



Trust-aware energy-efficient stable clustering approach using fuzzy type-2 Cuckoo search optimization algorithm for wireless sensor networks

Nitin Mittal¹ · Simrandeep Singh¹ · Urvinder Singh² · Rohit Salgotra²

Published online: 19 August 2020
© Springer Science+Business Media, LLC, part of Springer Nature 2020

Abstract

With the advancement of communication and sensor technologies, it has become possible to develop low-cost circuitry to sense and transmit the state of surroundings. Wireless networks of such circuitry, namely wireless sensor networks (WSNs), can be used in a multitude of applications like healthcare, intelligent sectors, environmental sensing, and military defense. The crucial problem of WSN is the reliable exchange of data between different sensors and efficient communication with the data collection center. Clustering is the most appropriate approach to prolong the performance parameters of WSN. To overcome the limitations in clustering algorithms such as reduced cluster head (CH) lifetime; an effective CH selection algorithm, optimized routing protocol, and trust management are required to design an effective WSN solution. In this paper, a Cuckoo search optimization algorithm using a fuzzy type-2 logic-based clustering strategy is suggested to extend the level of confidence and hence network lifespan. In intra-cluster communication, a threshold-based data transmission algorithm is used and a multi-hop routing scheme for inter-cluster communication is employed to decrease dissipated energy from CHs far away from BS. Simulation outcomes indicate that the proposed strategy outperforms other communication techniques in the context of the successful elimination of malicious nodes along with energy consumption, stability period, and network lifetime.

Keywords CS · WSN · Network lifetime · Stability period · Fuzzy logic · Trust-aware

1 Introduction

Advances in sensor technology popularize battery-powered wireless sensor networks (WSNs) in many industrial areas including vehicle traffic monitoring, smart factories, IoT, and public safety networks, etc. [1, 2]. WSNs are applied in many fields such as in health-care, environmental sensing, industrial monitoring [3, 4], and vehicle to vehicle communication [5–7]. A WSN is comprised of a base station (BS) and several distributed sensor nodes which, through the sensing of certain physical parameters, communicate

with the environment. The BS is tasked with receiving, processing, and providing data to the end-user for decision making [2]. Nodes in WSN rely on their on-board, limited, non-rechargeable, and non-changeable batteries. Additionally, sensor nodes are limited in storage, memory, and CPU processing capabilities [3].

As sensor nodes and BS use wireless radio signals to exchange packets, energy-efficient routing algorithms play a vital role in energy depletion and network lifetime [1–3]. Direct transmission to the BS consumes additional energy than sending the same data over the same distance in multiple stages of shorter distances. Accordingly, clustering has received attention from researchers, in which each member node communicates directly with a cluster head (CH). In turn, the CH aggregates, compresses and transmits the information to the BS or a neighbor CH [2–4].

Clustering allows multi-hop transmission, data aggregation, data compression, and redundant data elimination. The benefits from clustering depend on the perfection of

✉ Nitin Mittal
nitinmittal.me@cumail.in; mittal.nitin84@gmail.com

¹ Department of Electronics and Communication Engineering, Chandigarh University, Mohali, Punjab 140413, India

² Department of Electronics and Communication Engineering, Thapar University, Patiala, Punjab 147004, India

the clustering algorithm and the fitness of the exploited parameters. Unlike distributed clustering algorithms, which are performed by individual sensor nodes using their local information, the centralized clustering algorithms performed by the BS allow optimal clustering solutions, because the overall view of the WSN is available [4].

Taking the security of the network into account is a challenging task. Trustworthy data collection is a major topic that interests much research work. Trust plays an important role in military and other applications. Many algorithms do not take security into account while selecting CH for WSNs. Most of the existing security-aware protocols use the cryptographic method, which is not enough to overcome serious issues. The cryptographic technique causes complexity in the network, a large amount of overhead, and poor connectivity. Therefore, there is a requirement to consider a security-aware solution to WSN with low complexity and hence less overhead.

The metaheuristic approach is the preferable optimization scheme to enhance performance parameters in hierarchical clustering protocols [8]. In the field of intelligent systems, the fuzzy logic system (FLS) is also a dominant subject. In conjunction with the Cuckoo Search (CS) Algorithm, we suggest a novel clustering protocol using an interval type-2 fuzzy logic system (IT2FLS) [9, 10] for CH selection. CS algorithm since its inception has made a significant impact in the field of optimization research and has been found as one among the viable alternatives. Many improvements have been incorporated into the CS algorithm to demonstrate its significance and make it more of a standard benchmark. In present work, one such recent version of CS namely Cuckoo Search version 1.0 (CV1.0) proposed in [11] has been used to formulate the algorithm for CH selection. As far as CV1.0 is concerned, this algorithm is based on the concepts of division of population and generations. The algorithm also employs the concepts of Cauchy based exploration operation to improve the explorative tendencies of CS. In this work, we merge IT2FLS and CS into one methodology to acquire the power of these two techniques.

Certain factors affect the clustering algorithms in WSN, for example, the residual energy in the sensor nodes and their distances from the BS. Trust value can also be considered as a major parameter that affects the performance of nodes. However, if the problem is carefully analyzed, other factors can be considered. Obtaining an optimal clustering solution requires scaling each parameter by a weight corresponding to its influence on the dissipated energy and network lifetime. The Fuzzy Inference System (FIS) is an efficient modeling tool to combine parameters for better parameter integration results.

We introduce a fuzzy-based centralized clustering technique for energy-efficient routing protocols in WSN.

The proposed clustering technique uses fuzzy logic along with an optimization algorithm to select CHs and enforces a separation distance between them for even CH distribution through the covered area. The separation distance is estimated adaptively according to the number of remaining alive nodes, the dimensions of the area covered by these nodes, and the percentage of the desired CHs. The proposed fuzzy model uses four parameters: the residual energy, trust value, BS distance, and node density to prioritize opportunities of sensor nodes' secure CH choice.

The main contributions of this article can be summarized as follows:

- A trust-aware clustering scheme for WSNs, in which CV1.0 is utilized to optimize the fuzzy rule base table of the fuzzy system for CH selection, is suggested.
- CS-based clustering protocol is used along with energy-aware heuristics to have a longer stability period.
- Trust-aware data communication is employed in that all nodes deliver packets to their next node with the highest trust value.

The rest of the paper is structured as follows: the related work is provided in Sect. 2. Section 3 describes the optimization algorithm for Cuckoo Search (version 1.0), Fuzzy Inference systems and the radio energy dissipation model. Section 4 dealt in detail with the suggested methodology respectively. Section 5 describes the results of the suggested algorithm. Section 6 summarizes the conclusion of our work.

2 Related work

Hierarchical routing protocols (HRPs) for WSNs were introduced in the literature for various routing protocols. HRPs in WSN show higher energy and bandwidth efficiency over conventional routing protocols. Unlike flat routing protocols, where sensors transmit their data to the BS directly, HRPs allow sensors to transmit data via mediators. HRPs are either cluster-based or chain based. In former HRPs, sensors are organized into clusters, and transmissions go through CHs, while in chain-based HRPs, sensors are organized as chains through which the transmissions pass [4].

The clustering of WSN's is typically performed by balancing energy consumption to preserve network life. Most protocols of clustering are probable and CH is chosen on the grounds of maximum residual energy and its distance from BS which is not enough to select the best candidate. A lot of research on clustering protocols in WSN has been undertaken in the latest years for exploration and study. The main points of some common and latest clustering methods are discussed in this section.

2.1 Hierarchical clustering protocols

There have been several efforts to maximize the longevity of the WSN, such as mobile relays, optimal deployment of sensor nodes, and energy harvesting [2]. In terms of routing and clustering, LEACH protocol has played a trailblazing role in the energy consumption minimization of the network [12]. This protocol uses a hierarchical clustering-based routing strategy that forms multiple clusters, each with a single CH. In each group, member nodes transfer their information to the CH node, and the CH node is responsible for aggregating the whole received information and forwarding it to the BS. Besides single-hop clustering, there are several multi-hop hierarchical clustering based protocol. PEGASIS forms chains of sensor nodes such that each node transfers the data back and forth with adjacent nodes [13]. HEED considers remaining energy and proximity to adjacent nodes for the selection of the CH node [14]. Also, energy-efficient uneven clustering (EEUC) protocol partitions the set of sensor nodes into unequal sized clusters and uses multi-hop routing for inter-cluster communication to save the energy of the CH node located near the BS [15].

For the simplicity of the LEACH protocol, several variations have been developed for the minimization of the network energy consumption. In LEACH-C [16], a central coordinator exists to control the cluster formation. By collecting all the required network information, the central coordinator provides a solution for the optimized clusters. In LEACH-CE [17], a CH selection has been improved. Upon the formation of a cluster, the member node with the highest remaining energy is elected as a CH node. This enables to distribute the energy consumption load uniformly over the member nodes. In LEACH-CKM [18], a K-means (or K-medians) clustering algorithm has been adopted to enhance the clustering performance. With a sophisticated energy-efficient cluster formation, the network lifetime is improved.

Manjeshwar and Agrawal [19] classify sensor networks as proactive or reactive networks based on their functional mode. Nodes react to modifications with appropriate parameters of interest in reactive mode instantly, while in proactive mode sensors respond periodically. In the TEEN protocol, the sensed information is communicated to BS only if there is an occurrence based on soft and hard thresholds. TEEN is a reactive routing protocol; it reduces unnecessary or redundant transmissions. TEEN outperforms existing conventional WSN protocols in terms of energy efficiency. Manjeshwar et al. [20] also introduced APTEEN as an extension to TEEN suitable for sending regular information and responding to time-critical circumstances.

Enhanced-SEP (E-SEP) [21] implemented a three-level hierarchy similar to a two-level hierarchy in SEP [22]. E-SEP distributes sensors into three categories where, compared to intermediate and normal nodes, advanced nodes have higher energy. Kang et al. [23] suggested a protocol called LEACH with distance thresholds (LEACH-DT) for CHs selection. In LEACH-DT, CH selection probability depends upon its distance as a parameter from BS. BS determines the distance amongst all nodes and calculates the probability function. This information is broadcast by BS to all nodes. Sensors decide based on the following information about CH selection without any centralized control. LEACH-DT also proposed multi-hop routing in which nodes are divided into various groups depending on their distances from BS. Using multi-hop transmission, energy consumption is reduced to some extent in which the data is communicated from distant groups to the closer ones.

Cluster chain-weighted metrics (CCWM) [24] achieve energy efficiency and increase network performance based on weighted metrics. A set of CHs is selected depending on these metrics. Member nodes use direct communication for transferring data towards their respective CHs. A routing chain of elected CHs is constructed for inter clusters communication and each CH forwards data to its neighboring CH until it reaches BS. However, due to the non-optimized CH election, the reselection of CH results in network overheads. Moreover, intra-cluster communication is direct which leads to uneven energy consumption.

Tarhani et al. [25] introduced the SEECH protocol suitable for periodic data transmission applications. It makes use of a distributed approach in which CHs and relay nodes are selected separately [25]. The reason for different CH and relay node selection is to mitigate the energy burden of CHs. SEECH protocol performed well for large scale WSNs.

Mittal et al. offered two reactive clustering approaches suitable for event-based applications called DRESEP [26] and SEECP [27] in that CHs are chosen in a periodic and deterministic manner respectively to prolong the network lifetime.

2.2 Evolutionary hierarchical clustering protocols

Researchers have created cluster-based routing schemes using optimization algorithms to fix and find optimal solutions for this issue to ensure a longer lifespan for the network [28–38]. ERP [30], EAERP [31], SAERP [32], and STERP using DE [33], HSA [34], SMO [35] and GA [36] are recently developed optimization algorithms based clustering protocols. EAERP restructured substantial features of EAs that assures extended stability period and

prolonged lifetime. ERP overcame the shortcomings of the HCR algorithm [29] by improving the cluster quality of the network. SAERP based routing schemes (DESTERP, HSSTERP, SMTERP, and GASTERP) are inspired by SAERP to achieve extended stability period [33–36].

Mittal et al. introduced a Fuzzy cluster-based stable, energy-efficient, threshold-sensitive routing protocol called FESTERP [37] for applications such as forestry fire detection, suitable for event-driven purposes. The remaining power, node centrality, and distance to BS are considered in the protocol to choose the relevant CHs. CHs are selected with FIS as a fitness function for EFPA. In this approach, a longer stability period is achieved with an energy-based heuristic.

2.3 Fuzzy-based clustering protocols

Kim et al. suggested the CHEF protocol [39] with residual energy and local distance as input variables with FIS. For assessing fuzzy inputs and calculating the possibility of nodes being elected as cluster coordinators, nine fuzzy rules are accessible. LEACH-FL [40] believes the chances of CH candidature to be calculated by three descriptors of nodes (farness from BS, node density, and residual energy). In [41], a clustering algorithm with fuzzy logic is proposed to extend the network lifetime. EAUCF [42] proposes a fuzzy-based distributive clustering protocol based on remaining energy and distance from BS as the CH selection parameter. The tentative CH can be picked using nine IF–THEN fuzzy rules. Each tentative CH calculates the competitive radius for CH candidacy.

MOFCA [43] uses an additional way of selecting CH using the remaining energy and distance to BS parameters. It is mainly intended for two key variables: first, energy efficiency, and second, lightweight for real-time execution. If a CH is nearer to BS, it is more competitive and can accomplish more tasks, such as information collection and transfer.

Most of these fuzzy logic approaches for the CH election in cluster-based HRP uses the partial combination of the parameters, residual energy, BS proximity, local distance, concentration, centrality, etc., to select CHs, but none of them uses an effective combination. Fuzzy logic-based clustering approaches proposed in the literature vary among centralized, distributed, and hybrid. However, most of them are centralized because the fuzzy logic-based CH election requires high CPU cycles and high memory capacities. Furthermore, fuzzy logic-based clustering algorithms require global knowledge about sensors' attributes, which would be costly in terms of energy and bandwidth if exchanged via the sensors themselves. Therefore, for fuzzy logic-based clustering in WSN, the centralized approaches are preferred.

