and recovery to ensuring secured fault-tolerant routing in WSN, without increasing the overheads or complexities. The CH selection is based on fuzzy logic using three objective parameters as input membership functions. The MBOA is then used for route selection based on the following parameters: residual energy, delay, lifetime, reliability and hop count. This model also employs efficient fault detection with in-network verification and fault recovery process.

4.1 Ch selection using fuzzy logic

CH selection has been performed based on fuzzy logic to improve the network lifetime. In a deployed sensor network, the fuzzy logic sets fuzzy variables and the fuzzy rules to calculate the node priority to be selected as the CH. In each layer, the CH can initiate transmission with the CH of the different layers and does not communicate with the other CH in its same layer to maintain isotropic data transmission. The CH must be selected considering many factors and hence, the fuzzy logic model uses the node physical factors such as node relative energy, relative density and relative centrality as the fuzzy input variables i.e. the objective parameters to select CH. With these three input variables, the node priority will be the fuzzy output variable. Based on this output variable, the CH will be selected. The main advantage in selecting these three parameters as fuzzy input variables is that they consider all the vital properties involved. The lifetime of the network is increased when the overall CH consumes less energy. This is dependent directly on the closeness of the nodes to the CH. When there is sufficient distance between the CHs there is a balance in energy consumption and the centrality metric improves the load balancing in a cluster. Hence these three metrics were chosen as the objective parameters for CH selection.

The relative energy of a node i denotes the remaining energy of the node in the cluster and it is defined as the ratio of the node's residual energy to the maximum residual energy of the cluster. It is given as follows:

$$RE(i) = \frac{E_i}{E_{max}} \tag{6}$$

Where E_{max} denotes the maximum residual energy of cluster and E_i denotes the residual energy of the i-th node.

Node relative density is represented as the concentration of nodes mass and is formulated as the number of neighbour nodes in the interference range d_0 . When node density increases, the energy consumed by the neighbor nodes decreases. Node relative density of node i can be defined as the ratio

Received: July 10, 2021. Revised: September 4, 2021.

of i-th node's density to the maximum node density in a cluster. It is given as follows:

$$RD(i) = \frac{D_i}{D_{max}} \tag{7}$$

Where $\mathrm{D_{max}}$ denotes the maximum density of cluster and $\mathrm{D_i}$ denotes the density of the i-th node.

Centrality determines the closeness of a node with its neighbours and it is computed based on the coordinates and the neighbor nodes. It is given as follows:

$$C_i = \sqrt{\left(x_i - \frac{1}{n}\sum_{j=1}^{n} x_j\right)^2 + \left(y_i - \frac{1}{n}\sum_{j=1}^{n} y_j\right)^2} \tag{8}$$

Where C_i denotes i-th node centrality, n represents the number of neighbor nodes, x_i and y_i denote the horizontal and vertical coordinates of i-th node while x_j and y_j denote the horizontal and vertical coordinates of the j-th node. The relative centrality is computed based on this centrality equation and it determines that the energy consumption is less when the CH is closer to the nodes. The relative centrality of the i-th node is defined as the ratio of the minimum centrality (C_{min}) of the cluster to i-th node centrality and is given as

$$RC(i) = \frac{C_{min}}{C_i} \tag{9}$$

Based on these three metrics, fuzzy logic determines the node priority based on if-then fuzzy rules. The node priority then determines the selection of nodes to the CH role. The triangular and trapezoidal membership functions are used for this evaluation since these functions are effective for real-time sensor networks in a similar manner. The node priority can have the following values: very low, low, moderate-low, medium, moderate-high, high and very high. The fuzzy if-then rules for the three input functions to obtain the output node priority are given in Table 2.

The node priority status can be computed based on these fuzzy rules. Further, fuzzy logic also computes the actual value of the node priority using the center-of-gravity for de-fuzzification. The node priority (NP) thus computed can be expressed as follows:

$$NP = \frac{\int x\mu_{NP}(x)dx}{\int \mu_{NP}(x)dx} \tag{10}$$

Table 2. Fuzzy if-then rules for node priority

Rule	Relative energy	Relative density	Relative centrality	Node priority
1	Low	Low	Low	Very low
2	Low	Low	Medium	Very low
3	Low	Low	High	Very low
4	Low	Medium	Low	Very low
5	Low	Medium	Medium	Very low
6	Low	Medium	High	Very low
7	Low	High	Low	Very low
8	Low	High	Medium	Very low
9	Low	High	High	Very low
10	Medium	Low	Low	Moderate low
11	Medium	Low	Medium	Moderate low
12	Medium	Low	High	Medium
13	Medium	Medium	Low	Moderate low
14	Medium	Medium	Medium	Medium
15	Medium	Medium	High	Moderate high
16	Medium	High	Low	Moderate low
17	Medium	High	Medium	Medium
18	Medium	High	High	Moderate high
19	High	Low	Low	Moderate low
20	High	Low	Medium	Medium
21	High	Low	High	Moderate high
22	High	Medium	Low	Medium
23	High	Medium	Medium	Moderate high
24	High	Medium	High	High
25	High	High	Low	Medium
26	High	High	Medium	High
27	High	High	High	Very high

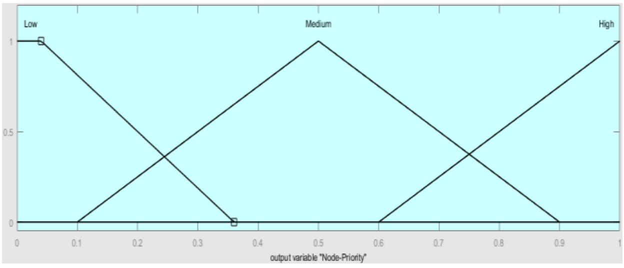

Figure. 1 Membership functions

Where x denotes the node and $\mu_{NP}(x)$ denote the fuzzy membership function for *NP*. This equation deduces the node priority value and during subsequent iterations when the selected CH drains out, that parameter node with the second-highest priority is evaluated first for the CH selection process to reduce the time taken for fuzzification. The node priority values thus obtained are analysed and the highest-ranking priority node in each layer is selected as the CH. The remaining nodes in the layer join the adjacent CHs to create the clusters. The fuzzy memberships are obtained for the given problem in the triangular and trapezoidal shapes shown in Fig. 1.

