

Interference-free Clustering Protocol for Large-Scale and Dense Wireless Sensor Networks

Zhihong Chen¹, Hai Lin^{1*}, Lusheng Wang² and Bo Zhao¹

¹ Key Laboratory of Aerospace Information Security and Trusted Computing, Ministry of Education

¹ School of Cyber Science and Engineering, WuHan University
Wu Han, HuBei 430074 -China

[e-mail : chenzhihong, lin.hai, zhaobo@whu.edu.cn]

² School of Computer and Information, HeFei University of Technology
HeFei, AnHui 230009-China

[e-mail: wanglusheng@hfut.edu.cn]

* Corresponding author: Hai Lin

*Received June 23, 2018; revised September 2, 2018; accepted October 13, 2018;
published March 31 2019*

Abstract

Saving energy is a big challenge for Wireless Sensor Networks (WSNs), which becomes even more critical in large-scale WSNs. Most energy waste is communication related, such as collision, overhearing and idle listening, so the schedule-based access which can avoid these wastes is preferred for WSNs. On the other hand, clustering technique is considered as the most promising solution for topology management in WSNs. Hence, providing interference-free clustering is vital for WSNs, especially for large-scale WSNs. However, schedule management in cluster-based networks is never a trivial work, since it requires inter-cluster cooperation. In this paper, we propose a clustering method, called Interference-Free Clustering Protocol (IFCP), to partition a WSN into interference-free clusters, making timeslot management much easier to achieve. Moreover, we model the clustering problem as a multi-objective optimization issue and use non-dominated sorting genetic algorithm II to solve it. Our proposal is finally compared with two adaptive clustering methods, HEED-CSMA and HEED-BMA, demonstrating that it achieves the good performance in terms of delay, packet delivery ratio, and energy consumption.

Keywords: Wireless sensor network, clustering, energy saving, interference-free

The research presented in this paper is supported by Natural Science Foundation of HuBei Province, China under grant No. 2016CFB453 and Applied Basic Research Program of WuHan City, China under grand No. 2017010201010117. Also partially supported by the National Natural Science Foundation of China under Grant No. 61501160.

1. Introduction

Energy saving is big challenge for WSNs, and it becomes more critical for large-scale WSNs that carry more load and deplete energy faster. It is well-known that topology management is an efficient method for energy-saving. The clustering technique has been widely accepted as topology management for WSNs [1]. On the other hand, schedule-based medium access is considered as the best access control protocol for WSNs requiring reliability and timeliness, e.g. industrial WSN (IWSN) [2]. Time division multiple access (TDMA) is a typical schedule-based access method for radio networks to avoid interference by scheduling nodes into different timeslots. Each node wakes up and transmits data only in its allocated timeslots and keeps in sleep state in the remaining timeslots [3]. TDMA has proved to achieve a high degree of reliability and energy efficiency for cluster-based networks. However, timeslot management for cluster-based networks is not a trivial task. Each cluster should cooperate with others for timeslot allocation. Otherwise, the allocated timeslot in a cluster is not exclusive and could be disturbed by the communication from other clusters. This motivates us to design a new clustering method which is inherently interference-free to facilitate timeslot management in TDMA.

In cluster-based networks, sensor nodes are grouped into clusters. Each cluster usually consists of a cluster head (CH) and other nodes called members. Each member senses information and sends it to its CH. Then the CH aggregates all the received information into one packet and then forwards it to the sink. Besides energy saving, another benefit of the clustering is the scalability [4]. These make this technique well matched to the requirement of large-scale sensor networks.

Various clustering techniques have been proposed previously [5]-[6], which try to save energy via CH selection, routing optimization, data aggregation, etc. Even though only a few works are dedicated to interference-free, people do realize the importance of providing interference-free communication for cluster-based networks. Most of previous works introduce interference-free based on multiple access technologies or complex timeslot allocation methods for clustered WSNs [16]-[21], but the clusters themselves are not interference-free. This makes them inefficient, especially for large-scale WSNs. On the contrary, our proposition partitions the network into clusters that are inherently interference-free, i.e. the communication within a cluster does not interfere with the communications of other clusters. In the large-scale WSNs, a sensor node only covers a small fraction of the whole sensor field. Hence, if the CHs are carefully selected, it is possible that interference between clusters can be avoided. However, for high-density networks, it is impossible to distribute all nodes to clusters while keeping interference-free. Those nodes failing to join any cluster should be scheduled to inactive, which are called isolated nodes in the paper. Fortunately, in the dense network, scheduling some nodes to sleep itself is a density control method whose objective is to extend the network lifetime, at the same time keeping the connectivity and coverage constraints satisfied [8][9]. Since some nodes are scheduled into sleeping state in our proposition, it is critical to select these nodes to keep the monitor area still covered. This inspires us to model both energy saving and coverage as the objective functions.

Our clustering is designed for energy-saving, so it can be applied to any battery-powered WSNs, such as environment monitoring, smart agriculture. Nevertheless, with its interference-free feature, it is more suitable for the WSNs requiring reliability and real-time communication, where TDMA is adopted for deterministic communication. IWSN is a such

network, in which most standards, such as WirelessHART, ISA 100.11a, use the TDMA medium access control protocol [34].

The main contributions of this paper are as follows:

- A new clustering method called Interference-Free Clustering Protocol (IFCP) is proposed, with which the generated clusters are inherently interference-free.
- Model the interference-free clustering into a multi-objective optimization problem. By solving this problem, an optimal clustering is obtained, which can minimize energy consumption and coverage cost while holding interference-free.

The remainder of this paper is organized as follows. Section 2 describes the clustering technique and its related works. Section 3 describes in detail our clustering protocol IFCP. Section 4 analyzes IFCP's performance by comparing with HEED protocol. Section 5 summarizes the work and gives the prospectives.

2. Clustering Technology and Related Works

2.1 Clustering Method

Besides energy saving, clustering is an effective means of scheduling and density control [22]. It includes some key issues [11]: clustering strategy, CH selection, relay selection, re-clustering, routing and hotspot issues.

The early work on clustering is LEACH [5], in which CHs are selected based on a pre-determined probability. Other nodes choose a CH to join by estimating which of the selected CHs is the closest. LEACH addresses only one-hop transmission between CHs and the sink, which is not suitable for the large-scale WSNs. HEED [12] is designed for multi-hop WSNs, which also focuses on CH selection by considering both the residual energy and load-balancing.

In SEECH [6], authors address both CH and relay selection issues. SEECH selects CHs and relays separately and based on nodes eligibilities. In their proposition, high degree nodes are selected as CHs while low degree nodes are selected as relays. This separation of CHs and relay nodes achieves load balancing to some extent.

