CPAC: Energy-Efficient Algorithm for IoT Sensor Networks Based on Enhanced Hybrid Intelligent Swarm

Qi Wang[1, *], Wei Liu[1], Hualong Yu[1], Shang Zheng[1], Shang Gao[1] and Fabrizio Granelli[2]

Abstract: The wireless sensor network (WSN) is widely employed in the application scenarios of the Internet of Things (IoT) in recent years. Extending the lifetime of the entire system had become a significant challenge due to the energy-constrained fundamental limits of sensor nodes on the perceptual layer of IoT. The clustering routing structures are currently the most popular solution, which can effectively reduce the energy consumption of the entire network and improve its reliability. This paper introduces an enhanced hybrid intelligential algorithm based on particle swarm optimization (PSO) and ant colony optimization (ACO) method. The enhanced PSO is deployed to select the optimal cluster heads for establishing the clustering architecture. An improved ACO is introduced to realize the data transmission from terminal sensor nodes to the base station. Our proposed algorithm can effectively reduce the entire energy consumption and extend the lifetime of IoT sensor networks. Compared with the traditional algorithms, the simulation results show that the presented novel algorithm in this paper has obvious optimization and improvement in network lifetime and energy utilization efficiency.

Keywords: Internet of Things, wireless sensor network, particle swarm optimization, ant colony optimization, energy efficiency.

1 Introduction

The wireless sensor network (WSN) is widely utilized in various application scenarios of the Internet of Things (IoT) in the current society, such as modern intelligent agriculture [Wang, Wang and Wang (2006)], environmental monitoring [Oliveira and Rodrigues (2011)], intelligent medical treatment [Alameen, Liu and Andkwak (2012)], early warning of natural disasters [Chen, Liu, Wang et al. (2013)], etc. The WSN is an important implementation of IoT and the architecture of WSN usually comprises various dynamic nodes and base station (BS), which cooperates to perform data acquisition, processing, and transmission tasks. Each node is mainly composed of sensor unit, wireless transmitting module, power module, data processing, and storage unit. Usually, the nodes are embedded micro-devices with limited processing, storage, and communication capabilities, and also it has to face the most critical challenge-energy limitation. Simultaneously, it is difficult to

---

[1] Jiangsu University of Science and Technology, Zhenjiang, 212003, China.

[2] University of Trento, Via Sommarive 9, Povo TN 38123, Italy.

[*] Corresponding Author: Qi Wang. Email: wangqi@just.edu.cn.

replenish the extra energy demand of the deployed nodes. Hence, the exhaustion of energy means the "death" of those nodes.

Usually, the IoT sensor nodes in the perception layer should transmit data to the upper layer-the application layer as soon as possible after obtaining information. Most of the energy consumption of nodes mainly occurs during the data transmission between nodes. That is why energy-efficient algorithms can significantly reduce the energy consumption of nodes. At present, the most popular and feasible energy-saving schemes are always to devote the clustering routing algorithms [Abbasi and Younsi (2007)], that is, the network is logically divided into several clusters, and each cluster consists of one cluster head (CH) and several member nodes. The member nodes are responsible for collecting and transmitting data to the CH. The CH is responsible for intra-cluster communication scheduling, fusing, sending data to BS. This architecture has the advantages of appropriate scalability, high energy efficiency, low latency, etc. [Afsar and Tayarani-H (2014)].

In the clustering structure, the communication mode comprises intra-cluster and inter-cluster. According to the actual communication environment, the wireless link can be generally divided into single-hop and multi-hop. In an intra-cluster manner, the single-hop link requires all member nodes to communicate directly with the CH. The member node that is farther from the CH consumes much more energy. Via employing the multi-hop link, the member node does not need to communicate with the CH directly. In the inter-cluster mode, the multi-hop link also could assist the CH that is far from the BS, to avoid establishing direct communication with the BS. Compared with the single-hop link, the multi-hop link can reduce the network energy consumption [Mhatre and Rosenberg (2004)]. However, the multi-hop link also results in overload and excessive energy consumption for the nodes close to the BS or the CHs that frequently forward the data, thereby causing a "hot-spot" problem [Perillo, Cheng and Heinzelman (2005)].

In this paper, we discuss a novel energy efficient algorithm named as Combined PSO-ACO Clustering (CPAC) based on the enhanced hybrid PSO-ACO method and have adopted a centralized clustering approach for the CHs selection. The BS only needs to collect the information of nodes remaining energy and their location in the network, and via employing the computing capacity of BS, thereby selecting the optimal CHs, forming a coherent cluster structure and determining the optimal data transmission path. Compared with the most state of the art and classical algorithms, our innovations are listed as follows:

1) In the stage of CHs election, based on the enhanced PSO, the multi-attribute optimization problem is transformed into the single-objective optimization problem for selecting the optimal CHs.

2) In the stage of clustering structure formation, the member nodes can choose the optimal CH according to our proposed weight function that takes into account the energy remaining and location of the target CH.

3) In the stage of data transmission, based on the enhanced ACO, the optimal multi-hop transmission path from the member node to the CH, and the CH to the BS is explored and settled.

The rest of this paper is organized as follows: Section 2 lists related dynamic algorithms based on the probability theories and intelligent methods. The detailed prerequisite

knowledge is discussed in Section 3. In Section 4, the details of enhanced algorithms are described. Section 5 presents the experimental setup and simulation results. Finally, Section 6 includes concluding remarks.

2 Related work

The main challenges of network clustering include how to determine the number of CHs, optimize the election of CH, the cluster structure, and the intra-cluster and inter-cluster data transmission. Till present, researchers have proposed various clustering algorithms, which can be usually divided into two types, one is based on probability, and the various intelligent algorithms dominate the other one.

2.1 Probability based clustering methods

In the probability-based clustering algorithms represented by LEACH [Heinzelman, Chandrakasan and Balakrishnan (2000)], PEGASIS [Lindsey and Raghavendra (2002)], HEED [Younis and Fahmy (2004)], and TEEN [Manjeshwar and Agarwal (2001)], whether the node can be elected as the CH depends on the generated random number according to its attribute is more significant than a certain preset threshold. Even though the mechanism of these algorithms is uncomplicated and easy to implement, these algorithms may still cause uneven distribution of CHs and unbalanced load. Literature [Heinzelman, Chandrakasan and Balakrishnan (2000)] proposed the well-known LEACH algorithm that uses the distributed CH election strategy. In this protocol, each node obtains the same probability to become the CH. Hence, the network performance can be significantly reinforced. Since the CH is randomly selected, the node with less remaining energy may be elected as the CH. A series of problems may arise from the choice based on this strategy. For instance, the CH location is unevenly distributed in the geospatial space, the cluster size may be unequal, and the load balance of CH may not be able to achieve.

