Received October 7, 2019, accepted October 18, 2019, date of publication October 31, 2019, date of current version November 20, 2019.
Digital Object Identifier 10.1109/ACCESS.2019.2950855

Research on Uneven Clustering APTEEN in CWSN Based on Ant Colony Algorithm

CAIQING WANG[1,2] AND SHUBIN WANG[1]
[1]College of Electronic Information Engineering, Inner Mongolia University, Hohhot 010021, China
[2]Inner Mongolia First Machinery Group Company, Ltd., Baotou 014032, China

Corresponding author: Shubin Wang (wangshubin@imu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61761034, and in part by the Natural Science Foundation of Inner Mongolia of China under Grant 2016MS0616.

ABSTRACT In this paper, APTEEN is applied to cognitive wireless sensor networks, and an ant colony-based uneven clustering APTEEN algorithm (ACUCAPTEEN) is proposed. The algorithm combines routing and spectrum allocation with cross-layer design method, improves energy efficient uneven clustering protocol and introduces it to APTEEN, and uses ant colony algorithm to complete inter-cluster path search, which reduces the task of cluster head. In the candidate cluster head selection process, the residual energy is not considered and the competition radius is fixed, so that the nodes with many idle channels but low residual energy become cluster heads, and the cluster heads far away from the base station still have more intra-cluster tasks with less residual energy. For these problems, an optimized ACUCAPTEEN algorithm(OACUCAPTEEN) is proposed to optimize the candidate cluster heads, the residual energy is taken as a factor of the candidate cluster head probability, and the competition radius is optimized. The simulation results show that the ACUCAPTEEN algorithm and optimization algorithm can reduce the node death rate and extend the network life cycle.

INDEX TERMS Cognitive wireless sensor network, APTEEN, ant colony algorithm, EEUC algorithm.

I. INTRODUCTION

The concept of cognitive radio was proposed by Dr. Joseph Mitola in 1999, the cognitive user perceives the idle spectrum [1], learns external environmental information and infers the information of the external environment. The detection of authorized users by cognitive users can use energy detection [2]. Cognitive Wireless Sensor Network (CWSN) is a distributed network of cognitive sensor nodes that can improve spectrum utilization [3]. The advantage of the cognitive wireless sensor network clustering routing protocol is that it facilitates topology management. The author S. Gao extends the life cycle of the sensor network by dynamically adjusting the size of the cluster [4]. Currently, Some researchers have applied the LEACH protocol to cognitive wireless sensor networks [5]. Reference [6] proposes a cluster-based spectrum-aware routing protocol (SCR) to improve energy efficiency. The spectrum changes dynamically in cognitive wireless sensor networks, researchers use a cross-layer approach to reduce the instability of network connections [7].

Adaptive Periodic Threshold-sensitive Energy Efficient Sensor Network Protocol (APTEEN) is a typical even clustering protocol, a ‘‘hot zone’’ phenomenon will occur when data is transmitted in multiple hops. APTEEN is often used in the environment for monitoring [8], and V. Devadevan and other authors use APTEEN to monitor forest fires [9]. The literature [10] uses APTEEN to regularly monitor sensor nodes and send the necessary parameters to farmers, thus helping to extend the network life cycle. The literature [11] proposed a distance adaptive threshold-sensitive energy efficient sensor network (DAPTEEN) to eliminate data redundancy and improve energy efficiency. The literature [12] proposes APTEEN with threshold energy, which improves packet loss and network lifetime. The literature [13] analytically determines the delays that occur when using enhanced APTEEN to handle various types of queries.

Energy Efficient Uneven Clustering (EEUC) can effectively reduce the task of cluster heads and thus extends the life cycle of the network. The literature [14]–[16] improved the EEUC protocol to balance the energy consumption of

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

cluster heads and extend the life of the network. Ant Colony Optimization (ACO) is used to optimize network routing. The choice of path for ants will be adjusted as the network structure changes, and no large search energy consumption will be generated. The literature [17] describes an ant colony-based cognitive radio network spectrum-aware routing (ASAR) that can adaptively resolve cognitive radio network routing. In literature [18], an extended ant colony-based cognitive wireless network routing algorithm is proposed, which combines the delay time with the idle frequency and to reduce the delay.

At present, there are few researches on clustering-based cognitive wireless sensor network routing protocols. APTEEN can respond to emergencies immediately, EEUC can avoid the ''hot zone'' phenomenon, but lacks the research of uneven clustering APTEEN applied to cognitive wireless sensor networks. The ACO algorithm has a small search energy consumption. For the above problem, this paper proposes an algorithm in the subsequent content. The novelty of the algorithm is to improve and apply the EEUC algorithm to APTEEN, and use the ant colony algorithm for cognitive wireless sensor networks to complete the path search.

The paper is organized as follows. Section II describes APTEEN and energy efficient uneven clustering algorithms. Section III proposes a uneven clustering APTEEN algorithm based on ant colony algorithm, the improved EEUC algorithm is introduced into APTEEN, and the path search between clusters is completed by ACO algorithm. Sections IV optimizes the ACUCAPTEEN algorithm and proposes the OACUCAPTEEN algorithm. First, the selection of the candidate cluster head of ACUCAPTEEN is optimized, and then the competition radius is optimized. Sections V analysis simulation results. Conclusions are in Section VI.

II. RELATED WORK

Adaptive threshold-sensitive energy efficient sensor network protocol (APTEEN) is a protocol for improving LEACH, it adds hard threshold (HT), soft threshold (ST), and counting time to the LEACH protocol, which can respond to emergencies and limit the amount of data transmitted by the node.

Energy efficient uneven clustering algorithm(EEUC) enables nodes to build clusters of different sizes, which can balance the amount of tasks undertaken by cluster heads away from the base station and close to the base station. The protocol stipulates that the nodes calculate the respective competition radius according to the distance to the base station. First, the candidate cluster head selects the cluster head within the competition radius, and then completes the cluster establishment. The cluster head near the base station has a smaller cluster radius, the cluster head away from the base station has a large cluster radius. Finally, the node enters the data transmission phase. The network model of the uneven clustering APTEEN is shown in Fig.1. Each candidate cluster head calculates the competition radius based on the distance

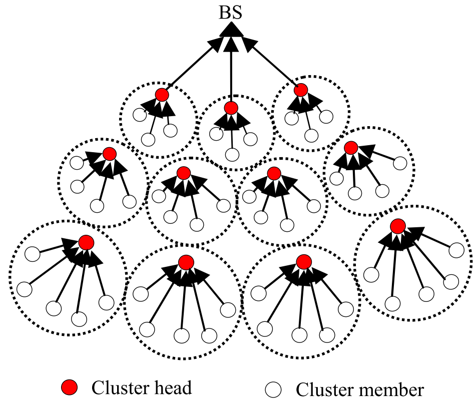

FIGURE 1. Network model based on uneven clustering APTEEN.

to the base station, the competition radius is obtained as

$$R_c = \left(1 - c\frac{d_{max} - d\left(s_i, BS\right)}{d_{max} - d_{min}}\right) R_c^0 \quad (1)$$

where d_{min},d_{max} respectively represents the minimum distance and maximum distance from the node to the base station, where $d(s_i, BS)$ represents the distance from the node to the base station. Where c represents the uneven coefficient from 0 to 1, which affects the degree of uneven of the cluster. where R_c^0 represents the maximum competition radius. The cluster head away from the base station has a large competition radius and contains many member nodes. The cluster head close to the base station contains fewer member nodes, which reduces the energy consumption in the cluster. In the election phase of the candidate cluster head, each node generates a random number from 0 to 1, if the random number is greater than 0.4, the node will be selected as the candidate cluster head, and the cluster head is selected from candidate cluster head.

