JOURNAL OF CRITICAL REVIEWS

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

ENERGY-EFFICIENT CLUSTERING-BASED TOPOLOGY CONTROL SCHEME AND FAULT TOLERANCE USING MODIFIED DEEP LEARNING NEURAL NETWORK IN WSN ENVIRONMENT

Hemantaraj M. Kelagadi[1], Priyatamkumar[2]

[1]Assistant Professor, School of Electronics and Communication Engineering, KLE Technological University, Hubli, Karnataka.
[2]Professor, School of Electronics and Communication Engineering, KLE Technological University, Hubli, Karnataka.

Email ID : hemantkelagadi@yahoo.com, priyatambvb@gmail.com

Received: 14 March 2020 Revised and Accepted: 8 July 2020

ABSTRACT: Today, Wireless Sensor Networks (WSN) takes an important part in different applications. Fault tolerance and Energy efficiency are notable issues in WSN. Though the existing research approaches concentrated on these issues, they could not resolve the issues completely. So, this work proposes an efficient topology control (TC) scheme centered on clustering and routing algorithms and performs fault tolerance with the utilization of the MDLNN algorithm (TCCR-FTM) in the WSN environment. First, the sensor nodes (SNs) are grouped, and subsequently, the cluster heads (CHs) are selected by utilizing the new CSFFC algorithm. Here, two CHs, such as SCH, and PCH, are selected for lessening the residual energy (RE) loss of the CH. Subsequently, in the clusters, the multipath routing (MPR) is created. And, the shortest path (i.e., alternatively three paths are chosen) is selected from the multipath by utilizing the E-AOMDV algorithm. Next, the fault node (FN) in the selected path of clusters is detected utilizing the MDLNN algorithm. If any links break and FNs occur in the clusters, then the FNs are isolated and then the alternate path is chosen for further communication. During experimental analysis, the proposed TCCR-FTM shows superior performance on considering the other methods in respect of performance metrics.

KEYWORDS: Modified Deep Learning Neural Network (MDLNN), Primary Cluster Head (PCH), Evolutionary based Ad-hoc On-Demand Multipath Distance Vector (E-AOMDV), Secondary Cluster Head (SCH), and Cosine Similarity-based Farthest First Clustering (CSFFC).

I. INTRODUCTION

Over the last '10' years, Wireless sensors network (WSN) is the utmost important technologies [1]. WSN has broad-spectrum applications, say military surveillance, healthcare applications, environment monitoring, along with target tracking [2-4]. These networks encompass sensor nodes (SN), which can monitor along with process the data as of a specific geographical location, in addition, send it to a remote location that is labelled Base Station (BS) [5]. The SN in WSN is prepared into clusters to preserve energy [6]. In the network design, faulty SNs are also rendered. Faulty SN is liable to report haphazard readings that do not send the realism of experiential physical processes to the BS [7].

Additionally, faulty SN cannot functions any monitoring task correctly but it must not affect the sensor network's overall task. This is the reliability or Fault Tolerances (FT) problem. FT is the capability to maintain sensor network functionalities devoid of any interruptions because of SN failures [8]. Thus, this faulty SN should be recognized at the correct time as well as separated as of the data collection procedure to assure the general data quality [9]. In large-scale WSN, it is not probable for the BS to collect data as of each SN as well as notice faulty SN in a centralized way [10]. In this situation, numerous protocols along with algorithms were suggested to augment the Network Life-Time (NLT), which is attained by lessening the redundant communication betwixt the nodes on the network through topology construction and maintenance.

Topology management is the important method engaged to essentially diminish battery power usage and also maintaining network-wide connectivity [11]. Nevertheless, the augmentation in the lifetime is at the rate of lessening the FT in the network [12]. FT has, thus, attained a lot of concentration as of researchers in the WSN field with reliability aimed at each layer of the communication stack [13]. Numerous fault detection along with recovery methods are presented in recent times, like superfluous node deployment techniques, attack analysis along with countermeasures, FT event boundary detection, FT target detection, as well as effective event query FT algorithm, to trounce the issues that are arisen because of faulty SN [14]. Moreover, merely fault diagnosis is also offered but that is not energy effectiveness. Consequently, this paper proposed a novel FT together with the TC scheme in the WSN.

The draft structure for this paper is systematized as: Section 2 surveys the associated works regarding the proposed method. In sections 3, a brief discussion about the proposed TC and FT routing methods are proffered. Section 4, elucidates the experimental outcome, and also section 5 concludes the paper.

II. RELATED WORK

Sasmita Acharya and C.R. Tripathy [15] rendered the Adaptive Neuro-Fuzzy Inferences System (ANFIS) estimator-centric data aggregation (DA) strategy termed Neuro-Fuzzy Optimization Model (NFOM) for the FT-WSN model. The ANFIS recognized the intra- and inter-cluster fault detection (FD) in the WSNs. The NFOM-type DA strategy was examined for "5" cases with disparate fault types and node densities, and the diverse performance measures were recorded. Its outcomes were then contrasted to that of Fuzzy Knowledge-centric FT (FTFK), Distributed FD (DFD), Low Energy DFD (LEDFD), and Majority Voting (MV) approaches. The ANFIS merely generated outcomes centered on knowledge, which did not acquire high-quality outcomes.

Shihong Hu and Guanghui Li [16] proffered a regular hexagonal-centric clustering (RHC) framework and a scale-free topology evolution (SFTE) strategy for WSN that sustained energy balance and elevated network survivability. This strategy examined the dynamic traits of SFTEM centered on the adopted mean-field theory. Its outcomes corroborated that the node degree dissemination of SFTE utilized the power-law distribution. The intrusion-tolerance (IT) along with fault-tolerance of RHC performed-well on considering other models. Nevertheless, the strategy failed to exactly determine the total faults in the WSN.

Haoran Liu *et al.* [17] recommended a scale-free topology, which simultaneously performed the processes like fault-tolerance in opposition to random faults (RFs) and IT in opposition to selective remove attacks. The system utilized the BA-E topology for degree dissemination. Then, the optimum parameter of the BA-E was attained, which was deployed via simulation for aforesaid processes. The performance shown by the BA-E model was contrasted to "3" prevailing models in experiments. The topology was not effectual that means the system failed to sustain the robustness in all scenarios.