4.2 Optimal routing path selection using mboa

For the optimal path selection, the multiple objective parameters considered include the residual energy, delay, lifetime, reliability and hop count. Before the selection of optimal paths, these parameters must be computed through the trial of sample data transmission in the paths formed by a greedy approach in the WSN deployed. The residual energy of the routes is computed based on the energy levels of the nodes evaluated by the energy model. The initial energy provided to a node tends to reduce gradually depending on the transmission and reception of each packet. The remaining energy or residual energy decides if the nodes are suitable for transmitting the data or if they would block the data transmission. It can be mathematically computed as follows:

$$E_{res,i}(t) = E_{0,i} - E_t \tag{11}$$

Where $E_{res,i}(t)$ is the residual energy at time *t*, $E_{0,i}$ the initial energy available at node *i* and E_t is the total energy consumption at time *t* computed based on energy model Eq. (1).

Delay is defined as the time for propagation of a packet k from node *i* to node *j* and is the sum of inter-cluster delay and intra-cluster delay. Inter-cluster and intra-cluster delays are computed as the end-to-end delay in transmitting a packet between the CHs to the sink, and the cluster node to the CH, respectively.

$$Delay = Inter - cluster\ delay + Intr - cluster\ delay$$

$$Delay = \left(\frac{(T_{rk,sink} - T_{sk,CH})}{k} + \frac{(T_{rk,CH} - T_{sk,i})}{k}\right) \tag{12}$$

Where $T_{sk,CH}$ and $T_{rk,CH}$ refer to the sending time and receiving time of CH for k-packets respectively, while $T_{sk,i}$ is the sending time of k packets by i-th node and $T_{rk,sink}$, the receiving time of k packets by the sink node.

Lifetime is the time a network operates until the death occurs of a node or a cluster of nodes and it is given as follows:

$$\mathbb{E}[L] = \frac{\varepsilon_0 - \mathbb{E}[E_w]}{P + \lambda \mathbb{E}[E_r]} \tag{13}$$

Where $\mathbb{E}[L]$ is the expected lifetime, P is the constant continuous power depletion, ε_0 is the total non-rechargeable initial energy, λ the average wavelength, $\mathbb{E}[E_w\ ]$ and $\mathbb{E}[E_r]$ are the expected wasted energy and expected reporting energy of the nodes, respectively.

The reliability of a link or route can be estimated based on the Received Signal Strength Metric (RSSM). RSSM value depends on two factors: the received signal strength (RSS_{iz}) of a node i at a distance z and a threshold (T_j). RSSM at node j for the link (i, j) is computed as follows:

$$RSSM = \begin{cases} 0 & RSS_{iz} < T_j \\ \left(1 - \frac{T_j}{RSS_{iz}}\right) & RSS_{iz} \geq T_j \end{cases} \tag{14}$$

Here RSS_{iz} and T_j are computed as

$$RSS_{iz} = \frac{G_a \times G_{tx} \times P_{t,max}}{\left(4\pi \times \frac{z}{\lambda}\right)^2} \tag{15}$$

$$T_j = \frac{G_{rx} \times G_{tx} \times P_{t,max}}{\left(4\pi \times \frac{0.9054R}{\lambda}\right)^2} \tag{16}$$

Where G_a , G_{tx} and G_{rx} refer to the average, transmitting and receiving antenna gains, respectively, while $P_{t,max}$ denotes the maximum transmission power of the antenna and R the maximum range of the antenna.

Hop count is computed as the number of hops taken by a packet to reach the destination. When the transmission range is *TR* and density of node deployment (ρ), hop count can be estimated as

$$Hop = \left\lceil \frac{D}{\frac{TR}{2}\cos(\frac{1}{2}\arcsin\frac{4}{\rho TR^2})} \right\rceil - 1 \tag{17}$$

Where $\left\lceil \frac{\mathrm{D}}{\frac{\mathrm{TR}}{2}\cos(\frac{1}{2}\arcsin\frac{4}{\rho\mathrm{TR}^2})} \right\rceil$ is the expected number of layers. Applying these parameters to the general multi-objective problem formulation, the fitness function or the objective function is determined.

$$F_i = w_1 \times E_{res,i}(t) + w_2 \times Delay + w_3 \times \mathbb{E}[L] + w_4 \times RSSM + w_5 \times Hop \tag{18}$$

Where w_1, w_2, w_3, w_4, w_5 are the weight values assigned to objective parameters to enhance energy-efficient and reliable route selection. Using this objective function, the MBOA selects the optimal routing paths. The Butterfly optimization algorithm (BOA) is based on the natural foraging and mating behaviours of butterflies [27]. Although it is significantly efficient than most of the existing optimization algorithms, this algorithm also comes with its own sets of limitations under the no free lunch theorem. In the sense that there could be limitations in the convergence rate of BOA. This limitation is due to the limited global search capability of BOA resulting in local optimum solutions. Through analysis, it has been found that improving both global and local search equations by modifying the randomly assigned numbers will be beneficial. Further, since BOA performs unidirectional searching it would also result in complexities and slow convergence. Hence, this unidirectional search has been modified to bidirectional searching. These two modifications are the major highlights which form the basis of the proposed MBOA.