Since CH depletes its energy faster than other nodes, the clustering process should be executed periodically to rotate the CH role on different nodes, which is called re-clustering [11]. To minimize energy consumption of re-clustering, on one hand, re-clustering frequency should be minimized. On the other hand, re-clustering should be localized, not the whole network.

In cluster-based networks, routing process is similar to relay selection process. Relay selection focuses on finding the next node which has for example the maximum residual energy, while routing has more large vision. It focuses on the whole forwarding path and tries to find a best path. In [14], authors propose a routing algorithm based on ant clustering technique, which utilizes the multi-path routing scheme to provide Quality of Service (QoS). LCM [10] introduces the predicted transmission count (PTX) as a primary metric for CH and gateway determination, so as to efficiently construct a persistent and reliable routing path.

The clusters close to the base station take a considerable load over multi-hop transmission and become the hotspots. To address this issue, networks are often partitioned into unequal clusters, e.g. clusters near the sink should be smaller than those far from the sink. In [13], the optimal unequal cluster size is obtained at different layers with the symmetrical and

unsymmetrical deployment of nodes. In [15], the network is partitioned into unequal grids according to the optimal cluster size and all the nodes of a grid are formed into a cluster.

2.2 Interference-Free Strategy

To eliminate the interference in cluster-based WSNs, hybrid medium access techniques are often applied. One solution is to combine TDMA and CDMA as done in LEACH. Within a cluster the TDMA is used while different clusters use different CDMA codes for inter-cluster interference avoidance. Similarly, Frequency Division Multiple Access (FDMA) together with TDMA scheme can also provide interference-free for cluster-based networks. In [16], the network is first transformed to a tree-cluster based topology and each cluster is assigned with a frequency. Then, the algorithm of graph coloration is used for reducing the total frequency number. Within a cluster, all nodes share the same frequency but use different timeslots for contention avoidance.

Due to limited capability of a sensor node, using two medium access techniques is considered impractical. Scheduling-based access is the best for interference avoidance. Without considering the interference from other clusters, the scheduling is a relative simple work. In [17], bit-map-assisted (BMA) timeslot allocation is proposed for intra-cluster communication, in which timeslots are allocated to each member according to members' requirements. A CH collects complete knowledge about which members have data to transmit during contention period. Then, the timeslots of transmission period are allocated accordingly. E-BMA [7] improves this strategy by collecting knowledge not only via the contention period, but via the piggybacking information of data packet as well.

To address the scheduling issue considering inter-cluster interference, [18][19] model the time slot allocation as a task (project) scheduling problem. In [18], authors first construct a graph of the communication tasks using the data flows and network topology. The interference is avoided by constraining the start time of each cluster, i.e. a cluster must wait for its competing clusters to finish communication. However, the construction of task graph requires the knowledge of network topology and data flows. Obviously, it is non-scalable and non-efficient. [19] aims to find an optimal scheduling scheme which can minimize the overall energy consumption and balance the workload of the system while meeting the application's deadline.

Alternatively, in [20], the communication period assignment for different clusters is considered as the coloring problem in graph theory. However, this solution requires all clusters to be synchronized. Paper [21] investigates the communication scheduling that can maximize the parallel communication. Authors first use clique-clustering algorithm to partition the network into maximal cliques (clusters). Those clusters having common nodes are defined as adjacent clusters. Then, a communication planning is constructed as follows: non-adjacent clusters can be scheduled independently, while adjacent clusters should be scheduled one after another.

2.3 Optimization Modeling/Solution

Most of the works presented in section 2.1 use a random method to partition the network, e.g. LEACH randomly selects nodes as CHs. The simplicity of this kind of clustering protocols is attractive, but obviously clustering is not optimal. To optimize the WSN clustering, some classic clustering algorithm or optimization algorithm can be used, such as k-means, affinity propagation [23], genetic algorithm [24] and fuzzy logic [25].

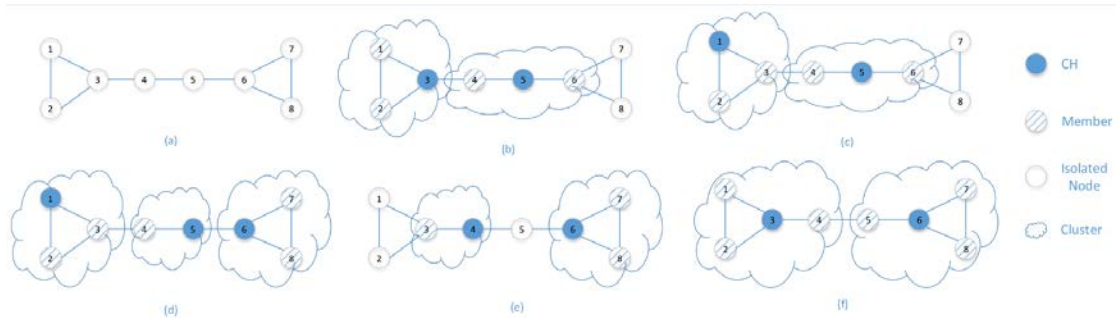


Fig. 1. Interference-free clustering

An interesting clustering optimization is proposed in [26], in which the clustering problem is modeled as hypergraph partitioning problem. In this model, the objective function is defined to minimize the energy consumption and subjected to partitioning constraint, location constraint and residual energy constraint, etc. To solve the model, authors use tabu search heuristic which consists of three types of moves that allow reassigning nodes to clusters, selecting cluster heads, and removing existing clusters.

To address both the topology control and routing problems in cluster-based WSNs, paper [27] develops three different mixed-integer linear programming (MILP) models, in which the third one minimizes both total energy consumption and range of remaining energy levels. The analysis demonstrates that it captures important characteristics of topology control and routing integration. Hui Lin [28] evolves this model by adding a new term with which a node with more residual energy has more chance to be selected as CH. In [29], the scheduling problem in cluster-tree topology WSNs is modeled as ILP. The goal of the scheduling is to grant the delay requirement of time-constrained data flow. Then, the authors use a graph theory based heuristic algorithm to address the problem.

Even though different criteria are considered in [27], still only one objective function is defined. Different from this kind of mono-objective approaches, paper [30] proposes the Multi-objective On line Hybrid Algorithm (MultiOnHA). This algorithm aims to minimize two objective functions, one for energy consumption and the other for 'non-coverage' penalty. The authors argue that joint optimizing both objectives brings better performance than the mono-objective approach where only energy consumption is set as objective function while the coverage is considered as a constraint.