Fatih Deniz *et al.* [18] built an energy-aware, distributed, and adaptive FT -TC algorithm termed Adaptive Disjoint Path Vector algorithm (ADPV) for the heterogeneous WSNs. The propounded ADPV algorithm embraced "2" phases: i) a single initialization phase, which transpired at the primary stage, and ii) restoration phases, which were done whenever the WSN's super-node connectivity was broken. Restoration phases employed variant routes that were developed at the initialization phase with the aid of an optimization grounded on the eminent set packing issue. Via simulations, the system corroborated that ADPV was good in preserved super node connectivity. But the SNs lost their life early.

M. Yuvaraja and M. Sabrigiriraj [19] put forward the FD and recovery framework in which the sink created an agent packet. The Agent developed a query path to the dead node or FN. The sink regularly broadcasted that Agent packet to the nearby nodes. The received node arbitrarily makes a decision of forwarding the packet, thereby, recognizing the dead nodes or FNs. After recognizing a dead node or node failure, the connectivity was re-stored grounded on Least-Disruptive Topology Repair (LeDiR), which was utilized to replace the FN with block movement. The experiments corroborated that the system acquired high-level performance on considering the prevailing methods.

Xue-Wei Wang *et al.* [20] rendered a double CH-centric FT-TC algorithm for WSN (DCHFT), which regarded the FT along with network life. Primarily, the network was partitioned into numerous subareas. Secondly, the main- and the vice- CH, which were respectively accountable for communication of clusters and the inter-clusters, were picked to build a network topology grounded on the node to node distance. Lastly, the outcomes corroborated that DCHFT improved fault tolerance, prolonged the NLT, and diminished energy consumption, contrasted to the typical clustering algorithms. The selection of CH nodes did not regard the RE, hence it influenced the NLTs.

TOPOLOGY CONTROL AND FAULT TOLERANCE FOR WSN

Fault-tolerant TC stands as a prime issue in WSN. It is indispensable for increasing reliability and NLT. Hence, an energy-efficient clustering-centric TC scheme and FT utilizing modified deep learning neural network (MDLNN) is proposed in this work. The phases like, i) network model, ii) cluster formation as well as CH selection, iii) MPR and shortest path selection, and iv) FN detection are performed by this work. Primarily, the network model is defined. Then, the cluster formation and CH selection are done by utilizing the CSFFC algorithm. And here, primary CH (PCH) is selected to monitor the environment and secondary CH (SCH) is chosen to monitor the cluster SNs. This sort of action offers a long life-time for the CH. Afterward, the MPR is developed in the clusters since each cluster encompasses more number of SNs. If the existing nodes are directly linked, the collision problem occurs and thereby the information gets collapsed. Hence, the existent nodes are not linked directly to the SCH. Each SN has a single or multi-path. If SNs have multipath, then the shortest path is chosen by utilizing the GKC-ACO algorithm. Alternatively, three paths are selected here grounded on the quality (that is, fitness), and they are scheduled. Then, the selected path is verified whether any FN is existent in the path or not with the aid of the MDLNN algorithm. If any link breaks or node failure is found in the selected path, the first selected path will be neglected and the second path is chosen for transmitting the packet to the SCH. Following this, the SCH delivers a packet to the PCH and it transmits the input packet to the BS. Here, the FN detection phase is indicated as the FT maintenance system and the remaining phases are signified as the TC system. Overall the proposed system is termed as TCCR-FTM, that is, "TC scheme centered on clustering and routing algorithms and FT maintained by utilizing the MDLNN algorithm". The proposed work could be comprehended utilizing the below block diagram proffered in Fig 1,

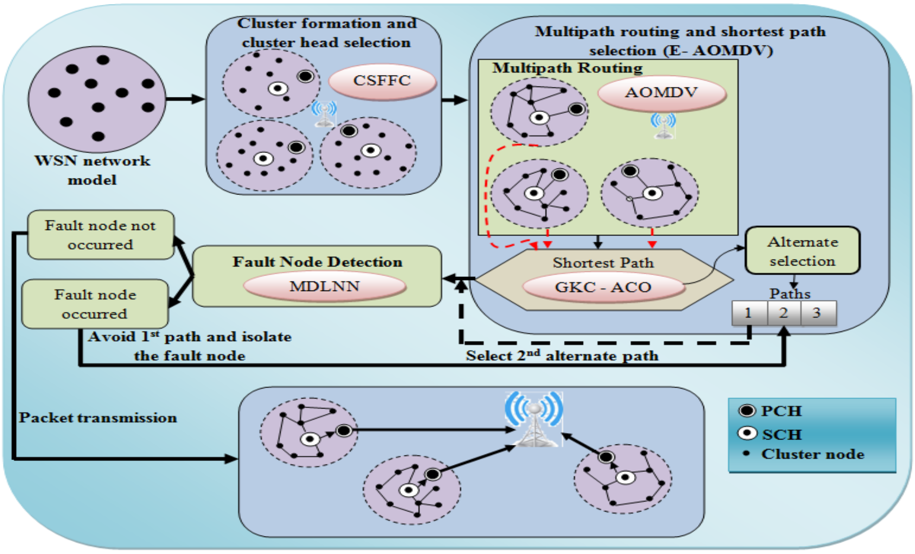

Figure 1: Block diagram for the TCCR-FTM scheme

WSN Network model

WSN has a load of nodes that are characterized by lower data transmission rate miniature, and inexpensive. The SNs monitor the environment information and deliver the sensing data to the SCH. The SCH transmits the packets to the PCH. The SCH is grounded on the center location of the cluster and the PCH is placed on the side corner of the cluster. The BS is fixed at a specific location. The SN of the WSN network model is written as,

$$N_s = \{o_1, o_2, o_3, \ldots\ldots\ldots, o_n\} \quad (1)$$

Where,

N_s - Network model

o_n - "n"-number of SNs in the WSN network model

Cluster Formation and Cluster Head Selection

Subsequent to stating the network model, the network's SN is grouped with the aid of the proposed cosine similarity centred farthest first clustering (CSFFC) algorithm. Initially, SN is grouped; then, the PCH and SCHs are chosen as of the cluster group. The CH is elected centred on their RE. Within the cluster group, the SN, which contains high RE, is signified as the PCH, in addition, the small difference as of the PCH nodes are chosen as the SCH and the remaining SN are signified as the cluster SN. Every SCH monitored the surroundings as well as transmits the information to the PCH, which it transmits to the BS (i.e, CH active mode). If the PCH energy is lessened (i.e sleeping mode) than the fixed threshold values, the new PCH is chosen as of the SCH. Here, the '2' CHs are elected for extending the CH's lifetime. Since, if only one CH is utilized, the CH will lose their energy. Thus, here, PCH along with SCH are chosen. On account of this, the proposed CSFFC offers improved performance.