In literature, there are lots of routing protocols existing but all are associated with one or a few types of parameters. Researchers have been demanding a generic protocol that is energy efficient, prolonged network lifetime, scalable, stable, and load balanced. Also, network formation and CH selection are two phases of network organization in cluster-based WSNs. Several cluster-based WSN protocols using nature-inspired optimization methods are proposed in the literature. Such cluster-based routing schemes attempt to optimize either cluster formation or optimal CH election to achieve energy efficiency. There is a requirement to consider both the aspects of network organization, i.e. optimal cluster formation and optimal CH election using an efficient optimization technique.

2.4 Secure clustering protocols

The advantage of selecting the best node as a CH is to enhance the network lifespan. If the security of the network is taken into account trustworthy CH selection is a challenging task. In most instances, CH selection in the network does not take the security into account. Many trust-based mechanisms to select secure CH in the network have been suggested.

LEACH-Mobile [44] is the LEACH variant that promotes node mobility. Every time the sensor node moves, clusters are reframed in this strategy, leading to a high overhead in the cluster but do not take account of network security. In LEACH-TM, CHs are elected using trust value [45]. This mechanism enhances the security of the network; reduce the packet loss by detecting the malicious node.

In a secure and energy-efficient algorithm, the appropriate trust model is set to identify the malicious nodes [46]. In a trust model, the direct and indirect trust calculations are performed by the neighbor monitoring mechanism and are combined to create a trust-aware model to find the malicious node. In this energy-balanced algorithm, remaining energy and node density is regarded as a routing choice to increase network lifespan.

Chen et al. [47] proposed a trust-aware and low energy consumption security topology (TLES) for WSN. The trust value, residual energy, and node density are considered as CH selection parameters in this algorithm. Based on the distance to BS, the node's degree, and remaining energy, the next-hop node is selected. In S-SEECH [48], an energy-aware routing protocol is suggested to provide energy efficiency and security in WSNs. Rehman et al. [49] presented the idea of a secure trust and energy-efficient based clustering algorithm for WSN. In this approach, trust-aware CH is elected by determining the weight of each node with low energy expenditure. The node weight implies the composite of different metrics such as trust metrics, which

enables a safe choice of a CH selection so that the malicious node is detected.

The literature shows that the primary objective of the methods mentioned above is to enhance the security and network lifespan, by applying efficient clustering and routing algorithms. FL appears to be a promising method to address some of such important decision-making aspects of WSNs. In present work, a relatively new CS has been used and exploited using IT2FS for solving the above-said problems in the clustering algorithm. CS is good at both exploration and exploitation. In this research, a clustering algorithm optimizes cluster formation and CH selection simultaneously keeping into account the energy efficiency and trust value of the node to find nearly optimal solutions for the organization of the cluster-based WSNs.

3 Background

3.1 Cuckoo search version 1.0 (CV1.0) algorithm

CS algorithm is a recently introduced algorithm in the field of global optimization and is based on global explorative Lévy flight based random walks and local uniformly distributed random number based exploitative searching paths [50]. Both these processes are controlled by randomly initialized switching probability. The algorithm is highly competitive but in its basic form suffers from the problems of poor exploration. Apart from poor exploration, most of the work done to date doesn't provide proper parametric studies and a lot of work is required to be done in this context. One such recent introduction is the CV1.0 algorithm [11]. This new algorithm is based on the properties of population division and generation division. These properties help the algorithm to improve the diversity required for exploration in the total search space and exploitation within the specified region.

Apart from this basic modification, the algorithm employs enhanced global search using the concepts of grey wolf optimization algorithm (GWO) [51] and dual division of local search. The concepts of GWO have been added to make the algorithm more efficient in explorative tendencies. The major reason for adding dual division in the local search stage is to provide different searching strategies for the exploitation part. It helps the algorithms to abruptly change their positions during the final stages and hence improve the algorithm performance.

The new CV1.0 algorithm starts by initializing random solutions and then employing the local and global search equations. The first step is to divide the total number of iterations into two parts. For the first half, the Cauchy based global search is employed. Original CS-based local search is used for the second half. This is because of the

fatter tail of Cauchy based search, the algorithms explorative tendencies improve gradually. The local search phase for the first half of the iterations remains similar to the original CS algorithm, as minimal exploitation is required during the initial stages.

In the global search phase, the general equation of CS remains the same with addition to Cauchy based mutation operator $C(\delta)$ [52] instead of Lévy mutation operator [11]. This mutation operator is based on Cauchy density function given by

$$f_{Cauchy(0,g)}(\delta) = \frac{1}{\pi} \frac{g}{g^2 + \delta^2} \quad (1)$$

And based on this density function, the distribution function is given by

$$y = \frac{1}{2} + \frac{1}{\pi} \arctan\left(\frac{\delta}{g}\right) \quad (2)$$

where $g = 1$ is the scale parameter and $y \in [0, 1]$. Solving above for δ , we get

$$\delta = \tan\left(\pi\left(y - \frac{1}{2}\right)\right) \quad (3)$$

The above equation will generate a Cauchy distributed random number in the range of 0–1.

$$x_i^{t+1} = x_i^t + \alpha \otimes Cauchy(\delta)(x_i^t - x_j^t) \quad (4)$$

Here x_i^t is the previous solution, x_j^t is a random solution from the population and \otimes is the multiplication factor.

For the second half of the generations, the global search phase is enhanced by using three random solutions from the whole population [11]. The three solutions required are generated based on the current best solutions and are given by:

$$\begin{aligned} x_1 &= x_i - A_1(C_1 \cdot x_{best} - x_i^t); \\ x_2 &= x_i - A_2(C_2 \cdot x_{best} - x_i^t); \\ x_3 &= x_i - A_3(C_3 \cdot x_{best} - x_i^t) \end{aligned} \quad (5)$$

$$x_{new} = \frac{x_1 + x_2 + x_3}{3} \quad (6)$$

Here x_{best} is the current best, x_{new} is the new solution generated by using x_i random solution. A_1, A_2, A_3 and $C_1, C_2, C_3 \in A$ and C respectively, and are given by $A = 2a \cdot r_1 - a; C = 2 \cdot r_2$. Here $a \in [0, 2]$, subject to $r_1, r_2 \in [0, 1]$ is a linearly decreasing random number.

For the local search phase, the population size is divided into two halves and two different search equations are used to find the final solution [11]. Here the search equation used is given by

$$x_i^{t+1} = x_i^t + \alpha \otimes H(p - \epsilon) \otimes (x_j^t - x_k^t) \quad (7)$$

and

$$x_i^{t+1} = x_i^t + F \cdot ((x_j^t - x_k^t) + (x_l^t - x_m^t)) \quad (8)$$

where Eqs. (7) and (8) both are used for only half of the population sizes and the final solution is a combined population from both the search equations. Also, F is a random solution in the range of $[0, 1]$ and solutions, x_j^t , x_k^t , x_l^t and x_m^t used are the random solution from the whole of the population. On a whole, all of these above said modifications have been added to improve the performance of the CS algorithm.

Further to solve the above said CH selection (a binary) problem for WSN, a binary version of CS is required. In this paper, when the position of the solution is updated, the equation used to discrete the position is as follows:

$$Flag_i(j) = \begin{cases} 1, & \text{if } (x_i(j) \geq 0.5) \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

where $x_i(j)$ indicates the j th position of i th solution.

The pseudo-code for the above-said algorithm is given by

Begin:

Initialize population: n of cuckoo
 Find current best and define fitness function $f(x)$
do Until generation counter < total generation size (t_{max})
 global Search: using equation (7)
 evaluate
 if new fitness better than old
 replace old solution by new
 end if
 local search: using (7) & (8).
 Find the current best.
end until
 update the final best

End

Algorithm 1: Pseudo-code of Cuckoo version 1.0 (CV 1.0)

3.2 Fuzzy inference systems

The fuzzy approach is preferred in applications when a lot of uncertainties are there for a single parameter. It checks a single parameter for a different number of conditions accounting for all the possibilities. It consists of four units; fuzzifier, defuzzifier, fuzzy rules, and an interface engine as shown in Fig. 1. A crisp value is given to the fuzzy system as input. It is further changed to the fuzzy input set value

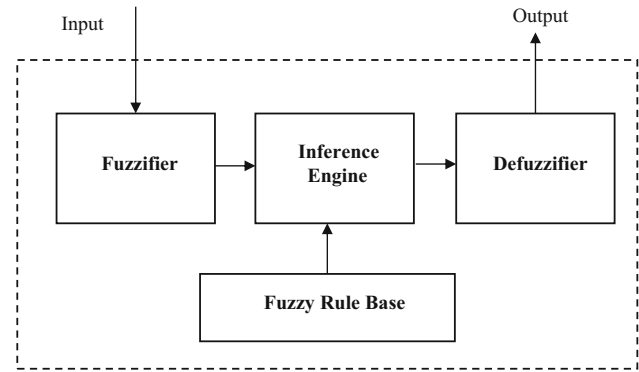


Fig. 1 Fuzzy inference systems

using Fuzzifier. To get back a crisp value at the output, a defuzzifier is used. An Interface engine is used to provide output. The output is obtained based on the rules that are defined for CH selection in this paper.

Type-2 fuzzy sets and systems generalize standard Type-1 fuzzy sets and systems so that more uncertainty can be handled. From the very beginning of fuzzy sets, criticism was made about the fact that the membership function of a type-1 fuzzy set has no uncertainty associated with it, something that seems to contradict the word fuzzy since

no uncertainty, then a type-2 fuzzy set reduces to a type-1 fuzzy set. Interval type-2 fuzzy sets have received the most attention because the mathematics that is needed for such sets—primarily Interval arithmetic—is much simpler than the mathematics that is needed for general type-2 fuzzy sets.

3.3 Radio energy dissipation model

Generally, WSN’s energy consumption consists of many parts such as monitoring, data storage, and data transmission [54]. However, a great percentage of total energy consumption is accounted for by the energy used for data transmission. In this work, the first order radio model [27] is considered for the energy consumption calculation used for data communication as shown in Fig. 2. For path loss calculations, two channel models are used in this energy consumption model: free space model and multipath fading model on the basis of transmitter and receiver separation. The free space model is used when the separation is less than or equal to the threshold d_0 , otherwise multipath fading model is selected.

In order to achieve our proposed protocol, a few assumptions that are adopted as follows:

1. All the chosen nodes are considered as static after deployment.
2. Two types of nodes are as follows: one is sensor node for sensing temperature monitoring environment and another type of node is sink or BS fixed in the center of the sensor network.
3. N Sensors are deployed randomly in the region A. The BS is deployed at the center of region A.
4. Sensors are location unaware i.e. they do not have any information about their location.
5. Sensors continuously sense the region and they send to CH or BS depending on some threshold value.
6. Battery of the sensors cannot be changed or recharged as the nodes are densely deployed in a harsh environment.
7. Network is homogeneous (or heterogeneous) i.e. the sensors may have same (or different) amount of energy and processing capabilities.

8. Finally, the BS is assumed to have wide transmission range cover, hence can use a single broadcast to reach all SNs.

The energy consumption for transmission of each packet is calculated by:

$$E_{TX}(l_{bits}, d) = \begin{cases} l_{bits}E_{elec} + l_{bits}\epsilon_{frees_amp}d^2, & \text{if } d < d_0 \\ l_{bits}E_{elec} + l_{bits}\epsilon_{two_ray_amp}d^4, & \text{if } d \geq d_0 \end{cases} \tag{10}$$

where E_{TX} is the energy consumption for transmission, l_{bits} is a length of the packet (i.e., number of bits in each packet), and d is the transmitter and receiver separation. E_{elec} is the energy consumption due to the transmitter and receiver circuit to process the data before sending or receiving while ϵ_{frees_amp} and $\epsilon_{two_ray_amp}$ are dependent on the transmitter amplifier model. The threshold value d_0 is used to judge which model should be adopted and it can be calculated as:

$$d_0 = \sqrt{\epsilon_{frees_amp} / \epsilon_{two_ray_amp}} \tag{11}$$

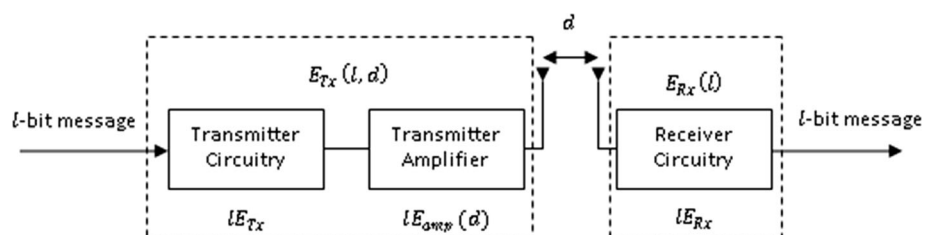
Energy consumption during the reception of data packet is calculated as:

$$E_{RX}(l_{bits}) = l_{bits}E_{elec} \tag{12}$$

4 Proposed clustering algorithm

Nodes in WSNs can be homogeneous or heterogeneous [3] and communicate with the BS individually or may form several clusters with CHs. This paper considers homogeneous WSNs for its purpose, heterogeneous WSNs can also be equally adopted in the proposed approach. In homogeneous WSNs, all nodes are considered to be the same in terms of residual energy, radio transmission capability, processing power, etc. However, heterogeneous WSNs consist of nodes with different residual energy, processing, and transmission power. As this proposal focuses on balancing residual energy strictly, its candidate CHs selection procedure is designed to choose only those nodes which are superior to others in terms of considered factors. In homogeneous WSNs, all nodes have the same capability of becoming a candidate CH at the beginning. Nodes served

Fig. 2 Radio energy dissipation model [27]



as CHs and left with less energy have a low chance of being a candidate CHs in the subsequent rounds until the residual energy of other nodes is reduced to the equivalent level. Similarly, in heterogeneous WSNs, all advanced nodes are forced to be selected as candidate CHs until their residual energy is reduced to the average. Again, candidates having values within a calculated range is considered only that guarantees the most resourceful nodes compete to be the final CHs.

HRPs generally follow a layer-based architecture, where CH election and cluster formation are accomplished in one layer, and routing is performed in another layer. In this work, interval type-2 FIS and CV1.0 were introduced to provide the fitness value for each node to cope with uncertainties during CH choice. The proposed protocol called a Trust-aware Energy-efficient fuzzy type-2 CV1.0 based routing protocol (TEFCSR) is divided into rounds consisting of set-up and data transmission phases as shown in Fig. 3.

In the set-up stage, BS uses FIS-based CV1.0 to select CHs from the alive SNs with remaining energy above a threshold energy level as shown in Fig. 4.