In BOA, the butterflies are attracted towards other butterflies through the emission of a fragrance fluid. Butterflies tend to move towards those butterflies that emit more high fragrance and the objective function senses as the determining factor of the butterfly stimulus intensity. In the proposed MBOA, for route selection, this fragrance is replaced by the objective function and the routing paths are mapped as butterflies. The initialization of the algorithm begins with the population of butterflies i.e. set of possible routing paths $B = \{b_1, b_2, \dots, b_n\}$. The fitness values of each path are estimated based on the objective function F_i. Following this, both local and global search processes are commenced to select the best butterfly (optimal routing path). The global search is performed by moving the butterfly towards the best position (g^*), based on fitness values. In BOA, it is given as follows:

$$b_i^{q+1} = b_i^q + (\mathrm{r}^2 \times \mathrm{g}^* - b_i^q) \times F_i \tag{19}$$

Where b_i^q and b_i^{q+1} refer to the current position and next positions of i-th butterfly, respectively. g^* denotes the best current position, q, the current iteration and F_i is the fitness values calculated using Eq. (18) and r, the randomly generated number, $r \in [0,1]$. Similarly, the local search is expressed as

$$b_i^{q+1} = b_i^q + (\mathrm{r}^2 \times b_j^q - b_k^q) \times F_i \tag{20}$$

Where $\mathrm{b}_\mathrm{j}^\mathrm{q}$ and $\mathrm{b}_\mathrm{k}^\mathrm{q}$ are the positions of butterflies j and k, respectively.

In the proposed MBOA, both global and local search equations have been improved by replacing the random number r with a parameter β_n, which is a chaotic parameter that controls the movement of the fireflies in the correct order and is expressed as follows:

$$\beta_n = \alpha\beta_{n-1}(1 - \beta_{n-1}) \tag{21}$$

Where α is the controlling parameter whose value is determined as $\alpha = 4$ to satisfy the chaotic sequence. This parameter is applied to the global and local search equations as given below:

$$b_i^{q+1} = b_i^q + (\beta_n^2 \times \mathrm{g}^* - b_i^q) \times F_i \tag{22}$$

$$b_i^{q+1} = b_i^q + (\beta_n^2 \times b_j^q - b_k^q) \times F_i \tag{23}$$

A probability parameter p is used as a control switch between the global and local search to set search operations during unfavourable conditions. The value of p is determined based on the current state of network nodes and it is compared with a threshold value between 0 and 1 to switch the search processes.

The second modification is the introduction of the bidirectional search process, replacing the less effective unidirectional search. To enable bidirectional search (both forward and backward), step size (S) has been introduced. The initial direction is selected by the greedy process in which the search continues in the direction where the solution is increasingly better and this is expressed as follows:

$$b = \begin{cases} b + S & if\ f(b) > f(b+S) \\ b - S & if\ f(b) > f(b-S) \\ b & otherwise \end{cases} \tag{24}$$

Where b refers to the present location of the butterfly and $f(b)$ denotes the fitness function value and S denotes the step size. These two modifications improve the BOA and enhance the optimal route selection process. The proposed MBOA is summarized in Algorithm 1.

Algorithm 1: MBOA based path selection

```
Begin
Set population of butterflies (routing paths) $B = \{b_1, b_2, \dots, b_n\}$
Calculate the objective parameters using Eq. (11-17)
Determine the value of probability parameter $p \in [0,1]$
Set iteration $q = 0$
While $q < q_{max}$, do
For each $b$ in B
Evaluate the fitness $F_i$ using Eq. (18)
End for
Analyse and determine the current best b
Select a threshold th; $th \in [0,1]$
```

```
If $th < p$ then
        Initiate global search using Eq. (22)
Else
        Initiate local search using Eq. (23)
End if
Initiate bidirectional search using Eq. (24)
Update $b, p$ and $q = q + 1$
End while
Return final best $b$
End
```

4.3 Fault detection and recovery mechanisms

The faults in a secured WSN deployment are either due to technical shortcomings or through external attackers. Therefore, detection of both these types of faults is necessary to maintain the health of the network. The technical faults can be detected easily through the non-response of the nodes in the links. However, these faults initiated by external attacker initiated faults require separate strategies for detection. Most studies ignore them due to the complexities involved in devising these strategies. However, the Elliptic Curve Digital Signature Algorithm (ECDSA) [28] is an efficient verification scheme for detecting faults. ECDSA also facilitates enhancing certain aspects of the process with less complexity and in a more compact manner in the WSN [13]. The Enhanced ECDSA (EECDSA) is built by overcoming the weakness of ECDSA to derive the signer's private key when it is used to generate two signatures for two different data [29]. This is achieved by introducing the unknown parameters of the signer, previously ignored by the verifier to increase verification performance.

First, two users are assigned. While namely UG and UV.UG generates the signature, UV performs the process verification. The elliptical curve parameters namely elliptical points (Ep), elliptic curve ($EC\ (a, b)$), point generator (PG) and a hash function (HF) are selected by the users. EECDSA performs the following three processes: key generation, signature generation and signature verification.

Key generation by UG:

Step 1a: Choose random integer $pk \in [1, pn - 1]$ where pn is the order of PG such that pn is a prime number and pk becomes the private key.