The farthest first clustering (FFC) is the customized adaptation of the K-means algorithm. The FFC is employed centred on the greedy algorithm. It also clusters the nodes centred on the Euclidean Distances (ED) formula. The ED is not well for a big number of data. Here, n-number of SN is taken, so, here, the cosine

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

similarity (CS) function is utilized for distance calculation, which highly supports a big number of data points. The explanation concerning the CSFFC is elucidated as follows,

First, initialize the clusters; the cluster centroid is chosen centred on the cluster count. Aimed at cluster centroid selection, the 1st center is chosen haphazardly. The 2nd center is selected as the point furthest as of the first. Every remaining center is ascertained via choosing the point furthest as of the compilation of previously selected centers. All the chosen centroid points are signified as,

$$C = \{c_1, c_2, c_3, ..., c_n\} \in N_s,\ (or)\ c_i \in N_s \qquad (2)$$

Where, C implies the centroid set, c_n signifies the n-number of chosen centroids and $i = 1,2,.....,n$. Subsequent to cluster centroid selection, the distance is computed betwixt the cluster centriods and SNs utilizing the CS function, which is expressed as,

$$s_c = \frac{c_i \cdot o_i}{\|c_i\| * \|o_i\|}, \quad i = 1,2,3,......., n \qquad (3)$$

Where, s_c implies the CS output, which is computed betwixt the clusters centroid and the SN. Centred on the above equation (3), the similarity is computed betwixt the entire network model's SN and the chosen centroid values.

Next, assigned the similarity points to the cluster, after that, check whether all the SNs are chosen or not. If it is chosen, leave it, otherwise, repeat the distance calculations i.e., scan the list of not-yet-chosen points to discover the not-grouped node that has the maximal distance as of the selected points. Then, remove a point as of the not-yet-chosen points and adjoin it to the sequence's end of chosen points. Subsequent to creating clusters, the CH is chosen by utilizing the RE of the SN. Gauging the RE is helpful for identifying the energy of the SN. The RE stands as a difference between initial energy and current energy of SN in the network, which is measured utilizing the following equation:

$$Rs_l = Il_l - Cu_l \qquad (4)$$

Where, Rs_l implies the SN's RE, Il_l and Cu_l implies the initial energy and current energy of the SN. Here, the high RE cluster node is signified as the PCH and the small difference between the PCH RE is declared as the SCH and the remaining nodes signify the cluster SN.

Multipath Routing and Shortest path selection

After selecting the SCH and PCH, the MPR discovery process is done for the cluster SNs to access the SCH. As all the clusters contain more SNs, each cluster does not directly access the SCH. So, the multipath route is first created. For the creation of the multi-path route, the protocol termed "Ad hoc on-demand Multi-path Distance Vector (AOMDV)" is utilized in this work. The AOMDV aids to assess the many link-disjoint as well as loop-free paths as of the source to the destination node (DN). In the multiple routes, the destination holds the subsequent-hops in conjunction with the equivalent hop counts (HCs) in the entries of the routing table. Consider that the sequence number of these subsequent-hops is the same. For all the paths, the advertised HC is concerned as the maximal HC. Route advertisement is sent to a DN by utilizing this HC value, and if it is a duplicate one, then the SN forwards the packet to the DN through some other path (alternate path). The loop freedom is facilitated by choosing the alternate path to reach the DN centered on the path's HC value, which is less on considering the advertised HC for that DN. With the maximal HC value, the DN sorted all the paths.

From all the paths, the best and shortest path is selected with the utilization of the Gaussian Kernel and Crossover-centric Ant Colony Optimization (GKC-ACO) algorithm. The normal ACO algorithm proffers an excellent result for the path selection but it embraces the issues of pre-mature convergence and convergence speed. Hence, this proposed paper utilizes the Gaussian kernel (GK) to avert the pre-mature problem and enlarge the ants, and prevent the convergence speed problem by using the crossover function. Here, the combination of the buffer size (b_s), received signal-strength (r_s) and mobility (m_y) is regarded as the fitness function "F_f"for the best shortest path selection, which is indicated as,

$$F_f = r_s + m_y + b_s \qquad (5)$$

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

Grounded on F_f , the shortest and best path in the clusters is selected utilizing the GKC-ACO algorithm. The ACO system is a framework grounded on “3” rules, such as i) pseudorandom proportional rule, ii) pheromone update rule, and iii) local pheromone trail updation, which handle the optimization issue. The ants are initialized prior to the deployment of these rules. The ants indicate the SN in the clusters $\{o_1, o_2, o_3, \ldots\ldots\ldots, o_n\}$. The GK function is evaluated for all the initialized ant since it averts the local optima and convergence premature problems, which is expressed as,

$$u_Q^p(o) = \frac{1}{\sigma_{p,Q}\sqrt{2\pi}} e^{-\frac{(o-\chi_{p,Q})^2}{2\sigma_{p,Q}^2}} \tag{6}$$

$$\sigma_{p,Q} = \varepsilon \sum_{k=1}^{K} \frac{|a_{v,k} - \bar{a}_v|}{K-1} \tag{7}$$

$$\chi_{p,Q} = \sum_{k=1}^{K} a_{v,k} \tag{8}$$

Where,

$u_Q^p(o)$ - Gaussian function’s final output for the SNs in the p^{th} dimension,

$\sigma_{p,Q}$ - Standard deviation,

$\chi_{p,Q}$ - Sample value,

a_v - The average value of the SNs in the p^{th} dimension

ε - Constant that controls the convergence rate of common ants

After finding the GK function, the first procedure termed the Tour Construction or pseudorandom proportional rule is executed as,

$$t = \begin{cases} \underset{l \in f_s^j}{\arg\max} \left\{ [\psi_{sl}] [\varphi_{sl}]^\beta \right\}, & \lambda \le \lambda_0 \\ E(\mu_{st}^j), & \lambda > \lambda_0 \end{cases} \tag{9}$$

Where, λ indicates an arbitrary variable with a uniform distribution [0, 1], λ_0 signifies a parameter for the best probable move $(0 \le \lambda_0 \le 1)$, j symbolizes an ant (that is, the GK of the initialized SNs), β indicates a parameter that determines the relative influence of the heuristic information, φ_{sl} signifies the heuristic information value, ψ_{sl} symbolizes the pheromone trail, s, t indicates the initial and the next choice or candidate, l signifies a candidate solution, f_s^j symbolizes the feasible neighborhood of ant j, and E indicates a variable (random) acquired as per the probability distribution μ_{st}^j wherein the ant j picks the next solution if $\lambda > \lambda_0$.

