Wireless Personal Communications (2021) 118:1535–1557
https://doi.org/10.1007/s11277-021-08101-2

A Novel QoS Based Secure Unequal Clustering Protocol with Intrusion Detection System in Wireless Sensor Networks

M. Maheswari[1,2] · R. A. Karthika[1]

Accepted: 10 January 2021 / Published online: 11 February 2021

Abstract

Wireless sensor network (WSN) becomes a hot research topic owing to its application in different fields. Minimizing the energy dissipation, maximizing the network lifetime, and security are considered as the major quality of service (QoS) factors in the design of WSN. Clustering is a commonly employed energy-efficient technique; however, it results in a hot spot issue. This paper develops a novel secure unequal clustering protocol with intrusion detection technique to achieve QoS parameters like energy, lifetime, and security. Initially, the proposed model uses adaptive neuro fuzzy based clustering technique to select the tentative cluster heads (TCHs) using three input parameters such as residual energy, distance to base station (BS), and distance to neighbors. Then, the TCHs compete for final CHs and the optimal CHs are selected using the deer hunting optimization (DHO) algorithm. The DHO based clustering technique derives a fitness function using residual energy, distance to BS, node degree, node centrality, and link quality. To further improve the performance of the proposed method, the cluster maintenance phase is utilized for load balancing. Finally, to achieve security in cluster based WSN, an effective intrusion detection system using a deep belief network is executed on the CHs to identify the presence of intruders in the network. An extensive set of experiments were performed to ensure the superior performance of the proposed method interms of energy efficiency, network lifetime, packet delivery ratio, average delay, and intrusion detection rate.

Keywords WSN · Unequal clustering · Fuzzy logic · Intrusion detection · QoS parameters

1 Introduction

The progressive development of information technology (IT) and integrated circuits (IC) intends to establish cost effective and compact-sized sensor nodes. Moreover, a wireless sensor network (WSN) is an internal portion of the internet of things (IoT) which helps

✉ M. Maheswari
mahesvarii.smuthu@gmail.com

R. A. Karthika
karthika.se@velsuniv.ac.in

[1] Vels Institute of Science Technology and Advanced Studies, Chennai, India

[2] SRM Institute of Science and Technology, Kattankulathur, India

in sharing a massive amount of information to enhance ecological user control. First, WSN has numerous sensor nodes organized in an adhoc manner to precede observation and communication with the external environment. The major limitations considered in WSNs are power consumption, bandwidth, and storage space while modeling the WSN system. Mostly, the sensors are placed in an unattended position where the battery recharge or replacement becomes impossible [1]. Followed by, the transmission cost is maximum when compared with sensing and computation processes in WSN [2]. To extend the network lifetime, an energy effective data transmission principle is highly essential for broadcasting the data from sensors to the base station (BS).

Clustering is a well-known energy efficient technique, where the sensor nodes are organized into clusters. Then, cluster members (CM) sense the real-time environment and forward the observed data to the concerned cluster head (CH). Next, the CH receives the aggregated data for the purpose of eliminating redundant data. Also, the CMs do not forward the data to BS directly, but sends it to the CH. Hence, the CH placed nearby BS depletes their energy at a faster rate. Consequently, the network connection has been interrupted and resulted in coverage problems named as hot spot issue; which commonly exists in the clusters exist in the clusters placed nearer to BS. To overcome the hot spot problem, an unequal clustering technique is applied to balance the overhead among CHs [3]. The structure of unequal clustering in WSN is illustrated in Fig. 1.

It reduces the cluster size which is placed nearer to BS and the size of a cluster is increased when the distance between CH and BS gets enhanced. Finally, the unequal clustering technique enables inter-cluster routing to consume minimum power. So, unequal clustering is efficient in resolving hot spot issues by the equal distribution of load. In past decades, diverse clustering and unequal clustering methodologies were projected to energy effective WSN. Even though numerous inclusive works of literature have been established for clustering models, only a few surveys were carried out in unequal clustering technologies. Probability related unequal clustering approaches are developed with respect to node deployment, position, mobility, location-awareness as well as data aggregation [4]. Various protocols are compared based on node count, power efficiency, stable cluster, location-awareness as well as heterogeneity level [5].

As defined previously, clustering is a significant optimization issue in WSN. A massive number of clustering related routing models was introduced. computational intelligence (CI) approaches like neural network (NN), reinforcement learning (RL), swarm intelligence (SI),

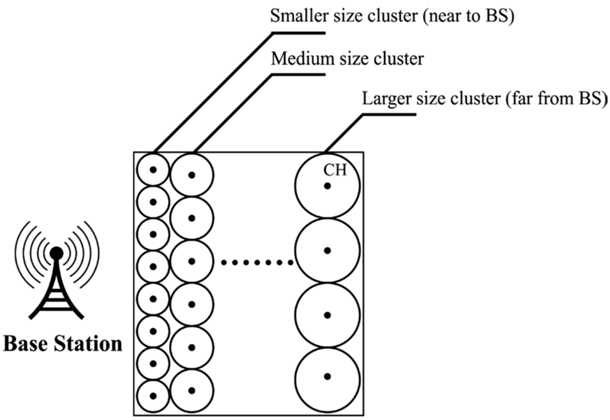


Fig. 1 Unequal clustering process in WSN

evolutionary algorithms (EA) and fuzzy logic (FL) have been employed for addressing the problems involved in WSN such as CH election, routing, privacy, data aggregation, and synchronization. Additionally, SI-centric clustering protocols are used for arranging the sensor nodes as clusters and minimize the power utilization of the complete system. For *N* sensors, around 2^{N-1} solutions have been attained and for each solution, a node is defined as CH or CM. Thus, clustering is meant to be an NP-hard problem. Followed by, the EA is used for effectively resolving diverse NP-hard issues. As the nodes are mutually dependent on interrelated measures, a cluster is formed utilizing previously determined rules. Afterward, FL is also an effective module in resolving maximum uncertainty [6, 7]. Recently, most of the developers applied SI based optimization methods as it is highly effective in resolving the optimization problems.

On the other hand, the intrusion detection system (IDS) mainly contributes to network security issues. It aspires to defend and track the anomalous actions in the network traffic which differentiates the normal and abnormal functions. In case of a higher level, the IDS are classified into 2 classes. Initially, the misuse-related IDS are also termed as signature-based IDS which predict the intrusion by monitoring the events similar to well-known attacks. But, the ability to predict known and unknown attacks is not so effective. Secondly, anomaly detection related IDS is operated by developing a profile of network behavior and identify the abnormal activities. Moreover, IDS predicts unseen attacks. The former IDS suffers due to a maximum false alarm rate, limited ability in predicting new attacks, and minimum prediction accuracy. Therefore, it is imperative to develop proficient IDSs with better prediction accuracy, reduced false alarm rate, and enhanced efficiency while predicting the known and unknown attacks. In order to satisfy the security demands, machine learning (ML) and deep learning (DL) models are found to be effective. Furthermore, ML techniques apply statistical principles for pattern identification. DL is an extended version of ML, which is derived from the artificial neural network (ANN). DL method is an essential mechanism in Artificial intelligence (AI), which is applicable in extracting features of the human eye and brain functions. Moreover, DL approaches are developed by ANN and numerous neuron connections, which performs higher-level abstractions of data feature extraction. Hence, the feature learned by neurons has been examined and computed by massive sub-neurons that results in the classification process. Conventional ANN can handle the nonlinear scenarios and the DL scheme is used for extracting the features and decides as similar as the human-brain.

This paper presents a new secure unequal clustering protocol with intrusion detection (SUCID) technique to satisfy QoS parameters namely energy, network lifetime, and security. The presented model comprises different processes namely node initialization, tentative CH (TCH) selection, final CH (FCH) selection, cluster maintenance, intrusion detection. The TCH process takes place using adaptive neuro fuzzy based clustering (ANFC) technique by the use of three parameters such as residual energy (RE), distance to BS (DBS), and distance to neighbors. In addition, the FCH process is carried out using the deer hunting optimization (DHO) algorithm to elect a final set of CHs. In order to balance the load between the clusters, the cluster maintenance phase is also executed. Finally, to achieve security in cluster based WSN, an effective IDS using a deep belief network (DBN) is executed on the CHs to identify the presence of intruders in the network.