2.2 Intelligent algorithm based clustering methods

Indeed, how to dynamically and efficiently elect CHs from the terminal nodes with gradually decreasing energy, meet uniform CHs distribution and minimize network energy consumption, which is recognized as the NP-hard problem [Kumar, Aseri and Patel (2009)]. As the emerging intelligent algorithms are widely known and accepted, swarm intelligence algorithm and evolutionary algorithm have been arranged to solve the clustering problem of sensor networks, and achieved expected excellent results. Typical swarm intelligence algorithms include PSO algorithm [Abdul Latiff, Tsimenidis and Sharif (2007); Kulkarni and Venayagamoorthy (2010); Zhang, Wang and Ji (2015); Wang, Zhou and Xiang (2016); Rao, Jana and Banka (2017)], artificial bee colony (ABC) algorithm [Karaboga and Basturk (2007); Karaboga, Gorkemli, Ozturk et al. (2014); Karaboga, Okdem and Ozturk (2012); Mann and Singh (2017)], ant colony optimization (ACO) algorithm [Camilo, Carreto, Silva et al. (2006); Wang, Li, Xiong et al. (2008); Cheng, Xun, Zhou et al. (2011)] and Genetic algorithm (GA) [Jin, Zhou and Wu (2003); Hussain, Matin and Islam (2007); Elhoseny, Yuan, Yu et al. (2015); Yuan, Elhoseny, El-Minir et al. (2017)] are recognized as the conventional evolutionary algorithms. Compared with the probabilistic-based clustering algorithms, the intelligent methods based clustering algorithms serve dynamic

CHs election modes. The intelligent algorithm based schemes consider the remaining energy, location, connectivity, coverage, and other properties of the nodes. Thus they can select the most appropriate CHs.

2.2.1 PSO algorithm based clustering methods

The PSO algorithm is a random search algorithm based on swarm collaboration developed by simulating the foraging behavior of birds. In the PSO algorithm, each bird is inspected as a particle. The particle has two properties: velocity and position. Each particle knows its optimal position and the excellent position so far among the entire swarm of particles, then employs Eq. (1) and Eq. (2) to adjust the particle's speed, and uses Eq. (3) to improve its position. The position of each particle represents a possible solution to the target problem. After several iterations, the optimal value can be obtained.

$$v_i^{k+1} = \omega \cdot v_i^k + c_1 \cdot r_1 \cdot \left(pb_i - s_i^k\right) + c_2 \cdot r_2 \cdot \left(gb - s_i^k\right) \tag{1}$$

$$\omega = (W - 0.4) \cdot (M_a - iter)/M_a + 0.4 \tag{2}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{3}$$

where v_i^k, x_i^k, v_i^{k+1}, x_i^{k+1} represent the velocity and position of the ith particle before and after adjustment, pb_i represents the optimal position of the ith particle, gb represents the optimal position of the current particle swarm, c_1 and c_2 are learning factors, r_1 and r_2 are the random numbers in $(0,1)$, ω is the time-involving inertia weight, M_a represents the maximum number of iterations, $iter$ represents the current number of iterations, and W indicates a fixed constant and the value is taken as 0.9.

A clustering algorithm based on PSO is proposed in Abdul Latiff et al. [Abdul Latiff, Tsimenidis and Sharif (2007)] (PSO-C). Only the nodes with the value of remaining energy higher than the amount of average energy in the network are eligible to qualify for the candidate CHs. The fitness function proposed in the literature takes into account the remaining energy of the node and the intra-cluster communication distance. The smaller the value of the fitness function, the elected CHs are better.

Wang et al. [Wang, Zhou and Xiang (2016)] (PSO-1) exploits the PSO algorithm to elect CHs and select the optimal relay node for each CH. The CH is responsible for collecting, fusing, and forwarding the data to the relay node. The relay node is responsible for forwarding the data to the BS, which could reduce the energy consumption of the CH itself. Therefore, when designing the fitness function of the CH and the relay node, factors such as the remaining energy of the node and its position should be carefully considered.

2.2.2 ABC algorithm based clustering methods

The ABC algorithm is an optimization algorithm based on the intelligent foraging behavior of honey bee swarm. In the ABC algorithm, the bee colonies consist of three groups of bees: employed bees, onlooker bees, and scout bees. The goal of the entire bee colony is to find the honey source with the most considerable nectar amount. Each employed bee corresponds to a target honey source location, and the employed bee determines a new food source within the neighborhood of the target honey source according to Eq. (4) and

calculates its nectar amount. When the employed bee completes the explore process, the nectar information is shared with the onlookers. According to the "Roulette wheel selection" method, the onlooker bee tends to select good food source from those who have higher nectar amount. Hence the food source with a large amount of nectar will have a higher probability of being chosen as the target. This process is performed multiple times in the neighborhood of the target honey source. If the search process is updated numerous times, and there is still no improvement, then the honey source is abandoned, and scout bees are about to start their mission. The scout bees are translated from part of employed bees, which leave their honey sources and search new ones via utilizing the Eq. (4). The honey source is considered as the CH, and finding the high-quality honey sources is to be recognized as searching a set of optimal CHs.

$$x_i^{k+1} = x_i^k + r \cdot \left(x_i^k - x_j^k \right) \tag{4}$$

where r is a random number in (-1, 1), x_i^k and x_j^k are the positions before the ith and j^{th} employed bees are adjusted, and x_i^{k+1} is the fixed position of the ith employed bee.

Karaboga et al. [Karaboga, Okdem and Ozturk (2012)] (ICWAQ) proposes a clustering algorithm based on ABC. The fitness function introduced in the literature fully premeditated the transmission distance, the remaining energy of the CHs and the QoS of the entire network. The higher the value of fitness is, the elected CHs are more reasonable.