μ_{st}^j is evaluated as,

$$\mu_{st}^j = \frac{[\psi_{st}]^\alpha [\varphi_{st}]^\beta}{\sum_{l \in f_s^j} [\psi_{st}]^\alpha [\varphi_{st}]^\beta} \tag{10}$$

Where, α indicates a parameter determining the relative influences of the ψ_{sl}.

After employing the first rule, execute the crossover operation on the SNs (i.e., ants) for averting the convergence speed problem. The crossover points in the population are arbitrarily extracted for executing crossover and are written as,

$$h_n \mapsto h_{n+1}(or)h_{n-1} \tag{11}$$

Where,

h_n - Current binary string of SNs selected for crossover operation

$\mapsto$ - Shift operator

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

h_{n+1} - Next binary string

h_{n-1} - Previous binary string

After crossover, the next rule "pheromone update rule" is performed. For improving the solution, the pheromone trails are ought to be updated and this updation embraces a) local updating and b) global updating. The local trail updating is proffered as:

$$\psi_u = (1-\rho)\psi(u_r, v_r) + \sum_{k=1}^{m} \Delta\psi_k(u_r, v_r) \tag{12}$$

Where, $\Delta\psi_k(u_r, v_r)$ signifies the quantity of pheromone trail included to the edges (u_r, v_r) by ant "k "between the time interval and is proffered as:

$$\Delta\psi_k(u_r, v_r) = \begin{cases} \dfrac{\omega}{L_k}, & (u_r, v_r) \in \pi_k \\ 0, & otherwise \end{cases} \tag{13}$$

Here,

ω - Constant parameter,

L_k - Distance of a sequence π_k toured by an ant in a time interval

The third rule "local pheromone trail updation" is employed during tour construction. The pheromone evaporation (PE) as well as a new pheromone deposit is updated while an ant explores or exploits the connection as per the pseudo-random proportional rule, which is proffered as:

$$\psi_{st} = (1-\gamma)\psi_{st} + \gamma\psi_0 \tag{14}$$

Where,

γ - Local PE rate, where $(0 < \gamma < 1)$

ψ_0 - Initial pheromone trails' value

With the GKC-ACO algorithm, the required shortest path is formed for all the clusters existent in the considered network model. Here, three alternate paths are chosen. The path with the topmost level centered on all the three metrics is picked as the first path. The path having the fitness value a little smaller than the initial one is picked as the second path. And, the selected third path has the fitness value smaller than both first and second paths.

$$Sl_p = \{\delta_1, \delta_2, \delta_3\} \tag{15}$$

Where,

Sl_p - Selected paths,

δ_1 - First path,

δ_2 - Second path

δ_2 - Third path

Thus, the MPR and the shortest path selection process by utilizing AOMDV and GKC-ACO are totally termed as E-AOMDV protocol. The term "E" in E-AOMDV indicates the evolutionary algorithm (like ACO).

Fault Node Detection

Subsequent to the shortest path creation, the 1st selected path is checked by FN (or) any other link break is present in the selected path utilizing the MDLNN. Normally, the Artificial Neural Networks (ANN) comprises an Inputs Layer (IL), a Hidden Layer (HL), as well as Outputs Layer (OL). In ANN, only one HL is considered for attaining the solution. Essentially, the deep analysis only gives better outcomes while the rough analysis may not cover all the parameters, which gives the poor result. So, here, the Deep Learning (DL) neural network is regarded, in this DL, more number of HL is taken for analyzing the data. Nevertheless, the DL also suffers concerning accurate FN detection together with speed. So, this paper modified the DL in the part weight calculation to enhance the accuracy in addition to lessen the execution time. Since in DL, the Weight Value (WV) is chosen arbitrarily. The arbitrary selection hasn't rendered a good outcome. The WV is modified centred on the targeted value. The targeted value is multiplied with the HL's inverse matrix value. The MDLNN's structure is evinced in Figure 2,

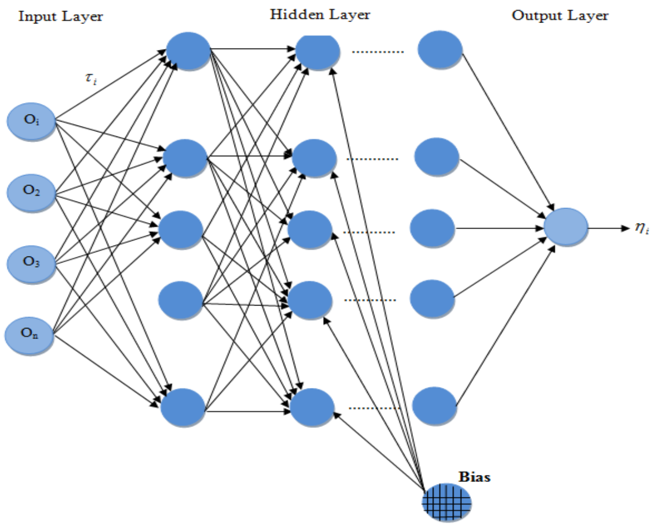

Figure 2: Structure of the MDLNN algorithm

Initially, the 1st path δ_1 is taken, which comprises of n-number of SN. All the SN in the chosen path is checked and also the FN is detected. The SN o_i of the chosen path is inputted to the IL. This IL renders the output to the hidden unit. In the HL, the SN and WV are multiplied and those outcomes are added with the bias function. First, for the hidden unit, the WV is arbitrarily initialized. The HL unit is computed as,

$$DD_i = bias + \sum_{i=1}^{n} o_i . \tau_i \tag{16}$$

Where, DD_i implies the hidden layer, τ_i signifies the specific layer's WV, and o_i implies the inputted data. After that, compute the output unit by adding up the hidden unit's output values. This is the computation to achieve the neuron's value in the OL.