2 Related Works

Recently, the FL model has gained massive attention from the developers while designing cluster based routing schemes in WSN. Numerous protocols utilize FL for resolving the problems in situations with high instability and uncertainties. As FL is flexible, fault-tolerant, minimum complexity and needs only low processing resources, different FL related protocols were established to report the clustering issues. Molay et al. [8] introduced a fuzzy centric clustering model with three input variables. Here, the concentration indicates the node count present in the vicinity while centrality is meant to be the rate of closeness to the intermediate cluster. However, it is operated as same as low energy adaptive clustering hierarchy (LEACH) only by excluding the CH election process. Followed by, the BS estimates the probability of becoming a CH with the help of Mamdani fuzzy inference system and nodes with maximum possibility would be selected as CH. Moreover, LEACH-FL is an extended version of [9], which applies RE, distance to BS (DBS), and node density (ND) as input parameters.

Cluster head election mechanism-based on fuzzy logic (CHEF) is defined as a distributed fuzzy relied mechanism without global data regarding a network [10]. It enhances the network lifespan with local CH selection using 2 input parameters like RE and adjacent distance. Besides, only the fuzzy input variables are modified, but the performance is similar for LEACH and CHEF. Energy aware fuzzy unequal clustering technique (EAUCF) is projected for extending the network stability and lifetime [11]. It can be meant to be a distributed technology that computes the competition range under the application of 2 input attributes like RE and DBS. Alternatively, the fuzzy related clustering approach is named as improved fuzzy unequal clustering method (IFUC) to expand network duration and to remove the hot spot problem [12]. It applies 3 fuzzy input parameters like RE, DBS, and ND for determining the possibility of becoming a CH and cluster size. In addition, an ant colony optimization (ACO) based routing technique is developed for inter-cluster data transmission. Besides, a relay node is chosen based on communication expense and rate of power utilization.

Fuzzy logic based unequal clustering (FBUC) [13] is a distributed fuzzy based clustering model that is an extended version of EAUCF. Then, tentative CHs has been elected using a probability based model as well as competition range determined using FL. Hence, fuzzy input variables are RE, node degree, and DBS for calculating the cluster radius; while RE and node degree have been employed for selecting tentative CHs. Furthermore, the distributed clustering framework is DUCF as proposed in [14]. It utilizes FL to selected CH and for determining the cluster size. Here, RE, node degree, and DBS were employed as fuzzy input variables for the CH election and to compute the competition radius. It manages the load between clusters and mitigates the power utilization with the help of multi-hop data communication among clusters. The CRT2FLACO is defined as kind 2 fuzzy based clustering protocols that apply ACO for inter-cluster routing. Next, 3 fuzzy input variables namely, RE, neighboring nodes, and DBS. Multi-hop data broadcast applies ACO for transmitting data from CH to BS. Additionally, FAMACROW [15] is applied as a fuzzy and ACO related model has been employed to report the challenging issues of clustering and routing process.

In [16], a new clustering technique has been presented based on adaptive neuro fuzzy clustering algorithm (ANFCA) for load balancing in WSN. The CH is elected based on different parameters such as energy level, distance, and density. A novel technique for defining the security breach by the use of the DBN model is presented [17]. It determines the occurrence of malicious activity that is active inside the network, and one tries to get its entry. In [18], an improved particle swarm optimization (IPSO) algorithm is introduced in the energy-balanced unequal clustering (EBUC) technique to attain maximum network lifetime in WSN. A new clustering technique based on the krill herd algorithm (KHA) [19] has been presented for WSN. This method is focused on the maximization of network lifetime by the consideration of energy level of the nodes. A 5-input fuzzy based unequal clustering protocol (F5NUCP) [20] has been presented by the use of non-probabilistic TCH selection and FL based CH selection. The non-probabilistic TCH selection process includes a backoff timer value with respect to residual energy. In [21], a fuzzy and ant colony optimization based combined mac, routing, and unequal clustering cross-layer protocol (FUCHAR) has been presented. This protocol involves three major phases such as FL based clustering, ACO based routing, and cluster maintenance. In [22], a new IDS has been presented in WSN to reduce the false alarms and increase the detection rate by the use of a multi-kernel kernel extreme learning machine. Though several algorithms are available in the literature, there is still a need to achieve a better tradeoff between network lifetime, energy efficiency, and detection rate.

3 The Proposed SUCID Model

The working principle involved in the presented SUCID model is shown in Fig. 2. Once the nodes are randomly placed in the network, the initialization process takes place to collect information related to neighbors. Then, the BS executes the ANFC technique to elect a primary set of TCHs. Followed by, the DHO algorithm is applied and the FCHs are chosen based on the fitness function (FF). After several rounds of operation, the CHs closer to BS tends to exhaust its energy. In such cases, the cluster maintenance phase is invoked to distribute the load uniformly. At last, the intrusion detection process is executed to identify the existence of intruders in the network. These processes are discussed in the subsequent sections.

3.1 System Model

Assume that there are N sensor nodes developed arbitrarily while sensing the atmosphere periodically. The sensor nodes develop clusters with the help of the presented system. Every cluster has a CH that receives the data from the CM. Basically; the sensing devices are constant by nature with similar energy and ability to sense the surrounding by data computation and data transmission. A radio link developed among the nodes is always symmetric. It refers that nodes necessities for homogeneous energy to perform the data forwarding in all directions. Also, the BS is located on the external side of a system. The sensor nodes are capable to change the transmission energy which depends upon the distance among the receiver nodes. Furthermore, the first order radio mechanism is employed for estimating the energy required in the newly deployed model. Consider the packet size as m bits. The overall energy applied in data transmission is m bits over l meter distance among a transmitter and receiver, which is expressed as,

$$E_{TNE}(m,l) = \begin{cases} m * E_{elect} + m * \varepsilon_{fsp} * l^2 & if \quad l < l_o \\ m * E_{elect} + m * \varepsilon_{mpf} * l^4 & if \quad l \geq l_o \end{cases} \tag{1}$$

The energy utilized for receiving a packet of m bits from the transmitter node is projected as,