Let $X_i = (X_{i1}, X_{i2}, \dots, X_{in})$ represent the i th population vector of n SNs, where $X_i(j) \in \{0, 1\}$. Alive SNs and CH nodes are represented by 0 and 1 respectively. For example, assuming a solution is (0, 0, 1, 0, 0, 0, 1, 0, 0, 1). There are 10 sensor nodes in the region and CH nodes are chosen as 3rd, 7th and 10th nodes.

The initial population of solution vectors is given by

$$X_i(j) = \begin{cases} 1, & \text{if } (rand \leq p) \\ 0, & \text{otherwise} \end{cases} \quad (13)$$

where p is the percentage of CH selection, and $rand$ is a uniform random number.

The fitness value of each solution is assessed to quantify the efficacy of each solution in the CH selection problem using FIS (in the next section). To generate an evolved population, the population passes through different operators (see Eqs. 1–9).

Finally, the fittest vector is used to seed the next phase where the non-CH nodes are associated with their CHs to form clusters. This process is repeated iteratively until the termination condition occurs.

4.1 CH selection using T2FLS

In this section, we present the proposed fuzzy model used for the CH election and a clustering technique based on that fuzzy model to accomplish optimal clustering in WSN. Different factors influence the CH election in WSN. Therefore, they must be combined appropriately for the best decisions. FIS is an efficient mechanism for such a purpose. It allows combining all input parameters in such a

way that reflects their effectiveness in the CH election. To achieve maximum benefits from fuzzy logic for CH election, it is necessary to explore the factors that have an impact on CH election, use effective means to measure each of these factors, and build an efficient fuzzy model characterized by the effective combination of fuzzy rules and the appropriate design for the fuzzy sets. Accordingly, the Type 2 Fuzzy Logic System (T2FLS) model scheme is built to meet the above-mentioned requirements to achieve an efficient CH election in WSN.

The lifetime of the WSN is considerably influenced by the technique used for the CH election, which in turn is influenced by many factors. These factors are expressed in the context of fuzzy logic as linguistic variables. Four linguistic variables are involved in the proposed fuzzy controller. They influence the network lifetime directly or indirectly by one of these aspects: energy consumed by CHs, total energy consumed by non-CH nodes (local consumed energy), the distribution of energy consumption loads through sensor nodes, or the trust value of each node. The following are the linguistic variables used in our proposed system:

Remaining Energy (RE): Selecting sensor nodes with higher energy as CHs improves network lifetime by balancing energy consumption through the WSN's nodes.

Distance from the BS (DBS): The lower the distance between CHs and the BS, the lower the consumed energy. Sensor nodes closer to the BS have to be given higher opportunities to be CHs over farther ones.

Density of surrounding nodes (D): Selecting CHs surrounded by dense nodes over CHs surrounded by sparse nodes improves the energy consumption by increasing the opportunity for nodes with more neighbors in their vicinity to become CHs. Thereby, the local consumed energy for the group members is decreased.

Trust Value (TV): The proposed approach is based on trust management in WSN. To find the malicious node, each SN observes activities of the surrounding node. These activities are used to describe the trust value. Trust value is of two types: direct trust and indirect trust values. Direct trust value is the value that is based on nodes self-monitoring only. Indirect trust value is the value that may rely on opinions provided by the neighbor nodes. The trust calculation is performed in each communication round. Trust value is calculated according to the threshold value. If the normal node's trust value is below a predefined threshold, then it is said to be a malicious SN.

4.1.1 Calculation of direct trust value

Nodes monitor each other's behavior between neighbors and use the direct and indirect trust value to get comprehensive trust values. If SN i and j are single-hop neighbor

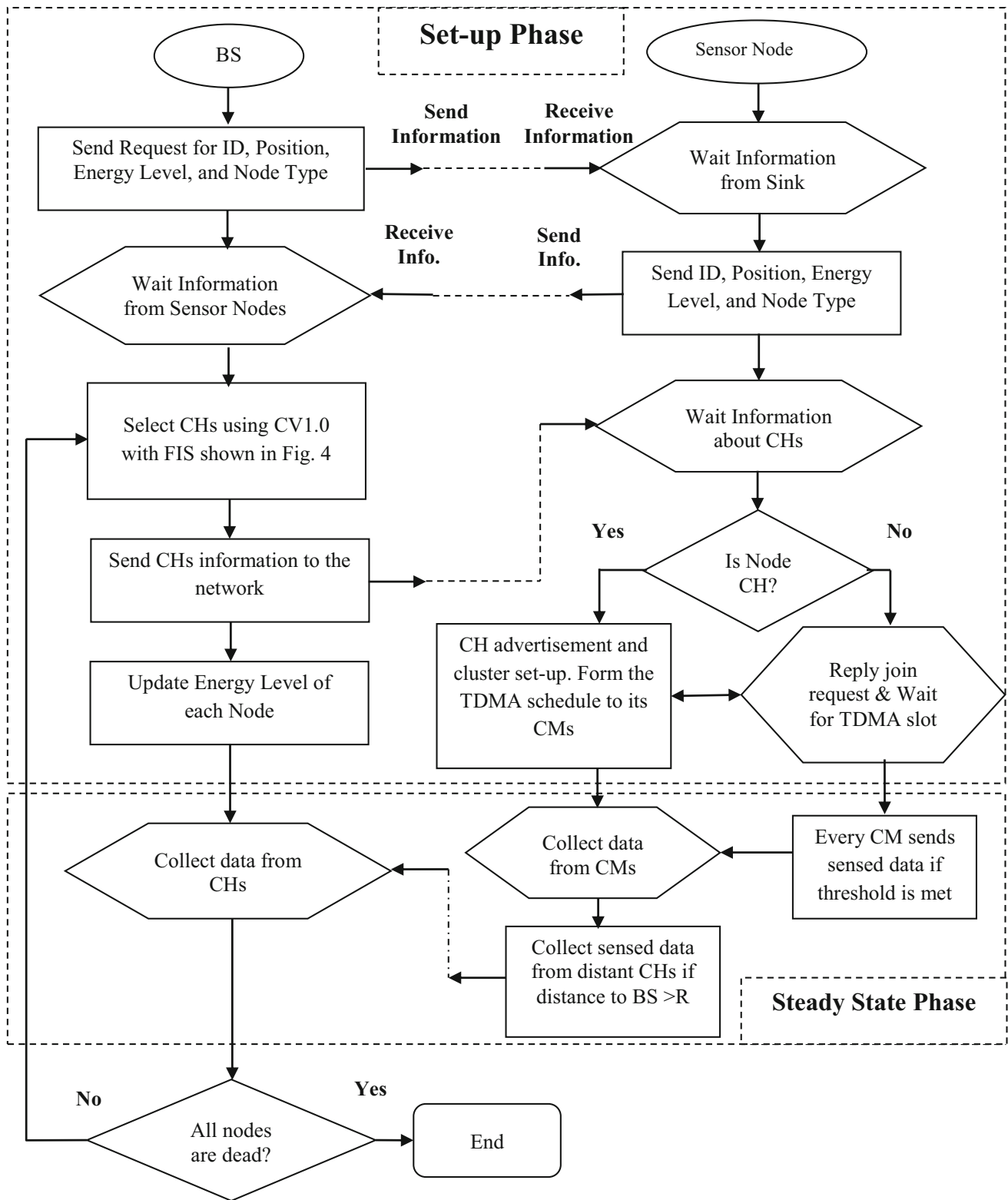


Fig. 3 Operation of proposed TEFCSRP

then direct trust value of SN j is calculated by SN i after every next communication round.

Sending Rate Factor $SF_{i,j}(t)$: Calculating SN i observe the amount sending of the calculated SN j . The SN can be called a self-seeking SN when the calculated value is below

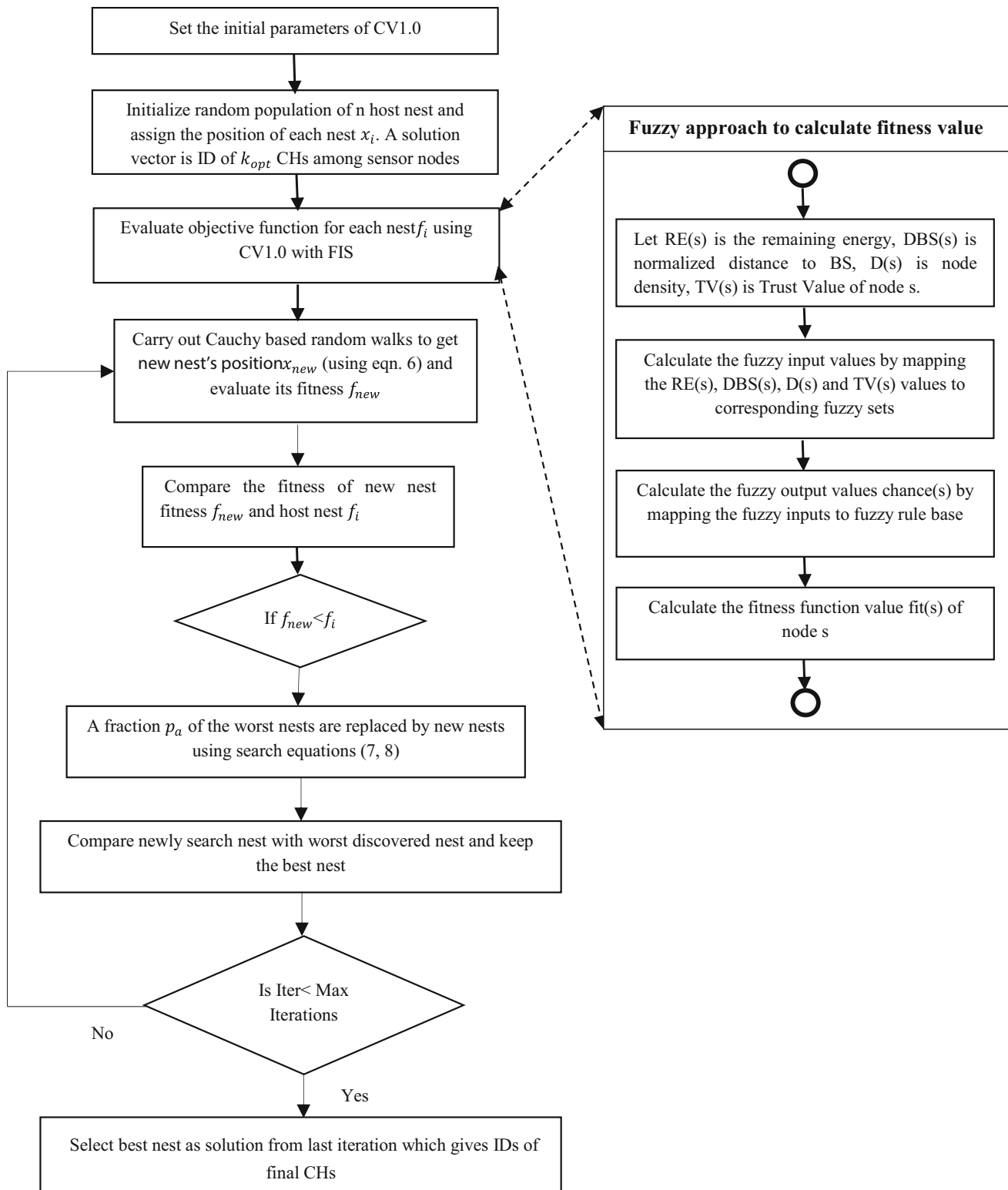


Fig. 4 CH election algorithm using CV1.0 with FIS

the lower-level threshold T_L . If the value exceeds the upper-level threshold T_H the SN performs the attack as the denial of service. Equation (14) evaluates the sending rate factor.

$$SF_{ij}(t) = \begin{cases} \frac{SP_{ij}(t) - T_L}{ES_{ij}(t) - T_L}, & SF_{ij}(t) \leq ES_{i,j}(t) \\ \frac{T_H - SP_{ij}(t)}{T_H - ES_{ij}(t)}, & SF_{ij}(t) > ES_{i,j}(t) \end{cases} \quad (14)$$

where $SP_{ij}(t)$ stands for the sending quantity at period t and $ES_{ij}(t)$ means evaluated value of the sending quantity. When $T_L = 300$, $T_H = 700$, and $ES_{ij}(t) = 500$, the changes of $SF_{ij}(t)$ are shown in Fig. 5.

If the value of $SP_{ij}(t)$ is nearer to $ES_{ij}(t)$, the value of $SF_{ij}(t)$ is nearer to 1, which means the nodes get higher trust value.

Packet Drop Rate Factor ($DF_{ij}(t)$): In MWSNs due to rapid topology changes high packet loss occurs. The packet drop exists during the transmission process, causing loss of data. Equation (15) evaluates the packet drop rate factor.

$$DF_{ij}(t) = R(t)/T(t) \quad (15)$$

Here, $R(t)$ stands for the amount of packet received by all SNs in time t . $T(t)$ stands for the amount of packet transmitted by all SNs in time t . Also, it varies from 0 to 1.

Consistency Factor ($CF_{ij}(t)$): To avoid malicious node’s fake packets due to wireless nature, it is required to compare the information of SNs by itself with the information observed by neighbor nodes. SN i observes the packet of calculated node j , and then compares the information observed by itself with the information observed by j . If the variation is within a specific range, the calculating SN i and calculated SN j have the same recommendation about the observed surrounding. Equation (16) calculates the consistency factor.

$$CF_{ij}(t) = \frac{CP_{ij}(t)}{CP_{ij}(t) + NCP_{ij}(t)} \quad (16)$$

$CP_{ij}(t)$ stands for the number of SNs having the same packet and $NCP_{ij}(t)$ is the number of the inconsistent packet. $CP_{ij}(t) + NCP_{ij}(t)$ is the number of all the packet that i received from its surrounding nodes. Firstly, we calculate SN direct trust value and after that indirect trust value via different SN m that connects both SNs i and j . Direct trust value is computed as follows:

$$Td_{ij}(t) = (1 - \alpha) * SF_{ij}(t) * CF_{ij}(t) * DF_{ij}(t) + \alpha * Td_{ij}(t - 1) \quad (17)$$

where $Td_{ij}(t)$ stands for direct trust value varying from 0 to 1, α is a constant and is set to 0.5 here. If $Td_{ij}(t)$ is 0, it means that SN is a not behaving well and hence is malicious and 1 represent that the SN is behaving well and is trustworthy.