2.2.3 ACO algorithm based clustering methods

The ACO is a global optimization algorithm for simulating the foraging behavior of ant colonies. Ant colonies can quickly find food by exchanging the foraging information by secreting a substance called pheromone when searching for food. The ant usually selects the shortest path among massive nodes, the kth ant calculates the probability from source node i to all next hop target node j using Eq. (5), the "Roulette wheel selection" method is used to select the final next node, then the pheromone τ_{ij}^{t+1} will be updated at the end of each searching period using Eq. (6), repeating this process until it reaches the destination. When the ant traveled down the entire path, the pheromone on the elapsed path is updated. During the same duration, the shorter path would remain more pheromone released by the ants, and then more ants would choose this path as the target. Finally, the ants in the entire ant colony will be concentrated on the shortest route. The ACO algorithms usually can be employed for path selection to quarry the shortest multi-hop path.

$$P_{ij}^k = \left(\tau_{ij}{}^\alpha \cdot L_{ij}{}^{-\beta} \right) / \left(\sum_{k \in M_k} \tau_{ij}{}^\alpha \cdot L_{ij}{}^{-\beta} \right) \tag{5}$$

$$\tau_{ij}^{t+1} = (1 - \rho)\tau_{ij}^t + Q/D_k \tag{6}$$

where P_{ij}^k denotes the probability that k^{th} ant chooses node j as the next hop, L_{ij} represents the distance between i and j, τ_{ij} denotes the pheromone between i and j, M_k is the set of nodes that have not been visited yet, α and β represent weighted values of pheromone and visibility, D_k is the obtained path length after the kth ant completes the entire path, ρ is the pheromone volatilization factor, and Q is the pheromone constant. The authors of the article Cheng et al. [Cheng, Xun, Zhou et al. (2011)] (EAACA) also

present a clustering algorithm based on ACO. The ant selects the next hop target node according to the remaining energy of the neighbor nodes and the pheromone concentration on the link. When the pheromone is updated, the path with the shorter length and higher residual energy of the nodes would be reinforced more pheromone.

2.2.4 GA based clustering methods

The GA is a heuristic search algorithm proposed by reference to the biological evolution process. Drawing on the theory of natural evolution, the GA simulates the problem to be solved into a process of natural evolution. Individuals in the population are used to represent feasible solutions to the problem. Each represents a potential solution to the problem. Individuals typically assess their performance through fitness function. Via applying the methods of selection, intersection, and mutation, the new population continuously iterates, and the higher the fitness function value, the higher the probability that the individual retains and finally the community evolves toward the optimal result, thus achieves the optimal solution.

The authors of literature Jin et al. [Jin, Zhou and Wu (2003)] (GAO) believe that fewer CHs will have higher energy efficiency, due to the CH consumes more energy than the member nodes, and the number of CHs should be limited as few as possible. Since energy consumption is positively related to the transmission distance, the transmission distance also should be as short as possible. Therefore, the fitness function proposed in this literature is to minimize the transmission distance and minimize the number of CHs. The larger the fitness function value, the elected CHs are more reasonable.

The fitness function proposed in Hussain et al. [Hussain, Matin and Islam (2007)] (GAHN) takes into account the current remaining energy in the network, the expected energy consumption in the current round, the transmission distance between the CH and the BS, and the density factor of the nodes, to select the optimal node as the CH. The larger the fitness function value, the elected CHs are more legitimate.

3 Prerequisite knowledge

3.1 Network model

Based on a large number of works of literature, our proposed network model has the following properties:

1) Each node has the same initial energy and cannot be re-charged or acquire other energy.

2) Each node has a unique ID and can obtain the information of location and remaining energy.

3) Each node can perform data collection as the cluster member node, or as the CH to perform data fusion.

4) Each node can adjust the transmit power, and the transmitted radio signals have the same energy loss in any direction.

5) The locations of the nodes and BS are fixed; the BS has sufficient power supply and dominant computing power.

3.2 Energy consumption model

Using the "First Order Radio Model" as the energy consumption model [Heinzelman,

Chandrakasan and Balakrishnan (2000)], which mainly considers the node energy consumption when transmitting and receiving data. The transmission distance of the data is equal to the Euclidean distance from the source node to the destination node. When it is less than the distance threshold, the free space channel model is adopted, and the multi-path propagation model is used instead.

3.2.1 Energy consumption model for transmission

The energy consumed by the source node to transmit k bit data to the destination node.

$$E_{TX}(k,d) = \begin{cases} E_{elec} \cdot k + E_{fs} \cdot k \cdot d^2 & d \leq d_0 \\ E_{elec} \cdot k + E_{mp} \cdot k \cdot d^4 & d > d_0 \end{cases} \tag{7}$$

where k is the length of data, d is the distance of transmission, $d_0 = \sqrt{E_{fs}/E_{mp}}$ is the distance threshold, E_{elec} is the energy required to transmit or receive unit data in the transmitting or receiving circuit, E_{fs} and E_{mp} are the energy consumption required to transmit the unit distance of the unit data in the various models.

3.2.2 Energy consumption model for receiving

The energy consumed by the destination node to receive k bit data.

$$E_{RX}(k) = E_{elec} \cdot k \tag{8}$$

3.2.3 Energy consumption model for data fusion

The CH performs data fusion, which can reduce redundant data and reduce the energy consumption of communication. If the number of cluster member nodes is m, each node transmits K_M bit data to the CH, and the CH fuses the data into K_P bit, meanwhile, the energy consumed by the CH to combine each bit of data is E_{DA}.

$$E_{PR}(m) = E_{DA} \cdot K_M \cdot m \tag{9}$$

3.3 Experimental scenario

Seeing Fig. 1 below. In the $500 \times 500 \text{ m}^2$ field, 100 nodes are randomly deployed, and the BS is located in the center of the area (250, 250).
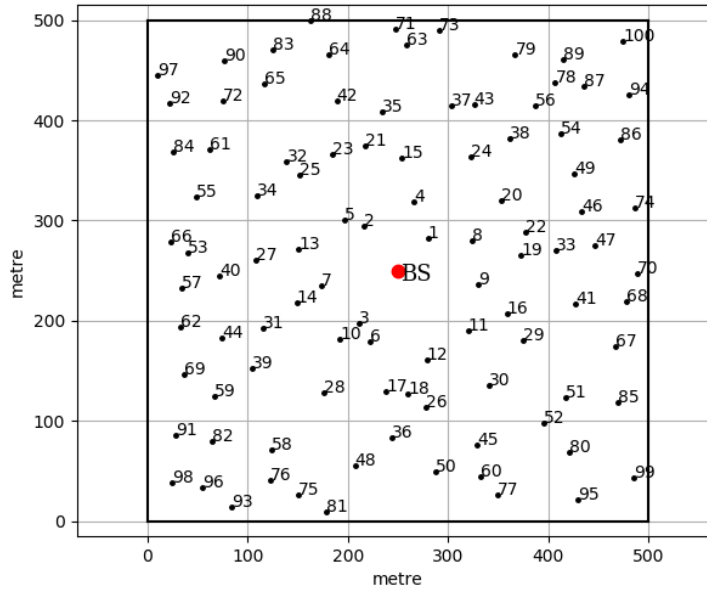
Figure 1: Node geolocation map

4 Enhanced clustering algorithm

The clustering scheme usually comprises the following three phases: the CHs election, the cluster formation, and the data transmission. In the CHs election phase, a PSO algorithm-based strategy is used to select the optimal CHs. In the cluster formation phase, the common node should choose the optimal CH according to the remaining energy of the CH and its distance to the CH. In the data transmission phase, an ACO algorithm-based mechanism is proposed. When the current node selects the next hop target node, the node's remaining energy and communication distance will be considered to determine the best multi-hop transmission path. The energy-saving strategies are utilized in each phase to effectively reduce node energy consumption and extend the lifetime of the entire network.