III. ANT COLONY BASED UNEVEN CLUSTERING APTEEN ROUTING ALGORITHM

When APTEEN transmits data in multi-hop format, the cluster head close to the base undertake more tasks, which will reduce the life cycle and increase the number of dead node. For the problem of APTEEN, this chapter proposes an ant colony based uneven clustering adaptive threshold-sensitive energy efficient sensor network protocol (ACUCAPTEEN). ACUCAPTEEN improves the EEUC algorithm. It is introduced into APTEEN, which reduces the tasks undertaken by the cluster head close to the base station. and uses the ant colony algorithm to search the path between cluster, thereby extending the life cycle of the network. The ACUCAPTEEN algorithm consists of four phases, namely the spectrum sensing phase, the cluster establishment phase, the ant colony-based path search and route update phase, and the data transmission phase.

A. SPECTRUM SENSING STAGE

The condition that node X_i communicates with neighboring node X_j in cognitive wireless sensor networks is that they have at least one identical idle channel. Before communication, X_i and X_j perform spectrum sensing, respectively obtains a set of idle channels. Then it should be determined whether there is an identical channel between the set of idle channels of X_i and the set of idle channels of X_j. If there is at least one same channel, then X_i can communicate with X_j, otherwise communication cannot be performed. The ACUCAPTEEN algorithm uses an energy detection method to detect authorized users at this stage.

The ACUCAPTEEN algorithm improves the election of candidate cluster heads. In the election of candidate cluster heads, the channel influence factor is added. The cluster establishment phase is divided into three steps. In the first step,the node selects the candidate cluster head according to the number of idle channels. In the second step, the final cluster head is selected from the candidate cluster head, and in the third step, the cluster head establishes a cluster with the surrounding nodes. The three stages are described separately below.

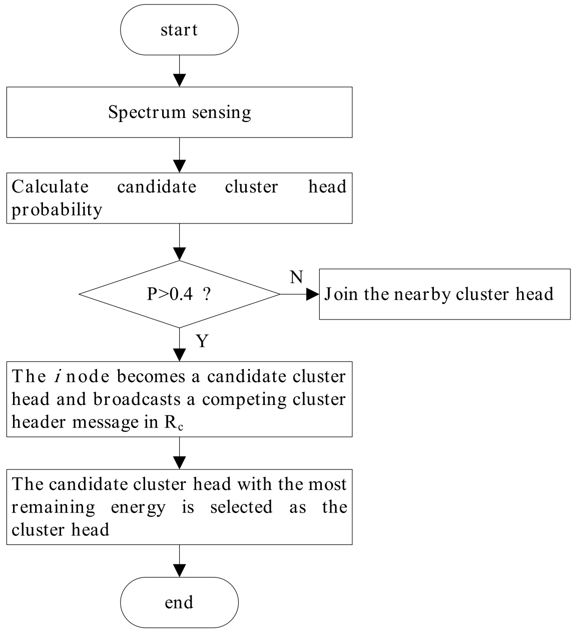

FIGURE 2. Cluster head election flow char.

B. CLUSTER ESTABLISHMENT PHASE

Step 1) In the candidate cluster head election stage, the selection of traditional candidate cluster heads is often applied to the wireless sensor network clustering routing protocol, which cannot be directly applied to cognitive wireless sensor networks, and should be improved to adapt to the changing spectrum of cognitive wireless sensor networks. In this chapter, the number of idle channels perceived by the cognitive node is taken as the factor that affects the candidate cluster head selection. The candidate cluster head probability formula is given as

$$P_i(t) = min\left(k\alpha\frac{c_i}{y}, 1\right) \tag{2}$$

where k represents the expected number of cluster heads, where y represents the total number of channels, where α represents a fixed parameter, and where c_i represents the number of idle channels perceived by the node. A node with $P_i(t) > 0.4$ becomes a candidate cluster head.

Step 2) In the election phase of the cluster head, candidate cluster heads are selected according to (2). Each candidate cluster head broadcasts a CH-Competition message on the common control channel, the content of the message contains the node ID, the competition radius, and the remaining energy. The cluster head election flow chart is shown in Fig.2. The competition radius is related to the uneven coefficient c. c affects the number of nodesjoining the cluster and the energy consumption of the cluster head. The more members in each cluster, the more energy thecluster head consumes. After receiving the CH-Competition message, the candidate cluster head establishes a set of neighboring candidate cluster heads, and selects the candidate cluster head with the most remaining energy as the cluster head. After the cluster head is determined, other candidate cluster heads in the competition radius do not perform the election of cluster head, and become ordinary nodes.

Step 3) In the establishment phase of the cluster, the cluster head broadcasts a Competition-success message on the common control channel, and the content includes ID, a list of idle channels. After receiving the message, the node in the competition radius compares its own channel list with the channel list of the cluster head, if it has the same channel as the cluster head, it sends the message CM-Request-Join to the cluster head, and the message content includes node ID and the idle channel list. Based on the received message, the cluster head selects the idle channel shared by most nodes as the channel, and broadcasts the message Final-Channel on the common control channel, the message includs the final channel ID. After receiving the message Final-Channel, the node joins the cluster at the lowest cost. After the establishment of the cluster, the cluster head sends a TDMA slot table to the member, broadcasting a hard threshold, a soft threshold, and a counting time.