3. Interference-Free Clustering Protocol

3.1 Interference Free

In this paper, we focus on: a) *avoiding the Interference for the communications from member to CH*, i.e. the interference of data collection transmission of different clusters. This is also the principal communication scenario (data collection) in WSNs. b) One-hop clusters, i.e. each member communicates directly with its CH. One-hop cluster simplified not only the intra-cluster communication, but timeslot allocation as well. In the following, we use a simple network topology (Fig. 1. a) to explain how to achieve interference-free clustering. In Fig. 1. a, each node is eligible to be CH, but they should be selected carefully to avoid interference. If node 3 and node 5 are selected as CHs (Fig. 1. b), the communication from node 4 to node 5 interferes with the intra-communication of node 3's cluster. To generate Interference-free cluster, the selected CHs should meet one of the following criteria:

1. if CHs are three-hop away or more, there is no interference between these CHs. For instance, if node 1 and node 5 are selected as CHs (Fig. 1. c), there is no interference between the two clusters. In this case, no more clusters can be formed, i.e. the rest nodes (node 7 and node 8) become isolated nodes.
2. if CHs are two-hop away, interference can be avoided if their common neighbors should be set as isolated nodes. For instance, node 4 and node 6 could be selected as CHs (Fig. 1. e), but node 5 should not be member of any clusters, i.e. node 5 becomes an isolated node.
3. if CHs are one-hop away, there is also no interference. For instance, node 1, node 5 and node 6 could be selected as CHs (Fig. 1. d), where node 5 and node 6 are direct neighbors.

These three criteria of interference-free rule can be summarized as: *Select the nodes that have no common neighbor nodes to serve as CHs. Or if the nodes with common neighbors are selected as CHs, the common neighbors should be set as isolated nodes.* The rule limits a node to cover only one CH, so a node's communication will never interfere with the CHs of other clusters. We also observe from Fig. 1 that different clustering results can be obtained. If the objective is to cover the sensor fields as much as possible while minimizing the cluster number, the clustering in Fig. 1.f is obviously the best.

As mentioned above, IFCP probably brings isolated nodes in high-density networks. Nevertheless, the isolated nodes can be used for density control. The fundamental problem in high-density WSNs is to minimize the number of active nodes to save energy, while keeping the area covered and the connectivity of nodes. Therefore, in a density control scheme, some nodes should be scheduled to sleep on purpose [31][32]. In our proposition, the isolated nodes are naturally used for density control, i.e. letting the isolated nodes in sleeping state. However, we should rotate the role of isolated node on different nodes for different round just like rotating CH role on different nodes. Although the isolated nodes cannot participate in data collection, they could be used for relaying the aggregated data to the sink, which is nevertheless out of scope of this paper.

Table 1. Optimization Parameters

Parameters	Description
S	set of sensor nodes
CH	set of cluster heads
D	set of monitoring points
r_i	node's role variable which assumes 0 if node i is an isolated node, 1 if node i is a member and 2 if node i is a CH
M_i	set of nodes which are the members of node i
N_i	set of nodes which are neighbors of node i
c_{ij}	binary variable which assumes 1 if node i covers the monitoring point j , 0 otherwise
v	energy for the transmit amplifier (J/bit/m ²)
w	energy for circuitry (J/bit)
pk	packet size
e_i	residual energy of node i
ag	energy for aggregation (J/bit)
d_{ij}	distance between node i and j
apk	aggregated packet size

3.2 Multi-objective Optimization Based Clustering

Most previous clustering methods merely focus on energy optimization, while IFCP also considers area coverage as an important objective. Therefore, our work optimizes energy consumption and area coverage simultaneously. The optimization parameters are defined in [Table 1](#). With these parameters, we devise the two objective functions and some constraints.

3.2.1 Energy Consumption Objective

Energy consumption for cluster-based networks comes from three parts: intra-communication (e_{intra}), inter-communication (e_{inter}) and data aggregation (e_{ag}). For the former two energy dissipation, we employ first order radio model. The objective function of energy consumption for each round is defined as:

$$F_e = e_{intra} + e_{inter} + e_{ag} \quad (1)$$

where

$$e_{intra} = \sum_{i \in CH} \sum_{j \in M_i} (w + v \cdot d_{ij}^2) \cdot pk + w \cdot pk \cdot (|S| - |CH|) \quad (2)$$

where $|S|$ is the number of nodes and $|CH|$ is the number of CHs. The first term refers to energy consumed for data sending of all member nodes and the second term is for data receiving of all CHs.

$$e_{inter} = \sum_{i \in CH} (w + v \cdot d_{is}^2) \cdot apk \quad (3)$$

where d_{is} is the distance between CH i and the sink. Here, we simplify the energy consumption model for inter-communication by only considering the distance between CH i and the sink, because it is complex to express the inter-communication energy from CH i to the sink considering the communication links between each pair of nodes found on the path. However, this simplification is justified by the fact that, in a dense network, routing protocols can find a path between CH i and the sink, which is similar to a line joining CH i and the sink on one hand. On the other hand, this simplification is also valid in single-hop communication models [17].

$$e_{ag} = \sum_{i \in CH} ag \cdot pk \cdot |M_i| \quad (4)$$

where $|M_i|$ is the member number of CH i .

3.2.2 Coverage Objective

The coverage objective aims to control the coverage of targets (called monitoring points in the paper). A sensor node can cover the monitoring points within its coverage radius. We use penalty function to describe coverage requirement, in which both insufficient coverage and extra coverage bring penalty. Supposing that a WSN application requires each monitoring point to be covered by k sensor nodes, the penalty function is described as:

$$F_c = \sum_{j \in D} \mathbf{1}(cv_j < k) P_{<} \cdot (k - cv_j) + \sum_{j \in D} \mathbf{1}(cv_j > k) P_{>} \cdot (cv_j - k) \quad (5)$$

where $\mathbf{1}(\cdot)$ is the indicator function; cv_j which represents the number of nodes actually covering the monitoring point j equals to $\sum_{i \in S} \mathbf{1}(r_i = 1 || r_i = 2) \cdot c_{ij}$; $P_{<}$ represents one-node

insufficient penalty for a monitoring point, while $P_>$ represents one-node extra penalty. Since insufficient coverage incurs data loss, while extra coverage only introduces redundancy and more energy consumption which is already considered in the first objective function, we set in our simulation $P_< \gg P_>$. It worth to note that a uncovered monitoring point introduces $k \cdot P_<$ penalty. Obviously, the number of uncovered monitoring points should be minimized, since it introduces lots of penalties.