$$\eta_i = bias + \sum_{i=1}^{n} DD_i . \tau_i(\eta) \tag{17}$$

Where, η_i implies the output unit, $bias$ functions as a constant which aids the design to fit the specified data, $\tau_i(\eta)$ implies the WV aimed at the output unit calculation, which is expressed as,

$$\tau_i(\eta) = \begin{bmatrix} DD_1 \\ DD_2 \\ . \\ . \\ DD_n \end{bmatrix}^T . \ x_t \tag{18}$$

Wherein, T implies the transverse function, which is taken for the entire hidden output, and it is multiplied with the targeted value x_t. By utilizing this MDLNN, the FN is found on the path. After that, gauge the Loss Function (f_e) utilizing the equation (19),

$$f_e = (x_t - \eta_i) \tag{19}$$

Wherein, x_t indicates the targeted output of the NN. After getting the final output, the f_e is checked. Formerly, the standard threshold value is fixed for the detection process. If the detected outcome meets the fixed threshold, the attained outcome is better. Otherwise, operate once more. The detected outcome comprises FNs (or) any other link break. Next, the alternate 2nd path is selected, and likewise, the FN is detected. If the detected outcome doesn't encompass the FN, then maintain the same path. The cluster nodes send the packet to the SCH and the SCH sends the packet to the PCH. This detection based communication system is signified as the FT scheme. Pseudocode for the proposed TCCR-FTM is exhibited in Figure 3,

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

Input: Sensor nodes, $N_s = \{o_1, o_2, o_3, ..., o_n\}$
Output: Efficient topology control and fault tolerance schemes

Begin

Initialize sensor nodes o_i, PCH, SCH, r_s, m_y, and b_s
Select centroid points in the sensor nodes, $c_i \in N_s$
Calculate CS between centroid and sensor nodes, $s_c = \frac{c_i \cdot o_i}{\|c_i\| * \|o_i\|}$
if $(\forall o_i == selected)$ {
cluster is formed
} **else** {
Again calculate distance & form the cluster
}
Select PCH & SCH based on Rs_l
Create multipath routing and **select** shortest paths in the cluster group

// cluster formation and cluster head selection

Select 3 alternate paths, $Sl_p = \{\delta_1, \delta_2, \delta_3\}$
if $(\delta_1 == fault\ node)$ {
Isolate δ_1
Select δ_2
} **else** {
Packet transmission process is carried out
}

// Fault tolerance

End

Figure 3: Pseudo code for the TCCR-FTM scheme for WSN

Figure 3 evinces the pseudocode for the complete work, i.e., the TCCR-FTM scheme for WSN. The step as of the cluster formation for alternating path selection is termed the TC scheme, and the FN detection procedure maintained the FT scheme for the WSN. Cluster formation in addition to CH selection is performed by means of CSFFC, MPR, and shortest path selection is processed via E-AOMDV, and FN is detected by the MDLNN algorithm. This sort of method extends the NLT, increases reliability, along with efficiency.

III. RESULT AND DISCUSSION

Here, the proposed TCCR-FTM is evaluated by contrasting it with the existing techniques, such as Low Energy Adaptive Clustering Hierarchy (LEACH), Two Level-LEACH (TL-LEACH), and Improvised TL-LEACH (ITL-LEACH). MATLAB is the platform used for implementing this proposed work. Table-1 is proffered below to elucidate the simulation parameters,

Table 1: Simulation parameters and its value

S.No	Parameters	Values
1	Network area	300m x 300m
2	Number of nodes	100
3	Speed of node	0-20 ms^{-1}
4	Simulation time	300 s
5	Protocol	E-AOMDV
6	BS selection	(150, 150)m
7	Packet size	512bytes
8	Percentage of CH	5%

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

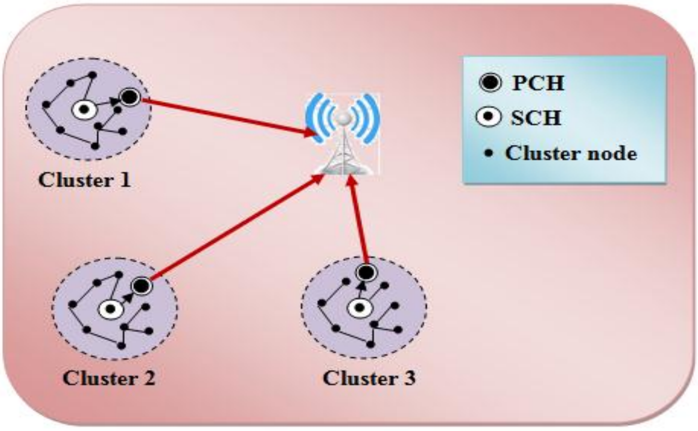

Figure 4: Clustered network communication of TCCR-FTM

The example of clustered network communication of the TCCR-FTM system could be elucidated by using Fig 4. Here, the SCH and PCH are declared and the MPR of the clustered SNs is shown. The SCH and PCH are chosen by utilizing the CSFFC algorithm (i.e., grounded on the RE). Then, MPR and shortest path are selected with the aid of the E-AOMDV algorithm.

Performance analysis

Here, the proposed TCCR-FTM is analyzed by contrasting it with the existing LEACH, TL-LEACH, and ITL-LEACH protocols centered on the performance in respect of metrics like, i) RE, ii) NLT, iii) reliability average end-to-end delay (EED), iv) throughput, along with v) packet delivery ratio (PDR). The metrics are explained below excluding RE (it is already given in eqn. (4)),

(a) Network Lifetime: The time elapsing as of initial deployment to the instant of the probability of connectivity attaining the preset threshold is regarded as NLT, which is written as,

$$N_l = \frac{o_{ie}}{o_{te}} \qquad (20)$$

Where,

N_l - NLT,

o_{ie} - Initial energy of SN

o_{te} - Total energy of SNs

(b) Packet delivery ratio: The PDR (P_r) is attained as of the total count of data packets arrived at SCH (Arr_{SCH}) divided by the totally transmitted data packets from cluster SNs ($Total_{CN}$), which is derived mathematically as,

$$P_r = \frac{Arr_{SCH}}{Total_{CN}} \qquad (21)$$

(c) Average EED: It is the time consumed by a packet to route via the network as of cluster SNs to its SCH. It is attained by computing the mean of EED of all correctly delivered packets.