C. ANT COLONY BASED PATH SEARCH AND UPDATE

After the establishment of the cluster, the ACO algorithm is used to search the path between the clusters, the number of idle channels and energy consumption are used as heuristic information. The transition probability is shown as

$$P_{ij}^k = \begin{cases} \dfrac{\tau_{ij}^\alpha \delta_{ij}^\gamma l_{ij}^\beta}{\sum \tau_{ij}^\alpha \delta_{ij}^\gamma l_{ij}^\beta} & \text{if } j \in allowed \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where P_{ij}^k represents the transition probability from node i to node j for the kth ant at time t, and where l_{ij} represents the number of idle channels perceived by the neighboring cluster head, that is, it represents the idle channel number heuristic information. Where $\delta_{ij} = 1/d_{ij}^2$ represents of energy consumption, where τ_{ij} represents the pheromone concentration on the path from node i to node j, where α, γ and β respectively pheromone weight value, heuristic factor weight value, idle channel weight value. where allowed represents the set of nodes that the kth ant can access. Ants are divided into forward ants and backward ants. The f-ant is transmitted on the control channel, CH_i and CH_j are adjacent, if the number of idle channels $l_{ij} > 0$,the address and the number of idle channels of CH_j are put into f-ant. The f-ant selects the next hop according to (3), and its function is to find a spectrum feasible path from CH_i to base station, there are common channels between adjacent nodes, and the number of path search times is reduced. Path search is performed before data are sent, the f-ant is released at the cluster head, b-ant organizes network information, and updates the cluster head routing table. The pheromone update rule is shown as

$$\tau_{CH_{ij}}(t+1) = (1-\rho)\cdot\tau_{CH_{ij}}(t)+\Delta\tau_{CH_{ij}}(t,t+1) \tag{4}$$

$$\Delta\tau_{CH_{ij}}(t,t+1) = \sum_{k=1}^{m}\Delta\tau_{CH_{ij}}^{k}(t,t+1) \tag{5}$$

where ρ represents a pheromone evaporation coefficient, where $\tau_{CH_{ij}}(t)$ represents pheromone concentration on the path from cluster head i to cluster head j, where $\Delta\tau_{CH_{ij}}(t, t+1)$ represents pheromone concentration released by all ants on the path (i, j) from time t to time $t+1$. After a f-ant arrives at base station from cluster head, its collected information is stored in base station. When all f-ant arrive at base station, the path taken by one of f-ant is the best path. The calculation formula of the best path is shown as

$$w = \frac{M}{L} \tag{6}$$

where M represents the total number of idle channels collected by the f-ant through multiple cluster heads, and where L represents the total path length from a cluster head to base station. The larger the optimal path value, the better the path. After all f-ant arrive at the base station, b-ant is sent to the best path selected by f-ant for global pheromone update. Where $\Delta\tau_{CH_{ij}}^{k}(t,t+1)$ represents the concentration of pheromone released for the kth ant from time t to time $t+1$. The formula is shown as

$$\Delta\tau_{CH_{ij}}^{k}(t,t+1) = \frac{M}{L^2} \tag{7}$$

D. DATA TRANSMISSION PHASE

After the establishment of the cluster is completed, the cluster head broadcasts the TDMA slot table, hard threshold, soft threshold, and the counting time to the members. When the member receives hard threshold, soft threshold, and the counting time, the data of the first sensing is first judged. If the data is greater than hard threshold, then a data is saved to the instantaneous variable, the data is sent to the cluster head. The cluster head transmits data to base station along the path selected by the ACO algorithm. In each subsequent transmission process, the data transmitted this time will be compared with the data stored the instantaneous variable last time. Only when the amount of change is greater than soft threshold, the member transmits some datas to the cluster head.

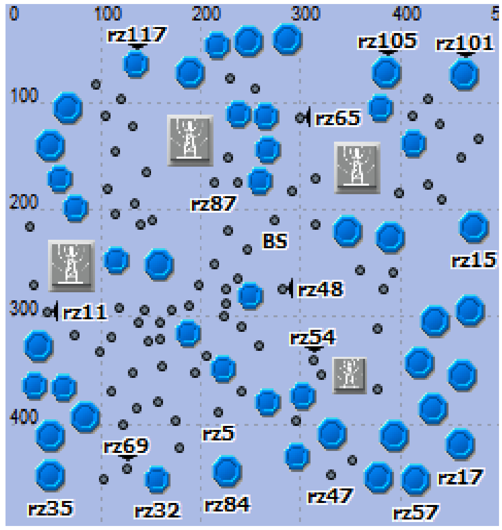

FIGURE 3. Cognitive wireless sensor network model, 121 cognitive users and 5 authorized users distribute randomly in a 500×500m² field.

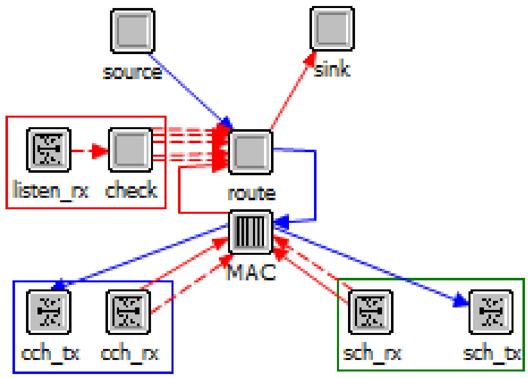

FIGURE 4. Node model cognitive user, the check module detects the spectrum and sends the result to the route module through the status line.

E. COGNITIVE WIRELESS SENSOR NETWORK NETWORK MODELING IN OPNET

OPNET is a software for communication network simulation. This section introduces the modeling of cognitive wireless sensor networks in OPNET according to a three-layer modeling mechanism. Know that the cognitive user is named rz_i, any authorized user is named SQ_i.

The network layer model of cognitive wireless sensor networks is shown in Fig.3. The node model of the cognitive user is shown in Fig.4. In the node model, a spectrum detection module is added to detect whether the licensed band is idle.

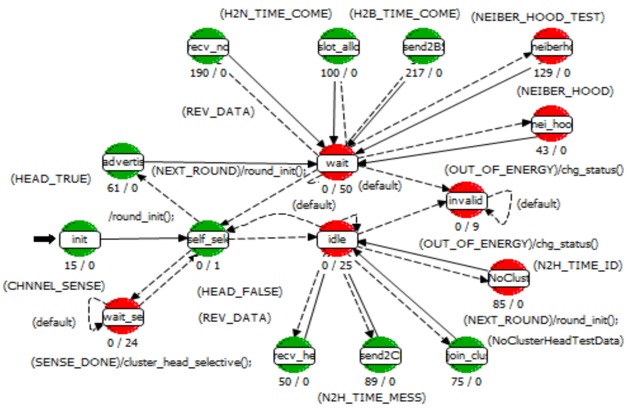

FIGURE 5. **Process model of route module, Cluster_head_selective() determines the number of idle channels, and the neiberhood_route state machine causes all neighbor cluster heads to broadcast f-ant.**

The source module implements a collection service source with a fixed arrival interval. The source module randomly generates arrivalinterval. The source module randomly generates data packets distributed from 0 to 100. The MAC module is interval. The source module randomly generates data packets distributed from 0 to 100. The MAC module is used for multi-channel packet access, transmitting control packets on the control channel, and transmitting data packets on the data channel. The cch_tx and cch_rx modules represent the wireless transceivers of the control channel, completing the transmission and reception of control packets. The sch_tx and sch_rx modules complete the transmission and reception of data packets. In the authorized user node model,authorized users can take packets and send packets through the ON-OFF model carried by OPNET. You can set the frequency band used by authorized users, the duration of the busy and idle states of the channel can be set in the ON-OFF model.