3.2.3 Constraints

The clustering aims to optimize the above two objectives, but it is subjected to some constraints. The first is interference-free constraint, which represents the three criteria of the interference-free rule described above (section 3.1):

$$\begin{cases} N_i \cap N_j = \emptyset & \forall i, j \in CH, i \neq j \\ or \\ N_i \cap N_j = \{n_i, n_j\} & \forall i, j \in CH, i \neq j \\ or \\ r_x = 0 & \forall x \in (N_i \cap N_j), i, j \in CH, i \neq j \end{cases}$$

The second is one-hop constraint, which imposes that only the neighbors of a CH can be its members:

$$M_i \subseteq N_i \quad \forall i \in CH$$

It limits the network to be partitioned into one-hop clusters. The third is residual energy constraint:

$$e_i \geq e_0$$

where e_0 is a predefined energy threshold. With this constraint, only a node with residual energy above a predefined threshold could be selected as CH. The last is role constraint, which limits a node to be one role: CH, member or isolated node, given by

$$r_i \in \{0, 1, 2\} \quad \forall i \in S$$

3.3 Solution

Our goal is to jointly minimize two objective functions defined previously. To do this, we use non-dominated sorting genetic algorithm II (NSGA-II) [33] which can optimize simultaneously multiple objectives. In this algorithm, an individual which represents a clustering result is defined as a binary chromosome. An individual has n bits (n is the number of nodes) and each bit represents a node: '1' is for CH and '0' is for non-CH (member or isolated node).

3.3.1 Initiation

The first step of the algorithm is to initiate individuals (chromosomes), i.e. partition the network into rough interference-free clusters. To do this, we simply select all CHs that are at least three-hop away. Fig. 2 shows an example for the initiation procedure:

1. Set all nodes with residual energy above e_0 as CH candidates (suppose all nodes have residual energy above e_0 in Fig. 2.a).

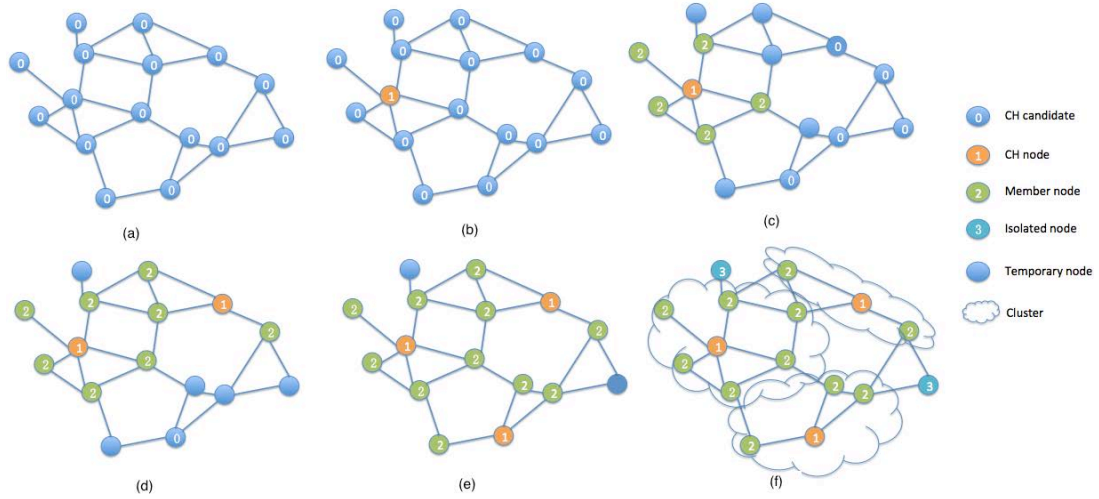


Fig. 2. Clustering Initiation

2. Randomly select a node from CH candidates as CH (suppose node A is selected in Fig. 2.b).
3. The direct neighbors of the selected CH become the members (Fig. 2.c). If these members are in candidate list, they are deleted from the list. Then, all two-hop neighbors are deleted from candidates, since they cannot play CH role anymore (become temporary nodes in Fig. 2.c).
4. Repeat the above two steps until no CH candidate is present. In the figure, node B is then selected as CH (Fig. 2.d). Afterwards, node C is selected as CH (Fig. 2.e).
5. Non-CH and non-member nodes are set to isolated nodes (Fig. 2.f) and clusters are formed (Fig. 2.f).

Clearly, the initiation procedure generates clusters in which CHs have no common neighbor nodes, so they are interference-free. Then the binary chromosome is generated according to the selection results. Namely, if a node is CH, its corresponding bit in chromosome is set to '1'. Otherwise, the bit is set to '0'. The initiation procedure is repeated so as to generate a number of individuals (suppose N individuals) which form the population P_0 .

3.3.2 Evolution and Validation

After the initiation procedure, the population P_0 uses binary crossover and mutation operation to create an offspring population Q_0 which also has N individuals. The crossover operation is as follows:

1. Randomly select two individuals from P_0 ;
2. Randomly select an integer m between 2 and $n - 1$;
3. Set m as crossover point and exchange the bits of two individuals after m ;
4. Repeat step1-3 until the number of crossover individuals reaching N .

After crossover operation, the mutation operation is executed, in which nodes could be altered with a small probability. That is, a CH (bit value=1) could become a non-CH node (bit value=0). However, due to the residual energy constraint, only non-CH nodes with residual energy above e_0 could be altered from '0' to '1' (become CHs). The crossover and mutation operation generates N new individuals (chromosomes) which form a new population Q_0 . Each chromosome corresponds a clustering result. The procedure to generate a corresponding clusters from a chromosome is: all nodes with bit '1' are set as CHs and its neighbors becomes its members (one-hop constraint). If two CHs have common neighbors, these neighbors are set to isolated nodes (interference-free constraint). All other nodes are also set as isolated nodes (role constraint).

The following step is to select N best individuals from population $R_0 = P_0 \cup Q_0$ as the new population P_1 . First, each individual of R_0 is assigned a rank equal to its non-domination level [33]. To do this, each individual is compared with other individual in the same population. The individuals are ranked with the first level are those individuals which are not dominated, i.e. there is no other individuals whose energy function (Equation (1)) and coverage function (Equation (5)) are both better than this individual. For the individual in the second level, the individuals of the first level are discounted temporarily and the same procedure is processed. The above procedure is repeated until all individuals being ranked. Afterwards, the individuals are selected according to their rank, i.e. individuals with rank 1 are selected first; those with rank 2 are selected second, until the last rank with which the number of all selected individuals is equal or larger than N . If N individuals are exactly selected, new population P_1 is formed. Otherwise, extra individuals should be deleted. This is done by sorting the individuals (according to crowding distance [33], the pseudocode of crowding-distance is shown at the bottom of the page) in the last rank and then the worst individuals are deleted.

After creating the population P_1 , Q_1 is generated similarly to Q_0 . The procedure continues until that the population P does not change from the previous one or the maximum iteration is attained. At the end, the algorithm delivers the final population P that includes a set of best individuals, which are efficient with regard to energy consumption and area coverage.