BS uses CV1.0 to select CHs based on the *chance* value of each node using T2FLS, depending on the data received from SNs id, location, residual energy and trust value). The node is chosen as CH that has a *chance* value greater than other SNs. The entire CH selection method is shown in Fig. 6. A T2FLS (Fig. 6a) is comparable to a Type-1 FLS (T1FLS) as shown in Fig. 6b, the T1FLS defuzzifier block is replaced by the T2FLS output processing block consisting of type reduction followed by defuzzification [9, 10].

As illustrated in Fig. 7, four input variables for T2FLS are the remaining energy (RE), distance to BS (DBS), the density of adjacent nodes (D) and trust value of nodes (TV), and CH node selection likelihood is the only output parameter called *chance*. The node option to be chosen as CH is more for greater *chance* values.

The universal discourse of RE , DBS , D , TV and fit variables is [0... 1], [0... 1], [0... 1], [0... 1], and [0... 1] respectively. The membership functions are shown in Fig. 7(a–d) for each of the four input linguistic variables.

Applying these features to fuzzy logic the resulting suggested T2FLS involves the following set of fuzzy input factors:

$$\text{Residual energy } RE \in \{\text{low, medium, high}\}, \quad (18)$$

$$\text{Distance to BS } DBS \in \{\text{nearby, average, far}\}, \quad (19)$$

$$\text{Density of surrounding nodes } D \in \{\text{low, medium, high}\}, \quad (20)$$

$$\text{Trust value of nodes } TV \in \{\text{low, medium, high}\}, \quad (21)$$

and the probability of a CH candidate election *chance* is the resulting output, shown in Fig. 7e.

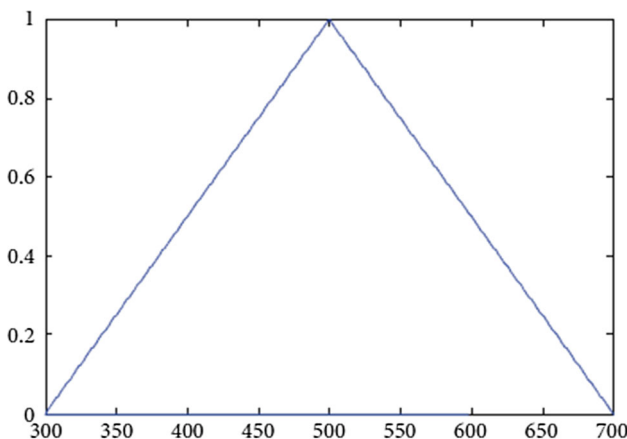


Fig. 5 The variation of sending rate factor

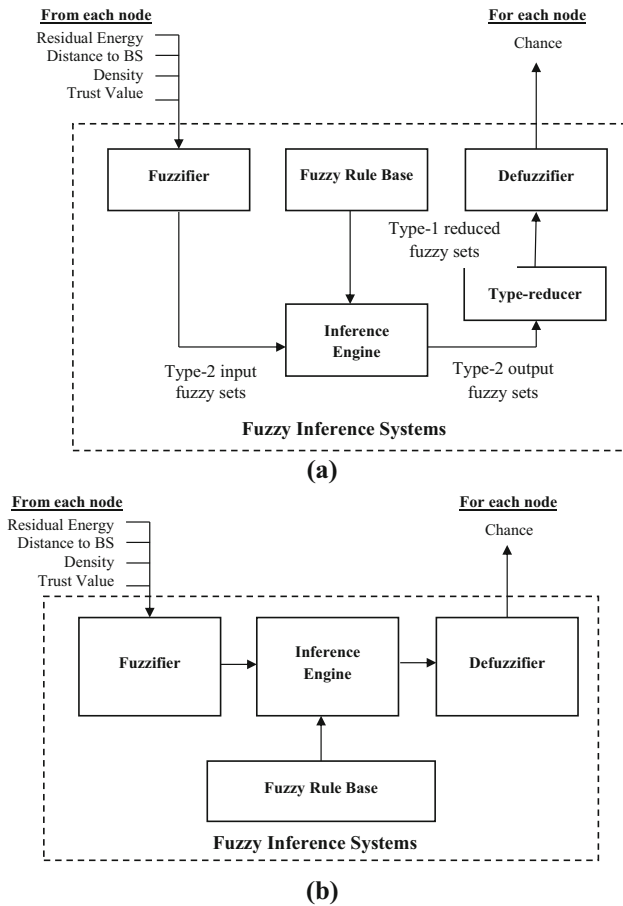


Fig. 6 Probabilistic model for CH selection. **a** Using fuzzy type-2 and **b** using fuzzy type-1

$$\begin{aligned}
 \text{chance} \in \{ & \text{very low (VL), low (L), rather - low (RL),} \\
 & \text{medium - low (ML), medium (M), medium - high} \\
 & \text{(MH), rather - high (RH), high (H), very high (VH)} \}.
 \end{aligned}
 \tag{22}$$

Table 1 shows a detailed set of fundamental rules ($3 \times 3 \times 3 \times 3 = 81$) for T2FLS. A node with greater residual energy, closer to BS, high density and high trust value is more likely to be chosen as a CH. In Table 1, Rule 63 is an optimistic illustration of the issue, while the contradictory one is Rule 19.

Steady-state phase: It is the data communication phase that represents the communication model for the transmission of sensed data to the BS. The steady-state phase is divided into two parts called an Intra-cluster data transmission phase and an Inter-cluster data transmission phase. Since the interval for data transmission in this phase is much longer than the set-up phase, so there is a scope of energy dissipation reduction in this phase as well. In intra-cluster data transmission, the member nodes send information to the CH at a certain time interval, being a part of

the reactive protocol [55]. During the inter-cluster data transmission phase, CH receives data from other CHs and sends the aggregated information to the next hop. Next hop depends on the distance in BS and CH at the slot of time that is allocated by the CH of the upper level.

The energy of active sensor nodes will be dissipated during the intra-cluster data transmission stage while sensing, packet transmission, receiving, and aggregation. The sensor nodes only communicate the data identified to the CH when the threshold values are met [34]. The reception and aggregation of packets will also consume energy for CHs. The energy of the MNs and CHs can, therefore, be formally altered in this stage according to the following expression:

$$E(\text{node}_j) = \begin{cases} E(\text{node}_j) - E_{\text{sensing}} - E_{TX_{\text{node}_j, CH_k}}, & \text{if sensed value} \geq \text{Threshold} \\ E(\text{node}_j) - E_{\text{sensing}}, & \text{if sensed value} < \text{Threshold} \end{cases}
 \tag{23}$$

$$E(CH_k) = E(CH_k) - (E_{RX} + E_{DA})
 \tag{24}$$

where $E(\text{node}_j)$ and $E(CH_k)$ denote the residual energy of sensor j and CH k respectively, $E_{TX_{\text{node}_a, \text{node}_b}}$ is energy cost for transmission from node_a to node_b , E_{RX} is energy cost for the reception of data and E_{DA} is the data aggregation energy expenditure.

In the inter-cluster data transmission phase, for next-hop selection indirect trust value is used. Equation (25) calculates indirect trust value

$$\begin{aligned}
 Tid_{i,j}(t) &= f_i(Td_{i,j}(t), Td_{k,j}(t)) \\
 &= a * Td_{i,j}(t) + b * Td_{k,j}(t)
 \end{aligned}
 \tag{25}$$

where $Td_{k,j}(t)$ is the direct trust value of evaluated SN by k . $f_i[\cdot]$ can be evaluated according to the needs of the actual network subject to $a + b = 1$. The value of a is set to be higher if the node trust value judgment by own is more important than the trust value of the other nodes. Here, we consider

$$Tid_{i,j}(t) = 0.5 * Td_{i,j}(t) + 0.5 * Td_{k,j}(t)
 \tag{26}$$

If this trust value is within the prescribed threshold limits, then only data is transmitted to the next hop. Also, CHs calculate the energy consumption expense of separate routing routes to select an ideal relay node (another CH) or transmit information directly to BS to prevent long-distance communication. Using direct communication, energy consumption in the routing path can be calculated as:

$$E(CH_k, BS) = \begin{cases} l_{\text{bits}} E_{\text{elec}} + l_{\text{bits}} \epsilon_{\text{fris_amp}} d_{CH_k, BS}^2, & \text{if } d_{CH_k, BS} < d_0 \\ l_{\text{bits}} E_{\text{elec}} + l_{\text{bits}} \epsilon_{\text{two_ray_amp}} d_{CH_k, BS}^4, & \text{if } d_{CH_k, BS} \geq d_0 \end{cases}
 \tag{27}$$

where $d_{CH_k, BS}$ represents the separation between CH k and BS. $E(CH_k, BS)$ represents the direct communication between CH and BS.

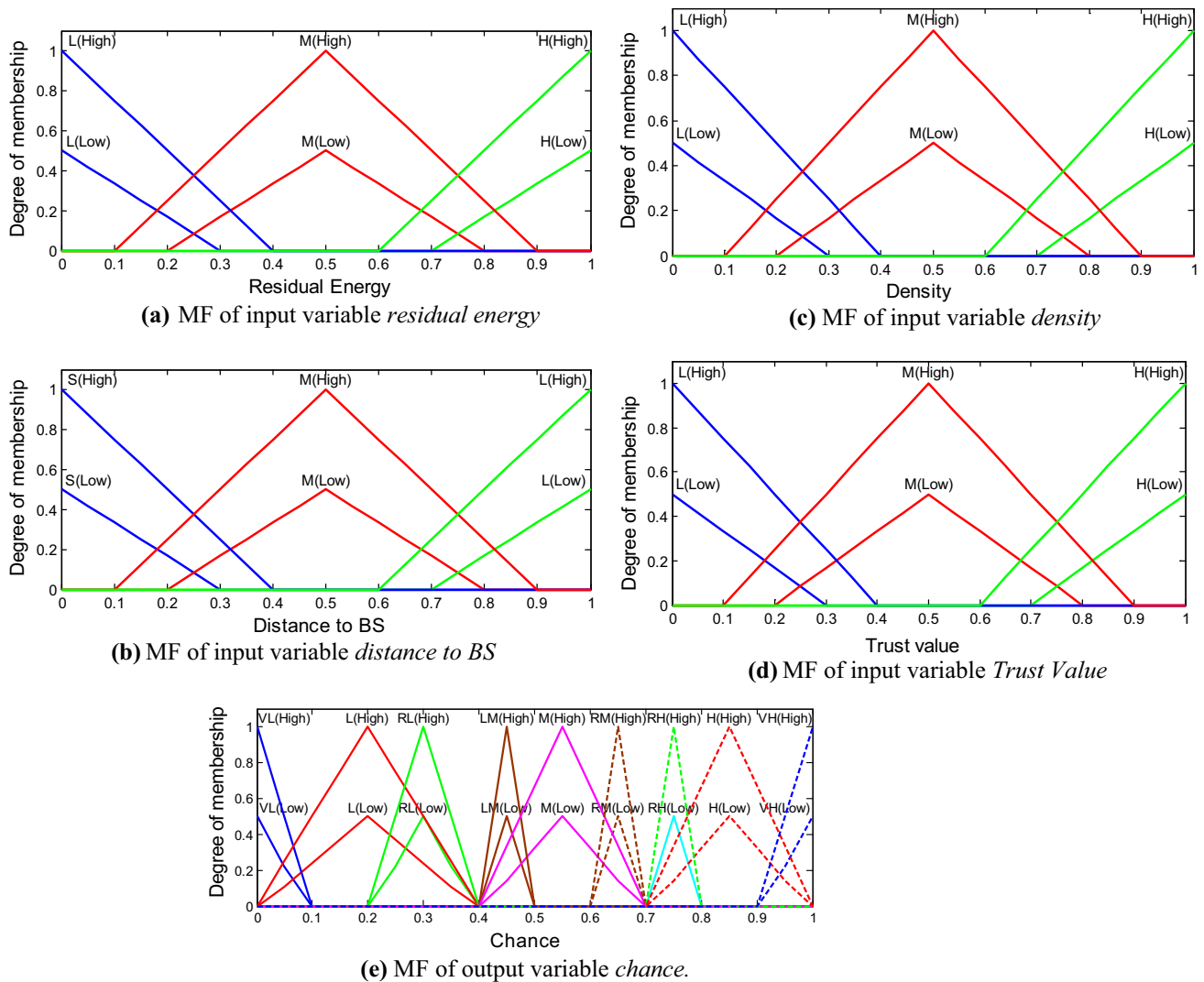


Fig. 7 Fuzzy sets for input and output variables

If the BS is far away from CH k , a relay CH m will be chosen as an intermediate node to transmit the packet. The total communication cost can be calculated as:

$$\begin{aligned}
 E(CH_k, CH_m, BS) &= E_{TX_{CH_k, CH_m}} + E_{RX} + E_{TX_{CH_m, BS}} \\
 &= 3l_{bits}E_{elec} + l_{bits}\epsilon_{fris_amp}d_{CH_k, CH_m}^2 \\
 &\quad + l_{bits}\epsilon_{fris_amp}d_{CH_m, BS}^2
 \end{aligned}
 \tag{28}$$

CHs compare $E(CH_k, BS)$ with $E(CH_k, CH_m, BS)$ and choose the low energy path for data transmission. Therefore, the inter-cluster transmission cost for CH k can be determined using:

$$E(CH_k) = \min(E(CH_k, BS), E(CH_k, CH_m, BS))
 \tag{29}$$

5 Simulation results

5.1 Simulation setup

This section evaluates the performance analysis of TEFCSRP and its competitive protocols. The simulations are conducted using MATLAB. In this simulation, for homogeneous setup 100 sensor nodes are initially scattered at random in a $100\text{ m} \times 100\text{ m}$ square region in between $(0, 0)$ and $(100, 100)$ having initial energy E_0 , with BS located at $(50, 50)$. Advanced and super nodes are set to 20% and 10% of total nodes having initial energy $2E_0$ and $3E_0$ respectively for heterogeneous setup. Initially, the value of each trusted node is set to 1. C_p is the percentage of malicious nodes that are introduced within the network. Simulation parameters are summarized in Table 2.