Step 1b: Estimate the public key $PG_u = pk \times PG$.

Signature generation by UG:

Step 2a: Choose two random integers $a_1, a_2 \in [1, pn - 1]$.

Step 2b: Estimate $A_1 = a_1(PG) = (x_1, y_1)$ and $A_2 = a_2(PG) = (x_2, y_2)$.

Step 2c: Estimate $rand_1 = x_1\ mod\ pn$ and $rand_2 = x_2\ mod\ pn$.

If $rand_1 = 0$ and $rand_2 = 0$, go to step 2a.

Step 2d: Estimate $a_1^{-1}\ mod\ pn$ and $e = HF(m)$.

Step 2e: Estimate $Sign = a_1^{-1}(ea_2 + pk\ (rand_1 + rand_2))\ mod\ pn$.

Step 2f: If $Sign = 0$, go to step 2a.

Step 2g: Send m and ($A_2, rand_1, Sign$) - the signature of UG for message m, to UV.

Signature Verification by UG:

Step 3a: Verify that if $rand_1, rand_2, Sign \in [1, pn - 1]$ or else the signature is invalid.

Step 3b: Estimate $e = HF(m)$.

Step 3c: Estimate $rand_2 = x_2\ mod\ pn$.

Step 3d: Estimate $O = S^{-1}\ mod\ pn$.

Step 3e: Estimate $u_1 = eO\ mod\ pn$ and $u_2 = (rand_1 + rand_2)O\ mod\ pn$

Step 3f: $Y = u_1 A_2 + u_2 PG_u = (x_3, y_3)$ and if $Y = 0$, the signature is invalid and stop verification.

Step 3g: Estimate $rd = x_3\ mod\ pn$.

Step 3h: Signature is recognised only if $rd = rand_1$.

In the signature verification process, the introduction of two unknown random integers will necessitate identifying them to extract the private key. In most cases, nodes that are initiated by the attackers are effectively tracked. This helps detect the faults created by external attackers. This approach seems to be much better than the agent-based methods of detecting faults through the message header. Security is also enhanced significantly in this approach.

Once the faults are detected, the nodes which caused the faults are further confirmed by the centralized authority by forwarding smaller message packets to the neighboring nodes of that affected node. This aids in assuring their functionality and also to the report changes to their corresponding neighbors at a one-hop distance. If the forwarded packets are missing, the faults are confirmed. The neighbor nodes of the detected fault nodes determine the connectivity critical level of the fault through the shortest routing table. This ensures that no node is wrongly identified as a fault. These nodes cause both link and network failure, which can be repaired and the connectivity can be restored using the Disruptive fault repair and elimination (DRFE). DRFE restores the connectivity without increasing the length of the shortest paths. It

Table 3. Simulation Parameters

No. of Nodes	100
Area Size	1000 X 1000 m
Channel type	Wireless Channel
Propagation model	Two Ray Ground
Link Layer	LL
Antenna model	Omni Antenna
Traffic type	CBR
Mobility model	Random Waypoint
MAC	IEEE 802.11
1Initial energy	100 Joules
Radio Range	250m
Simulation Time	1000 seconds
Number of packets	10000
Packet rate	8 packets/sec
Data payload	256 bytes/packet
Percentage of CH	10%
Round length	10 seconds
Initial node energy	0.5J
Packet transmission energy	50nJ/bit
Packet reception energy	50nJ/bit

also replaces the fault node by performing nodes block movement instead of individual node movements. The major advantage of DRFE is that it chooses the smallest disjoint blocks to perform fault recovery to reduce the recovery overheads.

The DRFE is installed in a centralized manner to associate with all nodes in the network and access the shortest routing table of each node. When a node is found to be a failure, the smallest block containing the node is determined by the fewest number of nodes. The possible disconnection that is likely to be created to the neighboring nodes' shortest path is also analysed. The critical node thus has the role of shortest path of two nodes and so, the set of suitable nodes to replace the fault is selected based on the shortest routing path after discounting the failed node. So after moving the blocks, two nodes can be connected in a path only if they belong to the same block. If a node *i* become a failed node, the block is moved and the one-hop neighbour *j* node is checked for being present in the same block to replace connectivity. This node *j* then becomes the parent node, while the two-hop and three-hop neighbor nodes become child and grand-child, respectively. As a result of moving the node *j* to replace the failed node, some child nodes may lose direction connection with *j*, thus referencing the paths near the failed node with the least changes to avoid path discovery overhead.

5. Experimental results

5.1 Simulation parameters

The proposed EESFTR protocol was simulated using MATLAB version 9.1. IEEE 802.11 was used as the MAC layer protocol to notify the network layer about possible link breaks. The simulation settings and parameters are provided in Table. 3.

5.2 Results and discussion

The evaluated performance of the proposed EESFTR protocol has been compared with that of the existing routing models in the literature. FTLEACH [8], HEED-FT [11], EABC-EELB-PWDGR [17] (referred to as EABC-PWDGR in graphs), PSO-FTR [18] and RCEEFT [26] have been compared with the proposed EESFTR in terms of end-to-end delay, time cost, energy consumption, lifetime, number of surviving nodes, packet delivery ratio, packet drop and hop count.

End-to-end delay: It is the time taken by the nodes to select the CH and transmit the data packets from the source to destination.

Time cost: it is the time spent until the selected CH nodes runs out-of-energy and becomes a fault.

Energy consumption: It is the amount of power consumed by the nodes in transmitting a packet from a source to destination.

Lifetime: It is defined as the maximum time the network nodes can survive until energy of the node becomes zero.