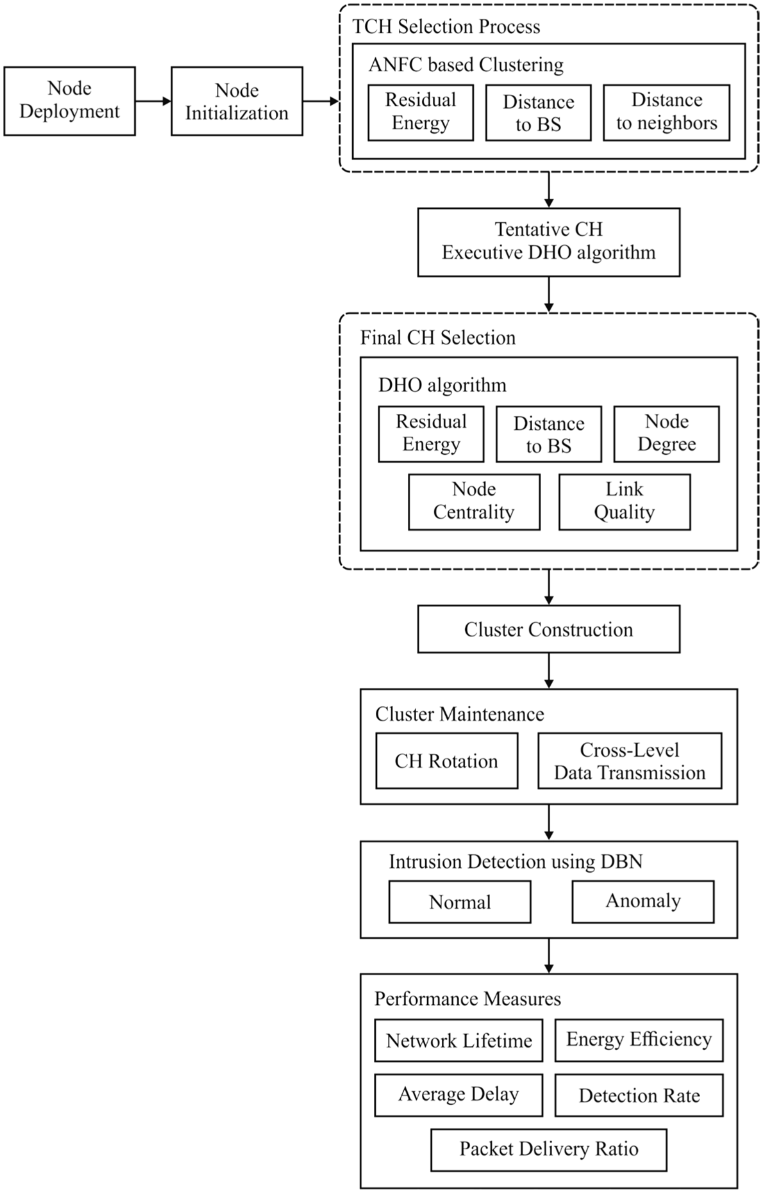


Fig. 2 Working process of SUCID model

$$E_{RCE}(m) = m * E_{elect} \tag{2}$$

where E_{elect} implies the data regarding electronic power dispersion. It is influenced by numerous factors like digital coding, manageable bit-rate, modulation, and so forth. The ε_{fsp} and ε_{mpf} are energy utilizing factors in a free space path as well as multipath fading. If the source and receiver nodes are divided with a specific threshold value

$l_o\left(l_o = \sqrt{\mathcal{E}_{fsp}/\mathcal{E}_{mp}}\right)$, then it applies a free space mechanism, else multipath fading channel has been applied to determine the power application.

3.2 TCH Selection Process

Here, FL and ANNs are effective modules to make the AI system more efficient due to the existence of generalization and non-linearity features. A hybrid approach has been deployed by the combination of soft-computing models like FL and ANN. This hybrid system is known as Adaptive Neuro Fuzzy Inference System (ANFIS) [16]. In ANFIS that executes the Takagi-Sugeno FIS of 5 layers. An important aspire of the ANFIS model is to enable the Membership Function (MF) and if-then rules are used interms of data collected from input and output data which is referred to as Adaptive Fuzzy Inference System (AFLIS). Additionally, the fuzzy rules are tuned in an automated manner that is applied in AFLIS under the application of supervised learning. Here, ANFIS applies the Trapezoidal MF (TMF) to weight modification. The MF is employed with inference rules in the fuzzification level. A newly deployed ANFIS method is comprised of inputs and output as illustrated in Fig. 3. All inputs have employed 3 MF and Takagi-Sugeno type method has 27 rules, where the nodes are maintained. An antecedent portion of rules showcases a fuzzy subspace while consecutive portions compute the external side of the fuzzy subspace.

An ANFIS model is defined as feed-forward NN (FFNN) with 5 layers with the help of a supervised learning mechanism. The layers are depicted as fuzzy, *T*-norm, normalized, defuzzy, and aggregated layers that imply 1st, 2nd, 3rd, 4th, and 5th layers. The 1st and 4th layers are composed of adaptive nodes, and residual layers are composed of permanent nodes. In this model, 3 inputs were applied namely, RE, DBS, and ND, and an output: chance (CH). Also, 27 rules of "*if–then*" has been developed for the ANFIS network according to the Takagi-Sugeno fuzzy inference method [16]. The rules are

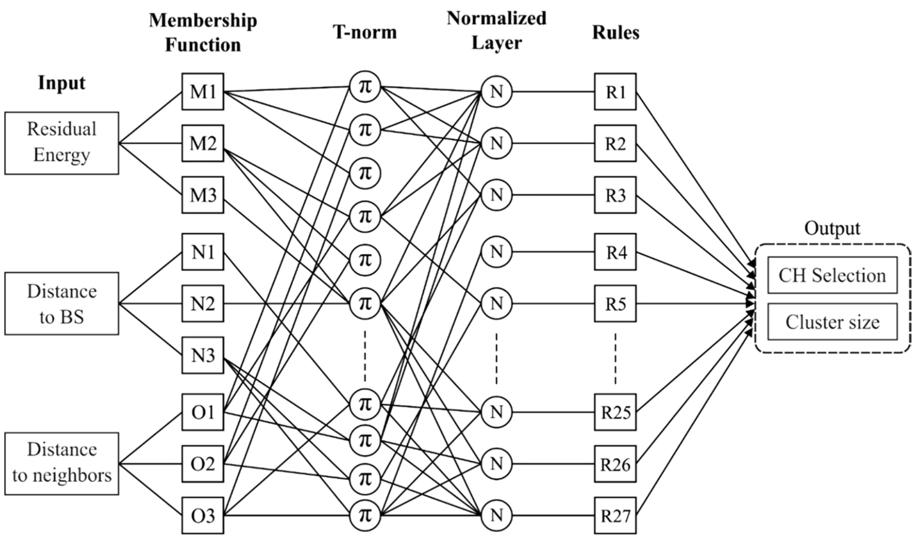


Fig. 3 Structure of TCH selection process

Rule 1 = *If* RE is low, DBS is near and ND is low Then $F_1 = S_1m + T_1n + U_1o + P_1$
Rule 2 = *If* RE is low, DBS is near and ND is medium Then $F_2 = S_2m + T_2n + U_2o + p_2$
Rule 3 = *If* RE is low, DBS is near and ND is high Then $F_3 = S_3m + T_3n + U_3o + P_3$
…
Rule 25 = *If* RE is high, DBS is distant and ND is low Then $F_{25} = S_{25}m + T_{25}n + U_{25}o + p_{25}$
Rule 26 = *If* RE is high, DBS is distant and ND is medium Then $F_{26} = S_{26}m + T_{26}n + U_{26}o + P_{26}$
Rule 27 = *If* RE is high, DBS is distant and ND is high Then $F_{27} = S_{27}m + T_{27}n + U_{27}o + p_{27}$

where low, near, and low defines MF of inputs RE, DBS, and ND, correspondingly, and S_i, T_j, U_k denotes linear parameters of Takagi-Sugeno fuzzy inference method. The linguistic variable for RE (M) = (*low*, fair, *high* is signified as (M_1, M_2, M_3), the DBS (N) = (near, midway, distant) is showcased a 0s (N_1, N_2, N_3) and ND (O) = (low, medium, high) is represented as (O_1, O_2, O_3).

1. **Fuzzy Layer** Here, the behavior of a node is flexible which depends upon the backward pass and mimics that each input variable is related to an MF. The MF graph is constructed over an adaptable node and defines the result. Also, ML applies Gaussian distribution as depicted in Eq. (3) and bell-shaped MF [Eq. (4)] which offers a value from 0 and 1.

$$\mu_{M\alpha}(M) = exp\left[-\left(\frac{m - f_\alpha}{2d_\alpha}\right)^2\right] \tag{3}$$

$$\mu_{M\alpha}(M) = \frac{1}{1 + \left|\frac{m-f_\alpha}{d}\right|^2 e_\alpha} \tag{4}$$

The result of the first layer is demonstrated as,

$$R_{1,\alpha} = \mu_{M\alpha}(M), \quad \alpha = 1, 2, 3$$
$$R_{1,\alpha} = \mu_{N\alpha}(N), \quad \alpha = 1, 2, 3$$
$$R_{1,\alpha} = \mu_{0\alpha}(0), \quad \alpha = 1, 2, 3$$

where M denotes the input node to α and μ_{Mi}, μ_{Ni}, μ_{oi} implies the degree of MF to linguistic variables M_i, N_i, and O_i and $\{d_i, e_i, f_i\}$ refers to the parameter set of MF. A bell-shaped MF differs from the measures of the premise parameter set. Furthermore, triangular and TMF are used for an input node which is referred to as valuable quantifiers for a node.