$$A_{(E)to(E)} = \frac{1}{n}\sum_{i=1}^{n}(Rt_i - St_i)*1000 \qquad (22)$$

Where,

$A_{(E)to(E)}$ - Average EED

Rt_i - Reception time

i - Packet identifier

St_i - Sending time

n - Number of correctly delivered packets

(d) Throughput: It is the ratio of the packet-count the receiver received to the packet transmission delay at the time of the process, and is mathematically determined as,

ISSN- 2394-5125 VOL 7, ISSUE 15, 2020

$$Th_t = \frac{C(R_p)}{Y_l} \tag{23}$$

Where,

Th_t - Throughput,

$C(R_p)$ - Number of successfully received packets

Y_l - Delay

(e) Reliability: It gauges the system's lifetime, and reliability and packet loss is inversely proportional to each other.

$$WO_y = 1 - \frac{t}{\zeta(t)_{(1,.....n)}} \tag{24}$$

Where,

WO_y - Reliability,

$\zeta(t)_{(1,.....n)}$ - Mean time between failure nodes

The performance of both existing and proposed approaches for the important metrics like NLT and the RE of nodes is analyzed using table 2,

Table 2: Analysis of the performance of the TCCR-FTM with ITL-LEACH, LEACH, and LEACH based on (a) residual energy, and (b) network lifetime

(a)

Number of rounds	Proposed TCCR-FTM	ITL-LEACH	TL-LEACH	LEACH
0	100	100	100	100
40	95	90	86	80
80	84	81	75	71
120	78	73	69	65
160	72	68	64	59
200	67	61	58	54

(b)

Number of rounds	Proposed TCCR-FTM	ITL-LEACH	TL-LEACH	LEACH
0	100	100	100	100
40	100	100	100	99
80	100	100	98	91
120	100	96	90	87
160	98	89	86	82
200	92	85	81	77

IV. DISCUSSION

Table 1 proffers the (a) RE and (b) NLT- values attained by the proposed TCCR-FTM based TC and FT schemes for WSN and the existing TL-LEACH, ITL-LEACH, and LEACH. The performance is examined for 0 to 200 rounds. Round 0 indicates the initial stage; at which, all the systems have 100J energy and 100 protocols. In the time of increasing the rounds, the SNs lose their RE for the existing protocols. In table 1(a), at round 200, the proposed TCCR-FTM has 67J RE. The proposed TCCR-FTM only has the topmost energy at this stage, but all the existing protocols have less energy. Likewise, in table 1(b), the NLT is better for the proposed TCCR-FTM. Grounded on both metrics, the LEACH renders the lower-most performance on considering the remaining protocols. The ITL-LEACH protocol attains a superior result when contrasted to the existing TL-LEACH and LEACH. Moreover, TCCR-FTM gains better RE and NLT.

V. COMPARATIVE ANALYSIS

Here, the performance rendered by the proposed TCCR-FTM is contrasted with the existing ITL-LEACH, TL-LEACH, and LEACH protocols regarding the RE, NLT, PDR, average EED, throughput, and reliability metrics, which are graphically explicated utilizing Figures 5 to 10,

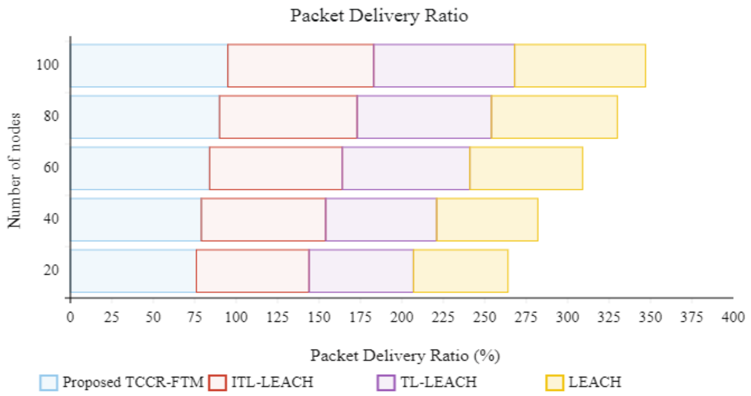

Figure 5: Packet delivery ratio analysis

DISCUSSION

Figure 5 compares the proposed TCCR-FTM scheme and the existing ITL-LEACH, TL-LEACH, and LEACH protocols regarding PDR. Here, the PDR is also a notable metric for estimating the system's quality and it is analyzed grounded on node counts. For 20 nodes, the TCCR-FTM has 76% PDR, ITL-LEACH, TL-LEACH, and LEACH protocols have 68%, 63%, and 57% PDR. For the remaining (40, 60, 80, and 100) nodes, the TCCR-FTM attains higher PDR. Here, the existing ITL-LEACH is much better on considering the other existing protocols, but, it has lower PDR than the TCCR-FTM. The PDR pictorial representation and the description corroborated that TCCR-FTM has higher performance when contrasted to the existing protocols.

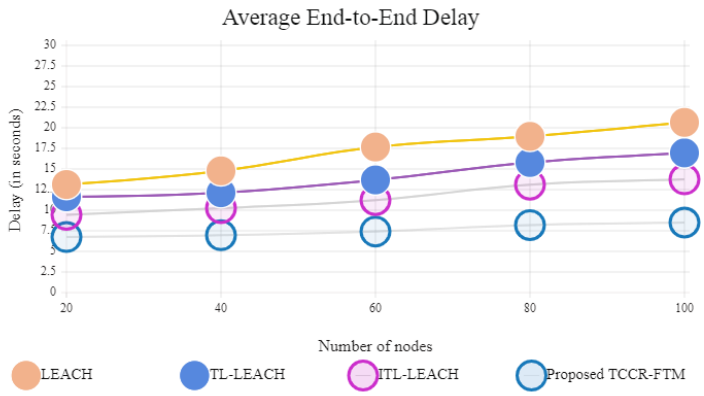

Figure 6: Average End-to-End delay analysis

DISCUSSION

Figures 6 compares the proposed TCCR-FTM and the existing ITL-LEACH, TL-LEACH, and LEACH grounded on the average EED metric (in seconds). Here, the delay is analyzed grounded on a disparate number of nodes (20, 40, 60, 80, and 100). The system is not suited for perfect communication if it has a higher delay. But, here, for 100 nodes, the TCCR-FTM has the 8.497s delay but the existing protocols, such as ITL-LEACH, TL-LEACH, and LEACH have 13.441s, 11.628s, and 13.121s delay, respectively. It corroborates that the existing protocols have a higher delay on considering the proposed TCCR-FTM. Likewise, the delays taken by the proposed and the existing protocols for the remaining node count are also evinced in above Fig 6.