In our proposed network, the network operating cycle is measured by "Round", the number of nodes is N and clusters are K, the number of surviving nodes in the current round is M, the number of nodes in the jth cluster is n_j , the node i in the jth cluster is represented as S_{ij}, the distance from S_{ij} to C_j is D_j, the distance from C_j to BS is D_j, and d_{jk} represents the distance between two different CHs j and k.

4.1 The CHs election

The PSO is a popular multi-dimensional optimization algorithm with the following features, such as easy implementation, high computational efficiency, fast convergence, and excellent solution accuracy [Abdul Latiff, Tsimenidis and Sharif (2007)]. So, this paper designs a PSO-based CHs election algorithm in our proposal. An optimal set of CHs should have the higher remaining energy, the well-knit cluster structure, and evenly distributed in the area.

Since the CHs need to collect, fuse, and transmit data, which require that the CHs should

have higher remaining energy; hence, not all nodes can be selected as CHs. In the current round, if the node's remaining energy is below a certain threshold, it only could be recognized as an ordinary node and cannot be selected as a CH. According to Eq. (10), the remaining energy of all currently surviving nodes is normalized and sets the threshold to 0.9 based on previous experience. After calculated by employing Eq. (11), only if FE_i is greater than 0.9, the node can be accepted as the candidate node of CH.

$$NE_i = \frac{RE_i - min_{l=1}^{M}(RE_l)}{max_{l=1}^{M}(RE_l) - min_{l=1}^{M}(RE_l)} \tag{10}$$

$$FE_i = 1 - e^{\frac{\mu \cdot NE_i}{1 - NE_i}} \tag{11}$$

where RE_i, NE_i, FE_i respectively represent the remaining energy of the node, the normalized value, and the final judgment value. $min_{l=1}^{M}(RE_l)$ and $max_{l=1}^{M}(RE_l)$ respectively represent the minimum and maximum values of the remaining energy of all surviving nodes in the current round. As shown in Fig. 2, when $NE_i = 0.1, FE_i = 0.9$, the μ is -20.72.

There are several principles could help us to verify whether the optimal CHs have been selected or not.
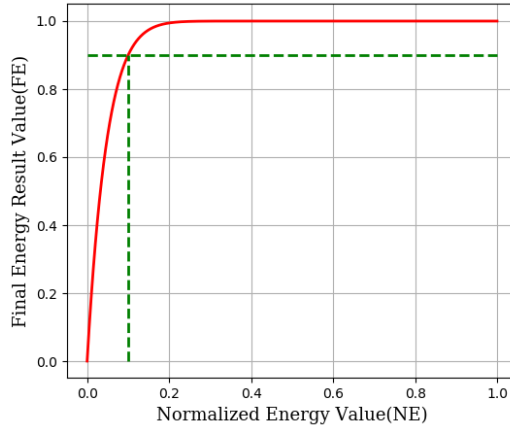
Figure 2: Node energy decision function

4.1.1 Higher remaining energy

As shown in Eq. (12), the node selected as a CH should retain higher remaining energy. C_1 represents the ratio of the remaining energy of all nodes in the current round to the remaining energy of the CHs set. The smaller the C_1, indicates that the remaining energy of selected CHs is higher.

$$C_1 = \sum_{i=1}^{M} E_i^{current} / \sum_{j=1}^{K} E_j^{current} \tag{12}$$

4.1.2 Compact cluster structure

The energy consumption of a cluster $E_{Cluster}$ is the sum of the energy consumed by the

cluster member nodes to transmit data E_{cmT} and the energy consumed by the CH to receive the data E_{chR}. Eq. (13) represents the energy consumed by a cluster, Derivation of Eq. (14) and Eq. (15), Eq. (16) shows that the entire energy consumption of a cluster is positively correlated with the communication distance, that is, the further the range, the more energy is consumed during the communication. Hence, an optimal CH should minimize the intra-cluster communication distance.

$$E_{Cluster} = E_{cmT} + E_{chR} \tag{13}$$

$$E_{Cluster} = \sum_{i=1}^{n_j}(E_{elec} \cdot k + E' \cdot k \cdot D) + k \cdot n_j \cdot E_{elec} \tag{14}$$

$$E_{Cluster} = 2 \cdot n_j \cdot E_{elec} \cdot k + E' \cdot k \cdot \sum_{i=1}^{n_j}(D) \tag{15}$$

$$E_{Cluster} \propto \sum_{i=1}^{n_j}(D) \tag{16}$$

where E' is E_{fs} or E_{mp}, and D is $(D_{ij})^2$ or $(D_{ij})^4$. Thus, introducing a new indicator called "compactness" presented by C_2. The more compact the geometry of the cluster, the smaller the distance from the cluster member node to the CH, the less energy is required for data transmission. The lesser the value of the C_2, the lower the energy consumption of the intra-cluster communication.

$$C_2 = max_{j=1}^K \sum_{i=1}^{n_j} D_{ij}/|n_j| \tag{17}$$

4.1.3 Evenly distributed

Ideally, an optimal set of CHs should be evenly distributed in the target surveillance area. So, using the following two methods to determine whether the distribution position of the selected CHs is infinitely close to the ideal CH position. Assuming the number of CHs is i. In the ideal state, the distance from each CH to the BS should be L. Thus, the distance sum of all CHs to the BS is $i * L$. In the non-ideal state, the distance from the j to the BS is l_j, and the distance between all CHs and the BS is $\sum_{j=1}^i l_j$. Introducing the Eq. (18) to measure the error values of these two states.

$$\varepsilon = lim|i * L - \sum_{j=1}^i l_j| \tag{18}$$
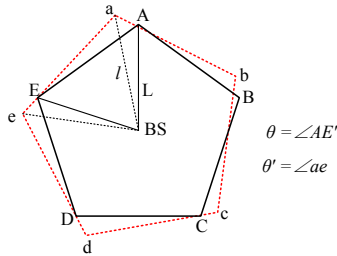
Figure 3: When i = 5, contrast map of CH distribution between ideal state (black) and non-ideal state (red)

when the ε approaches infinity to 0, the actual selected CHs distribution is closest to the CHs distribution in the ideal state.