3.3.3 Best Individual Selection

Finally, we need to select the best individual from the final population. If there exists only one individual that belongs to rank 1, it is selected. Otherwise, we need to select one from individuals with rank 1. To do this, we define the parameter Dif_{num} which indicates the difference between the largest member number and the smallest member number, as follows:

$$Dif_{num} = \max |M_i| - \min |M_i| \quad \forall i \in CH$$

Smaller value of Dif_{num} means closer member number for clusters, which in turn means better load balancing. Moreover, similar member number improves the scheduling performance. In summary, besides addressing energy and coverage objectives, our proposition additionally

Crowding-distance assignment

```

for each  $i$  in  $R$ , set  $Cdist[i]=0$  //for each individual in population  $R$ , initialize crowding distance to 0
for each objective function  $m$ 
     $Cdist = \text{sort}(Cdist, m)$  //sort using each objective value
     $Cdist[1] = Cdist[N] = \infty$  //the crowding distance of boundary individual set to infinity
    for  $i=2$  to  $(N-1)$ 
         $Cdist[i] = Cdist[i] + (Cdist[i+1].m - Cdist[i-1].m) / (\max(f[m]) - \min(f[m]))$  //the crowding
        distance equals to the normalized difference of two adjacent individuals.

```

tackles the issue of load balance by selecting a final individual with minimum value of Dif_{num} .

4. Simulation and Performance Discussion

In this section, we first investigate how different parameters impact the clustering results of our algorithm. Then, a detailed analysis on comparison between IFCP, HEED-CSMA and HEED-BMA is carried out.

4.1 Analysis of IFCP Clustering Results

The configuration of IFCP refers to [Table 2](#) (genetic algorithm related configuration) and [Table 3](#) (energy evaluation related configuration).

Table 2. Configuration for IFCP

Parameters	Values
Population size (N)	10
Mutation probability	0.04
Maximum iteration	20
Required coverage number for each monitoring point (k)	3
Insufficient penalty for coverage ($P_{<}$)	10
extra penalty for coverage ($P_{>}$)	1

The studied parameters include:

- Number of total nodes,
- Number of monitoring points,
- Node's coverage radius,
- Node's communication radius.

The coverage radius represents a node's monitor capability (a monitor area is a circular area with a node as the center and its coverage radius as the radius), while the communication radius represents a node's communication capability. The proposed mechanism is analyzed by MATLAB simulation. The network field is defined as a square area 1000*1000 where nodes are randomly deployed. The monitoring points are also randomly set on the area.

First, we fix the number of monitoring points, node's coverage radius and node's communication radius to 100, 30m and 50m respectively, and vary the number of total nodes from 500 to 2300 to analyze the clustering performance. In [Fig. 3.a](#), three curves black, blue and red represent the number of clusters (CHs), member nodes and isolated nodes respectively. An interesting observation is that the number of clusters almost stays stable with the increase of total nodes, i.e. around 150. But the number of member nodes and isolated nodes increases with the number of total nodes. Since the coverage is one objective of IFCP, it always partitions the network field into maximum clusters to cover the field even when the number of nodes is 500. Therefore, the increase of nodes cannot bring more clusters. Otherwise, the interference is introduced. Whereas the increase of nodes augments both node density within clusters (member node) and outside of clusters (isolated node). Then, we investigate the criterion of the number of uncovered points which are defined as the number of monitoring points covered by none of nodes. In [Fig. 3.b](#), two curves, red and black, represent the number of uncovered points before and after clustering respectively. Obviously, after clustering, some nodes are classified to isolated nodes, the number of uncovered points increases. The figure

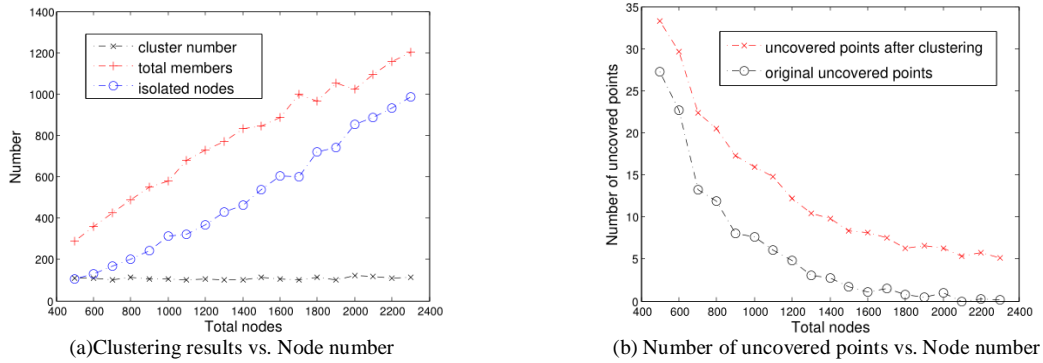


Fig. 3. Number of nodes impacts on performance

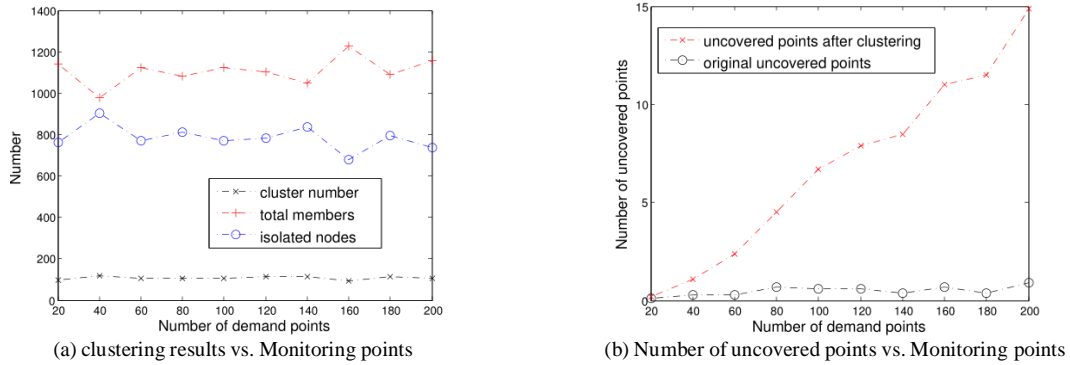


Fig. 4. Number of monitoring points impacts on performance

shows that, compared to the original uncovered points, the clustering algorithm incurs about 10 extra-uncovered points when the total nodes is about 500. Then this number decreases to 5 when the number of nodes increases to 2300. Both Fig. 3.a and Fig. 3.b demonstrate that our algorithm works better for dense networks.

Then, we fix the total node number to 2000, and vary the number of monitoring points from 20 to 200. We observe that in Fig. 4.a the clustering result remains largely unchanged. The increase of monitoring points does not lead to the increase of the active nodes (CH or member). The reason is probably that activating more nodes will violate interference-free property. Obviously, this characteristic brings more uncovered points when the number of monitoring points increases, which is demonstrated in Fig. 4.b. But, still the extra uncovered points caused by our algorithm are not significant.