Table 1 Fuzzy Inference Rules

| S. No. | Residual energy | Distance to BS | Density | Trust value | Chance |
|--------|-----------------|----------------|---------|-------------|---------------|
| 1 | Low | Nearby | Low | Low | Rather low |
| 2 | Low | Nearby | Low | Medium | Low medium |
| 3 | Low | Nearby | Low | High | Rather low |
| 4 | Low | Nearby | Medium | Low | Low medium |
| 5 | Low | Nearby | Medium | Medium | Low medium |
| 6 | Low | Nearby | Medium | High | Low medium |
| 7 | Low | Nearby | High | Low | Medium |
| 8 | Low | Nearby | High | Medium | Rather medium |
| 9 | Low | Nearby | High | High | Rather high |
| 10 | Low | Average | Low | Low | Low |
| 11 | Low | Average | Low | Medium | Rather low |
| ... | ... | ... | ... | ... | ... |
| 18 | Low | Average | High | High | Rather medium |
| 19 | Low | Far | Low | Low | Very low |
| 20 | Low | Far | Low | Medium | Low |
| ... | ... | ... | ... | ... | ... |
| 27 | Low | Far | High | High | Medium |
| 28 | Medium | Nearby | Low | Low | Rather low |
| 29 | Medium | Nearby | Low | Medium | Low medium |
| 30 | Medium | Nearby | Low | High | Medium |
| 31 | Medium | Nearby | Medium | Low | Medium |
| 32 | Medium | Nearby | Medium | Medium | Rather medium |
| ... | ... | ... | ... | ... | ... |
| 61 | High | Nearby | High | Low | Rather high |
| 62 | High | Nearby | High | Medium | High |
| 63 | High | Nearby | High | High | Very high |
| 64 | High | Average | Low | Low | Low medium |
| ... | ... | ... | ... | ... | ... |
| 78 | High | Far | Medium | High | Rather medium |
| 79 | High | Far | High | Low | Medium |
| 80 | High | Far | High | Medium | Rather medium |
| 81 | High | Far | High | High | Rather high |

The interval between consecutive cluster reformations is referred to as a single round. The predetermined cluster number required by the existing algorithms for comparison is set to 5%. Results are averaged over 20 random deployments of WSNs. The lifetime of the WSN is calculated by the number of live nodes, which is evaluated at each round. The network lifetime and total amount of transferred data are measured by evaluating the number of rounds until the death of the last node. Performance of TEFCSR has been critically analyzed by comparing it with the existing state of the art routing protocols including LEACH, SEP-E, HCR, ERP, DRESEP, HSTERP, DETERP, and FESTERP. The parameters setting for simulated protocols are given in Table 3. We considered the metrics of network lifetime and total consumed energy to

Table 2 Network parameters

| Parameter | Value |
|---|------------------------------|
| Initial number of sensors | 100 |
| Initial energy of normal node, E_0 | 0.25 J, 0.5 J, 1 J |
| CH election probability | 0.05 |
| Radio electronics energy, $E_{Tx} = E_{Rx}$ | 50 nJ/bit |
| Energy for data-aggregation, E_{DA} | 5 nJ/bit |
| Free space fading coefficient, ϵ_{free_amp} | 100 pJ/bit/m ² |
| Multi-path fading coefficient, $\epsilon_{two_ray_amp}$ | 0.0013 pJ/bit/m ⁴ |
| Temperature range on the field | 0–200 °F |
| Hard threshold | 50 °F |
| Soft threshold | 2 °F |

evaluate the schemes and methods proposed by this research.

5.2 Analysis of simulation results

5.2.1 Trust-based security

Randomly few malicious SNs ($C_p = 0.1, 0.2, \text{ and } 0.3$) are deployed with in the area of interest. Malicious SNs have features like some of them have bad behaviors like packet drop, too large or too small amount of transmitted packet, and transmitting false information. In this section, the detection accuracy of malicious SN is analyzed first by taking the different levels of threshold ($R_0 = 0.1, 0.2, \text{ and } 0.3$). After that, we use improved threshold obtained in the previous result to examine the variation of average sending ratio, the variation of average consistency ratio, and the variation of average packet delivery ratio concerning the number of rounds, to detect whether the proposed approach can eliminate the malicious SNs successfully and also enhance the average sending ratio, the average consistency ratio, and average packet delivery ratio of the sensor network.

In Fig. 8, the horizontal axis represents the number of communication round and the vertical axis represents the percentage of different threshold values ($R_0 = 0.1, 0.2, \text{ and } 0.3$) to detect malicious SNs when first SN dies in the entire region. Various graphs are obtained by setting a different level of threshold. From Fig. 9 it is clear that all malicious SNs can be detected when threshold R_0 is 0.3. Hence, by setting the threshold level R_0 as 0.3, the malicious SNs can easily be detected under experimental environments. Initially, all SNs have the same trust value and malicious SNs are not eliminated, therefore, the average consistency ratio, the sending ratio, and average packet delivery ratio of the entire network are 1. As malicious SNs remain in the network having some abnormal behaviors, all the three trust factors mentioned above will decline slowly. As the number of rounds increases, the malicious SN will be recognized and is eliminated slowly and these bad behaviors will reduce approximately. In the next stage of the whole network, above mentioned three trust factors will

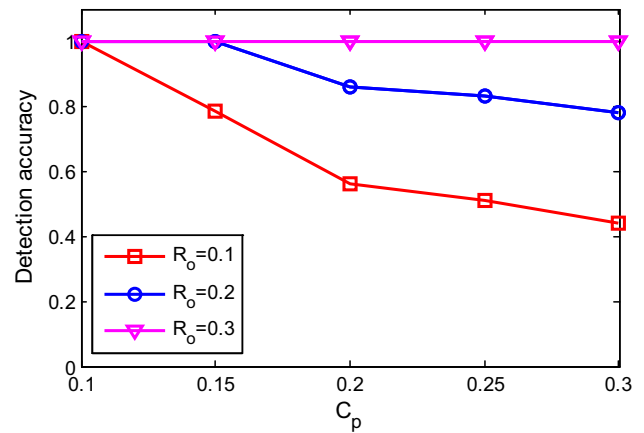


Fig. 8 The proportion of malicious nodes and detection accuracy

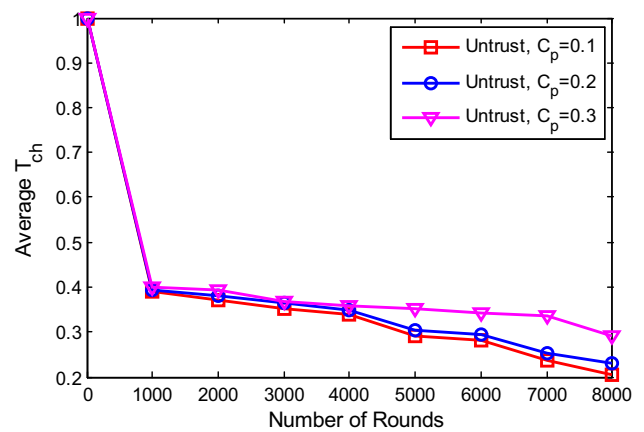


Fig. 9 The average value of a malicious node’s trust

be increased with the increased communication round as shown in Figs. 10, 11, and 12 respectively. The horizontal axis in Fig. 9 represents the number of communication round and the vertical axis stand for the average sending ratio. In the initial stages, the variation of the average sending ratio in the sensor network increases with increasing communication round no matter how many malicious SNs are present so, at the first stage, the network suffers degradation. As the rate of malicious SNs increases, the rate of failure goes faster in the down phase.

Table 3 Parameters setting for simulated algorithms

| Protocol | Parameters |
|----------|--|
| HCR | $NP = 30; D = 100; G_{max} = 200; P_m = 0.03; P_c = 0.7$ |
| ERP | $NP = 30; D = 100; G_{max} = 200; P_m = 0.03; P_c = 0.7$ |
| SAERP | $NP = 30; D = 100; G_{max} = 200; P_m = 0.03; P_c = 0.7$ |
| DESTERP | $NP = 30; D = 100; G_{max} = 200; CR = 0.5$ |
| HSSTERP | $HMS = 30; D = 100; NI = 200; HMCR = 0.7; PAR = 0.1$ |
| FESTERP | $NP = 30; D = 100; G_{max} = 100; p_{ds} = \text{Dynamic and linearly decreasing}$ |
| TEFCSR | $NP = 30; D = 100; G_{max} = 200$ |

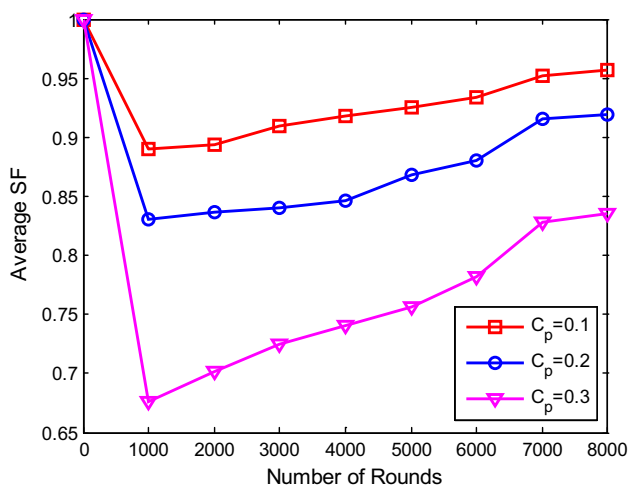


Fig. 10 The average sending ratio

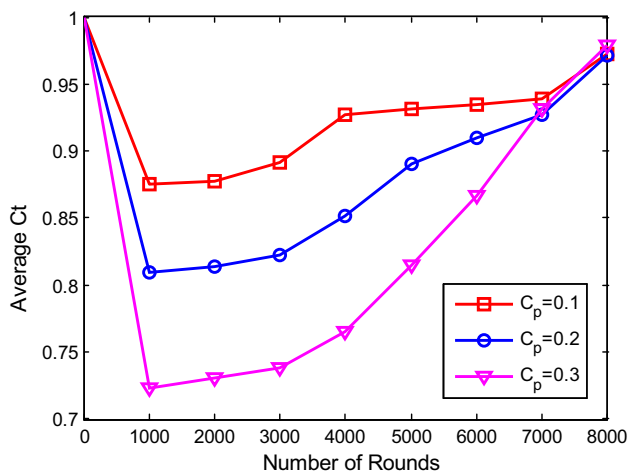


Fig. 11 The average consistency ratio

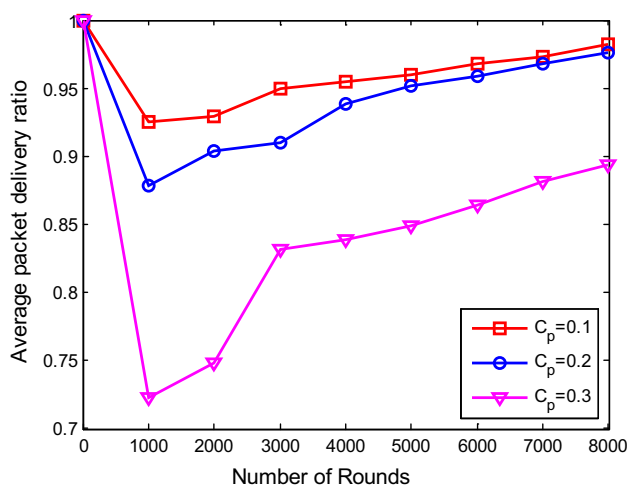


Fig. 12 Average packet delivery ratio

In Fig. 11 horizontal axis represents the number of rounds and the vertical axis represents the average consistency ratio and in Fig. 12 horizontal axis represents the average packet delivery ratio and the horizontal axis represents communication round. The variation in the consistency ratio and average packet delivery ratio is equal to that of variation in the average sending ratio. Initially, they undergo degradation and then increase accordingly, irrespective of malicious SN quantity. They also have the feature such as the rate of failure in the down phase goes faster when the amount of malicious SNs is greater. On comparing the results of Figs. 10, 11, and 12 it is clear that firstly value of the trust factor decreases and then increases with an increase in the communication round. Sending factor's change is slow, and inconsistency factor and packet loss factors changes are approximately larger because, in each round, one more or one less packet is allowed by malicious SNs to send than normal node. Sending rate variation of SN's is not very large, so the variation of the transmitting factor is slow.

5.2.2 Energy consumption and network lifetime

Figure 13 shows the alive nodes over the communication rounds for a homogeneous setup. TEFCSR improves the network lifespan, as it considers the remaining energy and trust value of SNs for the CH election. The improvements achieved by the TEFCSR scheme point to the ability to balance the energy through the nodes. In TEFCSR, the node with the higher remaining energy, nearer to BS, higher density, and higher trust value has the best chance to become the CH. This improved network lifetime is a result of a better selection of the CHs and interchanging the load over the nodes in a more balanced approach.

Also as shown in Tables 4, 5, and 6 for homogeneous setup for $E_0 = 0.25$ J, 0.5 J, and 1 J respectively, the improvement of network lifetime is for FND and also for HND and LND, respectively. This sequence is evidence of the strength of collaboration between different sensors to take loads of CH operations. Thus, most of the nodes operate together for the longest possible duration and then almost die together. In other words, they tend to die in groups rather than individually. This is contrary to the case of less balancing of remaining energy between different sensors.