Determining from the angle θ formed by the adjacent CH and the BS whether the selected CH has reached the even distribution state. In the ideal state, the angle formed by the jth, k^{th} adjacent CHs and the BS is θ. In the non-ideal state, the angle formed by the jth, k^{th} adjacent CHs and the BS is θ'_{jk}. Introducing the Eq. (19) and Eq. (20) to measure the error values of these two states.

$$k = \begin{cases} j+1 & j < i \\ 1 & j = i \end{cases} \tag{19}$$

$$\delta = lim \left| i \cdot tan\,\theta - \sum_{j=1}^{i} tan\,\theta'_{jk} \right|, \ \theta = 2\pi/i \tag{20}$$

when the δ approaches infinity to 0, the selected CHs distribution is closest to the CHs distribution in the ideal state. It can be known from the Eq. (21). whether the result of the CHs election is the optimal solution closest to the ideal state or not. At the same time, the smaller the C_3, the more balanced the CHs distribution.

$$C_3 = min\,(\varepsilon + \delta) \tag{21}$$

The performance of the fitness function determines the result of the CHs selection. Eq. (22) represents the proposed fitness function in our strategies. The fitness function considers the remaining energy and position of the node, and converts the multi-objective optimization problem into a single-objective optimization problem by weighted sum, where φ_1, φ_2 and φ_3 are values between (0, 1).

$$C = \varphi_1 \cdot C_1 + \varphi_2 \cdot C_2 + \varphi_3 \cdot C_3 \tag{22}$$

4.2 The cluster formation

When the CHs election is completed, if the non-CH nodes select the corresponding CH and join that cluster structure simply based on the "nearest distance" principle, it may result in uneven cluster size and unbalanced energy consumption. So, this paper provides the optimal solution for each non-CH node based on the remaining energy and location of the CHs.

The node preferentially selects the CH with higher remaining energy.

$$S_i^{CH_j} \propto RE_{CH_j} \tag{23}$$

The node preferentially selects the CH that is closer to the node itself.

$$S_i^{CH_j} \propto 1/D_{ij} \tag{24}$$

The node i selects the corresponding CH based on the weight function of the Eq. (25), hence the larger the S_i^{CH} value, the result is better.

$$S_i^{CH} = max_{j=1}^{K} \left(RE_{CH_j}/D_{ij} \right) \tag{25}$$

where RE_{CH_j} is the remaining energy of the jth CH.

4.3 The data transmission

In the small-scale network, the single-hop mode is usually employed to reduce transmission delay for the intra-cluster and inter-cluster communication. However, in a large-scale network, owing to the energy consumption of transmitting data by the node far away from the CH or the CH far away from the BS, is following the 2nd or 4th order relationship with the distance, multi-hop routing should be adopted for the intra-cluster or inter-cluster communication. To meet the above challenges, the ACO algorithm has been widely used in WSN routing protocols as the optimal path selection [Mohajerani and Gharavian (2015)].

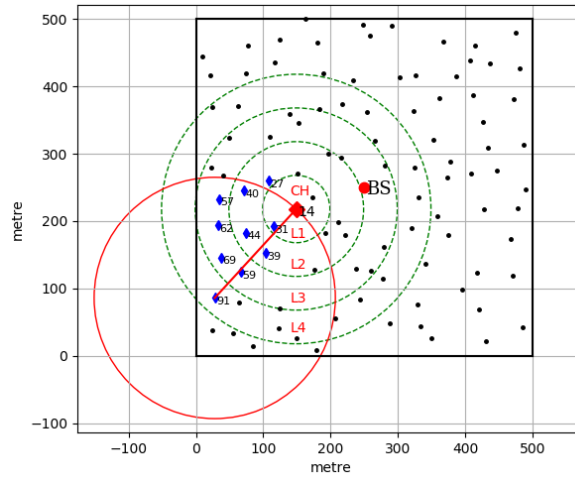### 4.3.1 Intra-cluster nodes hierarchy

Assuming node numbered 14 is a CH, node numbered 91 is a source node, and all of the blue square nodes are the intra-cluster nodes. Settling the CH as the center of the circle and draw a set of concentric circles $O_i (i = 1,2, ...)$ by the radius R_1 multiplication until all the nodes in the cluster are included in this set of concentric circles. If the source node supposes to communicate with the CH, drawing another circle O' with the source node as the center, and settle the radius equals the distance from the source node to the CH. As shown in Fig. 4, the intersection of O_i and O' divides the nodes in the cluster into four layers. The L1 layer includes node 31, the L2 layer comprises nodes 39, 40, 44, the L3 layer involves nodes 57, 59, 62, 69, and the L4 layer contains node 91. The source node transmits data to the CH by following this order: L4→ L3→ L2 →L1→CH.

4.3.2 Inter-cluster nodes hierarchy

Assuming node numbered 14 is a CH, and blue square node is an intra-cluster node, and the BS is the center of the circle. Drawing a set of concentric circles $O_i (i = 1,2, ...)$ by the radius R_1 multiplication until the node CH in the cluster is included in this set of concentric circles. If CH wants to communicate with the BS, then we draw another circle O' with the CH as the center and settle the radius equals to the distance from the CH to the BS. As shown in Fig. 5, the intersection of O_i and O' divides the nodes in the cluster into 3 layers. There is no node located in the L1 layer. The L2 layer has nodes 2, 3, 5, 6, 7, 10, and the L3 layer has nodes 13, 14. The CH sends data to the BS in the following order: L3 →L2→BS.
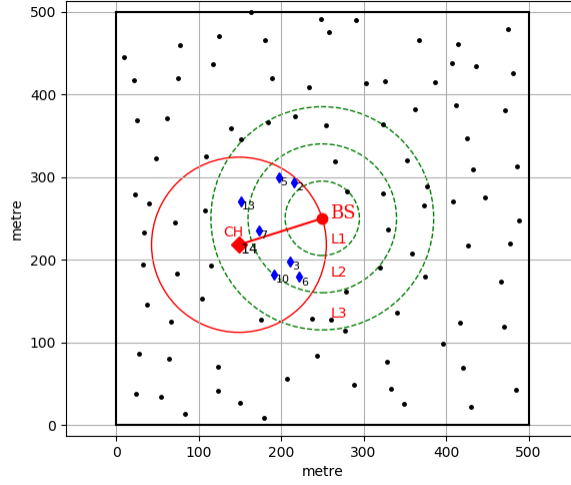
Figure 4: Intra-cluster nodes hierarchy

Figure 5: Inter-cluster nodes hierarchy

4.3.3 Enhanced ACO algorithm

How to select the most suitable forwarding node from the various nodes located in each layer is another challenge. The ACO algorithm is utilized to plan the shortest path from the cluster member node to the CH and the CH to the BS. To avoid the "hot spot" problem, each ant calculates the probability of selecting all next hop paths according to Eq. (26), and finally determines the target next hop via exploiting the "Roulette Wheel Selection" algorithm. This selection process fully considers the remaining energy of the neighbor nodes, distance, and pheromone concentration on the associated link. The purpose is to select the node with the closer range and higher remaining energy as the target next hop node.

$$P_k(r,s) = \frac{[\tau(r,s)]^\alpha \cdot (E_s/d_{rs})^\beta}{\sum_{u \notin M_k}[\tau(r,u)]^\alpha \cdot (E_s/d_{rs})^\beta} \tag{26}$$

where M_k is the set of the nodes that ant k has visited, $\tau(r, s)$ is the concentration of pheromone between the two nodes, d_{rs} represents the distance between nodes, E_s denotes the remaining energy of the node s.