We also study the clustering results by varying node’s communication radius from 20m to 100m (the number of monitoring points is set to 100). As expected, the number of clusters decreases with the increase of node’s communication radius (Fig. 5.a). In fact, our clustering method partitions the network into one-hop clusters without overlapping. The increase of communication radius makes the coverage area of one cluster increased, and so the cluster number decreases. Fig. 5.b shows that the number of uncovered points increases with the increase of node’s communication radius. This is because: first, the number of clusters decreases with the increase of radius, which makes the number of total active node decreased. Hence, the number of uncovered points increases. Second, the increase of the communication radius leads to reductions in active area which we define as the area covered by all clusters. We take two extreme cases. When the radius is 20m, our method has 648 clusters (Fig. 5.a). Each cluster covers $\pi r^2 = 1256m^2$. So the active area is $648 * 1256 = 813888m^2$. When the radius is 100m, we have 23 clusters. Each cluster covers $31415m^2$. The total active area is $23 * 31415 =$

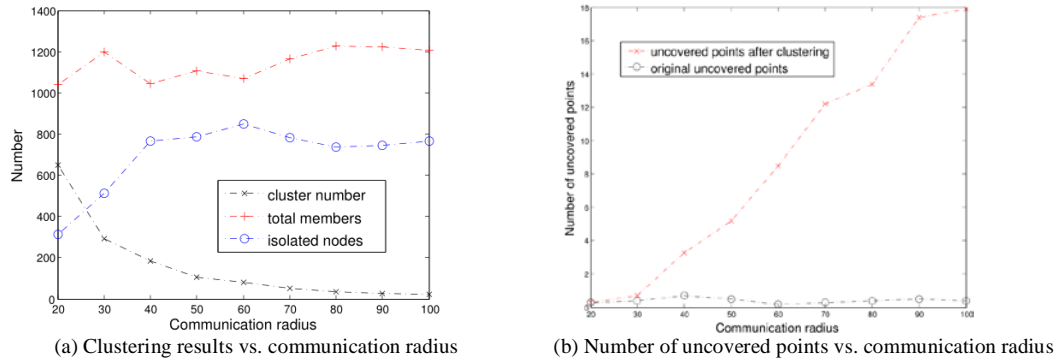


Fig. 5. Node's communication radius impacts on performance

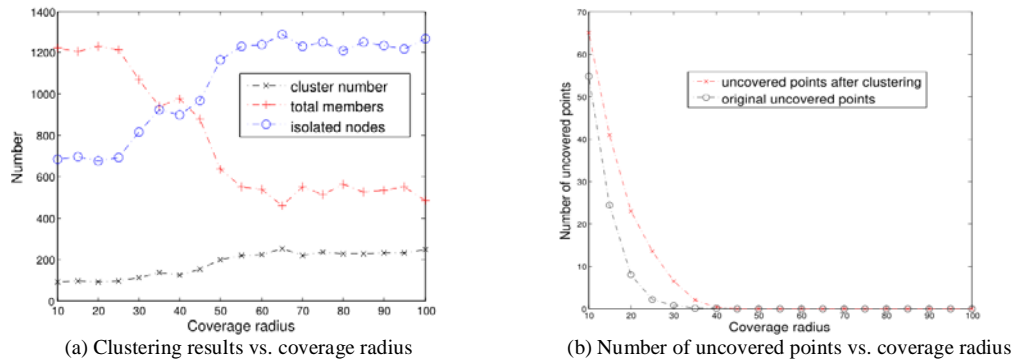


Fig. 6. Node's coverage radius impacts on performance

722545m^2 . This reduction in active area also makes the number of uncovered points increased.

Finally, we analyze how node's coverage radius impacts the clustering results by fixing node's communication radius to 50m. Previous analysis illustrates that the cluster number mainly depends on node's communication radius. **Fig. 6.a** shows that it is also impacted by node's coverage radius. The cluster number increases slightly with the coverage radius. This is probably because when the coverage radius increases, IFCP has more chances to decide which nodes should be active and which should be isolated. To minimize communication cost, the nodes close to the CH are selected to be active and form a cluster. That is, a cluster area is decreased, which makes the cluster number increased. Thanks to the increase of the coverage radius, the number of uncovered points decreases (**Fig. 6.b**). We observe when the coverage radius reaches 40m, all monitoring points are covered.

4.2 Comparison Analysis

To further analyze the performance of IFCP, we compare it with two adaptive protocols which are both based on the important clustering protocol HEED. To do this, we endow HEED with two access protocols, BMA [17] and CSMA. In HEED-BMA, after carrying out clustering with HEED, each cluster manages timeslots independently. So it does not take into account of inter-cluster interference. HEED-CSMA just uses CSMA for contention avoidance, neither intra-cluster nor inter-cluster interference is considered. HEED-BMA divide timeframe into two periods: contention period and transmission period. In contention period, a node sends a bit '1' to its CH on its slot if it has data to transmit. Otherwise, no action is taken. Based on this information, CH allocates slots in transmission period to corresponding nodes. HEED-CSMA uses classic CSMA for medium access. That is, each node senses the channel before sending

data to its CH. Data can be sent only when the channel is idle. In IFCP, each cluster allocates slots to its member rotationally. For example, if a cluster has 5 members, divides a timeframe to 5 slots for each members. IFCP avoids both intra-cluster interference and inter-cluster interference. For all three protocols, each node collects data periodically, and a CH aggregates data from all its members before forwarding to the sink.

Parameter setting is listed in **Table 3**. The performance analysis is carried out on three criteria: delay, receiving rate and energy. We will analyze two types of delays, one for transmission delay and the other for collection delay. The transmission delay is the time interval from the moment when a packet is sent by a member to the moment when the packet being received by the sink, while the collection delay is the time interval from the moment when a packet is collected to the moment when the packet is received by the sink. The gap of the two time intervals represents the waiting time for a packet on a node before being sent out. Similarly, two types of receiving rate are analyzed, one for collection rate and the other for transmission rate. The collection rate is defined as the number of received data by the sink divided by the total number of collected data, while the transmission rate is defined as the total number of received data divided by the total data sent out. The energy criterion also has two types. One for total residual energy of the network after running for a while; the other for energy efficiency which is defined as consumed energy per received packet, i.e. total consumed energy divided by total received packets. For fairness reason, we let all nodes enter sleep state if it has no data to send no matter CSMA or TDMA is used.