Referring to Fig. 13, we see that the line representing the live nodes of TEFCSR takes the form of a step function with a sudden drop, while the line representing the live nodes of LEACH-like protocols tend to decrease gradually, and the drops are larger in the earlier period of network lifetime. Thus, most of the nodes in LEACH die through the earlier period of network lifetime. In contrast, the TEFCSR overcomes this shortcoming and always

Fig. 13 Number of survival nodes changes with rounds updating for homogeneous setup for $E_0 = 1$ J

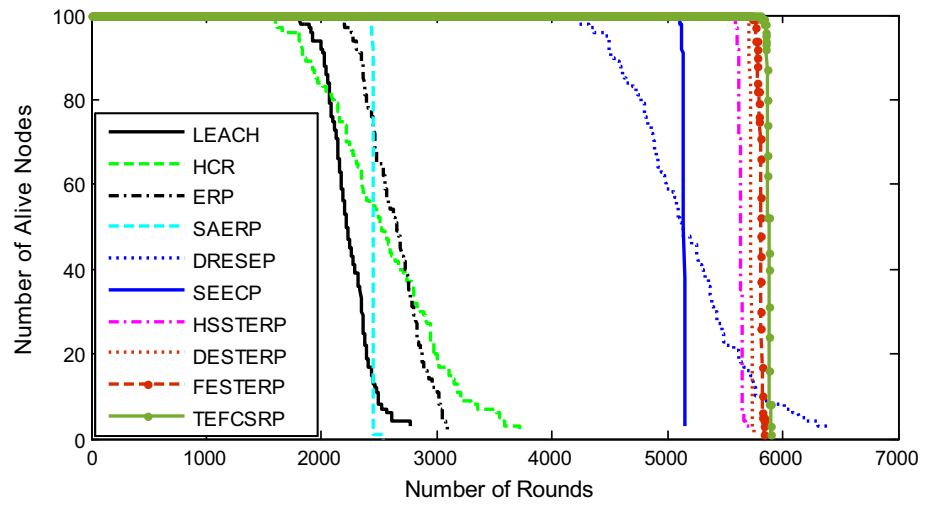


Table 4 Comparison of network lifetime for homogeneous setup for $E_0 = 0.25$ J

| % Dead nodes | LEACH | HCR | ERP | SAERP | DRESEP | SEECP | HSSTERP | DESTERP | FESTERP | TEFCSR |
|--------------|-------|-------|-------|-------|---------------|--------|---------|---------|---------|---------------|
| 1 (FND) | 461.4 | 441.2 | 510.2 | 610.2 | 782.6 | 1265.9 | 1110.8 | 1370.4 | 1374.5 | 1388.6 |
| 10 | 488.5 | 498.1 | 588.9 | 611.5 | 1133.5 | 1282.1 | 1137.2 | 1393.1 | 1397.8 | 1411.8 |
| 20 | 506.2 | 516.3 | 608.1 | 611.9 | 1233.1 | 1287.8 | 1146.5 | 1403.4 | 1405.5 | 1421.9 |
| 30 | 515.5 | 537.2 | 627.9 | 612.1 | 1282.1 | 1295.2 | 1155.7 | 1409.1 | 1414.8 | 1427.7 |
| 40 | 523.2 | 561.4 | 641.2 | 613 | 1339.8 | 1301.8 | 1158.9 | 1414.9 | 1420.7 | 1433.5 |
| 50 (HND) | 568 | 598.1 | 648.3 | 613.3 | 1395.7 | 1306.3 | 1174.1 | 1416.8 | 1422.6 | 1435.6 |
| 60 | 588.9 | 607.9 | 658.9 | 613.6 | 1419.8 | 1309.1 | 1182.3 | 1419.1 | 1430.5 | 1437.8 |
| 70 | 588.7 | 617.8 | 674.1 | 613.8 | 1443.1 | 1309.2 | 1189.8 | 1424.8 | 1432.6 | 1444.7 |
| 80 | 607.2 | 625.3 | 701.4 | 614.5 | 1489.7 | 1310 | 1217.4 | 1427.1 | 1436.2 | 1445.9 |
| 90 | 630.7 | 636.5 | 732.7 | 615.1 | 1539.8 | 1310.3 | 1222.2 | 1429.6 | 1442.5 | 1448.3 |
| 100 (LND) | 745.2 | 848.2 | 784.2 | 617.2 | 1603.4 | 1316.2 | 1406.3 | 1432.2 | 1447.8 | 1451.3 |

Bold values indicate the best value with respect to others

Table 5 Comparison of network lifetime for homogeneous setup for $E_0 = 0.5$ J

| % Dead nodes | LEACH | HCR | ERP | SAERP | DRESEP | SEECP | HSSTERP | DESTERP | FESTERP | TEFCSR |
|--------------|--------|--------|--------|--------|---------------|--------|---------|---------|---------|---------------|
| 1 (FND) | 971.2 | 873.3 | 1037.9 | 1217.9 | 1562.8 | 2543.2 | 2399.7 | 2795.3 | 2836.6 | 2859.6 |
| 10 | 1006.8 | 1010.2 | 1159.1 | 1221.8 | 2262.4 | 2563.9 | 2421.3 | 2821.4 | 2885.3 | 2908.9 |
| 20 | 1040.1 | 1063.7 | 1199.3 | 1223.9 | 2463.9 | 2572.1 | 2441.6 | 2833.1 | 2895.9 | 2917.8 |
| 30 | 1061.2 | 1114.8 | 1238.2 | 1225.4 | 2562.4 | 2579.2 | 2453.9 | 2838.8 | 2899.9 | 2922.6 |
| 40 | 1074.1 | 1166.4 | 1268.2 | 1227.9 | 2682.6 | 2588.4 | 2458.3 | 2845.9 | 2908.7 | 2931.6 |
| 50 (HND) | 1169.6 | 1228.6 | 1292.9 | 1229.4 | 2782 | 2592.3 | 2460.3 | 2848.9 | 2910.8 | 2933.8 |
| 60 | 1206.9 | 1265.9 | 1319.8 | 1230.6 | 2841.9 | 2594.8 | 2470.1 | 2850.5 | 2918.1 | 2940.9 |
| 70 | 1265.6 | 1306.2 | 1360.1 | 1231.2 | 2882.4 | 2595.5 | 2485.1 | 2857.8 | 2920.6 | 2943.8 |
| 80 | 1316.5 | 1353.6 | 1410 | 1232.8 | 2982.9 | 2596 | 2487.3 | 2860 | 2921.6 | 2944.6 |
| 90 | 1369.2 | 1411.9 | 1477.3 | 1234.3 | 3082.5 | 2597.6 | 2503.7 | 2863.4 | 2927.1 | 2948.3 |
| 100 (LND) | 1672.8 | 1741.3 | 1609.8 | 1243.6 | 3202.3 | 2604.4 | 2553.3 | 2866.3 | 2931.1 | 2952.5 |

Bold values indicate the best value with respect to others

Table 6 Comparison of network lifetime for homogeneous setup for $E_0 = 1$ J

| % Dead nodes | LEACH | HCR | ERP | SAERP | DRESEP | SEEC | HSSTERP | DESTERP | FESTERP | TEFCSR |
|--------------|--------|--------|--------|--------|---------------|--------|---------|---------|---------|---------------|
| 1 (FND) | 1805.1 | 1726.3 | 2113.2 | 2437.7 | 4101.4 | 5109.4 | 5635.7 | 5699.4 | 5742.7 | 5816.2 |
| 10 | 2022.7 | 2048.5 | 2276.1 | 2443.6 | 4504.1 | 5131.2 | 5655.1 | 5712.5 | 5780.8 | 5873.5 |
| 20 | 2069.3 | 2189.6 | 2364.4 | 2445.6 | 4769.9 | 5134.5 | 5666.5 | 5718.5 | 5794.6 | 5875.3 |
| 30 | 2141 | 2315.3 | 2438 | 2447.1 | 4881.4 | 5135.6 | 5672.7 | 5722.4 | 5807.3 | 5879.3 |
| 40 | 2169.4 | 2420.2 | 2509.5 | 2448.5 | 4983.5 | 5141.8 | 5675.8 | 5723.1 | 5809.4 | 5886.7 |
| 50 (HND) | 2215.2 | 2524.6 | 2580 | 2449.6 | 5126.7 | 5143.3 | 5680.3 | 5726 | 5812.6 | 5888.4 |
| 60 | 2280.1 | 2629.9 | 2648.9 | 2450.7 | 5294.2 | 5144.5 | 5682.4 | 5726.6 | 5816.1 | 5890.3 |
| 70 | 2346.3 | 2752.2 | 2744.8 | 2451.4 | 5395.4 | 5146.9 | 5687.8 | 5729.3 | 5816.8 | 5897.5 |
| 80 | 2394.8 | 2916.8 | 2837.3 | 2452 | 5621.1 | 5149.1 | 5689.6 | 5732.8 | 5819.8 | 5898.3 |
| 90 | 2485.6 | 3107.1 | 2983.3 | 2453.3 | 5770.7 | 5151.5 | 5693.1 | 5736.2 | 5822.6 | 5901.4 |
| 100 (LND) | 2763.5 | 3574.3 | 3305.9 | 2455.2 | 6402.2 | 5154.8 | 5715.2 | 5737.9 | 5836.3 | 5905.6 |

Bold values indicate the best value with respect to others

works to prolong sensors lifetime when it is worthy to keep them alive.

Figure 14 depicts the percentage of total remaining energy for the rounds before LND. The figure shows that the TEFCSR conserves more total energy than the other schemes. It also shows that TEFCSR tends to consume energy gradually per round in equal amounts.

5.2.3 Reduction of blind spot problem

At the beginning of the operation, the network shows good performance by capturing the desired number of events per unit of time. However, towards the end, the network experiences the death of its constituent nodes which results in an inability to capture events in certain places. This scenario is described as a blind spot problem that results in the degradation of network performance in terms of the number of captured events per unit of time. The sole reason for this problem lies in the fact that a few nodes die out

quickly in the network due to unbalanced energy consumption. The more unbalanced the energy consumption is, the more quickly nodes start dying in the network. On the other hand, the majority of the recent clustering approaches focus on enlarging the network lifetime without incorporating a robust energy-balancing technique. This results in a long duration between FND and LND, i.e., the network suffers from event capturing inability or blind spot problem for a long time. To reduce this sufferance, this paper focuses on adopting a mechanism that forces a balanced energy consumption in each round for both nodes and clusters.

Figure 13 also shows the duration of the blind spot problem in the network with the proposed approach and with the competitive approaches. From the figure, it is seen that the round count for FND in TEFCSR is higher in comparison to LEACH, SEP-E, HCR, ERP, DRESEP, HSTERP, DETERP, and FESTERP. On the contrary, the network experienced a difference in FND and LND is less

Fig. 14 The residual energy of the network changes over time for homogeneous setup

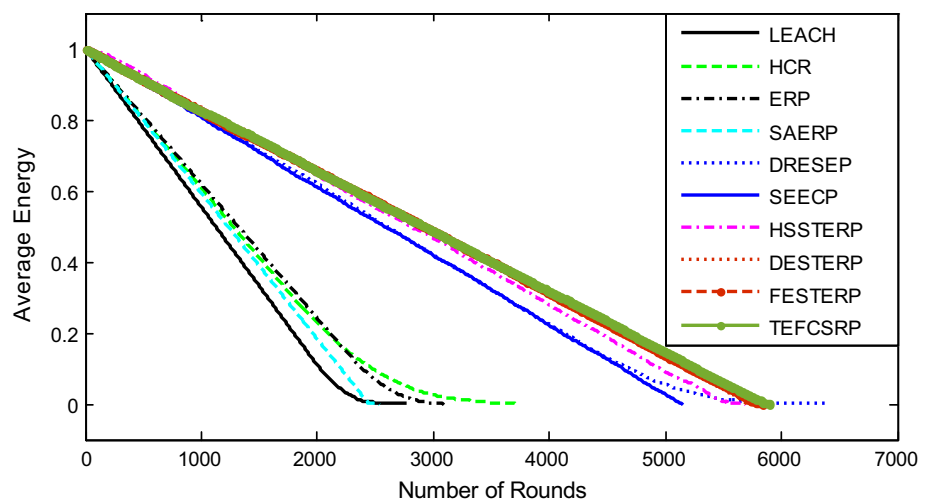


Fig. 15 No. of survival nodes changes with rounds updating for heterogeneous setup for $E_0 = 1 \text{ J}$

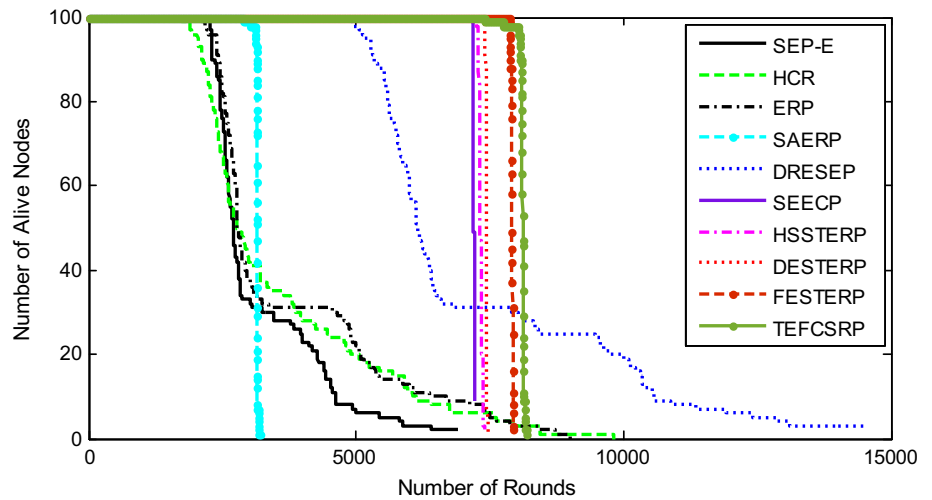


Fig. 16 The residual energy of the network changes over time for heterogeneous setup