The pheromone is updated by following Eq. (27), to obtain more pheromones for the paths with the shorter length and higher remaining energy nodes.

$$T_k(r, s) = (1 - \rho)T_k(r, s) + EAvg_k/L_k \qquad (27)$$

where L_k denotes the total length of the path that the ant k has visited, $EAvg_k$ represents the average energy of all nodes located on the path.

4.4 CPAC algorithm

We firstly introduce the procedure of CPAC algorithm in Tab. 1, then introduce modified and enhanced PSO and ACO algorithm in Tab. 2 and Tab. 3 respectively.

Table 1: The flowchart of CPAC Algorithm

Procedure CPAC routing protocol

The BS acquires the residual energy and location of each SN.

for round=1 to MaxRound **do**

1. Checking whether the SN is dead.
2. Executing PSO algorithm to select CHs (Shown in Tab. 2).
3. The common node selects CH to form cluster.
4. Executing ACO algorithm to determine path (Shown in Tab. 3).
5. Executing data transmission.

end for

Table 2: The flowchart of enhanced PSO algorithm

Procedure enhanced PSO algorithm

1. **Initialization**
    1.1 Find candidate CHs using equation Eqs. (10)-(11).
    1.2 Initialize position and velocity of each particle.
2. **for** iter=1 to M_a **do**
    2.1 Evaluate fitness of each particle using Eq. (22).
    2.2 Find the personal and global best for each particle.
    2.3 Update velocity and position using Eqs. (1)-(3).
    2.4 Limit the change in the particle's position value.
    2.5 Map particle position with the closest (x, y) coordinates.
    end for
3. Output global best of particle swarm.

Table 3: The flowchart of enhanced ACO algorithm

Procedure enhanced ACO algorithm

---

1. **Initialization**

   Set initial pheromone value for each edge.

2. **For CMs**

   for i=1 to K **do** // K is the number of CHs.

     for j=1 to N$_j$ **do** // the number of CMs per cluster are N$_j$ perform layering between CM and CM.

       for iter=1 to MAXITER **do**

         for ant=1 to M **do** // M is the number of ants, each ant use Eq. (26) to calculate probabilities. Using roulette to select the next node.

           While (arriving CH)

         end for

         Store the shortest path of this CM.

         Update pheromone value using Eq. (27).

       end for

     end for

   end for

3. **For CHs**

   for i=1 to K **do**

     Perform layering between CH and BS.

     for iter=1 to MAXITER **do**

       for ant=1 to M do // M is the number of ants, each ant use Eq. (26) to calculate probabilities.

       Using roulette to select the next node.

         while (arriving BS)

       end for

       Store the shortest path of this CH.

       Update pheromone value using Eq. (27).

     end for

   end for

---

5 Simulation result

In this paper, we set the number of CHs as 5 [Abdul, Tsimenidis and Sharif (2007); [Karaboga and Okdem (2012)]. The network model and energy consumption model are described in Section 3. The energy consumption parameters of the "First Order Radio Model" are listed as follow. $K_P = 4000bit$，$K_M = 200bit$，$E_{elec} = 50nJ/bit$，$E_{fs} = 10pJ/bit/m^2$，$E_{mp} = 0.0013pJ/bit/m^4$，$E_{DA} = 5nJ/bit$. In the PSO algorithm, the number of particles is set to 30, $c_1 = c_2 = 1.5$, $V_{max} = 200$, $V_{min} = -200$, $X_{max} = 500$，$M_a = 100$，Based on various trials and verifications, $\varphi_1 = 0.3$，$\varphi_2 = 0.6$，$\varphi_3 = 0.4$. In the ACO algorithm, the number of ant colonies is set to 20，$\alpha = 1.5$, $\beta = 2$，$\rho = 0.8$，$MAXITER = 20$.

Two scenarios are assumed as follows:

Scenario 1: There are 20% heterogeneous nodes are located near the BS, the initial energy of the heterogeneous nodes is 1J, and the initial energy of rest nodes is 0.5 J.

Scenario 2: The initial energy of each node in the network is 0.5 J.

In each round, the topology is firstly generated by the CPAC algorithm, then the "First Order Radio Model" is used to calculate the residual energy of each node. Finally, the process is iterated continuously. To evaluate the effectiveness and usability of different clustering algorithms, the network lifetime and energy consumption are used as evaluation criteria for network performance.

5.1 Lifetime

Using the number of rounds corresponding to the first node dead (FND) as the lifetime of the network [Mohajerani and Gharavian (2015); Handy, Haase and Timmermann (2002); Chen and Zhao (2005)]. As shown in Fig. 6, the lifetime of the CPAC algorithm is 3334 in scenario 1, and 1919 in Scenario 2. Compared with the traditional algorithms, the lifetime of CPAC is much longer and showing both better performance in Scenarios 1 and 2. Meanwhile, we can get a conclusion that adding a certain number of heterogeneous nodes to the network can achieve a longer network lifetime.
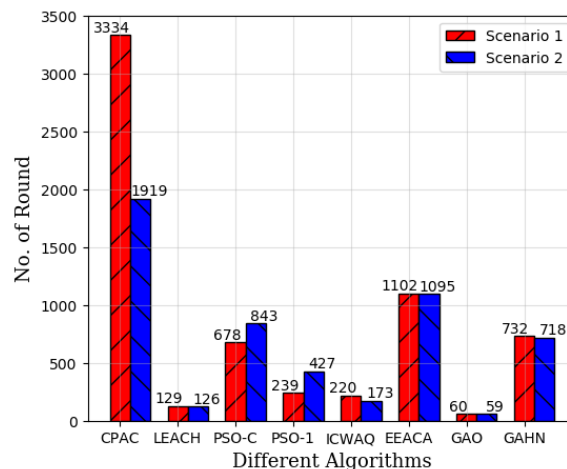
**Figure 6:** The network lifetime of different algorithms

5.2 Energy consumption

Energy consumption is another important indicator reflecting the network performance and is also an essential basis for evaluating the superiority of clustering algorithms. An optimal clustering algorithm should adjust all nodes load balance in each round. Hence, exploring the performance of network energy consumption of different routing algorithms, when the number of the round is progressively increased.