Table 3. Simulation Parameters

Parameters	Values
Simulation area	1000 * 1000m
Node number	2000
Node initial energy	2J
Circuit energy cost for one bit sending/receiving	$0.5e - 007J$
Contention Packet length	100bits
Data packet length	2000bits
Aggregated packet length	$2000 + 20 \cdot members $
Node's transmission radius	50m
Node's coverage radius	30m
Monitoring points	1004
Slot for contention period	5ms
Slot for transmission period	50ms

First, we analyze the performance under different data collection periods, especially under high collection rate. To do this, we vary the data collection period from 5*timeslot (node collects data every 5*50ms) to 100*timeslot. Note, longer collection period means lower collection frequency and less network load. **Fig. 7** shows the comparison results. In **Fig. 7.a**, we can see that, for all protocols, the transmission delay increases when the collection frequency decreases. It seems that this contradicts with the common sense that less network load brings less delay. In fact, in cluster-based networks, the transmission delay depends not only on network load, but also on aggregation waiting time. The latter is the time taken by each CH to collect data of all its members before aggregating it. When the collection period increases (the collection frequency decreases), more time required for CHs to collect all data. This is why the transmission delay increases as the collection period. As for collection delay, all protocols behave alike, i.e. first decrease then increase. At high collection rate, especially when the collection frequency is higher than the sending rate (the maximum packets a node

can

send

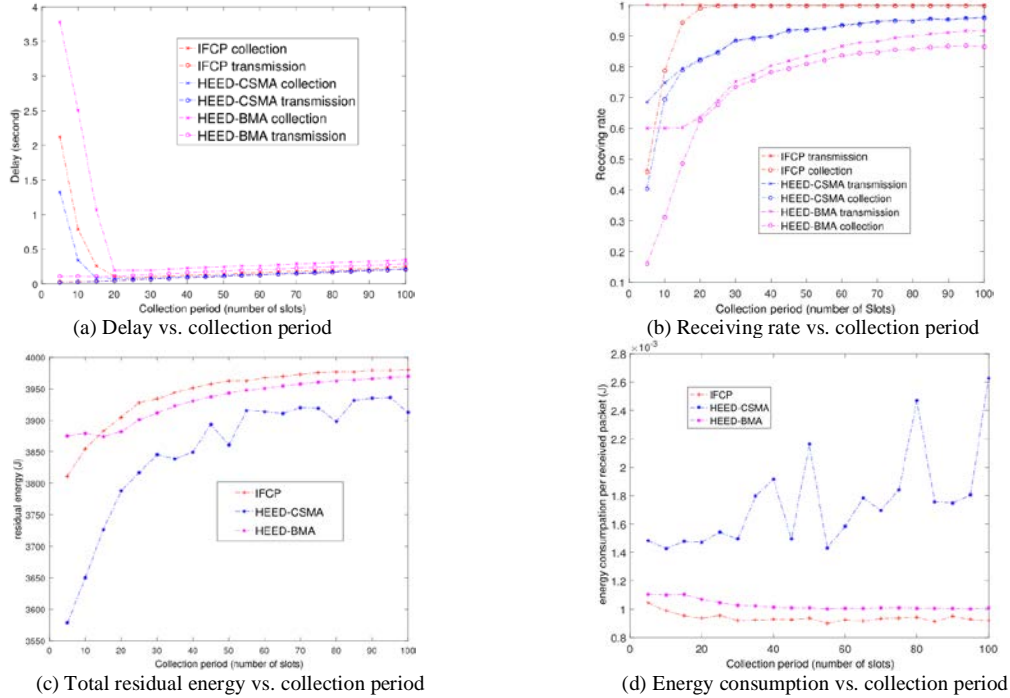


Fig. 7. Performance analysis by varying data collection period

out), collected data accumulates at the node, which makes the collection delay increased. Then, this delay decreases with the collection frequency until the collection frequency is equal or less than the sending rate. Afterwards, the collection delay increases with the collection period. This is due to the increase of aggregation waiting time that we just describe above. For all protocols, HEED-CSMA performs the best in terms of delay. This is because that HEED-CSMA sends out data immediately if channel is available. But for TDMA based protocols, a node should wait for its timeslot to send out data. Our IFCP has similar delay performance as HEED-CSMA even though a node should wait for its timeslot. HEED-BMA has the worst performance, because it has longer period which consists of contention period and transmission period.

Fig. 7.b shows the comparison result of collection rate and transmission rate. We observe that IFCP has the best performance. Thanks to the interference-free, IFCP's transmission rate reaches 100%. At low collection frequency (collection period $> 20 \times \text{timeslot}$), IFCP's collection rate is almost 100%. At high collection frequency (collection period $< 20 \times \text{timeslot}$), some data does not arrive at the sink. This is not because of data loss, but of data accumulation at a node, i.e. data has been collected but not yet sent out. The interesting observation is that although TDMA is used by HEED-BMA, its collection rate and transmission rate are even less than HEED-CSMA. BMA does not consider the inter-cluster interference, which leads to more data loss than CSMA. The figure also illustrates that both rates increases with the decrease of the collection frequency. It is easy to understand, since both rates are in inverse proportion of network load.

Fig. 7.c shows the residual energy of the network after running $2000 \times \text{timeslot}$. We can see that IFCP outperforms HEED-CSMA and HEED-BMA in most cases, except when the collection frequency is very high, where HEED-BMA incurs high collision, so transmits fewer