The process model of the cognitive user node model route module is shown in Fig.5. When establishing the cognitive node model, the model should include a spectrum detection device, including the check module and the listen_tx module. The check module detects the energy of each channel and determines whether the current channel is busy by the energy estimation. The status line of the listen_tx module to the check module represents the statistic, and a rising edge trigger mechanism is used to send energy information to the check state machine. The check module detects the spectrum and sends the result to the route module through the status line. After the init state machine generates the spectrum start detection interrupt event and the spectrum detection end interrupt event, the initialization registration statistic is performed, and then the state machine enters the self_select state. When the program performs spectrum detection to start an interrupt response, the process state transitions from the self_select state to the wait_sense state. When the program performs the spectrum detection end interrupt response, the process state is transferred from wait_sense to the self_select state, and the cluster_head_selective() is executed. Cluster_head_selective() determines the number of idle channels, selects the candidate cluster head according to (1), and the candidate cluster head selects the cluster head according to the remaining energy. When the node becomes the cluster head, the state machine self_select enters the advertis state, and the advertis state machine broadcasts a packet containing a uneven radius, a free channel, and a cluster head ID. After the cluster member enters the idle by self_select, it enters the join_cluster state. The join_cluster state machine parses the uneven radius and the idle channel in the received broadcast packet to determine whether the node becomes a cluster member of the cluster in which the cluster head is located. If it becomes a cluster member, the package containing its own ID is unicast to the cluster head. In the nei_hood state, the cluster head broadcasts a packet with a uneven radius. Only after the cluster head receives the broadcast packet, it parses, and selects a neighbor cluster head list according to the uneven radius selection. Neiberhood_route first broadcasts neighbor cluster head table to forward ants. When all forward ants arrive at the base station, base station selects an optimal path. The base station sends the backward ants along the best path to update the pheromone concentration through the node. The wait_sense state machine is used to process interrupt events. The channel busy state is changed by the trigger of the interrupt event. The Wait state machine takes a change in the channel and broadcasts a notification packet. The send2BS state machine is used by the cluster head to send data to the base station. send2CH completes the data transfer within the cluster. The slot_allol state machine obtains the member list size by the op_prg_list_size().

IV. OPTIMIZATION OF ANT COLONY BASED UNEVEN CLUSTERING APTEEN IN CWSN

The ACUCAPTEEN algorithm uses the idle channel as a factor of the candidate cluster head probability. Only the idle channel is considered. Without considering the residual energy, the node with many idle channels but low residual energy becomes the cluster head, which reduces the network life cycle. A fixed competition radius results in cluster heads far from the base station still having more intra-cluster tasks with less residual energy. Aiming at these problems, this paper optimizes the ACUCAPTEEN algorithm, which is optimized in two stages. In the first stage, during the ACUCAPTEEN candidate cluster head election process, the candidate cluster head is optimized, and the residual energy is taken as a factor of the candidate cluster head probability. The improved ant colony based uneven clustering APTEEN algorithm (IACUCAPTEEN) is proposed. In the second stage, the IACUCAPTEEN competition radius is optimized, further improved ant colony based uneven clustering APTEEN algorithm (FIACUCAPTEEN) is proposed.

A. CANDIDATE CLUSTER HEAD OPTIMIZATION

The ACUCAPTEEN algorithm divides the cluster head election into two phases. In the first phase, the candidate cluster head election process uses the number of idle channels as the impact factor. In the second phase, the candidate cluster heads from its neighbor candidate cluster heads. The candidate cluster head with the most remaining energy is selected as the cluster head, and the inter-cluster path search is performed by the ACO algorithm. ACUCAPTEEN considers the channel in the candidate cluster head selection, and does not consider the residual energy at the same time, so that the node with low residual energy but perceives the number of idle channels becomes the candidate cluster head, which leads to the selection of residual energy from the candidate cluster head. The low cluster head makes the cluster head with lower residual energy die earlier.

For the problem that ACUCAPTEEN does not consider residual energy in the candidate cluster head election process, this section optimizes the candidate cluster head selection and proposes IACUCAPTEEN, which establishes a new candidate cluster head selection formula, The formula is shown as

$$P_i(t) = \frac{E_{cur}}{E_{int}} min\left(k\alpha\frac{c_i}{y}, 1\right) \tag{8}$$

where E_{cur} represents the residual energy of the node, where E_{int} represents the starting energy of the node. The improved candidate cluster head selection formula adds the residual energy impact factor, that is, the candidate cluster head selection probability $P_i(t)$ is related to the remaining energy and the number of idle channels, and the number of channels is perceived to be more and remaining. A node with a large amount of energy is likely to be a candidate cluster head, so that a candidate cluster head having a large number of idle channels and a large amount of remaining energy is selected as a cluster head. The IACUCAPTEEN algorithm still uses the node with $P_i(t) > 0.4$ as the candidate cluster head. The candidate cluster head selects the candidate cluster head with the most remaining energy from the neighbor candidate cluster head set as the cluster head, then completes the cluster establishment and uses the ACO algorithm to perform the path. Search, and finally enter the data transfer phase.

The IACUCAPTEEN algorithm adds energy factors to the candidate cluster head elections, so that nodes with more channels and more residual energy have a greater probability of becoming candidate cluster heads, while cluster heads are under more idle channels and residual energy. The probability that the same node has the same channel is larger, so the possibility of stable communication between the cluster head and the nearby nodes in the clustered cognitive wireless sensor network is increased, thereby extending the life cycle of the network.

B. COMPETITION RADIUS OPTIMIZATION

The above IACUCAPTEEN algorithm changes the candidate cluster head selection probability formula, so that the candidate cluster head selects two influence factors including channel and residual energy. This document specifies that the location of node in CWSN will not change. The competition radius of the node is related to c and the distance to the base station. the IACUCAPTEEN algorithm takes c = 0.5.

The fixed radius of competition will make the cluster head away from the base station still have a large competition radius even if there is less residual energy, which leads to more members in the cluster head, which increases the task of the cluster head in the cluster, making it more Large clusters of energy consumption die. It can be seen from the above analysis that the IACUCAPTEEN algorithm is insufficient. This section optimizes the competition radius for the problem of fixed IACUCAPTEEN competition radius, and proposes the FIACUCAPTEEN algorithm. The calculation formula of the competition radius of the algorithm is shown as

$$R_c = (1 - exp(-\frac{E_{cur}}{E_{ini}})\frac{d_{max} - d(s_i, BS)}{d_{max} - d_{min}})R_c^0 \tag{9}$$

where E_{cur} represents the residual energy of the node, where E_{ini} represents the starting energy of the node. The FIACUCAPTEEN algorithm makes the competition radius of the cluster head change with the change of residual energy. When the cluster head has less residual energy, it has a smaller competition radius, which can reduce the intra-cluster tasks undertaken by the cluster head and extend the life cycle of the network. The FIACUCAPTEEN algorithm only changes the competitive radius formula, it still includes the spectrum sensing phase, the candidate cluster head election phase, the cluster head election phase, the cluster establishment phase, the path search phase, and the data transmission phase. The path search phase still uses the ACO algorithm in IACUCAPTEEN.