2. **T-Norm Layer** Here, a node is non-adaptive by nature, and termed as rule nodes that are illustrated using circle shaped structure π (as shown in Fig. 3). Such nodes imply the firing strength of a rule linked. To compute the outcome of a node, consolidate the MFs that emerged into a node. The T-norm operator applies generalized AND for estimating the antecedents at the 2nd layer of a rule.

$$R_{2\alpha} = T_\alpha = \mu_{M\alpha}(M) * \mu_{N\alpha}(N) * \mu_{o\alpha}(O), \quad \alpha = 1, 2, 3 \tag{5}$$

where T_α refers to the outcome of a node that represents a rule's firing strength.

3. **Normalized Layer** Non-adaptive nodes are projected in this layer that is denoted by circles as N (see Fig. 3). The simulation outcome of a node is the calculation of proportion among αth rule's firing strength to all rules. The result of a 3rd layer is depicted as,

$$R_{3\alpha} = T_{n\alpha} = \frac{T_\alpha}{\Sigma_\alpha T_\alpha}, \quad \alpha = 1, 2, 3 \tag{6}$$

4. **Defuzzy Layer** In this layer, nodes with adaptive essence are illustrated as a square (see Fig. 3). The outcome of the node is referred to as the multiplication of normalized firing strength as well as an individual rule. The simulation output of the 4th layer is expressed as,

$$R_{4\alpha} = T_{n\alpha} f_\alpha = T_{n\alpha}\left(s_\alpha m + t_\alpha n + u_\alpha o + p_\alpha\right) \tag{7}$$

   where $T_{n\alpha(s_\alpha m + t_\alpha n + u_\alpha o + p_\alpha)}$ means the normalized firing strength from the normalized layer and $\left(s_\alpha m + r_\alpha n + u_\alpha o + p_\alpha\right)$ implies a parameter in the node. Defuzzy layer parameters are named as the consequent parameter.
5. **Aggregated Output Layer** It is comprised of a node with a non-adaptive mechanism. This non-adaptive node offers the details regarding system performance estimated by including approaching signals from the former node. Summation sign Σ is applied within a circle to denote the aggregated output node. The outcome of the 5th layer is determined as,

$$R_{5\alpha} = \sum_\alpha T_{n\alpha} f_\alpha = \frac{\Sigma_\alpha W_\alpha f_\alpha}{\Sigma_\alpha w_\alpha} \tag{8}$$

   The ANFIS is applied for training the premise and subsequent parameters. An initial layer mimics the adaptive node with a non-linear premise parameter, and the 4th layer is comprised of linear consequent parameters.

3.3 FCH Selection Process

Once the TCHs are chosen, they compete with one another using the DHO algorithm for becoming FCHs. The DHO algorithm is a metaheuristic technique, based on the hunting nature of humans towards deer [23]. Although the actions of the hunters may vary, a method of attacking the buck/deer mainly depends upon the hunting mechanism. Owing to the peculiar capabilities of deer, it can be escaped easily. The hunting mechanism is based on the motion of 2 hunters in their optimal positions called leader and successor. For deer hunting, the hunters encircle it and move towards it. Followed by, every hunter updates the position till they reach the deer. Cooperative nature between the hunters is also essential to make the hunting process proficient. At last, they have reached the target depends on the position of leader and successor.

In this model, an optimal position has been selected for hunting a deer where the behavior of a deer has to be studied properly. There are a few characteristics that make hunting a complex process for the attackers. When compared with massive features, visual power is highly effective when compared with human beings. But it suffers from color deficiency where red and green colors are insignificant to a deer. A deer is also named as a buck that is capable to watch even a small move and scientists reveal which a white-tailed deer has peripheral vision ranging from 250° to 270°. It is helpful for a buck to predict the hunter's action; however, it is possible only for the defined range. The sensibility of a white-tailed deer is superior in sensing even a small change in the environment. The olfactory sensors of bucks are highly effective when compared with human beings.

When any danger is predicted by a deer, then it makes an alert signal for other bucks by heavily treading and loud sniffing. Also, the hearing intensity of a deer is not so effective as humans, and literature recommends that reasonable sensitivity of a deer ranges from 3000 to 8000 Hz whereas human hearing sensitivity ranges from 20 to 20,000 Hz. One of the remarkable talents of a deer is that, sensing ultra-high-frequency sounds that are impossible for human beings. The ears of a deer mimic the satellite chips which capture the signals and sounds that exist in the environment. Here, the numerical method of the DHO algorithm is defined below. Initially, the population of hunters is depicted as given in the following,

$$Y = \left\{Y_1, Y_2, \dots, Y_n\right\}; \quad 1 < j \leq n \tag{9}$$

where n implies the count of hunters which are considered as solutions in population Y. Once the population is initialized, wind angles, as well as the position of deer, are the 2 significant attributes while computing optimal positions of hunters. Mostly, the search space is assumed in the form of a circle, and then the wind angle follows the circumference of the circle.

$$\theta_i = 2\pi r \tag{10}$$

where r implies a random value from [0, 1] and i denotes the present iteration. Simultaneously, the position angle of a deer is expressed as,

$$f_i = \theta + \pi \tag{11}$$

where θ defines a wind angle.

3.3.1 Position Propagation

While the position of optimal space is undefined, then a candidate solution is placed nearby the optimal one and determines based on the FF, referred to as the best solution. In this model, 2 solutions have been considered namely.

Leader position ($\mathbf{Y}^{lead}$) It is an initial and optimal position of the hunter.
Successor position ($\mathbf{Y}^{successor}$) It defines the position of the following hunter.

(i) Propagation through a leader's position: Once the optimal positions are invoked, every individual of a population attempts to reach a successive place and update the place iteratively. Followed by, encircling behavior is labeled as provided in the given below,

$$Y_{i+1} = Y^{lead} - X * p * \left| L * Y^{lead} - Y_i \right| \tag{12}$$

where Y_i represents the place in recent iteration, Y_{i+1} denotes the location at upcoming iteration, X and L implies the coefficient vectors and p defines a random value deployed using the wind speed where the value ranges within 0 to 2. The coefficient vectors are evaluated as,

$$X = \frac{1}{4} \log \left(i + \frac{1}{i_{\max}} \right) b \tag{13}$$

$$L = 2 * c \tag{14}$$

where $j_{\max}$ refers to higher iteration, b denotes a parameter ranges from − 1 and 1 and c signifies an arbitrary value from [0, 1]. Here, (Y, Z) implies the initial position of a hunter that is upgraded with the prey's position. Followed by, the position of an agent is modified until reaching an effective position (Y^*, Z^*) and change the place of X and L. Position update is operated using Eq. (13) only if $p < 1$, which refers that an individual is allowed to move in a random direction irrespective of position angle. Hence, Eqs. (12) and (13) shows the position update of a hunter randomly within a definite space.

(ii) Propagation through position angle: To improve the search space, the model is updated by angle position. The angle calculation is highly significant for evaluating the position of a hunter where the victim is unaware of the danger and make the hunting method further effective. The angle of visualization of a deer is determined as,

$$a_j = \frac{\pi}{8} * r \tag{15}$$

According to the variations among the wind angle as well as the visual angle of a deer, a novel attribute is determined which is applied for upgrading the position angle.

$$d_i = \theta_i - a_i \tag{16}$$

where θ implies the wind angle. Next, a position angle is upgraded for upcoming iteration using the given function,

$$f_{i+1} = f_i + d_i \tag{17}$$

By considering the position angle, it is upgraded using the given following,

$$Y_{i+1} = Y^{lead} - p * \left|\cos(v) * Y^{lead} - Y_i\right| \tag{18}$$

where $A = f_{i+1}, Y_i^*$ defines an optimal position and p represents the arbitrary value. The place of an individual is adjacent to the inverse position angle, thus the hunter moves from the deer's view.