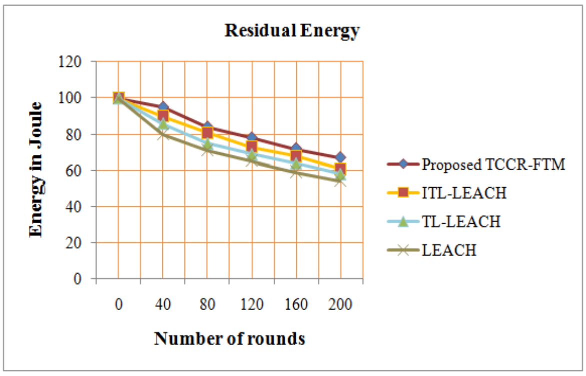

Figure 7: Comparison analysis of the TCCR-FTM with the ITL-LEACH, TL-LEACH, and LEACH based on the residual energy metric

DISCUSSION

Figure 7 demonstrates the performance rendered by the TCCR-FTM with the existing TL-LEACH, ITL-LEACH, and LEACH grounded on the RE metric. After completing every round, the RE of the nodes in the clusters is analyzed. Hence, the RE of the CHs and nodes are analyzed. After 40 rounds, the TCCR-FTM system has 95J RE, only 5J is lost and the existing ITL-LEACH, TL-LEACH, and LEACH have 90J, 86J, and 80J RE. This corroborates that, in the existing protocols, the cluster nodes lose more energy on comparing to the proposed system. Thus, the proposed TCCR-FTM attains the superior performance on contrasting with the existing protocols.

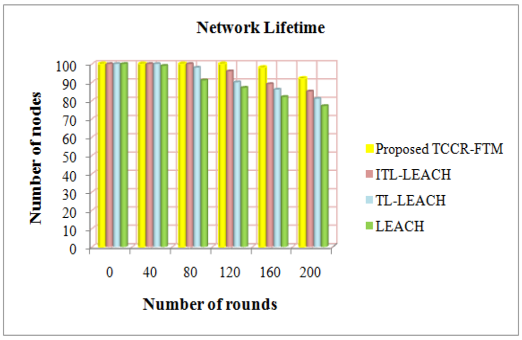

Figure 8: Illustrate the performance of the TCCR-FTM with the ITL-LEACH, TL-LEACH, and LEACH based on the network lifetime

DISCUSSION

Figure 8 analyzes the NLT of TCCR-FTM, ITL-LEACH, TL-LEACH, and LEACH. Regarding the NLT metric, for the 0, 40, and 80 rounds, most techniques generate a better result. After the completion of those rounds, the NLT of the systems is very poor. For example, after 160 rounds, the LEACH has 82 nodes, which is bad for the network quality. This LEACH protocol offers the worst result on considering the TCCR-FTM, ITL-LEACH, and TL-LEACH. Likewise, after 200 rounds, the LEACH attains a very poor result. But, the proposed have a better result on considering the existing methods. Hence, it infers that the proposed TCCR-FTM system acquires the longer NLT.

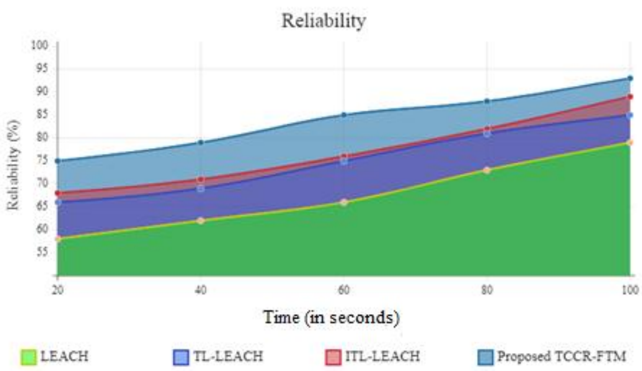

Figure 9: Reliability analysis

DISCUSSION

Figure 9 analyzes the reliability values attained by the proposed and existing systems. The reliability metric indicates how reliable the communication system is for users. Here, the reliability is analyzed between every 20s. For the first 20s, the proposed TCCR-FTM has 75% reliability but the existing ITL-LEACH, TL-LEACH, and LEACH have 68%, 66%, and 58% reliability, respectively. Similarly, for the remaining time intervals, the proposed TCCR-FTM gives higher reliability. At the 100s, the TCCR-FTM has 93% reliability, which corroborates that the proposed TCCR-FTM attains better performance on considering the existing systems.

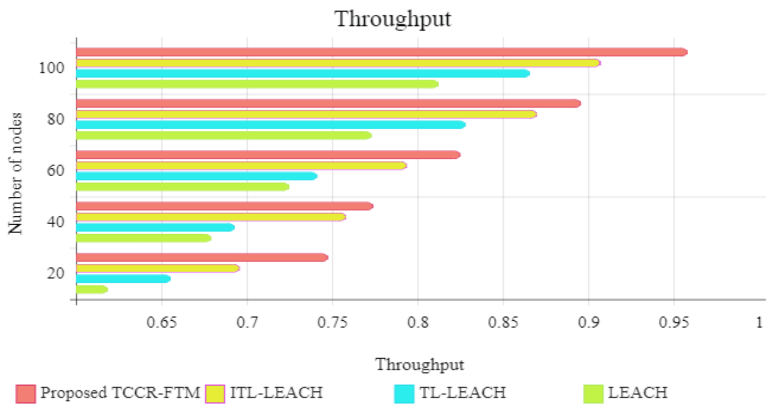

Figure 10: Compared the performance of the TCCR-FTM with the existing ITL-LEACH, TL-LEACH, and LEACH in terms of throughput

DISCUSSION

Fig 10 compared the proposed and the existing systems grounded on the throughput metric. Here, the performance is gauged centered on the node counts. For 100 nodes, the TCCR-FTM has 0.9571-throughput and the existing ITL-LEACH has 0.9062-throughput, TL-LEACH has 0.8648-throughput, and the existing LEACH has 0.8112-throughput. The proposed TCCR-FTM and the ITL-LEACH are better on considering the existing LEACH and TL-LEACH methods. But, the ITL-LEACH also has poor performance on considering the proposed TCCR-FTM system. Lastly, the proposed TCCR-FTM is found to attain superior performance on considering the existing systems.