V. SIMULATION ANALYSIS

APTEEN is a uniform clustering protocol. In this paper, the ant colony algorithm is introduced into APTEEN and uneven clustering APTEEN for simulation comparison. It is represented by ACAPTEEN algorithm and ACUCAPTEEN algorithm respectively. The purpose is to compare the ant colony-based uniform clustering APTEEN algorithm (ACAPTEEN) with the network life cycle based on the ant colony uneven clustering APTEEN algorithm. In this paper, the cognitive wireless sensor network using ACUCAPTEEN algorithm, ACAPTEEN algorithm and APTEEN algorithm is modeled by OPNET simulation software. The cognitive wireless sensor network has a topological range of 500m $*$500m, and has 121 cognitive users distributed. There are 5 authorized users, and any authorized user has an authorization channel with a total of 5 authorized channels. Detailed simulation parameters are shown in Table 1.

The simulation graph of the number of dead node is shown in Fig.6. The number of dead node in the cognitive wireless sensor network using the ACUCAPTEEN algorithm is simulated. After the simulation is over, the number of dead node in the network is observed. The horizontal axis time represents time, and the vertical axis Death Node Num represents the number of node deaths. Death Node Num increases with time. At t = 1762s, node in ACAPTEEN begins to die. At t = 2668s, Death Node Num is equal to 6, at t = 3212s, Death Node Num is equal to 11, at t = 4363s, The

TABLE 1. Experimental simulation parameter.

Parameter	Value
topological range	500m*500m
BS	(275m, 218m)
cognitive user	121
authorized user	5
channel	5
R_c^0	90
α	2
β	4
γ	3
ρ	0.5
c	0.5

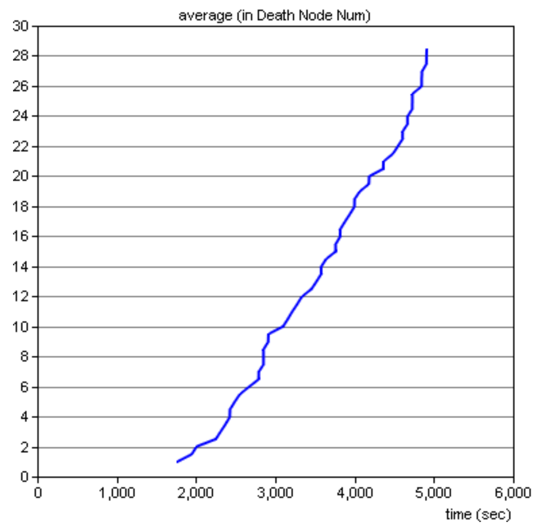

FIGURE 6. Number of dead node of ACUCAPTEEN.

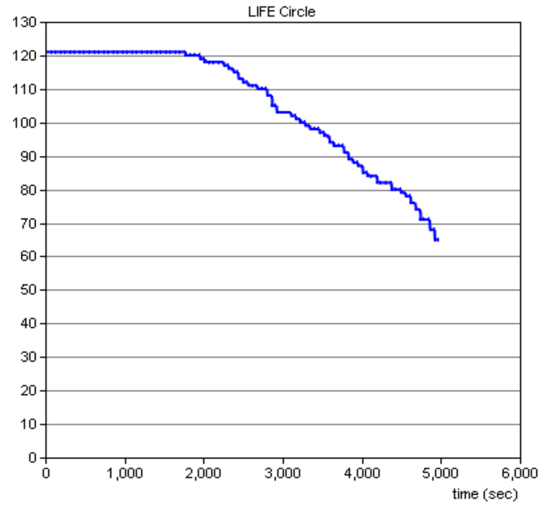

FIGURE 7. Life cycle of ACUCAPTEEN.

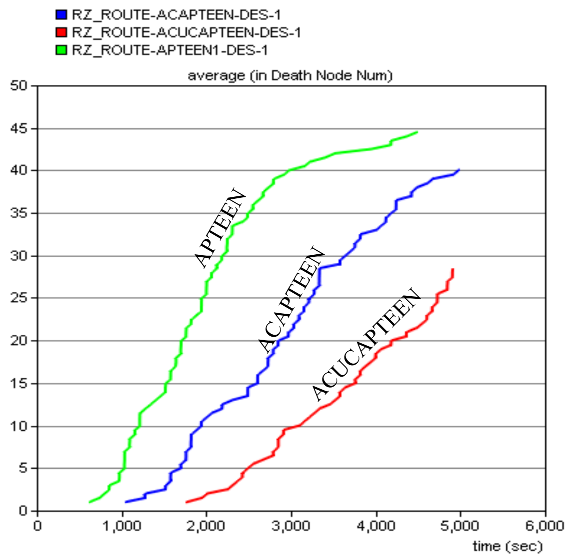

FIGURE 8. Comparison of the number of death node.

Death Node Num is equal to 21. Observing the data, the Death Node Num slowly rises as time goes by. Because the node transmits and receives data, and energy consumption is generated. Therefore, the energy of the node continues to decrease with the continuation of time. The life cycle of ACUCAPTEEN is shown in Fig.7. It can be seen from the figure that the life cycle decreases with time. The horizontal axis time represents time, and the vertical axis LIFE Circle represents the life cycle of the node. It can be obtained from the figure. At t = 1762s, LIFE Circle starts to fall and the curve slowly decreases. At t = 2650s, LIFE Circle is equal to 111. At t = 3350s, LIFE Circle is 99, at t = 4350s, LIFE Circle is equal to 82. Observing the data, LIFE Circle keeps decreasing with time. Because the life cycle of the network is represented by the number of nodes surviving. As the simulation progresses, the energy of the node decreases and the number of dead node gradually increases. The number of remaining node is getting smaller and smaller. Comparing the number of dead node in ACUCAPTEEN, ACAPTEEN, and APTEEN, as shown in Fig.8 At t =615s, APTEEN's Death Node Num begins to increase. The Death Node Num of ACUCAPTEEN and ACAPTEEN began to increase at t =1762s and t =1038s, respectively. Compared with ACAPTEEN, the number of death node in ACUCAPTEEN decreased by an average of 10.3%. The ACUCAPTEEN node dies slowly, because ACUCAPTEEN uses a uneven clustering algorith-m, which reduces the amount of tasks undertaken by the cluster head close to the base station and extends the life of the cluster head. Comparing the life cycle of ACUCAPTEEN, ACAPTEEN and APTEEN, as shown in Fig.9. APTEEN's LIFE Circle has the fastest drop rate. Compared with APTEEN, ACUCAPTEEN's LIFE Circle and ACAPTEEN's LIFE Circle fall slower. Because APTEEN enables the cluster head to transmit a large amount of data at a time to generate a large energy consumption,which reduces the LIFE Circle. ACUCAPTEEN and ACAPTEEN transmit data in a

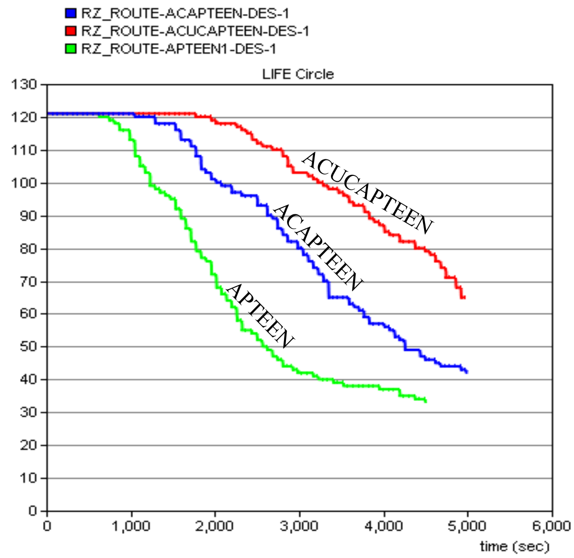

FIGURE 9. Comparison of life cycle.