(iii) Propagation through the position of the successor: In this approach, a similar procedure of encircling behavior is applied by extending the vector L. Considering the search space as a random position, then the value of a vector L is minimum than 1. So, the position update depends upon a successor position. It enables a global search as depicted using the given notion,

$$Y_{i+1} = Y^{successor} - X * p * \left|L * Y^{successor} - Y_i\right| \tag{19}$$

where $Y^{successor}$ denotes the successor position of a search agent from a recent population. From the arbitrary initialization of solutions, the position update is carried out in search agents on the basis of the optimal solution. If $|L| < 1$, a search agent has been selected randomly while an effective solution is selected only if $|L| \geq 1$ upgrades the position of all agents. Therefore, using an adaptive difference of a vector L, the newly projected technology moves from Exploration and Exploitation stages. Furthermore, parameters have to be extended based on X and L, which is a major advantage for the presented approach. The position update has been performed in all iterations till reaching the optimal position that depends upon the objective function. It is initialized using an arbitrary group of solutions.

Hunters change the positions under the application of the update rule. Figure 4 demonstrates the flowchart of the presented DHO algorithm.

The cluster construction process takes place after the election of FCHs by the DHO algorithm. The BS executes the following steps to initiate the clustering process.

Step 1 Transformation of problem domain into DHO space where the hunter position has two dimensions namely leader and successor positions.
Step 2 Determine the fitness value (FV) by the use of FF. The FF of the FCH selection algorithm intends to optimize the five parameters namely RE, DBS, ND, NC, and link quality (LQ). The FV is determined for every hunter using Eq. (20):

$$\begin{aligned} FV = {} & \alpha_1 * \frac{\sum_{i=0}^{n} d(PN, memberi)}{n} + \alpha_2 * \frac{\sum_{i=0}^{n} RE(memberi)}{RE(PN)} + \alpha_3 \\ & * \frac{\sum_{i=0}^{n} ND(memberi)}{n} + \alpha_4 * \frac{\sum_{i=0}^{n} NC(memberi)}{NC(PN)} + \alpha_5 * \frac{\sum_{i=0}^{n} LQ(memberi)}{L(PN)} \\ & + \left(1 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 - \alpha_5\right). \frac{1}{No.\ of\ members\ covered\ by\ PN} \end{aligned} \tag{20}$$

where α_1, α_2, α_3, α_4 and α_5 are weighing parameters and n refers to the number of members covered in a cluster.

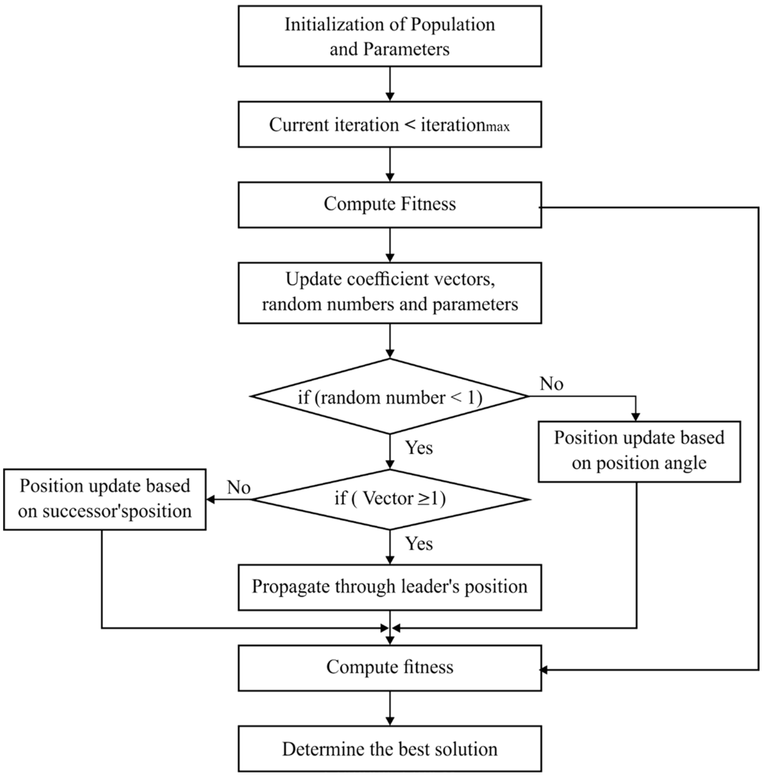


Fig. 4 Flowchart of DHO algorithm

Step 3 Generate a new position of hunters from the primary solution. The generation of the new position of the hunter from the previous one is the creation of a novel hunter.

Step 3.1 Determination of novel hunter position: the present position of the hunter is assumed as the rate at which the hunter's position is altered. The new hunter position is determined as given below.

$$\begin{aligned} new\_position = \omega * old\_position + w_1(local\_best\_position\, current\_best\_position) \\ + w_2(global\_best\_position\, current\_best\_position), \end{aligned} \tag{21}$$

where ω is inertia weight and w_1 and w_2 are basic DHO tunable parameters.
Step 3.2 Determination of the novel position of the hunter using Eq. (20). At last, the new position gets achieved.

Step 4 Calculate the FV of the new hunter position. The FV of the new hunter is determined in Step 2 with a novel hunter position.
Step 5 Compare the FV of old and new hunter positions and an optimal one is chosen for the subsequent round:

If new $FV > oldFV$.
choose new hunter position;
else.
the old one is propagated to the subsequent round.

Step 6 For each round, an optimal solution is chosen as the local best one.
Step 7 The local best solution from every iteration of the hunter which is maximum over the earlier solution is selected as the global best solution.

The BS organizes the clusters by the use of the DHO algorithm and advertises a message to the nodes. Every node saves the message and executes the CH selection process to select the FCHs.

3.4 Cluster Maintenance Process

The cluster maintenance process becomes essential for load balancing between the clusters. Next to certain rounds of operation, the clusters which are near to the BS are loaded with intercluster data traffic and deplete the energy rapidly. Therefore, a cluster maintenance phase is required to uniform distribution of load, resolve hot spot issues, and improve the network lifetime. A cluster maintenance process involves 2 stages namely CH rotation and cross-level data broadcast. The CH rotation process is carried out if the RE of the CH comes under a threshold value (15% of primary value). Once the remaining energy exceeds a threshold value, the novel CH would be chosen using the possibility of becoming CH. To uniformly distribute the load and make every CH exploit an identical amount of energy, cross-level data transmission is utilized. Once 15% of nodes act as CHs in a cluster, the BS is aware that fewer nodes are left to acts as the CHs. The BS transmits a message to the subsequent level CHs for direct data transmission. This procedure gets iterated until every next level CHs positioned far away from the BS. It leads to uniform energy dissipation and improves the network lifetime in a significant way.

3.5 Intrusion Detection Process

In this study, DBN is applied as an intrusion detection technique to determine the presence of intruders. DBN has been utilized for the purpose of classification. In training and learning phases, the DBN computes the data pre-processing and eliminates the noise present in the data [17]. The process of normalization removes the misguided solution. DBN applies the principle of probabilistic redevelopment of inputs, thus the layer showcases the feature predictors. Figure 5 illustrates the architecture of the DBN model.

DBN is designed with massive layers with a restricted boltzmann machine (RBM) that is organized in multi-phases. DBN is composed of hidden layers in order to create the procedure more effective. RBM is developed on the basis of Markov random field (MRF) named as log-linear. The energy function of RBM is comprised of numerous parameters to enhance accuracy. Hence, the RBM interacts with an alternate RBM for interchanging the learning features. It applies the possibility of distribution to complete the learning cycle. Thus, the energy function of the hidden unit is expressed as,

$$E(V, HL) = -b'V - c'HL - HL'WV \tag{22}$$

Using the energy function, the free energy derivations are computed as provided in:

$$F(V) = -b'V - \sum_{i} \log \sum_{H_i} e^{HL_i(C_i + W_i V)} \tag{23}$$

where W implies the weighted medium to the hidden layer as well as visible layer. b, c were allocated as an offset of visible and hidden layers correspondingly. A hidden layer is often independent of alternate layers. If the RBM applies the probability distribution function, then the signal's nature is modified as sinusoidal, as given below.

$$P\left(HL_i = 1 - V\right) = sigm\left(c_i + W_i V\right) \tag{24}$$

$$P\left(V_j = 1 - HL\right) = sigm\left(b_j + W_j' HL\right) \tag{25}$$

At last, the free energy of the hidden layer is extended as,

$$F(V) = -b'V - \sum_{i} \log\left(1 + e^{(C_i + W_i V)}\right) \tag{26}$$

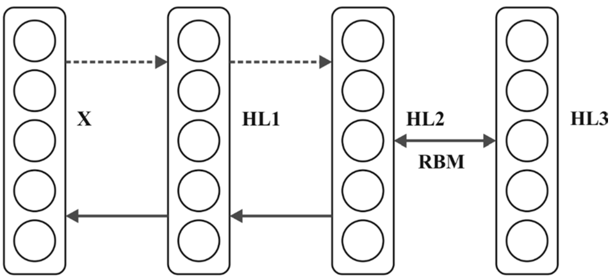


Fig. 5 Structure of DBN model

Using Eq. (5), the hidden layer is applicable to capture the directional details of data where the edge data is extracted applied to the trained method. Since the RBM interchanges the features and learning knowledge to alternate RBM, the network deeply learns the data.