VI. CONCLUSION

The faults may occur in WSN due to diverse reasons, which may adversely affect the reliability and NLT of WSNs. In this work, the efficient TC and FT structures are proposed for WSN. In this proposed work, the TC is done by cluster formation and CH selection, and MPR and shortest path selection. Then, the FT is done by utilizing the MDLNN algorithm; overall the system is termed as the TCCR-FTM system. The performance rendered by the proposed TCCR-FTM is analyzed utilizing the number of SNs and the round completions. The proposed TCCR-FTM is contrasted with the existing ITL-LEACH, TL-LEACH, and LEACH protocols in terms of RE, NLT, PDR, average EED, throughput, and reliability metrics. NLT, reliability, and RE are the utmost notable metrics for proving the communication system's performance. For all the metrics, the proposed TCCR-FTM system proffers better performance on considering the other protocols. Also, in the cluster group, this proposed system considers the MPR and shortest path selection, which is highly helpful for the big communication systems. In the future, this proposed TCCR-FTM system could be extended by considering the security for defending the system from the attackers.

VII. REFERENCES

[1] Kaur, T, & Kumar, D, "A survey on QoS mechanisms in WSN for computational intelligence based routing protocols", Wireless Networks, 2019, pp. 1-22, 10.1007/s11276-019-01978-9

[2] Hadjidj, A., Bouabdallah, A., & Challal, Y, "HDMRP: An Efficient Fault-Tolerant Multipath Routing Protocol for Heterogeneous Wireless Sensor Networks", Quality, Reliability, Security and Robustness in Heterogeneous Networks, 2012, pp. 469–482. 10.1007/978-3-642-29222-4_33.

[3] Sivakumar, S., & Vivekanandan, P, "Efficient fault-tolerant routing in IoT wireless sensor networks based on path graph flow modeling with Marchenko–Pastur distribution (EFT-PMD)", Wireless Networks, 2020,10.1007/s11276-020-02359-3

[4] Ahmed Khedr M, Zaher Al Aghbari, and PV Pravija Raj, "Coverage aware face topology structure for wireless sensor network applications", 2020, 10.1007/s11276-020-02347-7.

[5] Tyagi S, & Kumar, N, "A systematic review on clustering and routing techniques based upon LEACH protocol for wireless sensor networks", Journal of Network and Computer Applications, vol. 36, no. 2, pp. 623–645, 2013.
[6] Acharya, S, & Tripathy, C. R, "A reliable fault-tolerant ANFIS model based data aggregation scheme for Wireless Sensor Networks", Journal of King Saud University - Computer and Information Sciences, 2017, 10.1016/j.jksuci.2017.11.001
[7] Zeyu Zhang, Amjad Mehmood, Lei Shu, Zhiqiang Huo, Yu Zhang, and Mithun Mukherjee, "A survey on fault diagnosis in wireless sensor networks", IEEE Access, vol. 6, pp. 11349-11364, 2018.
[8] Angadi B M, Kakkasageri, M S, & Kori, G. S, "Topology control scheme for fault tolerance in wireless sensor network", International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), 2016, 10.1109/scopes.2016.7955661
[9] Kakamanshadi, Gholamreza, Savita Gupta, and Sukhwinder Singh, "A survey on fault tolerance techniques in wireless sensor networks", In international conference on green computing and internet of things (ICGCIoT), IEEE, pp. 168-173, 2015.
[10] Gupta Kakamanshadi, G, S., & Singh, S, "A survey on fault tolerance techniques in Wireless Sensor Networks", International Conference on Green Computing and Internet of Things (ICGCIoT), 2015, 10.1109/icgciot.2015.7380451
[11] Nayak, M. R., Tripathy, G., & Rath, A. K, "A Distributed Transmission Power Efficient Fault-Tolerant Topology Management Mechanism for Nonhomogeneous Wireless Sensor Network", Progress in Advanced Computing and Intelligent Engineering, pp. 481–493, 2018, 10.1007/978-981-10-6872-0_45.
[12] Qureshi, H. K., Iqbal, A., & Asif, W, "Topology Control for Harvesting Enabled Wireless Sensor Networks: A Design Approach", Wireless Personal Communications, vol. 82, no. 1, pp. 81–101, 2014.
[13] Papadopoulos, G. Z., Pappas, N., Gallais, A., Noel, T., & Angelakis, V, "Distributed adaptive scheme for reliable data collection in fault tolerant WSNs", IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015,10.1109/wf-iot.2015.7389037.
[14] Panigrahi, T., Panda, M., & Panda, G, "Fault tolerant distributed estimation in wireless sensor networks", Journal of Network and Computer Applications, vol. 69, pp. 27–39, 2016.
[15] Acharya, S., & Tripathy, C. R, "An ANFIS estimator based data aggregation scheme for fault tolerant Wireless Sensor Networks", Journal of King Saud University - Computer and Information Sciences, vol. 30, no. 3, pp. 334–348, 2018.
[16] Hu, S., & Li, G, "Fault-Tolerant Clustering Topology Evolution Mechanism of Wireless Sensor Networks", IEEE Access, vol. 6, pp. 28085–28096.
[17] Liu, H., Yin, R., Liu, B., & Li, Y, "A scale-free topology model with fault-tolerance and intrusion-tolerance in wireless sensor networks", Computers & Electrical Engineering, vol. 56, pp. 533–543, 2016,10.1016/j.compeleceng.2016.01.003
[18] Deniz, F., Bagci, H., Korpeoglu, I., & Yazıcı, A, "An adaptive, energy-aware and distributed fault-tolerant topology-control algorithm for heterogeneous wireless sensor networks", Ad Hoc Networks, vol. 44, pp. 104–117, 2016, 10.1016/j.adhoc.2016.02.018
[19] Yuvaraja, M., & Sabrigiriraj, M, "Fault detection and recovery scheme for routing and lifetime enhancement in WSN", Wireless Networks, vol. 23, no. 1, pp. 267–277, 2015.
[20] Xue-Wei Wang, Sang-Yang Liu, and Zhao-Hui Zhang, "Double Cluster Head-based Fault-tolerant Topology Control Algorithm for Wireless Sensor Networks", Journal of Computers, vol. 29, no. 3, pp. 19-27, 2018.