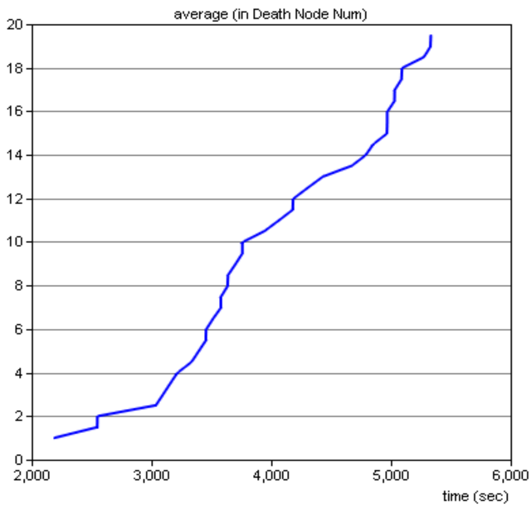

FIGURE 10. Number of dead node of IACUCAPTEEN.

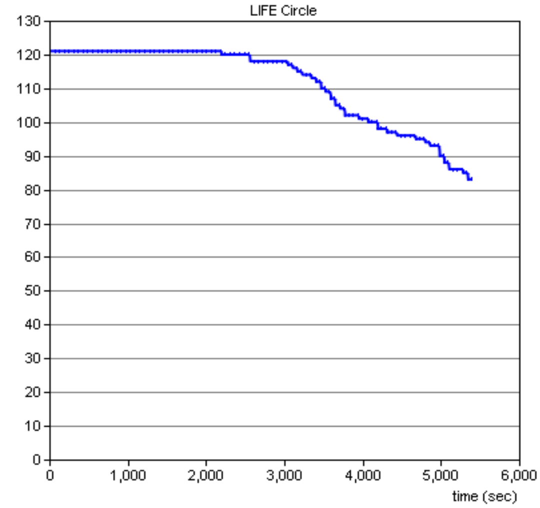

FIGURE 11. Life cycle of IACUCAPTEEN.

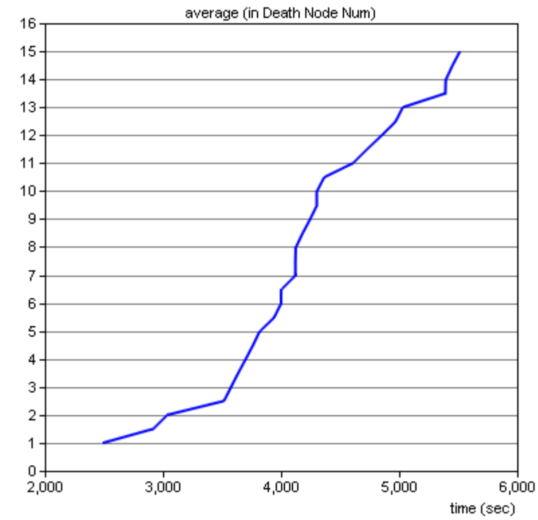

FIGURE 12. Number of dead node of FIACUCAPTEEN.

multi-hop format. the cluster head transmits the data to the base station through multiple intermediate cluster heads. This can effectively reduce the energy consumed by the cluster head for a single transmission of data, so that LIFE Circle has a larger value. Compared with the ACAPTEEN algorithm, the network life cycle of ACUCAPTEEN is extended by an average of 20.2%. According to the above simulation, Fig.6-9 show that the number of dead node and life cycle change with time after using the ACUCAPTEEN algorithm, that is, Fig.6-9 give a comparison of life cycle to demonstrate that the ACUCAPTEEN algorithm can extend the network life cycle.

The number of dead node of IACUCAPTEEN is shown in Fig.10. The life cycle of IACUCAPTEEN is shown in Fig.11. The life cycle continues to decrease over time. The horizontal axis represents time and the vertical axis LIFE Circle represents the life cycle of the network. At $t = 2183$s, LIFE Circle begins to fall and the curve slowly decreases. At $t = 2750$s, LIFE Circle is equal to 117, at $t = 3600$s, LIFE Circle is equal to 105, and at $t = 4250$s, LIFE Circle is equal to 97. Observing the data, LIFE Circle keeps decreasing with time. Because the life cycle of the network is represented by the number of survival points. As the simulation progresses, the number of remaining nodes is reduced. The number of FIACUCAPTEEN's dead node is shown in Fig.12. At $t = 2488$s, FIACUCAPTEEN's node begin to die. At $t = 4500$s, Death Node Num is equal to 11, at $t = 5028$s, Death Node Num is equal to 13, andat $t = 5512$s, Death Node Num is equal to 15. Observing the data, the Death Node Num curve rises slowly as time goes on. Because the node transmits and receives data, and energy consumption is generated. Therefore, the energy of the node continues to decrease with time, and when the node energy is completely consumed. At the end of the day, the node was declared dead.

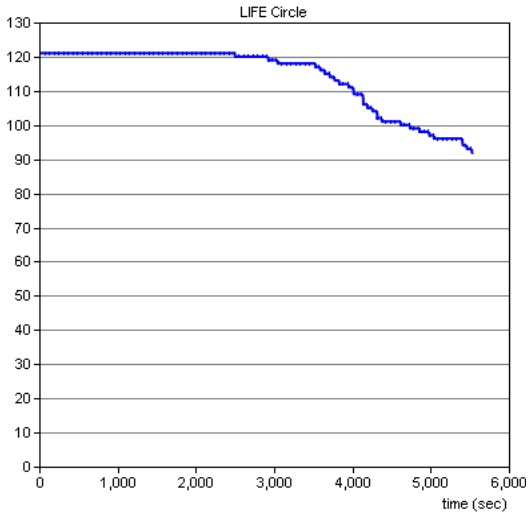

FIGURE 13. Life cycle of FIACUCAPTEEN.