As depicted in Fig. 6, the regularized flow of DBN for intrusion detection has an output target layer. Based on the DBN structure, the inputs are utilized for data preprocessing and derives the related primary data. In training, the model is composed of prior knowledge of networks like information of attacks employed. Followed by, the features have been examined into diverse forms that are induced for the upcoming hidden layer. Similarly, 2nd and 3rd layers derive the data for computing the learning process. The output layer matches the final decision obtained from the classification process. Since the resultant layer of a network is defined as the binary decision network, 0 is allocated for the protective network and 1 decided for intrusion prediction. The binary classifier ensures the action of a third party emerging into the network. Followed by, binary cross-entropy limits the overall cost of a network. Till reaching considerable results, DBN has to be trained using diverse values. The system is trained by isolating the data set for network qualification, web sampling, and finally, validate the system. When the function is not accomplished, then cross-validation (CV) is deployed using distinct integration of training and testing data.

- The Features are gathered from the input of normal and abnormal conditions.
- Pre-processing is performed to generalize the data and extract the essential features. Thus, high dimensions are minimized.
- Followed by, elected features are used for dividing the training and testing data.
- Initialization of the compile binary method of cross-entropy has been applied for classifying output for DBN.
- Under the application of training and testing data, recursive iterations are depicted until accomplishing the required performance.
- With the help of testing data, the testing process is invoked for learning the function of the trained network.
- Until reaching the desired performance, a mitigation process is performed. Hence, training for the DBN classifier is accelerated and the trained model is applied for learn-

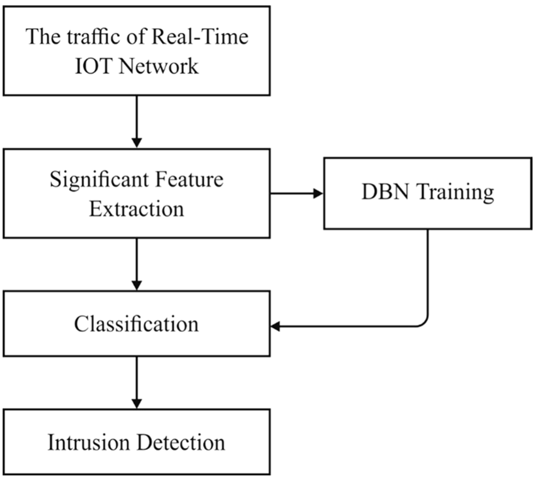


Fig. 6 Flow diagram of DBN for IDS

Table 1 Simulation parameters

Parameters	Value
Area	$100 \times 100\ m^2$
E_0	1 J
Node count	1000
E_{elec}	50 nJ/bit
ε_{fs}	10 pJ/bit/m^2
ε_{mp}	0.0013 pJ/bit/m^4
Packet size	4000 bits

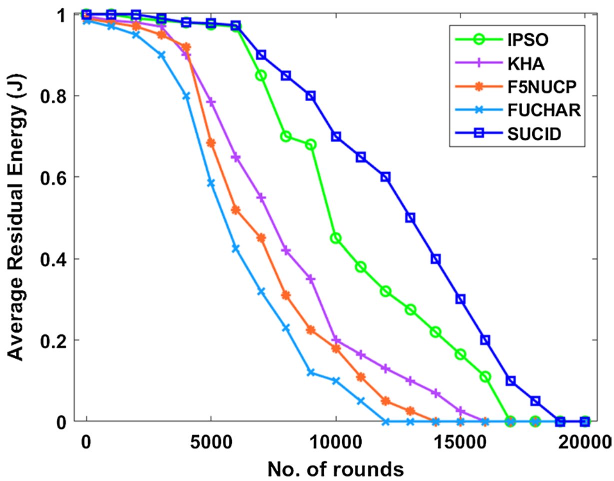


Fig. 7 Energy efficiency analysis of SUCID with existing models

ing and predicting the intrusions of a WSN. The network symptoms are detected and examined.

Based on the architecture, IDS monitors the attacks using the prior knowledge learned by the DBN.

4 Performance Validation

The performance of the presented SUCID algorithm has been simulated using MATLAB R2014a. The parameter setting of the presented model is given in Table 1. For comparison purposes, a set of algorithms namely IPSO algorithm [18], KHA [19], F5NUCP [20], and FUCHAR [21] are utilized.

4.1 Energy Efficiency Analysis

Figure 7 demonstrates the energy efficiency analysis of the SUCID model with respect to average RE under different iterations. From the figure, it is apparent that the FUCHAR is a worse performer than alternate models by showcasing high power utilization.

Simultaneously, the F5NUCP has outperformed the FUCHAR protocol and achieved moderate and identical RE. Concurrently, the KHA mechanism has illustrated considerable average RE when compared with previous approaches. Followed by, the IPSO technology has implied competing for RE than earlier models by excluding the SUCID framework. Finally, the SUCID technique has implied proficient function by gaining a higher value of RE. It refers that the SUCID scheme has applied a minimum amount of power and it is limited by enhancing the operating iterations. For sample, under a round value of 15,000, the FUCHAR and F5NUCP protocols have released maximum energy and average RE of 0 J while the SCE-PSO, KHA and IPSO methods have illustrated considerable and average RE of 0.02 J and 0.16 J correspondingly. However, the newly projected SUCID approach has confirmed optimal energy efficiency by reaching a better average RE of 0.31 J.

4.2 Network Lifetime Analysis

Figures 8 and 9 depicted the network lifetime analysis of the SUCID approach than compared models under distinct iterations. From the figure, it is evident that the count of active nodes is limited in FUCHAR protocol in previous operating iterations. At the same time, the count of active nodes is minimized gradually in the F5NUCP approach. In line with this, the KHA technique has represented a moderate outcome than the former models. Meantime, the IPSO scheme has delayed the expiry of a sensor node and reach an efficient network lifetime. However, the projected SUCID framework has demonstrated high network stability as well as maximized network duration to a greater extent.

An alternative way of analyzing the network lifetime is using First Node Die (FND), Half Node Die (HND), and Last Node Die (LND). The model that delays FND, HND, and LND shows moderate network stability. On calculating the network lifetime in light of FND, the previous mechanism has resulted in FND at earlier iterations. Hence, the SUCID approach has delayed the FND in secondary iterations. Likewise, on measuring the network lifetime by means of HND and FND, the related models have showcased poor outcomes with previous iterations. Therefore, the SUCID framework has delayed the LND to a specific extent.