Number of surviving nodes: It is the remaining alive nodes in a group of network nodes after the specified time.

Packet delivery ratio: It is the ratio of number of successful delivery of packets to the total number of transmitted packets.

Packet drop: It is the number of packets dropped during the transmission process or the rate of failed packets.

Hop count: Hop count is the number of hops taken by the packets to transmit from the source to the destination at a specified time.

Fig. 2 demonstrates the end-to-end delay comparison of the proposed EESFTR with the routing models in the literature. It can be seen that the proposed EESFTR protocol has outperformed other models in terms of reduced delay. EESFTR separated a delay of 25.68ms for 100 nodes, which is 12.6%, 18.9%, 14%, 28.9% and 33.5% lesser than those

Received: July 10, 2021. Revised: September 4, 2021.

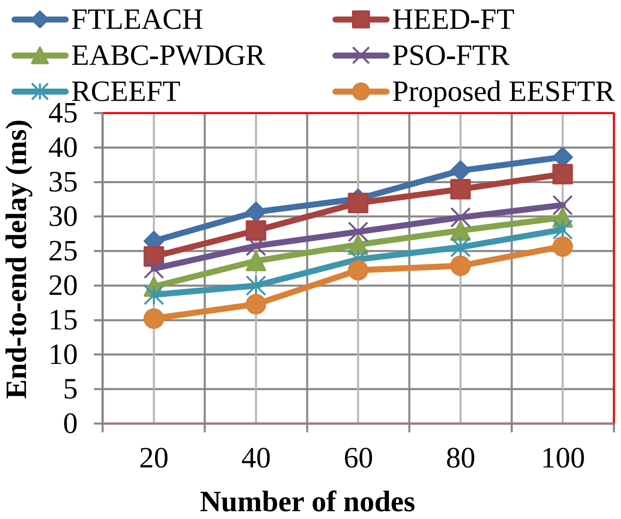

Figure. 2 End-to-end delay comparison

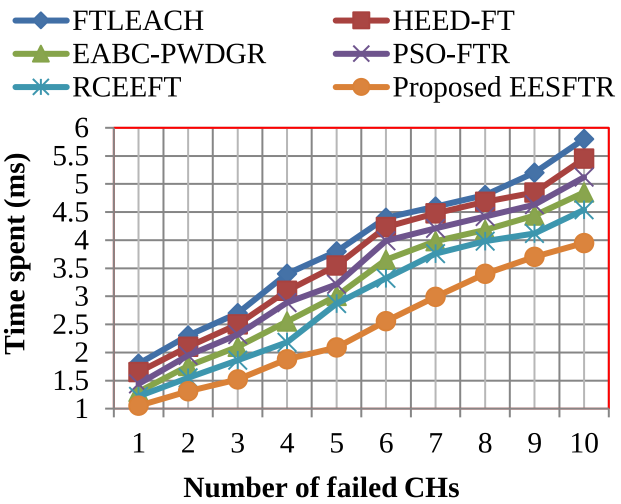

Figure. 3 Time cost of fault tolerance

experienced with FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively. The utilization of fast converging MBOA for route selection and reduced time to replace the faults has considerably minimized end-to-end delay.

Fig. 3 shows the time cost comparison for fault tolerance of the proposed EESFTR with the routing models in the literature. It can be seen that the proposed EESFTR protocol has outperformed other models in terms of less time taken when the failures occurred. EESFTR was found to have consumed 3.95ms when 10 CHs failed, which is 13%, 22.8%, 18.5%, 27.5% and 31.9% lesser than that of FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively. Faster detection and removal of faults using EECDSA and DFRE seem to have influenced the reduction in time cost.

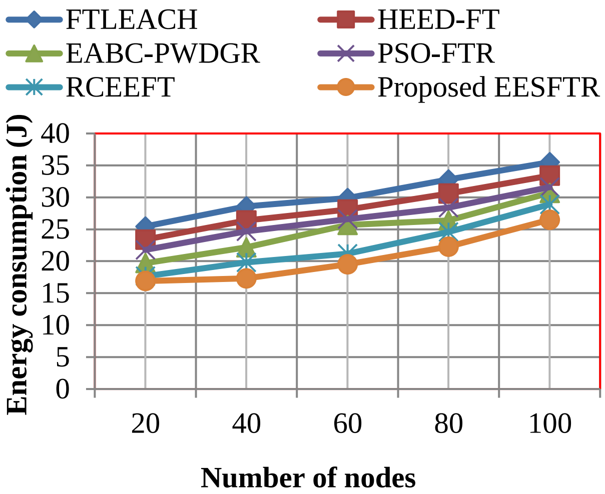

Figure. 4 Energy consumption comparison

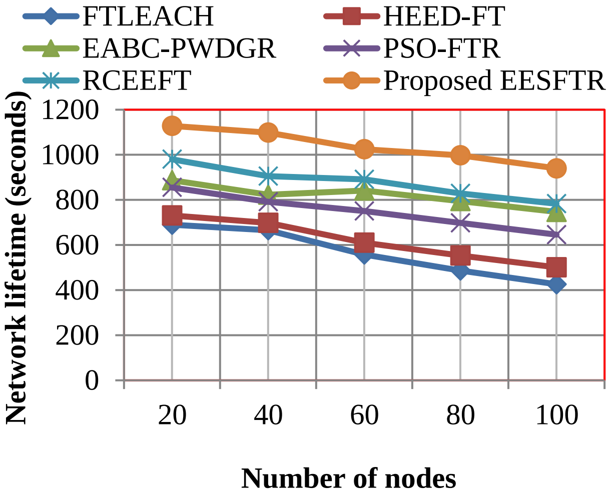

Figure. 5 Network lifetime comparison

Fig. 4 illustrates the energy comparison of the proposed EESFTR with the routing models in the literature. The proposed EESFTR protocol appears to have outperformed other models due to the energy-efficient CH selection and routing path selection. This has resulted in lesser energy consumption of 26.5J for 100 nodes network, which is 8.3%, 16%, 13.6%, 20.6% and 25.35% lesser than those consumed by FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively.