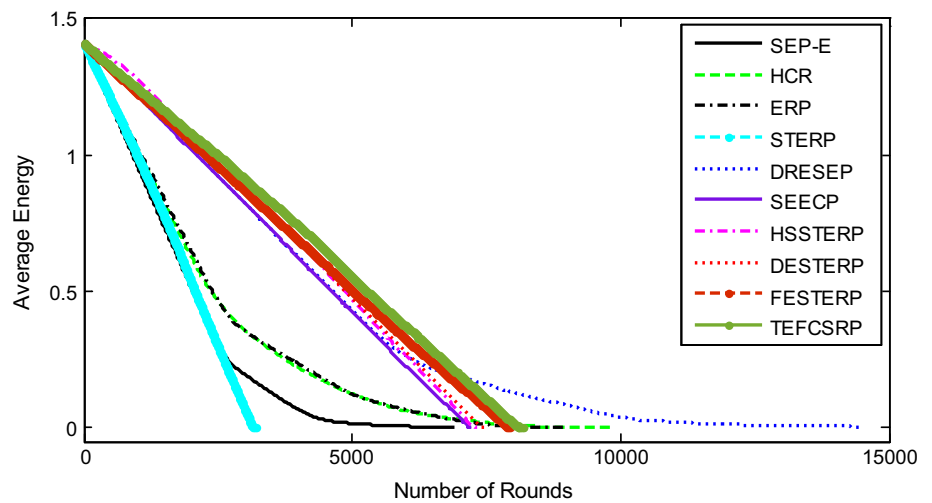


Table 7 Comparison of network lifetime for heterogeneous setup for $E_0 = 0.25 \text{ J}$

| % Dead nodes | SEP-E | HCR | ERP | SAERP | DRESEP | SEECP | HSSTERP | DESTERP | FESTERP | TEFCSRP |
|--------------|--------|--------|--------|-------|---------------|--------|---------|---------|---------------|---------------|
| 1 (FND) | 570.3 | 380.4 | 510.6 | 747.9 | 1214.7 | 1794.9 | 1823.6 | 1847.5 | 1973.7 | 1967.8 |
| 10 | 581.4 | 490.2 | 573.3 | 793.2 | 1336.2 | 1798 | 1831.8 | 1857.8 | 1976.8 | 2011.4 |
| 20 | 616.1 | 558.5 | 608.5 | 794.8 | 1402.3 | 1801.3 | 1834.3 | 1860 | 1980.2 | 2014.3 |
| 30 | 641.2 | 615.6 | 647.6 | 795.6 | 1450.6 | 1801.9 | 1836.7 | 1861.3 | 1980.8 | 2015.5 |
| 40 | 655.3 | 702.3 | 670.3 | 796.1 | 1502.2 | 1802.5 | 1837.8 | 1862.8 | 1982.4 | 2016.8 |
| 50 (HND) | 678.5 | 758.2 | 693.7 | 796.8 | 1538 | 1803.1 | 1839.1 | 1862.9 | 1982.9 | 2017.9 |
| 60 | 700.6 | 812.4 | 777.1 | 797.2 | 1597.4 | 1804.5 | 1844.7 | 1863 | 1985.5 | 2019.5 |
| 70 | 816.2 | 1010.7 | 1123.5 | 797.6 | 1983.6 | 1805.2 | 1846.4 | 1864.4 | 1986.1 | 2020.7 |
| 80 | 1071.3 | 1201.6 | 1280.3 | 798.4 | 2443.1 | 1805.6 | 1847.9 | 1864.8 | 1986.8 | 2021.3 |
| 90 | 1154.6 | 1549.3 | 1573.4 | 799.3 | 2645.7 | 1805.7 | 1849.3 | 1865.3 | 1987.7 | 2021.9 |
| 100 (LND) | 1471.4 | 1845 | 2110.7 | 802.7 | 3635.4 | 1805.8 | 1851.4 | 1867.4 | 1988.3 | 2022.4 |

Bold values indicate the best value with respect to others

Table 8 Comparison of network lifetime for heterogeneous setup for $E_0 = 0.5$ J

| % Dead nodes | SEP-E | HCR | ERP | SAERP | DRESEP | SEEC | HSSTERP | DESTERP | FESTERP | TEFCSR |
|--------------|--------|--------|--------|--------|---------------|--------|---------|---------|---------------|---------------|
| 1 (FND) | 1152.4 | 772.1 | 1030.2 | 1602.1 | 2442.1 | 3601.6 | 3657.9 | 3703.7 | 3956.5 | 3931.7 |
| 10 | 1172.9 | 992.3 | 1154.8 | 1623.4 | 2685.2 | 3609.4 | 3673.9 | 3724.5 | 3962.8 | 4018.5 |
| 20 | 1245.1 | 1127.6 | 1225.6 | 1626.1 | 2817.4 | 3613.7 | 3678.8 | 3729.1 | 3969.1 | 4024.2 |
| 30 | 1295.2 | 1242.8 | 1303.2 | 1627.7 | 2913.8 | 3615.2 | 3683.4 | 3731.4 | 3970.4 | 4025.6 |
| 40 | 1322.6 | 1415.9 | 1340.1 | 1629.6 | 3016.7 | 3616.1 | 3685.7 | 3734.5 | 3974 | 4029.3 |
| 50 (HND) | 1369.2 | 1527.6 | 1396.2 | 1630.8 | 3089.2 | 3617.6 | 3688.5 | 3734.4 | 3975.3 | 4030.9 |
| 60 | 1414.1 | 1636.1 | 1563.1 | 1630.9 | 3207.1 | 3620.5 | 3699.3 | 3734.9 | 3980.3 | 4035.8 |
| 70 | 1644.7 | 2032.8 | 2255.6 | 1632.2 | 3979.5 | 3621.9 | 3702.8 | 3737.7 | 3981.4 | 4036.6 |
| 80 | 2154.8 | 2415.2 | 2568.2 | 1633.5 | 4898.8 | 3622.3 | 3705.7 | 3738.4 | 3982.1 | 4037.9 |
| 90 | 2321.7 | 3110.4 | 3155.2 | 1634.9 | 5303.6 | 3622.8 | 3709.3 | 3739.3 | 3984.7 | 4040.6 |
| 100 (LND) | 2954.8 | 3701.2 | 4229.4 | 1640.8 | 7283.2 | 3622.9 | 3713.2 | 3743.7 | 3985.9 | 4041.3 |

Bold values indicate the best value with respect to others

Table 9 Comparison of network lifetime for heterogeneous setup for $E_0 = 1$ J

| % Dead nodes | SEP-E | HCR | ERP | SAERP | DRESEP | SEEC | HSSTERP | DESTERP | FESTERP | TEFCSR |
|--------------|--------|--------|--------|--------|-----------------|--------|---------|---------|---------------|---------------|
| 1 (FND) | 2269.5 | 1509.9 | 2034.5 | 2887.6 | 4846.6 | 7168.6 | 7284.6 | 7380.7 | 7878.4 | 7425.4 |
| 10 | 2309.4 | 1950 | 2284.5 | 3135.5 | 5331.7 | 7184.1 | 7317.5 | 7422.6 | 7890.9 | 8093.2 |
| 20 | 2452.5 | 2220.9 | 2425.9 | 3141.9 | 5597.1 | 7192.7 | 7326.7 | 7431.3 | 7903.6 | 8109.7 |
| 30 | 2552.6 | 2451.6 | 2581.7 | 3145.4 | 5789.5 | 7195.5 | 7336.2 | 7435.9 | 7906.3 | 8118.3 |
| 40 | 2607.8 | 2798.2 | 2673.2 | 3147.9 | 5995.4 | 7198.1 | 7340.5 | 7441.9 | 7913.4 | 8123.6 |
| 50 (HND) | 2701.3 | 3021 | 2766.8 | 3150.6 | 6140.1 | 7200.4 | 7345.9 | 7442.6 | 7914.5 | 8127.5 |
| 60 | 2790.5 | 3238 | 3100.6 | 3152.8 | 6376 | 7206 | 7368.2 | 7442.9 | 7925.9 | 8132.9 |
| 70 | 3251.9 | 4032.4 | 4485.9 | 3154.3 | 7920.6 | 7208.9 | 7374.4 | 7448.4 | 7928.2 | 8135.4 |
| 80 | 4272.1 | 4796.2 | 5111.7 | 3157.7 | 9759.3 | 7210.3 | 7381.1 | 7449.8 | 7929.5 | 8141.6 |
| 90 | 4605.5 | 6186.4 | 6284.8 | 3161.8 | 10,568.9 | 7210.6 | 7387.6 | 7451.9 | 7934.7 | 8149.2 |
| 100 (LND) | 5872.8 | 7368.7 | 8433.5 | 3174.3 | 14,528 | 7211.4 | 7395.8 | 7460.5 | 7937.1 | 8179.7 |

Bold values indicate the best value with respect to others

than allowed the blind spot problem to persist for a smaller number of rounds. Thus, the proposed approach reduces the blind spot problem in the network.

The behavior of TEFCSR for heterogeneous setup is shown in Figs. 15 and 16, and the statistics are given in Tables 7, 8, and 9.

Table 10 presents the comparative analysis for FND, HND, and LND together with stability and instability periods of competitive algorithms for $E_0 = 1$ J. The Table shows there is significant progress in the stability period for TEFCSR.

5.2.4 Effect of node density

A comparative evaluation of TEFCSR is performed using a varying number of nodes in the network (from 100 to 500) as given in Tables 11 and 12, to illustrate and validate its behavior under different densities, sparse, moderate, or

dense. The comparison is based on the metric of energy balancing and network lifetime in terms of FND, HND, and LND. All nodes in the scenarios are randomly distributed over an area of 100×100 meters. The performance improvement of the proposed TEFCSR over competitive algorithms become noticeable as the network size increases, in that the proposed TEFCSR finds a more energy-efficient solution than others from the consideration of the optimum fuzzy-based energy consumption model for CH nodes.

6 Conclusion

FISs are the best choice for building effective clustering algorithms/techniques for energy-efficient routing protocols in WSN, due to its high ability to combine and effectively blending input parameters to produce proper

Table 10 Comparison of network lifespan of simulated algorithms along with stability and instability periods for $E_0 = 1$ J

| Setup no. | Protocol | FND | HND | LND | Stability period | Instability period |
|---------------------|----------|---------------|---------------|---------------|------------------|--------------------|
| Homogeneous setup | LEACH | 1805.1 | 2215.2 | 2763.5 | 1805.1 | 958.4 |
| | HCR | 1726 | 2524.6 | 3574.3 | 1726 | 1848.3 |
| | ERP | 2113.3 | 2580 | 3305.9 | 2113.3 | 1192.6 |
| | SAERP | 2437.7 | 2449.6 | 2455.2 | 2437.7 | 17.5 |
| | DRESEP | 4101.4 | 5126.7 | 6402.2 | 4101.4 | 2300.8 |
| | SEECP | 5109.4 | 5143.3 | 5154.8 | 5109.4 | 45.4 |
| | HSSTERP | 5635.7 | 5680.3 | 5715.2 | 5635.7 | 79.5 |
| | DESTERP | 5699.4 | 5726 | 5737.9 | 5699.4 | 38.5 |
| | FESTERP | 5742.7 | 5812.6 | 5836.3 | 5742.7 | 93.6 |
| | TEFCSR | 5816.2 | 5888.4 | 5905.6 | 5816.2 | 89.4 |
| Heterogeneous setup | SEP-E | 2269.5 | 2701.3 | 5872.8 | 2269.5 | 3603.3 |
| | HCR | 1509.9 | 3021 | 7368.7 | 1509.9 | 5858.8 |
| | ERP | 2034.5 | 2766.8 | 8433.5 | 2034.5 | 6399 |
| | SAERP | 2887.6 | 3150.6 | 3174.3 | 2887.6 | 286.7 |
| | DRESEP | 4846.2 | 6140.1 | 14,528 | 4846.2 | 9681.8 |
| | SEECP | 7168.6 | 7200.4 | 7211.4 | 7168.6 | 42.8 |
| | HSSTERP | 7284.6 | 7345.9 | 7395.8 | 7284.6 | 111.2 |
| | DESTERP | 7380.7 | 7442.6 | 7460.5 | 7380.7 | 79.8 |
| | FESTERP | 7878.4 | 7914.5 | 7937.1 | 7878.4 | 58.7 |
| | TEFCSR | 7425.4 | 8127.5 | 8179.7 | 7425.4 | 754.3 |

Bold values indicate the best value with respect to others

Table 11 Effect of node density on the performance of TEFCSR for homogeneous setup

| Protocol | 100 | 200 | 300 | 400 | 500 |
|----------|--------|--------|--------|--------|----------|
| LEACH | 2763.5 | 3040.1 | 3316.3 | 3592.6 | 3847.9 |
| HCR | 3574.3 | 4003.4 | 4217.8 | 4431.9 | 4682.5 |
| ERP | 3305.9 | 3570.3 | 3712.9 | 3824.8 | 4015.6 |
| SAERP | 2455.2 | 2529.1 | 2604.5 | 2669.4 | 2736.9 |
| DRESEP | 6402.2 | 7490.5 | 8015.1 | 8976.8 | 10,143.7 |
| SEECP | 5154.8 | 5279.8 | 5385.6 | 5514.9 | 5625.2 |
| HSSTERP | 5715.2 | 5943.6 | 6062.8 | 6207.8 | 6332.2 |
| DESTERP | 5737.9 | 5967.2 | 6087.1 | 6231.1 | 6357.7 |
| FESTERP | 5836.3 | 6135.1 | 6338.6 | 6571.4 | 6789.5 |
| TEFCSR | 5905.6 | 6193.1 | 6419.4 | 6662.8 | 68,752.5 |

Table 12 Effect of node density on the performance of TEFCSR for heterogeneous setup

| Protocol | 100 | 200 | 300 | 400 | 500 |
|----------|--------|----------|----------|----------|----------|
| SEP-E | 5872.8 | 6343.2 | 6459.8 | 6694.7 | 6459.5 |
| HCR | 7368.7 | 7712.5 | 8232.6 | 8752.9 | 9359.9 |
| ERP | 8433.5 | 8971 | 9329.9 | 9609.8 | 10,090.2 |
| SAERP | 3174.3 | 3269.5 | 3367.6 | 3451.8 | 3607.1 |
| DRESEP | 14,528 | 16,997.9 | 18,187.7 | 20,370.3 | 23,018.4 |
| SEECP | 7211.4 | 7788.5 | 7943.8 | 8135.2 | 8297.8 |
| HSSTERP | 7395.8 | 7691.8 | 7845.3 | 8033.6 | 8195.2 |
| DESTERP | 7460.5 | 7759.1 | 7913.9 | 8103.7 | 8266.1 |
| FESTERP | 7937.1 | 8341.5 | 8616.3 | 8935.5 | 9229.9 |
| TEFCSR | 8179.7 | 8582.7 | 8859.3 | 9186.4 | 9527.3 |

decisions about CH selections. To achieve the best possible results of energy-efficient routing protocols in WSN, it is recommended to utilize every parameter affecting the energy efficiency of the WSN routing protocol. Furthermore, it is recommended to integrate them in a way that reflects the extent to which each affects the energy efficiency of the WSN. In this work, we introduced the TEFCSR clustering technique for energy-efficient routing protocols using fuzzy type-2 based on CS to perform the CH election. This fuzzy logic utilizes four parameters to determine the strength of each sensor’s chance to be a CH.