Fig. 8 Number of alive node analysis of SUCID with existing models

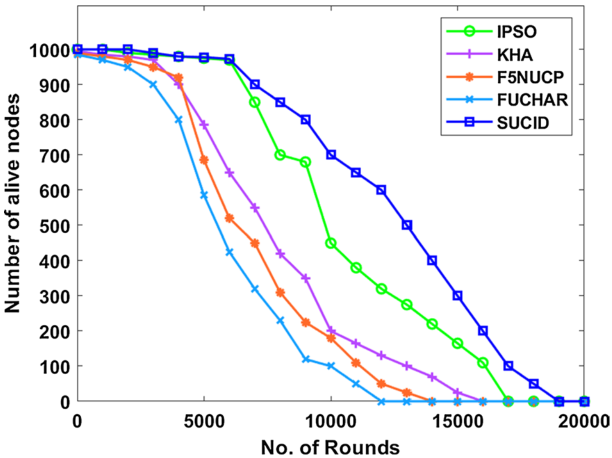
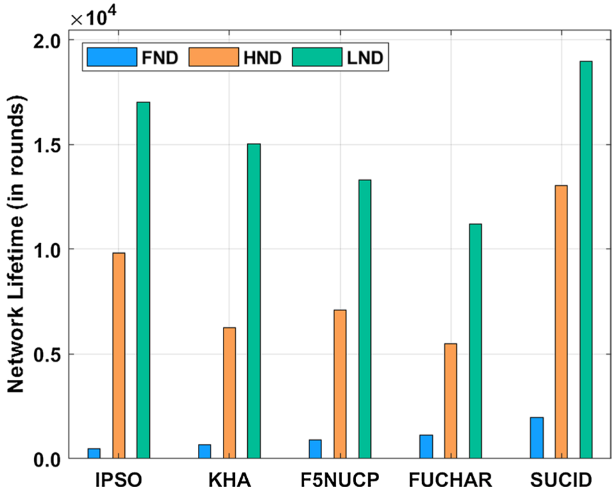


Fig. 9 Network lifetime analysis of SUCID with existing models

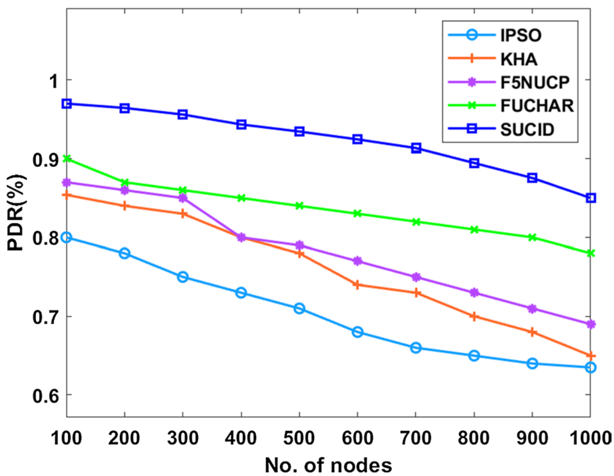


4.3 PDR Analysis

Figure 10 examines the results analysis of the SUCID model using the packet delivery ratio (PDR) under diverse rounds. From the figure, it ensures that the IPSO is an insignificant model and attained low PDR. Then, the KHA method has surpassed the IPSO protocol and gained considerable PDR.

Meantime, F5NUCP technology has accomplished improved PDR when compared with traditional approaches. Then, the FUCHAR scheme has depicted moderate outcome over the former models except for SUCID technology. Lastly, the SUCID framework has reached supreme results by accomplishing higher PDR. It means that the SUCID method has received a massive number of packets than alternate models effectively.

Fig. 10 PDR analysis of SUCID with existing models



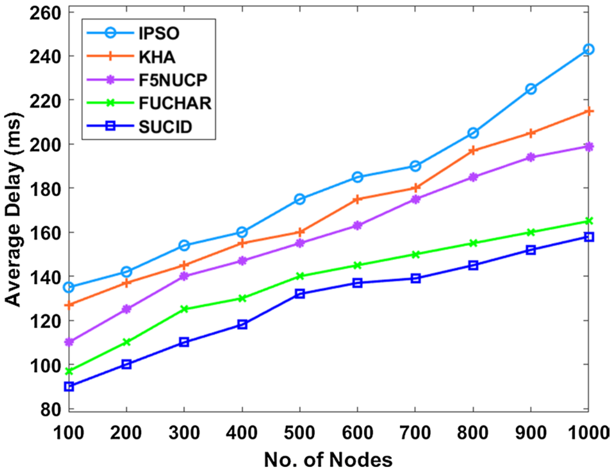


Fig. 11 Average delay analysis of SUCID with existing models

Table 2 Dataset Description

Dataset	No. of instances	No. of attributes	No. of classes	Normal/anomaly
NSL-KDD 2015	125,973	41	2	67,343/58,630
CICIDS 2017	2,830,743	80	2	2,273,097/557,646

4.4 Average Delay Analysis

Figure 11 showcased the average delay analysis of SUCID model under diverse node count. The figure depicted that the IPSO protocol gains maximum average delay because of single hop communication as well as random CH selection. In line with this, the KHA approach has demonstrated slightly better average delay than IPSO, except for alternate models. Likewise, the F5NUCP framework has illustrated a low average delay than previous methodologies. However, the FUCHAR scheme requires minimum and identical average delay under diverse node count. Therefore, the SUCID model has represented supreme performance by exhibiting low delay under diverse node count.

4.5 Intrusion Detection Analysis

The performance of the DBN model is examined using two dataset namely NSL-KDD 2015 and CICIDS 2017 dataset. The NSL-KDD 2015 dataset comprises 125,973 instances with 41 attributes. At the same time, the CICIDS 2017 contains a collection of 2,830,743 instances with 80 attributes. The results are examined interms of different measures, as illustrated in Table 2.

On investigating the intrusion detection results analysis of the DBN model on the applied NSL-KDD 2015 dataset in Table 3 and Fig. 12, the DBN model has achieved superior performance with the TNR of 99.98%, AUC of 99.97%, precision of 99.98%, recall

Table 3 Performance analysis of intrusion detection dataset for proposed DBN method

Measures	NSL-KDD 2015	CICIDS 2017
TNR	99.98	99.95
AUC	99.97	99.96
Precision	99.98	99.95
Recall	99.96	99.97
Accuracy	99.96	99.95
F-score	99.97	99.96
Error Rate	0.04	0.05

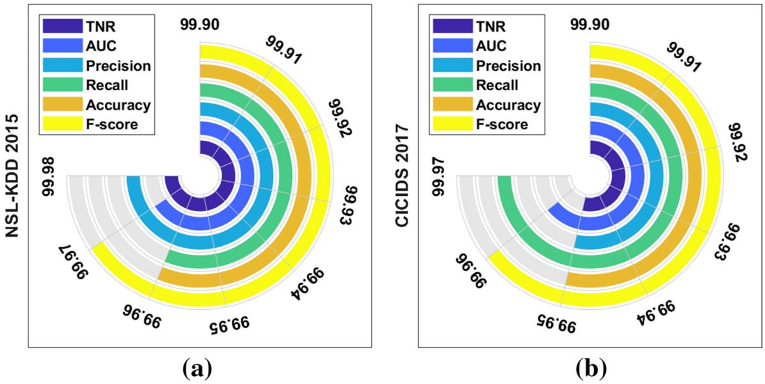


Fig. 12 Performance analysis of DBN on intrusion detection: **a** NSL-KDD 2015, **b** CICIDS 2017

of 99.96%, accuracy of 99.96%, and F-score of 99.97%. Similarly, on the applied CICIDS 2017 dataset, the DBN model has resulted in an effective classification performance with the TNR of 99.95%, AUC of 99.96%, precision of 99.95%, recall of 99.97%, accuracy of 99.95%, and F-score of 99.96%. From the table, it is also evident that the presented model has resulted in a minimum error rate of 0.04 and 0.05 on the applied NSL-KDD 2015 and CICIDS 2017 dataset respectively.

To validate the proficient performance of the proposed DBN method over the existing methods in Fig. 13, CS-PSO algorithm has resulted in unsuccessful classifier results with the minimum detection accuracy of 75.51%. In addition, the Gradient boosting model has showcased slightly better results with the minimum accuracy of 84.25%. Besides, the Gaussian process model has led to the moderate classification accuracy of 91.06%.