Fig. 5 compares the proposed EESFTR with the routing models in literature in terms of network lifetime. Due to the selection of energy-efficient CH and subsequent balancing of energy in the nodes, the network lifetime seems to have significantly improved in the EESFTR. For 100 nodes network, EESFTR has increased the lifetime of the network to 940 seconds, which is 20%, 45%, 25.8%, 87% and

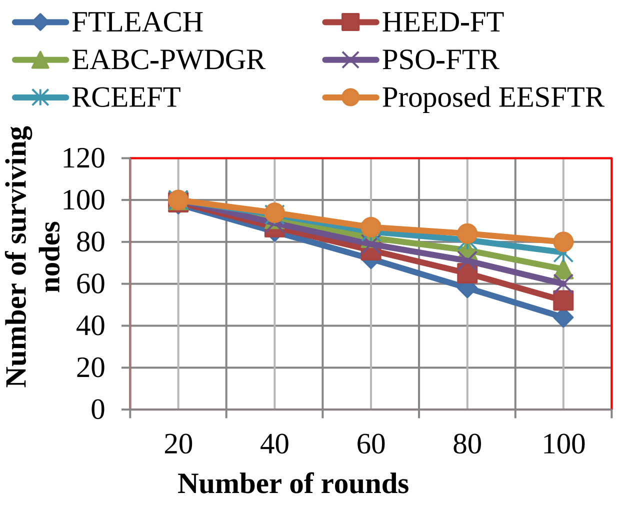

Figure. 6 Number of surviving nodes

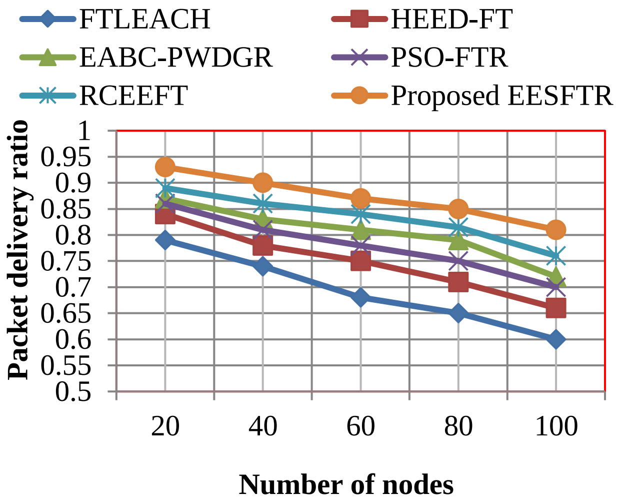

Figure. 7 Packet delivery ratio comparison

120% longer than those achieved by FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively.

Fig. 6 compares the proposed EESFTR with the routing models in literature in terms of the number of surviving nodes. As a result of the increased lifetime and energy-efficient CH selection in EESFTR, the number of dead nodes in the network is reduced even after numerous rounds. For 100 rounds of simulation, EESFTR has 80 active surviving nodes which are 6.67%, 33%, 19%, 53% and 81.8% more than that observed in FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively.

Fig. 7 shows the packet delivery ratio comparison of the proposed EESFTR with the routing models in the literature. As the optimal CH and routes have been selected with a faster and efficient fault recovery process, the packet delivery ratio appears to be significantly higher in EESFTR. For 100 nodes network, EESFTR reports a high delivery ratio of 0.81 which is 6.5%, 15.7%, 12.5%, 22.7% and 35% greater than the delivery ratio of FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively.

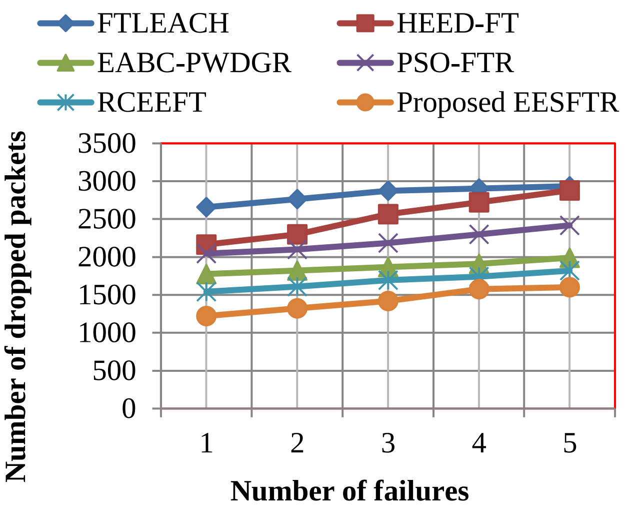

Figure. 8 Packet Drop during Faults

Fig. 8 compares the proposed EESFTR protocol with the routing models in literature in terms of the number of dropped packets during failures. The proposed EESFTR protocol seems to have outperformed other models with a lesser number of packet drops despite the presence of multiple failures. When there are 5 failures, EESFTR has dropped to an average of 1602 packets. This is about 12%, 33.7%, 19.5%, 44.3% and 45.4% lesser packets dropped than that observed in FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively. The DFRE seems to have repaired faults quickly and restored connectivity, which is a significant improvement in packet drop reduction.