These parameters are the remaining energy of the given sensor node, the distance of sensor nodes from the BS, density of other surrounding sensor nodes around the candidate CH, and trust values of nodes to transmit the data to next hop. To control the distribution of CHs over the WSN area, a condition is set by forcing minimum separation distance between CHs to guarantee their even distribution. A threshold-based data transmission algorithm is used in inter-cluster communication and a multi-hop routing scheme in which only CH nearer to BS transmits its

information directly to BS to decrease dissipated energy from CHs far away from BS.

TEFCSR was assessed relatively by simulating different WSN scenarios against LEACH, SEP-E, HCR, ERP, DRESEP, HSTERP, DETERP, and FESTERP for the metrics of total energy consumption, energy balancing, and network lifetime in terms of FND, HND, and LND. TEFCSR significantly outperforms these approaches in trust management metrics, network lifetime, and energy consumption effectiveness.

References

- Akyildiz, I. F., Su, W., Sankarasubramanian, Y., & Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4), 393–422.
- Anisi, M. H., Abdul-Salaam, G., Idris, M. Y. I., Wahab, A. W. A., & Ahmedy, I. (2017). Energy harvesting and battery power based routing in wireless sensor networks. *Wireless Networks*, 23, 249–266.
- Pantazis, N. A., Nikolidakis, S. A., & Vergados, D. D. (2013). Energy-efficient routing protocols in wireless sensor networks: A survey. *IEEE Communications, Surveys & Tutorials*, 15(2), 551–591.
- Halawani, S., & Khan, A. W. (2010). Sensors lifetime enhancement techniques in wireless sensor networks—A survey. *Journal of Computing*, 2(5), 34–47.
- Memon, I., & et al. (2019). Smart intelligent system for mobile travelers based on fuzzy logic in IoT communication technology. In *International conference on intelligent technologies and applications* (pp 22–31).
- Memon, I., & Mirza, H. T. (2018). MADPTM: Mix zones and dynamic pseudonym trust management system for location privacy. *International Journal of Communication Systems*, 31(17), e3795.
- Memon, I. (2015). A secure and efficient communication scheme with authenticated key establishment protocol for road networks. *Wireless Personal Communications*, 85(3), 1167–1191.
- Purkar, S. V., & Deshpande, R. S. (2018). Energy efficient clustering protocol to enhance performance of heterogeneous wireless sensor network: EECPEP-HWSN. *Journal of Computer Networks and Communications*. <https://doi.org/10.1155/2018/2078627>.
- Liang, Q., & Mendel, J. M. (2000). Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems*, 8(5), 535–550.
- Hwang, J. H., Kwak, H. J., & Park, G. T. (2011). Adaptive interval type-2 fuzzy sliding mode control for unknown chaotic system. *Nonlinear Dynamics*, 63(3), 491–502.
- Salgotra, R., Singh, U., & Saha, S. (2018). New cuckoo search algorithms with enhanced exploration and exploitation properties. *Expert Systems with Applications*, 95, 384–420.
- Heinzelman, W. B., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of 33rd annual Hawaii international conference on system sciences (HICSS-33)*. IEEE (p. 223). <https://doi.org/10.1109/hicss.2000.926982>.
- Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power-efficient gathering in sensor information systems. In *Proceedings of the IEEE AEROSPACE CONFERENCE, Big Sky, MT, USA, 9–16 March 2002* (vol. 3, pp. 1125–1130).
- Younis, O., & Fahmy, S. (2004). HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, 2004(3), 366–379.
- Li, C., Ye, M., Chen, G., & Wu, J. (2005). An energy-efficient unequal clustering mechanism for wireless sensor networks. In *Proceedings of the IEEE international conference on mobile adhoc and sensor systems, Washington, DC, USA, 7–10 November 2005* (pp. 598–604).
- Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. *IEEE Transactions on Wireless Communications*, 1(4), 660–670. <https://doi.org/10.1109/TWC.2002.804190>.
- Tripathi, M., Battula, R. B., Gaur, M. S., & Laxmi, V. (2013). Energy efficient clustered routing for wireless sensor network. In *Proceedings of the 2013 IEEE 9th international conference on mobile ad hoc and sensor networks, Dalian, China, 11–13 December 2013* (pp. 330–335).
- Mechta, D., Harous, S., Alem, I., & Khebbab, D. (2014). LEACH-CKM: Low energy adaptive clustering hierarchy protocol with K-means and MTE. In *Proceedings of the 2014 10th international conference on innovations in information technology (IIT), Al Ain, UAE, 9–11 November 2014* (pp. 99–103).
- Manjeshwar, A., & Agrawal, D. P. (2001). TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. In *15th international parallel and distributed processing symposium (IPDPS'01) Workshops, USA, California* (pp. 2009–2015).
- Manjeshwar, A., & Agrawal, D. P. (2002). APTTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In *International parallel and distributed processing symposium, Florida* (pp. 195–202).
- Aderohunmu, F. A., & Deng, J. D. (2009). *An enhanced stable election protocol (E-SEP) for clustered heterogeneous WSN*, Department of Information Science, University of Otago, Dunedin 9054, New Zealand.
- Smaragdakis, G., Matta, I., & Bestavros, A. (2004). SEP: A stable election protocol for clustered heterogeneous wireless sensor networks. In *Proceedings of international workshop on SANPA*. <http://open.bu.edu/xmlui/bitstream/handle/2144/1548/2004-022-sep.pdf?sequence=1>.
- Kang, S. H., & Nguyen, T. (2012). Distance based thresholds for cluster head selection in wireless sensor networks. *IEEE Communications Letters*, 16(9), 1396–1399. <https://doi.org/10.1109/LCOMM.2012.073112.120450>.
- Mahajan, S., Malhotra, J., & Sharma, S. (2014). An energy balanced QoS based cluster head selection strategy for WSN. *Egyptian Informatics Journal*, 15(3), 189–199.
- Tarhani, M., Kaviani, Y. S., & Siavoshi, S. (2014). SEECH: Scalable energy efficient clustering hierarchy protocol in wireless sensor networks. *IEEE Sensors Journal*, 14(11), 3944–3954. <https://doi.org/10.1109/JSEN.2014.2358567>.
- Mittal, N., & Singh, U. (2015). Distance-based residual energy-efficient stable election protocol for WSNs. *Arabian Journal of Science and Engineering*, 40(6), 1637–1646. <https://doi.org/10.1007/s13369-015-1641-x>.
- Mittal, N., Singh, U., & Sohi, B. S. (2017). A stable energy efficient clustering protocol for wireless sensor networks. *Wireless Networks*, 23(6), 1809–1821. <https://doi.org/10.1007/s11276-016-1255-6>.
- Adnan, M. A., Razzaque, M. A., Ahmed, I., & Isnin, I. F. (2014). Bio-mimic optimization strategies in wireless sensor networks: A survey. *Sensors*, 14, 299–345. <https://doi.org/10.3390/s140100299>.

29. Hussain, S., & Matin, A. W. (2006). Hierarchical cluster-based routing in wireless sensor networks. In *IEEE/ACM international conference on information processing in sensor networks, IPSN*.
30. Attea, B. A., & Khalil, E. A. (2012). A new evolutionary based routing protocol for clustered heterogeneous wireless sensor networks. *Applied Soft Computing*, 12, 1950–1957. <https://doi.org/10.1016/j.asoc.2011.04.007>.
31. Khalil, E. A., & Attea, B. A. (2011). Energy-aware evolutionary routing protocol for dynamic clustering of wireless sensor networks. *Swarm and Evolutionary Computation*. <https://doi.org/10.1016/j.swevo.2011.06.004>.
32. Khalil, E. A., & Attea, B. A. (2013). Stable-aware evolutionary routing protocol for wireless sensor networks. *Wireless Personal Communications*, 69(4), 1799–1817.
33. Mittal, N., Singh, U., & Sohi, B. S. (2017). A novel energy efficient stable clustering approach for wireless sensor networks. *Wireless Personal Communications*, 95, 2947–2971.
34. Mittal, N., Singh, U., & Sohi, B. S. (2017). Harmony search algorithm based threshold-sensitive energy-efficient clustering protocols for WSNs. *Ad Hoc & Sensor Wireless Networks*, 36, 149–174.
35. Mittal, N., Singh, U., Salgotra, R., & Sohi, B. S. (2018). A Boolean spider monkey optimization based energy efficient clustering approach for WSNs. *Wireless Networks*, 24(6), 2093–2109.
36. Mittal, N., Singh, U., Sohi, B. S. (2018). An energy aware cluster-based stable protocol for wireless sensor networks. In *Neural computing and applications (NCAA)* (pp 1–18).
37. Mittal N., Singh U., Salgotra R., & Bansal M. (2019) An energy efficient stable clustering approach using fuzzy enhanced flower pollination algorithm for WSNs. *Neural computing and applications (NCAA)* (pp 1–25). <https://doi.org/10.1007/s00521-019-04251-4>.
38. Mittal N., Singh U., Sohi B. S. (2016). Modified grey wolf optimizer for global engineering optimization. *Applied Computational Intelligence and Soft Computing* 1–13.
39. Kim J. M., Park S. H., Han Y. J., Chung T. M. (2008). CHEF: Cluster head election mechanism using fuzzy logic in wireless sensor networks. In *10th international conference on advanced communication technology* (Vol. 1, pp. 654–659).
40. Ran, G., Zhang, H., & Gong, S. (2010). Improving on LEACH protocol of wireless sensor networks using fuzzy logic. *Journal of Information & Computational Science*, 7, 767–775.
41. Lee, J. S., & Cheng, W. L. (2012). Fuzzy-logic-based clustering approach for wireless sensor networks using energy predication. *IEEE Sensors Journal*, 12, 2891–2897.
42. Bagci, H., & Yazici, A. (2013). An energy aware fuzzy approach to unequal clustering in wireless sensor networks. *Applied Soft Computing*, 13, 1741–1749.
43. Sert, S. A., Bagci, H., & Yazici, A. (2015). MOFCA: Multi-objective fuzzy clustering algorithm for wireless sensor networks. *Applied Soft Computing*, 30, 151–165.
44. Kumar G. S., Vinu P. M., & Jacob K. P. (2008). Mobility metric based leach-mobile protocol. In *16th International conference on advanced computing and communications* (pp. 248–253).
45. Wang, W., Du, F., & Xu, Q. (2009). An improvement of LEACH routing protocol based on trust for wireless sensor networks. In *5th international conference on wireless communications, networking and mobile computing* (pp. 1–4).
46. Liu, B., & Wu, Y. (2015). A secure and energy-balanced routing scheme for mobile wireless sensor network. *Wireless Sensor Network*, 7(11), 137.
47. Chen, Z., He, M., Liang, W., & Chen, K. (2015). Trust-aware and low energy consumption security topology protocol of wireless sensor network. *Journal of Sensors*. <https://doi.org/10.1155/2015/716468>.
48. Sandhya R., & Sengottaiyan N. (2016). S-SEECH secured-scalable energy efficient clustering hierarchy protocol for wireless sensor network. In *International conference on data mining and advanced computing (SAPIENCE)* (pp. 306–309).
49. Rehman, E., Sher, M., Naqvi, S. H. A., Badar, Khan K., & Ullah, K. (2017). Energy efficient secure trust based clustering algorithm for mobile wireless sensor network. *Journal of Computer Networks and Communications*. <https://doi.org/10.1155/2017/1630673>.
50. Yang, X. S., & Deb, S. (2009). Cuckoo search via Lévy flights. In *World congress on Nature and biologically inspired computing, 2009* (pp. 210–214). NaBIC 2009. IEEE.
51. Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69, 46–61.
52. Yao, X., Liu, Y., & Lin, G. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, 3(2), 82–102.
53. Zadeh, L. A. (1975). The concept of a linguistic variable and its application to approximate reasoning–1. *Information Sciences*, 8, 199–249.
54. Arain, Q. A., et al. (2016). Clustering based energy efficient and communication protocol for multiple mix-zones over road networks. *Wireless Personal Communications*, 95(2), 411–428.
55. Mittal, N. (2018). Moth flame optimization based energy efficient stable clustered routing approach for wireless sensor networks. *Wireless Personal Communications*. <https://doi.org/10.1007/s11277-018-6043-4>.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Nitin Mittal received his B.Tech. and M.Tech. degree in Electronics and Communication Engineering (ECE) from Kurukshetra University, Kurukshetra, India in 2006 and 2009 respectively. He has completed his Ph.D. in ECE from Chandigarh University, Mohali, India in 2017. He has more than 12 years of experience and is presently working as Associate Professor with Chandigarh University, Mohali, India. His research interests include wireless sensor networks, digital signal processing and soft computing.



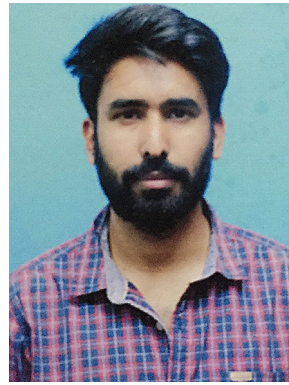
Simrandeep Singh received the B.Tech and M.Tech degree in Electronics & Communication engineering from the Punjab Technical University in years 2008 and 2015, respectively. He is pursuing Ph.D. from Chandigarh University, Gharuan, Punjab India. He is working as Assistant Professor in the Department of Electronics and Communication Engineering, Chandigarh University, Gharuan, India. He has 11 years of teaching experience. He has more than 15 research publications in various fields. His current

research interests include diatom analysis based on multifocus image fusion, wireless sensor networks and digital signal processing.



Urvinder Singh received his B.Tech. in Electronics and Communication Engineering (ECE) from Guru Nanak Dev University, Punjab, India in 1998. He received his M.Tech and Ph.D. in ECE from Punjab Technical University, Jalandhar, India in 2007 and 2013 respectively. He has more than 15 years of experience and is now working as Associate Professor with Thapar Institute of Engineering and Technology, Patiala, India. His research

interests include antennas, computational electromagnetics, filter design and wireless sensor networks.



Rohit Salgotra is currently a DST, Govt. of India sponsored PhD scholar, under INSPIRE fellowship, at the Thapar Institute of Engineering and Technology, Patiala, India. He has published around 16 high impact SCI journals and has reviewed articles for various journals of Elsevier, Springer and IEEE. He has received various travel grants for international travel including IEEE Computational Intelligence Society travel grant for CEC

2018 and CEC 2019 respectively. His research interests include evolutionary computing, optimization and antenna array design applications.