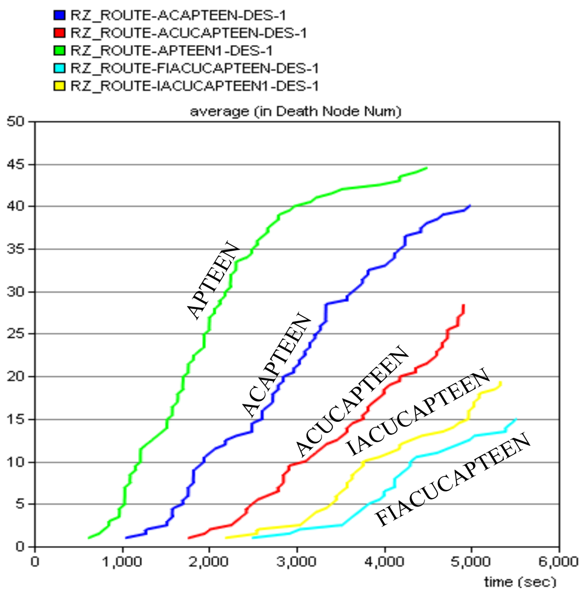

FIGURE 14. Comparison of the number of death node.

The life circle of FIACUCAPTEEN is shown in Fig.13. At t = 3750s, LIFE Circle is equal to 114. At t = 4250s, LIFE Circle is equal to 106, At t = 5000s, and LIFE Circle is equal to 97. It can be seen from the figure that at t = 2488s, LIFE Circle starts to fall and the curve slowly declines. Because the life cycle of the network is reprsented by the number of nodes surviving. As the simulation progresses, the energy of the node decreases continuously, and the dead node number gradually increases, resulting in fewer and fewer remaining node. The comparison of the number of dead node of APTEEN, ACAPTEEN, ACUCAPTEEN, IACUCAPTEEN and IACUCAPTEEN is shown in Fig.14. The number of death nodes and life cycle of APTEEN, ACAPTEEN, ACUCAPTEEN, FIACUCAPTEEN and IACUCAPTEEN are simulated and

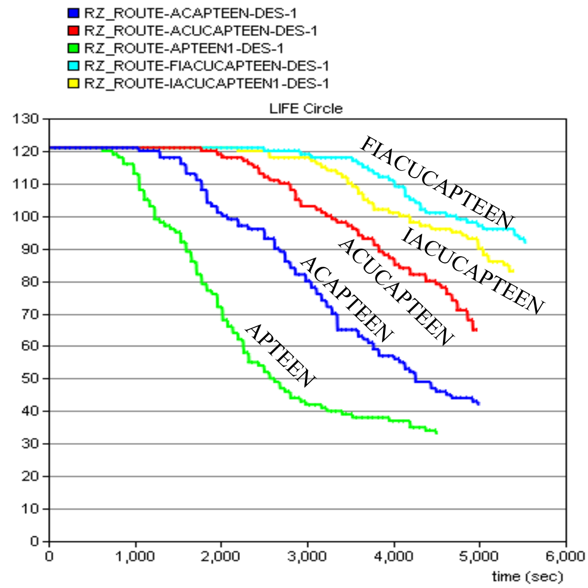

FIGURE 15. Comparison of life cycle.

compared respectively. At t = 1762s, the Death Node Num of ACUCAPTEEN begins to increase. At t = 2183s, the IACUCAPTEN's Death Node Num begins to increase. At t = 2488s,the FIACUCAPTEEN's Death Node Num begins to increase. In addition, compare ACUCAPTEEN, IACUCAPTEEN, and FIACUCAPTEEN at the same time as the Death Node Num. At t = 4302s, ACUCAPTEEN's Death Node Num is 20, IACUCAPTEEN's Death Node Num is 13, and FIACUCAPTEEN's Death Node Num is 10. Therefore, according to the data comparison in the figure, IACUCAPTEEN extends the node's initial death time, ACUCAPTEEN's node death rate is faster, and IACUCAPTEEN's node death rate is slower. Because IACUCAPTEEN considers the remaining energy of the candidate cluster head, it makes the selected cluster head have more residual energy, which reduces the death speed of the cluster head. The number of death nodes of FIACUCAPTEEN decreased by 5% on average compared with IACUCAPTEEN. FIACUCAPTEEN extends the node's initial death time, the death rate is slower, and the IACUCAPTEEN's nodes die faster. Because FIACUCAPTEEN has a smaller competition radius when the cluster head has less residual energy, which reduces the cluster head death rate. Compared with ACUCAPTEEN, the number of death node of FIACUCAPTEEN decreased by an average of 11%. Because FIACUCAPTEEN considers the remaining energy of the candidate cluster head and balances the remaining energy and the competition radius of the cluster head.

The life cycle comparison between APTEEN, ACAPTEEN, ACUCAPTEEN, FIACUCAPTEEN and IACUCAPTEEN is shown in Fig.15. At t = 1762s, the LIFE Circle of ACUCAPTEEN began to drop. At t = 2183s, the LIFE Circle of IACUCAPTEEN began to drop. At t = 2488s, the LIFE Circle of FIACUCAPTEEN began to drop. In addition, comparing ACUCAPTEEN, IACUCAPTEEN, and FIACUCAPTEEN at the same time LIFE Circle, at t = 3900s, ACUCAPTEEN's LIFE Circle is 89, IACUCAPTEEN's LIFE Circle is 102, and FIACUCAPTEEN's LIFE Circle is 114. According to the comparison of simulation results, compared with ACUCAPTEEN, the LIFE Circle curve of IACUCAPTEEN drops slowly, and the LIFE Circle of ACUCAPTEEN declines faster. Because IACUCAPTEEN adds energy factor in the candidate cluster head election process, IACUCAPTEEN adds energy factor in the candidate cluster head election process, and selects more idle. The candidate cluster heads of the channel and residual energy are cluster heads, which reduces the energy consumption of the cluster head, reduces the number of node deaths, and extends the life cycle of the network. Compared to IACUCAPTEEN, the network life cycle of FIACUCAPTEEN is extended by an average of 9.9%. The LIFE Circle curve of FIACUCAPTEEN drops slowly and the LIFE Circle of IACUCAPTEEN drops faster. Because FIACUCAPTEEN has a smaller competition radius when the cluster head has less residual energy, which reduces the energy consumption in the cluster head. The rate of node death in the network. Compared with ACUCAPTEEN, the network life cycle of FIACUCAPTEEN is extended by 20.6%. Because FIACUCAPTEEN balances the residual energy and competition radius of the cluster head. When the cluster head has less residual energy, it has a smaller competition radius and can be extended. The life cycle of the network reduces the death rate of nodes. According to the above simulation, Fig.10-15 show that the number of death node and the life cycle change with time after using the proposed algorithms, that is, Fig.10-15 give a comparison of life cycle to demonstrate that the proposal algorithms can extend the network life cycle.