Fig. 9 illustrates the hop count comparison of the proposed EESFTR with the routing models in the literature. As the optimal CH and routes were selected with faster fault detection and alternate path determination in EESFTR, the data packets seem to have taken fewer hops to reach the destination. For 100 nodes network, EESFTR achieved data transmission in 8 hops which are 11%, 33%, 42.8%, 50% and 55.5% lesser than in the case of FTLEACH, HEED-FT, EABC-EELB-PWDGR, PSO-FTR and RCEEFT protocols, respectively. Thus from the simulation results, can be concluded that the proposed EESFTR protocol with fuzzy logic based CH selection, MBOA based routing path selection and EECDSA and DFRE based fault detection and recovery mechanism has provided better

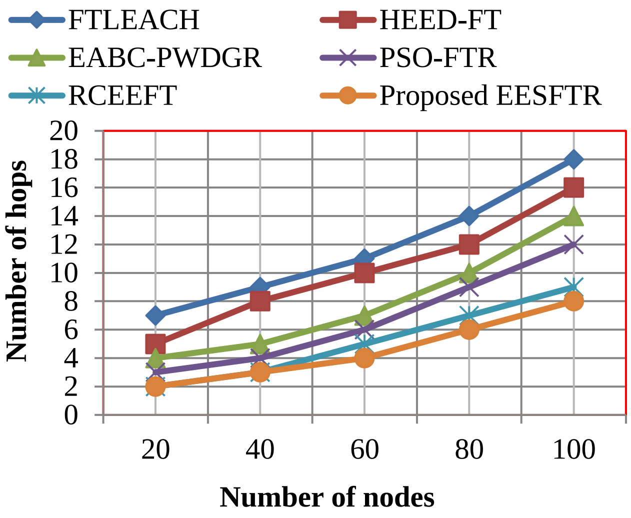

Figure. 9 Hop count comparison

performance when compared to those models.

6. Conclusion

Energy efficiency, security and reliability are the three major factors in ensuring optimal and proficient routing in WSN. A high-performance routing approach using EESFTR protocol has been presented in this paper to ensure the objective of energy-efficient secured fault-tolerant routing. Employment of fuzzy logic to select energy-efficient priority nodes as CH and MBOA to select energy surplus reliable routing paths has also been described in detail. The node and link failures were detected effectively using EECDSA and those faults were effectively repaired connectivity was also recovered using DFRE mechanism. Using these efficient strategies, the proposed EESFTR protocol appears to have provided high performance and high stable routing in WSN. Experimental results indicate that the EESFTR protocol has significantly improved performance with 12.6% lesser delay, 13% lesser time cost, 8.3% lesser energy consumption, 20% increased lifetime, 6.67% increased surviving nodes, 6.5% increased delivery ratio, 12% reduced packet drops and 11% lesser hops when the compared to the performance of the existing efficient routing protocols. In future, the proposed routing protocol will be tested in a real-world implementation. The possibility of resolving the problem of thermal dissipation and high-temperature hotspot problems in the WSN during data transmission in adverse environments will also be investigated. Yet another possible direction is exploring the case of collaborative routing algorithms to reduce faults through pre-detection strategies.

Conflicts of interest

The authors declare no conflict of interest and there are no ethical issues related to this research.

Author contributions

The first author, Hasib Daowd Esmail Al-Ariki is contributed to the paper conceptualization, the implementation of algorithms, the conduct of experiments, validation, formal analysis, the paper investigation and writing-original draft preparation, have been done by 1st author. The paper methodology, writing review and editing, visualization, and supervision and project administration, have been done by 2nd author.

References

[1] V. Kumar, A. Jain, and P. N. Barwal, “Wireless sensor networks: security issues, challenges and solutions”, *International Journal of Information and Computation Technology*, Vol. 4, No.8, pp. 859-868, 2014.

[2] M. Abdullah and A. Ehsan, “Routing protocols for wireless sensor networks: classifications and challenges”, *Journal of Electronics and Communication Engineering Research*, Vol. 2, No. 2, pp. 5-15, 2014.

[3] A. Dâmaso, N. Rosa, and P. Maciel, “Reliability of wireless sensor networks”, *Sensors*, Vol. 14, No. 9, pp. 15760-15785, 2014.

[4] H. D. E. A. Ariki and M. N. Swamy, “A survey and analysis of multipath routing protocols in wireless multimedia sensor networks”, *Wireless Networks*, Vol. 23, No. 6, pp. 1823-1835, 2017.

[5] A. Ahmed, K. A. Bakar, M. I. Channa, and A. W. Khan, “A secure routing protocol with trust and energy awareness for wireless sensor network”, *Mobile Networks and Applications*, Vol. 21, No. 2, pp. 272-285, 2016.

[6] V. W. Mahyastuty and A. A. Pramudita, “Low energy adaptive clustering hierarchy routing protocol for wireless sensor network”, *Telkomnika*, Vol. 12, No. 4, pp. 963, 2014.

[7] H. Y. Min and W. Zaw, “Energy efficient fault tolerant routing LEACH (EF-LEACH) protocol for wireless sensor networks”, In: *Proc. of International Conference on Advances in Engineering & Technology (ICAET 2014)*, 2014.

[8] M. Shelke, G. Tefera, A. Malhotra, and P. Mahalle, “Fuzzy-based fault-tolerant low-energy adaptive clustering hierarchy routing protocol for wireless sensor network”, *International Journal of Wireless and Mobile Computing*, Vol. 11, No. 2, pp. 117-123, 2016.