In ACO, the source node has the characteristics of short data transmission distance and a small amount of data. The relay node only plays the role of data forwarding, so the node load is quite less in each round, and the network energy consumption is also incredibly

balanced. Fig. 7 shows the difference in energy consumption of the CPAC and others in different scenarios. By observing the energy consumption curve, the curve of the CPAC algorithm has been kept at a constant slow linear drop. In Scenario 1, the energy consumption of CPAC is approximately equal to EEACA before 1500 rounds. In Scenario 2, before 500 rounds, the energy consumption of CPAC is similar to EEACA, indicating that the energy of the nodes in the network is evenly consumed. The energy curves of other algorithms are steep firstly, indicating the energy attenuation is too fast, and the number of the dead nodes is more. In the later stage, the energy curves are more gentle, showing that the individual nodes still have higher remaining energy.
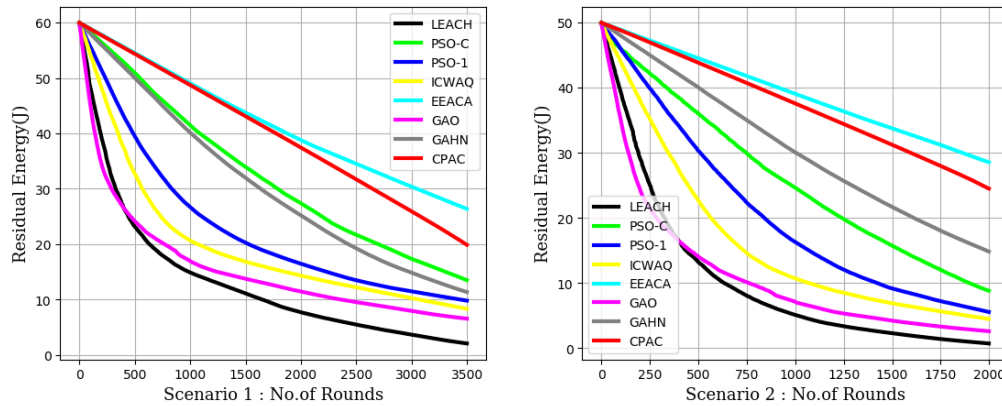
**Figure 7:** The energy consumption of different protocols in Scenario 1 and Scenario 2

5.3 Compared with the PSO and ACO

The CPAC algorithm employs dynamic clustering methods at various stages to improve network performance. Evaluating the difference in network lifetime and energy consumption metrics when the PSO, ACO and CPAC algorithm is separately adopted. As shown in Fig. 8, analyzing the results of the comparison of ACO, PSO, and CPAC algorithm, the lifetime is 3334, 1095, 573 in Scenario 1, and 1919, 1095, 542 in Scenario 2 in turn. The network lifetime of CAPC is three times more than ACO in Scenario 1, and nearly two times more in Scenario 2. Also, the lifetime of CPAC is almost six times more than PSO in Scenario 1, and nearly four times more in Scenario 2.
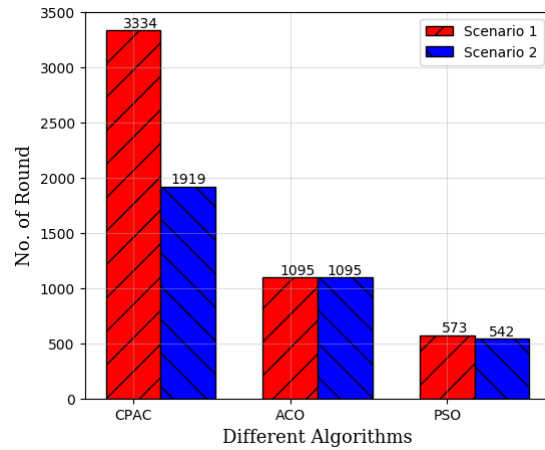
Figure 8: The lifetime of CPAC, PSO, and ACO

As shown in Fig. 9, the CPAC algorithm exhibits similar performance to the ACO algorithm in energy consumption, and its performance is superior to the PSO in different scenarios.
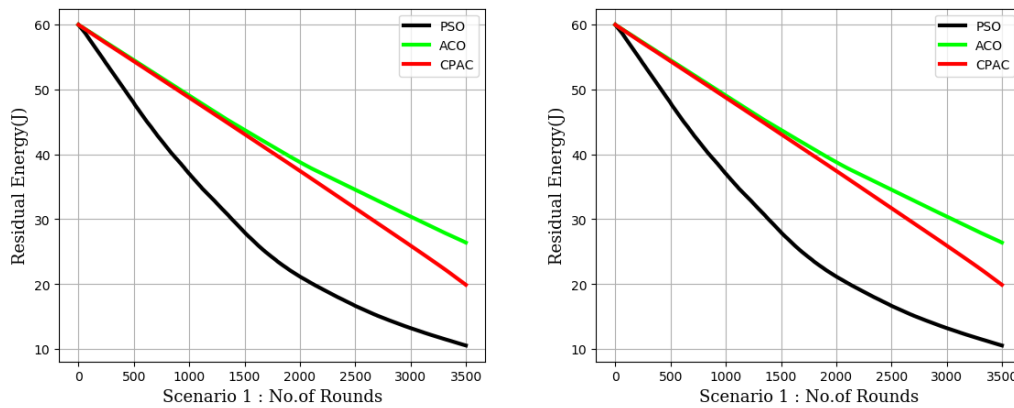
Figure 9: The energy consumption of CPAC, PSO, and ACO in Scenario 1 and Scenario 2

6 Conclusion

In this paper, we propose an energy-efficient IoT sensor network clustering algorithm named as CPAC based on the enhanced joint PSO-ACO algorithm. The PSO algorithm is adopted in the CHs election stage. The fitness function is constructed based on the remaining energy and location of the nodes. The ACO algorithm is employed during the transmission phase, based on the communication distance and remaining energy of the nodes. Besides, we choose the optimal relay node for each selected CH to achieve more efficient energy consumption in communication between the selected CH and BS. Simulation results show that our proposed CPAC protocol optimizes CHs distribution, cluster formation, the energy consumed by intra-cluster and inter-cluster data transmission, can effectively reduce network energy consumption, extend network lifetime, and improve

network reliability.

References

Abbasi, A. A.; Younsi, M. (2007): A survey on clustering algorithms for wireless sensor networks. *Computer Communications*, vol. 30, no. 14-15, pp. 2826-2841.