VI. CONCLUSION

In this paper, the ant colony-based uneven clustering APTEEN algorithm ACUCAPTEEN, and the optimization algorithms IACUCAPTEEN and FIACUCAPTEEN are proposed. The OPNET simulation tool verifies that the ACUCAPTEEN algorithm can reduce the workload of cluster heads close to the base station and extend the network life cycle. The ACUCAPTEEN algorithm is optimized for the ACUCAPTEEN candidate cluster head without considering the residual energy and fixed competition radius. The algorithm is optimized in two stages. In the first stage,the ACUCAPTEEN candidate cluster head election process optimizes the candidate through the IACUCAPTEEN algorithm. Cluster head, the second stage optimizes the IACUCAPTEEN competition radius by the FIACUCAPTEEN algorithm. The OPNET simulation tool verifies that FIACUCAPTEEN can balance the residual energy and competition radius of the cluster head. When the cluster head has less residual energy, it has a smaller competition radius, extends the network life cycle and reduces the node death speed.

The cognitive wireless sensor network clustering routing algorithm studied in this paper is not a routing algorithm suitable for any requirement. This paper proposes the ACUCAPTEEN algorithm and optimization algorithm, which is only applicable to the occasions of balanced spectrum utilization and network life cycle. It is not suitable for occasions with high spectrum utilization but short network life cycle. When studying cognitive wireless sensor networks, a large number of nodes should be arranged in the network. The cognitive wireless sensor network topology range is not large enough and the number of node is not enough. Therefore, the cognitive wireless sensor network topology range and the number of node should be increased, which is the next research content.

REFERENCES

[1] J. Mitola and G. Q. Maguire, "Cognitive radio: Making software radios more personal," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13–18, Aug. 1999.

[2] S. C. Shinde and A. N. Jadhav, "Centralized cooperative spectrum sensing with energy detecion in cognitive radio and optimization," in *Proc. IEEE Int. Conf. Recent Trends Electron., Inf. Commun. Technol. (RTEICT)*, May 2016, pp. 1002–1006.

[3] P. Spachos and D. Hantzinakos, "Scalable dynamic routing protocol for cognitive radio sensor networks," *IEEE Sensors J.*, vol. 14, no. 7, pp. 2257–2266, Jul. 2014.

[4] S. Gao, L. Qian, D. R. Vaman, and Q. Qu, "Energy efficient adaptive modulation in wireless cognitive radio sensor networks," in *Proc. IEEE Int. Conf. Commun.*, Glasgow, U.K.: IEEE Press, Jun. 2007, pp. 3980–3986.

[5] R. M. Eletreby, H. M. Elsayed, and M. M. Khairy, "CogLEACH: A spectrum aware clustering protocol for cognitive radio sensor networks," in *Proc. 9th Int. Conf. Cogn. Radio Oriented Wireless Netw. Commun. (CROWNCOM)*, Jun. 2014, pp. 179–184.

[6] G. A. Shah and O. B. Akan, "Spectrum-aware cluster-based routing for cognitive radio sensor networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 2885–2889.

[7] G. P. Joshi, S. W. Kim, and S. Y. Nam, "Routing layer solution for mitigating frequent channel switching in ad hoc cognitive radio networks," *IEEE Commun. Lett.*, vol. 19, no. 11, pp. 1917–1920, Nov. 2015.

[8] J. Ma, S. Wang, C. Meng, Y. Ge, and J. Du, "Hybrid energy-efficient APTEEN protocol based on ant colony algorithm in wireless sensor network," *EURASIP J. Wireless Commun. Netw.*, vol. 102, pp. 1–13, Dec. 2018.

[9] V. Devadevan and S. Suresh, "Energy efficient routing protocol in forest fire detection system," in *Proc. IEEE 6th Int. Conf. Adv. Comput. (IACC)*, Feb. 2016, pp. 618–622.

[10] S. Bhagyashree, S. Prashanthi, and K. M. Anandkumar, "Enhancing network lifetime in precision agriculture using apteen protocol," in *Proc. IEEE Technol. Innov. ICT Agricult. Rural Develop. (TIAR)*, Jul. 2015, pp. 44–48.

[11] A. Garg, "Distance adaptive threshold sensitive energy efficient sensor network (DAPTEEN) protocol in WSN," in *Proc. Int. Conf. Signal Process., Comput. Control (ISPCC)*, Sep. 2015, pp. 114–119.

[12] S. S. More and K. M. Nighot, "Optimization of wireless sensor networks using artificial intelligence and ant colony optimization for minimizing energy of network and increasing network lifetime," in *Proc. Int. Conf. Comput., Commun., Control Automat. (ICCUBEA)*, Aug. 2017, pp. 1–6.

[13] A. Manjeshwar, Q.-A. Zeng, and D. P. Agrawal, "An analytical model for information retrieval in wireless sensor networks using enhanced APTEEN protocol," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 12, pp. 1290–1302, Dec. 2002.

[14] C.-F. Li, G.-H. Chen, M. Ye, and J. Wu, "An uneven cluster-based routing protocol for wireless sensor networks," *J. Comput.*, vol. 30, no. 1, pp. 27–36, 2007.

[15] C.-J. Jiang, W.-R. Shi, M. Xiang, and X.-L. Tang, "Energy-balanced unequal clustering routing protocol for wireless sensor networks," *J. Softw.*, vol. 23, no. 5, pp. 1222–1232, 2012.

[16] D. Min and X. Chun, "An improved cluster protocol design method for low energy uneven wireless sensor network," in *Proc. Int. Conf. Comput. Comput. Sci. (ICCCS)*, Jan. 2015, pp. 189–192.

[17] B. Li, D. Li, Q.-H. Wu, and H. Li, "ASAR: Ant-based spectrum aware routing for cognitive radio networks," in *Proc. Int. Conf. Wireless Commun. Signal Process.*, vol. 5, Nov. 2009, pp. 1–5.
[18] Q. Hu, S. Zhang, and Z. Zhou, "An extended ant colony based routing algorithm for cognitive radio networks," in *Proc. World Automat. Congr.*, Jun. 2012, pp. 301–303.

CAIQING WANG was born in Inner Mongolia, China, in 1991. She received the M.S. degree in electronic information engineering from Inner Mongolia University, China, in 2019. Her research interests include communication channels and cognitive wireless sensor networks.

SHUBIN WANG was born in Chifeng, Inner Mongolia, China, in 1971. He received the B.S. and M.S. degrees in physics and electronic engineering from Inner Mongolia University, China, in 1996 and 2003, respectively, and the Ph.D. degree from the Beijing University of Posts and Telecommunications, in 2010.

From 2009 to 2010, he was a Research Fellow with Inha University, Inchon, South Korea, by the supervision of Prof. K. S. Kwak. Since 2013, he has been a Professor with the College of Electronic Information Engineering, Inner Mongolia University. He is the author of more than 40 articles, more than seven inventions, and holds four Chinese patents. His research interests include wireless sensor networks, cognitive radio networks, machine vision, binocular stereo vision, and signal processing.

Dr. Wang was a recipient of the Inner Mongolia Autonomous Region Science and Technology Progress Award, in 2009, and the eighth China Inner Mongolia Autonomous Region Youth Science and Technology Award, in 2011.