[9] O. Younis and S. Fahmy, “HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks”, *IEEE Transactions on Mobile Computing*, Vol. 3, No. 4, pp. 366-379, 2004.

[10] S. Lindsey and C. S. Raghavendra, “PEGASIS: Power-efficient gathering in sensor information systems”, *IEEE Aerospace Conference*, Vol. 3, pp. 3-13, 2002.

[11] Y. Zhou, X. Wang, T. Wang, B. Liu, and W. Sun, “Fault-tolerant multi-path routing protocol for WSN based on HEED”, *Journal of Sensor Networks*, Vol. 20, No. 1, pp. 37-45, 2016.

[12] S. Abba and J. A. Lee, “An autonomous self-aware and adaptive fault tolerant routing technique for wireless sensor networks”, *Sensors*, Vol. 15, No. 8, pp. 20316-20354, 2015.

[13] J. Xu, K. Wang, C. Wang, F. Hu, Z. Zhang, S. Xu, and J. Wu, “Byzantine fault-tolerant routing for large-scale wireless sensor networks based on fast ECDSA”, *Tsinghua Science and Technology*, Vol. 20, No. 6, pp. 627-633, 2015.

[14] J. Wang, Y. Zhang, J. Wang, Y. Ma, and M. Chen, “PWDGR: pair-wise directional geographical routing based on wireless sensor network”, *IEEE Internet of Things Journal*, Vol. 2, No. 1, pp. 14-22, 2014.

[15] H. D. E. A. Ariki and M. N. Swamy, “Energy-Efficient Geographical Multi-path Routing Protocol with Adaptive Load Balancing for Wireless Multimedia Sensor Networks”, *Journal of Advanced Research in Dynamical and Control Systems*, Vol. 9, No. 9, pp. 148-166, 2017.

[16] H. D. E. A. Ariki, A. S. Mohammed, and M. N. Swamy, “A novel approach to energy-efficiency in geographical multi-path routing protocol with adaptive load balancing for wireless multimedia sensor networks”, *International Journal of Engineering Intelligent Systems (EIS)*, Vol. 25, No. 2, pp. 77-92, 2017.

[17] H. D. E. A. Ariki, M. A. Alareqi, and M. N. Swamy, “An Enhanced Artificial Bee Colony Based EELB-PWDGR for Optimized Route Selection in Wireless Multimedia Sensor Networks”, *Pertanika Journal of Science & Technology*, Vol. 26, No. 4, pp. 1951-1974, 2018.

[18] M. Azharuddin and P. K. Jana, “A PSO based fault tolerant routing algorithm for wireless sensor networks”, *Information Systems Design and Intelligent Applications, Springer, New Delhi*, pp. 329-336, 2015.

[19] Z. Ye, T. Wen, Z. Liu, X. Song, and C. Fu, “A security fault-tolerant routing for multi-layer non-uniform clustered WSNs”, *EURASIP Journal on Wireless Communications and Networking*, Vol. 2016, No. 1, pp. 192-204, 2016.

[20] Y. Yue, L. Cao, B. Hang, and Z. Luo, “A swarm intelligence algorithm for routing recovery strategy in wireless sensor networks with mobile sink”, *IEEE Access*, Vol. 6, No. 1, pp. 67434-67445, 2018.

[21] M. Khabiri and A. Ghaffari, “Energy-aware clustering-based routing in wireless sensor networks using cuckoo optimization algorithm”, *Wireless Personal Communications*, Vol. 98, No. 3, pp. 2473-2495, 2018.

[22] K. Haseeb, K. A. Bakar, A. H. Abdullah, A. Ahmed, T. Darwish, and F. Ullah, “A dynamic Energy-aware fault tolerant routing protocol for wireless sensor networks”, *Computers & Electrical Engineering*, Vol. 56, No. 1, pp. 557-575, 2016.

[23] J. W. Lin, P. R. Chelliah, M. C. Hsu, and J. X. Hou, “Efficient fault-tolerant routing in IoT wireless sensor networks based on bipartite-flow graph modeling”, *IEEE Access*, Vol. 7, No. 1, pp. 14022-14034, 2019.

[24] P. Maratha, K. Gupta, and A. K. Luhach, “Improved fault-tolerant optimal route reconstruction approach for energy consumed areas in wireless sensor networks”, *IET Wireless Sensor Systems*, Vol. 10, No. 3, pp. 112-116, 2020.

[25] R. Talmale, M. N. Bhat, and N. Thakare, “Energy Attentive Pre-fault Detection Mechanism with Multilevel Transmission for Distributed Wireless Sensor Network”, *Revue d'Intelligence Artificielle*, Vol. 33, No. 2, pp. 97-103, 2019.

[26] E. Effah and O. Thiare, “Realistic Cluster-Based Energy-Efficient and Fault-Tolerant (RCEEFT) Routing Protocol for Wireless Sensor Networks (WSNs)”, in *Future of Information and Communication Conference, Springer, Cham*, pp. 320-337, 2020.

[27] S. Arora and S. Singh, “Butterfly optimization algorithm: a novel approach for global optimization”, *Soft Computing*, Vol. 23, No. 3, pp. 715-734, 2019.

[28] D. Johnson, A. Menezes, and S. Vanstone, “The elliptic curve digital signature algorithm (ECDSA)”, *International Journal of Information Security*, Vol. 1, No. 1, pp. 36-63, 2001.

[29] P. Q. Nguyen and I. E. Shparlinski, “The insecurity of the elliptic curve digital signature algorithm with partially known nonces”,

Designs, Codes and Cryptography, Vol. 30, No. 2, pp. 201-217, 2003.