Abdul Latiff, N. M.; Tsimenidis, C. C.; Sharif, B. S. (2007): Energy aware clustering for wireless sensor networks using particle swarm optimization. *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 1-5.

Afsar, M. M.; Tayarani-H, M. H. (2014): Clustering in sensor networks: a literature survey. *Journal of Network and Computer Applications*, vol. 46, pp. 198-226.

Alameen, M.; Liu, J.; Andkwak, K. (2012): Security and privacy issues in wireless sensor networks for health care applications. *Journal of Medical Systems*, vol. 36, no. 1, pp. 93-101.

Camilo, T.; Carreto, C.; Silva, J. S.; Boavida, F. (2006): An energy-efficient ant-based routing algorithm for wireless sensor networks. *5th International Workshop*, pp. 49-59.

Chen, D.; Liu, Z. X.; Wang, L. Z.; Dou, M. G.; Chen, J. Y. et al. (2013): Natural disaster monitoring with wireless sensor networks: a case study of data-intensive applications upon low-cost scalable systems. *Mobile Networks and Applications*, vol. 18, no. 5, pp. 651-663.

Chen, Y. X.; Zhao, Q. (2005): On the lifetime of wireless sensor networks. *IEEE Communications Letters*, vol. 9, no. 11, pp. 976-978.

Cheng, D. Q.; Xun, Y. Y.; Zhou, T.; Li, W. J. (2011): An energy aware ant colony routing algorithms for the routing of wireless sensor networks. *International Conference on Intelligent Computing and Information Science*, pp. 395-401.

Dietrich, I.; Dressler, F. (2009): On the lifetime of wireless sensor networks. *ACM Transactions on Sensor Networks*, vol. 5, no. 1, pp. 1-39.

Elhoseny, M. X.; Yuan, H.; Yu, Z. T.; Mao, C. L.; El-Minir, H. K. et al. (2015): Balancing energy consumption in heterogeneous wireless sensor networks using genetic algorithm. *IEEE Communications Letters*, vol. 19, no. 12, pp. 2194-2197.

Handy, M. J.; Haase, M.; Timmermann, D. (2002): Low energy adaptive clustering hierarchy with deterministic cluster-head selection. *4th International Workshop on Mobile and Wireless Communication Network, Stockholm, Sweden*, pp. 368-372.

Heinzelman, W. R.; Chandrakasan, A.; Balakrishnan, H. (2000): Energy-efficient communication protocol for wireless microsensor networks. *33rd Annual Hawaii International Conference on System Sciences*, pp. 1-10.

Hussain, S.; Matin, A. W.; Islam, O. (2007): Genetic algorithm for energy efficient clusters in wireless sensor networks. *4th International Conference on Information Technology*, pp. 147-154.

Jin, S. Y.; Zhou, M.; Wu, A. S. (2003): Sensor network optimization using a genetic algorithm. *7th World Multiconference on Systemics, Cybernetics and Informatics*, pp. 109-116.

Karaboga, D.; Basturk, B. (2007): A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, vol. 39, no. 3, pp. 459-471.

Karaboga, D.; Gorkemli, B.; Ozturk, C.; Karaboga, N. (2014): A comprehensive survey: artificial bee colony (ABC) algorithm and applications. *Artificial Intelligence Review*, vol. 42, no. 1, pp. 21-57.

Karaboga, D.; Okdem, S.; Ozturk, C. (2012): Cluster based wireless sensor network routing using artificial bee colony algorithm. *Wireless Networks*, vol. 18, no. 7, pp. 847-860.

Kumar, D. T.; Aseri, C.; Patel, R. (2009): EEHC: energy efficient heterogeneous clustered scheme for wireless sensor networks. *Computer Communications*, vol. 32, no. 4, pp. 662-667.

Kulkarni, R. V.; Venayagamoorthy, G. K. (2010): Particle swarm optimization in wireless-sensor networks: a brief survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 2, pp. 262-267.

Lindsey, S.; Raghavendra, C. S. (2002): PEGASIS: Power efficient gathering in sensor information systems. *IEEE Aerospace Conference on Big Sky*, pp. 1125-1130.

Mann, P. S.; Singh, S. (2017): Artificial bee colony metaheuristic for energy-efficient clustering and routing in wireless sensor networks. *Soft Computing*, vol. 21, no. 22, pp. 6699-6712.

Manjeshwar, A.; Agarwal, D. P. (2001): TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. http://www.nhu.edu.tw/~cmwu/Lab/TEEN.pdf.

Mhatre, V.; Rosenberg, C. (2004): Design guidelines for wireless sensor networks: communication, clustering and aggregation. *Ad Hoc Networks*, vol. 2, no. 1, pp. 45-63.

Mohajerani, A.; Gharavian, D. (2015): An ant colony optimization based routing algorithm for extending network lifetime in wireless sensor networks. *Wireless Networks*, vol. 22, no. 8, pp. 2637-2647.

Oliveira, L. M.; Rodrigues, J. J. (2011): Wireless sensor networks: a survey on environmental monitoring. *Journal of Communications*, vol. 6, no. 2, pp. 143-151.

Perillo, M.; Cheng, Z.; Heinzelman, W. (2005): An analysis of strategies for mitigating the sensor network hot spot problem. *Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*, pp. 474-478.

Rao, P. C. S.; Jana, P. K.; Banka, H. (2017): A particle swarm optimization based energy efficient cluster head selection algorithm for wireless sensor networks. *Wireless Networks*, vol. 23, no. 7, pp. 2005-2020.

Wang, N.; Zhou, Y.; Xiang, W. (2016): An energy efficient clustering protocol for lifetime maximization in wireless sensor networks. *IEEE Global Communications Conference*, pp. 1-6.

Wang, X. M.; Li, L. Q.; Xiong, N. X.; Pan, Y. (2008): Ant colony optimization-based location-aware routing for wireless sensor networks. *International Conference on Wireless Algorithms, Systems, and Applications*, pp. 109-120.

Wang, N.; Wang, N. Q.; Wang, M. H. (2006): Wireless sensors in agriculture and food industry: recent development and future perspective. *Computers and Electronics in Agriculture*, vol. 50, no. 1, pp. 1-14.

Yuan, X. H.; Elhoseny, M.; El-Minir H. K.; Riad, A. M. (2017): A genetic algorithm-based, dynamic clustering method towards improved WSN longevity. *Journal of Network and Systems Management*, vol. 25, no. 1, pp. 21-46.

Younis, O.; Fahmy, S. (2004): HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. *IEEE Transactions on Mobile Computing*, vol. 3, no. 4, pp. 366-379.

Zhang, Y. D.; Wang, S. H.; Ji, G. L. (2015): A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, pp. 1-38.