In the same way, the DNN-SVM model has achieved a classification accuracy of 92.03%. Moreover, the Fuzzy C-means, GA-Fuzzy, and Cuckoo optimization models have resulted in moderate performance with the accuracy of 95.3%, 96.53%, and 96.88% respectively. Furthermore, the behavior based IDS and PSO-SVM methods have exhibited manageable results with the results with the accuracy of 98.89% and 99.1%. Followed by, the ANN-IDS and MLIDS models have exhibited near-optimal performance with the accuracy of 99.39% and 99.93% respectively. At last, the DBN model has outperformed the previous models with a maximum accuracy of 99.96%. From the detailed experimental validation, the comparative analysis of the proposed with existing models showcased that the presented model has outperformed the other methods interms of different aspects.

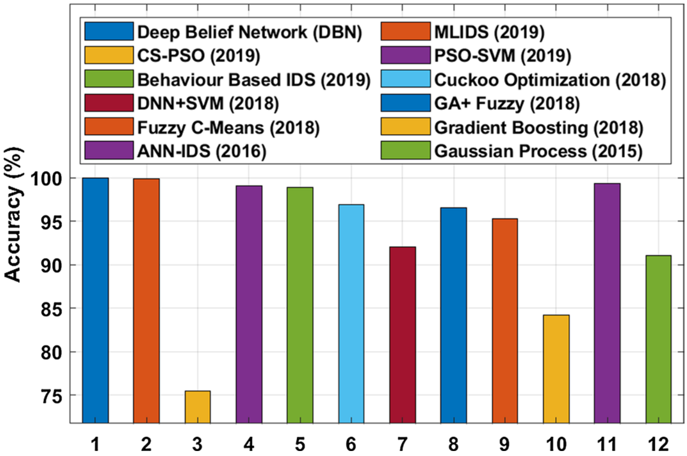


Fig. 13 Comparative results analysis of DBN with existing models

5 Conclusion

This paper has presented an unequal clustering with an intrusion detection technique, called SUCID for WSN. The presented model comprises different processes namely node initialization, TCH selection, FCH selection, cluster maintenance, intrusion detection. The ANFC based clustering technique is employed for TCHs and DHO algorithm is utilized for FCH. The incorporation of five input parameters for CH selection results in satisfactory performance. In addition, the involvement of the cluster maintenance phase greatly helps to achieve load balancing and maximum lifetime. At last, the application of the DBN model for IDS helps to identify the existence of intruders in the network. The detailed simulation analysis ensured the effective performance of the proposed method under diverse aspects. The experimental outcome stated that the SUCID model has attained maximum energy efficiency, network lifetime, PDR, average delay, and detection rate. In future, the performance of the DBN model can be improved by the use of the hyperparameter tuning process to determine the batch size, epoch count, and learning rate.

Funding Not applicable.

Compliance with Ethical Standards

Conflict of interest The authors declare that they have no conflict of interest.

References

1. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks, 38,* 393–422.
2. Raghavendra, C. S., Sivalingam, K. M., & Znati, T. (2004). *Wireless sensor networks*. USA: Springer.

3. Soro, S., & Heinzelman, W. B. (2005). Prolonging the lifetime of wireless sensor networks via prolonging the lifetime of wireless sensor networks via. In *19th IEEE international parallel distributed processing symposium*.
4. Kaur, G., & Varsha, . (2016). Review on hierarchical unequal clustering based protocols in wireless sensor network. *The International Journal on Recent and Innovation Trends in Computing and Communication, 4,* 96–99.
5. Vennira Selvi, G., & Manoharan, R. (2013). A survey of energy efficient unequal clustering algorithms for wireless sensor networks. *International Journal of Computers and Applications, 79,* 0975–8887.
6. Zadeh, L. A. (1965). Fuzzy sets. *Information Control, 8,* 338–353.
7. Zadeh, L. A. (1973). Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews, 3,* 28–44.
8. Molay, Z., Akbari, R., Shokouhifar, M., & Safaei, F. (2016). Swarm intelligence based fuzzy routing protocol for clustered wireless sensor networks. *Expert Systems with Applications, 55,* 313–328. https://doi.org/10.1016/jeswa201602016.
9. Ran, G., et al. (2010). Improving on LEACH protocol of wireless sensor networks using fuzzy logic. *Journal of Information and Computing Science, 7,* 767–775.
10. Singh, M., Gaurav Kumar, V., & Soni, S. (2016). Clustering using fuzzy logic in wireless sensor network. In *The 3rd international conference on computing for sustainable global development (INDIACom) 2016, New Delhi.*
11. Bagci, H., & Yazici, A. (2010). An energy aware fuzzy unequal clustering algorithm for wireless sensor networks. In *2010 IEEE international conference on fuzzy system (FUZZ), 18–23 July 2010* (pp. 1–8).
12. Mao, S., Zhao, C., Zhou, Z., & Ye, Y. (2012). An improved fuzzy unequal clustering algorithm for wireless sensor Network. *Mobile Networks Applications, 4,* 206–214. https://doi.org/10.1007/s11036-012-0356-4.
13. Logambigai, R., & Kannan, A. (2016). Fuzzy logic based unequal clustering for wireless sensor networks. *Wireless Networks, 22,* 945–957. https://doi.org/10.1007/s11276-015-1013-1.
14. Baranidharan, B., & Santhi, B. (2016). DUCF: Distributed load balancing Unequal Clustering in wireless sensor networks using Fuzzy approach. *Applied Soft Computing Journal, 40,* 495–506. https://doi.org/10.1016/jasoc201511044.
15. Gajjar, S., Sarkar, M., & Dasgupta, K. (2014). FAMACRO: Fuzzy and ant colony optimization based MAC/routing cross-layer protocol for wireless sensor networks. *Applied Soft Computing, 6,* 235–247.
16. Kumar Kashyap, P., Kumar, S., Dohare, U., Kumar, V., & Kharel, R. (2019). Green computing in sensors-enabled internet of things: neuro fuzzy logic-based load balancing. *Electronics, 8*(4), 384.
17. Balakrishnan, N., Rajendran, A., Pelusi, D., & Ponnusamy, V. (2019). Deep belief network enhanced intrusion detection system to prevent security breach in the internet of things. In *Internet of things* (p.100112).
18. Salehian, S., & Subraminiam, S. K. (2015). Unequal clustering by improved particle swarm optimization in wireless sensor network. *Procedia Computer Science, 62,* 403–409.
19. Karthick, P. T., & Palanisamy, C. (2019). Optimized cluster head selection using krill herd algorithm for wireless sensor network. *Automatika, 60*(3), 340–348.
20. Arjunan, S., Pothula, S., & Ponnurangam, D. (2018). F5N-based unequal clustering protocol (F5NUCP) for wireless sensor networks. *International Journal of Communication Systems, 31*(17), e3811.
21. Arjunan, S., & Sujatha, P. (2018). Lifetime maximization of wireless sensor network using fuzzy based unequal clustering and ACO based routing hybrid protocol. *Applied Intelligence, 48*(8), 2229–2246.
22. Zhang, W., Han, D., Li, K. C., & Massetto, F. I. (2020). Wireless sensor network intrusion detection system based on MK-ELM. *Soft Computing, 76,* 1–14.
23. Brammya, G., Praveena, S., Ninu Preetha, N. S., Ramya, R., Rajakumar, B. R., & Binu, D. (2019). Deer hunting optimization algorithm: A new nature-inspired meta-heuristic paradigm. *The Computer Journal*. https://doi.org/10.1093/comjnl/bxy133.

M. Maheswari is currently working as a Assistant Professor in the Department of Software Engineering, SRM Institute of Science and Technology, KTR Campus Chennai, India. She is pursuing her Ph.D. degree in the Department of Computer Science and Engineering, in Vels Institute of Science ,Technology and Advanced Studies, Chennai, India. Currently she is working in the field of Wireless sensor communication, Machine learning, Data storage and compression, security.

R. A. Karthika is currently working as a Associate Professor, in the Department of Computer Science and Engineering, Vels Institute of Science, Technology and Advanced Studies, Chennai, India. She obtained her Ph.D. degree from NoorulIslam University,Kanyakumari, India (2015). She has around 14 years of teaching experience and her areas of interest include networking, IOT, cloud computing, Big data. She has published around 34 papers in many national and international journals and also she is the member of the various international accreditation bodies like CSI.