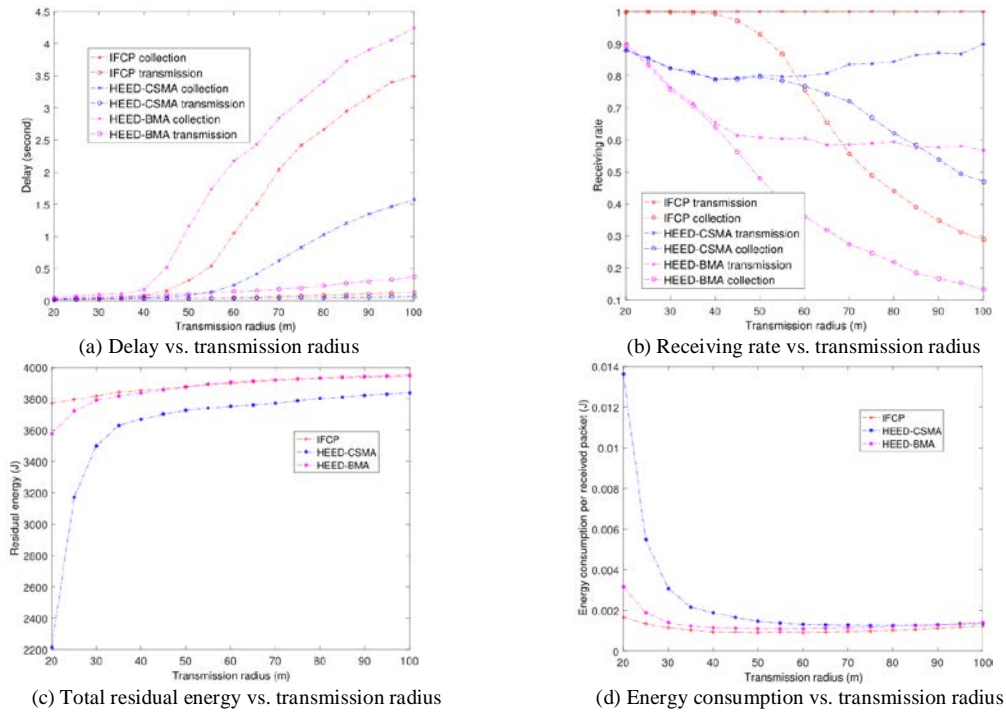


Fig. 8. Performance analysis by varying node's transmission radius

packets. In consequence, less energy is consumed. IFCP brings less energy consumption: first, not all nodes in IFCP participate in data collection (average 60% nodes). Second, IFCP's energy consumption is only due to data sending/receiving during the transmission period, while HEED-BMA also consumes energy during the contention period. As for HEED-CSMA, it consumes the most energy as expected. The energy consumed by HEED-CSMA consists of data sending/receiving, carrier sense and the back-off process. Then, with the decrease of collection frequency, the residual energy increases for all protocols. From Fig. 7.d, we observe that IFCP has the most efficient energy consumption, i.e. for each received packet, it consume the least energy. On the contrary, HEED-CSMA expends the most energy due to collision and back-off procedure.

In the following, we compare the performance by varying node's transmission radius from 20m to 100m, and fix the collection period to $15 \times \text{timeslot}$. Fig. 8.a shows the comparison result of packet delay. As explained above, HEED-CSMA brings less delay. On the contrary, HEED-BMA has the largest delay because of its longest period. Both delays of the three protocols increase with the increase of the transmission radius. As mentioned above, the delay depends not only on network load, but also on aggregation waiting time. When the transmission radius increases, the cluster area becomes larger and so the number of members increases. This makes the aggregation waiting time increased, which leads to larger delay.

Fig. 8.b shows the receiving rate by varying transmission radius. Still, thanks to the interference-free characteristic, IFCP reaches 100% transmission ratio. We also observe that the collection rate of IFCP decreases quickly. HEED-CSMA's collection rate is even better than that of IFCP when the transmission radius is larger than 68. This is because of high collection frequency. When the members increase due to the increase of the transmission radius, more and more collected data accumulate on the nodes. HEED-CSMA which can send out immediately the collected data brings less data accumulation. Furthermore,

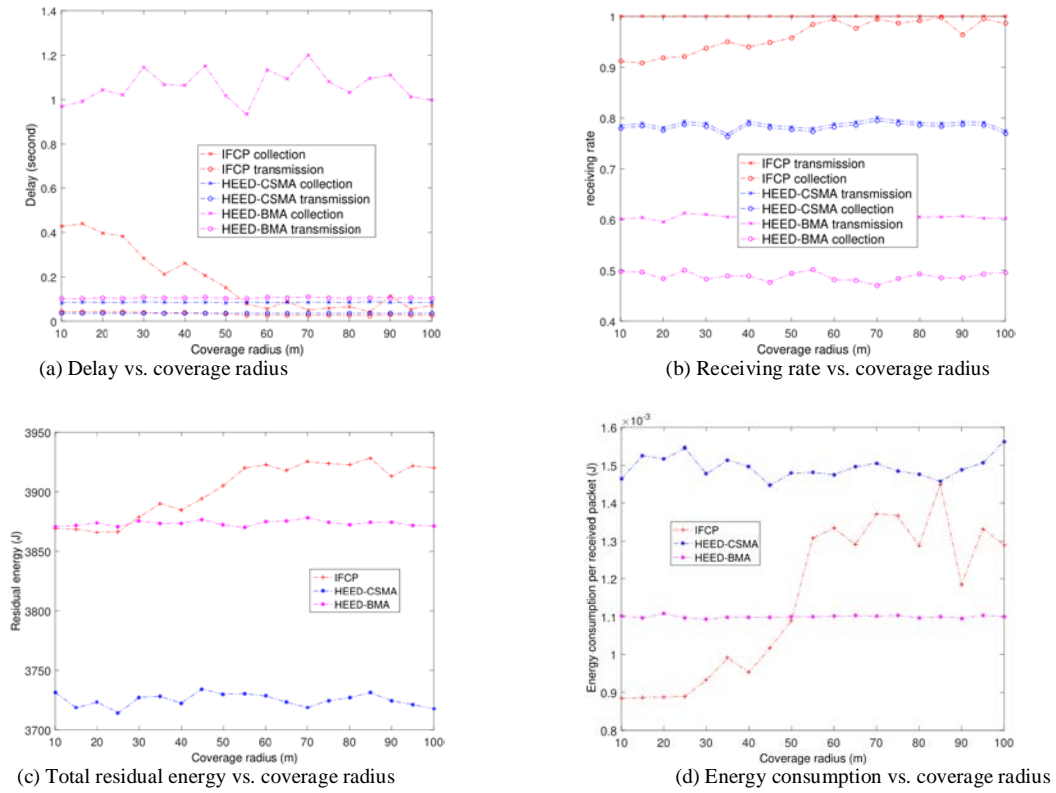


Fig. 9. Performance analysis by varying node's coverage radius

the larger transmission radius reduces data collision for CSMA, since a node has more chance to detect another ongoing transmission. This is also the reason why the transmission rate of HEED-CSMA does not decrease as others, but increases instead. As for HEED-BMA, it suffers from not only data loss, but also data accumulation, so it performs the worst. In **Fig. 8.c**, IFCP is the best in term of residual energy, for the same reason we described for **Fig. 7.c**. We also observe that all protocols increase the residual energy with the transmission radius. This is because, the increase of transmission radius makes a cluster have more members, which leads more data accumulated on nodes. In consequence, less data is forwarded on the network, so less energy is consumed. In **Fig. 8.d**, we still observe that IFCP is of the most efficient ways for energy consumption. We also observe that all protocols have better energy efficiency when the transmission radius increases. This is because, the increase of cluster members makes more data aggregated into a packet.

Now, we fix the transmission radius to 50m and vary the coverage radius. From **Fig. 9.a**, both the collection delay and transmission delay of HEED-CSMA and HEED-BMA are almost unchanged, since the coverage radius has no impact on HEED clustering result. But we observe that the delay of IFCP varies a lot, especially for the collection delay. This is because that the increase of coverage radius makes the number of cluster members decreased (**Fig. 6.a**), which in turn makes the aggregation waiting time decreased, so the delay decreases. As for the collection delay of IFCP, it decreases dramatically, not only because of the decrease of the aggregation waiting time, but also of the accumulation. Fewer cluster members bring less data accumulated. In consequence, the collection delay decreases.

As explained above, HEED clustering is not impacted by the coverage radius, so HEED-CSMA and HEED-BMA keep unchanged in terms of receiving rate (**Fig. 9.b**), total

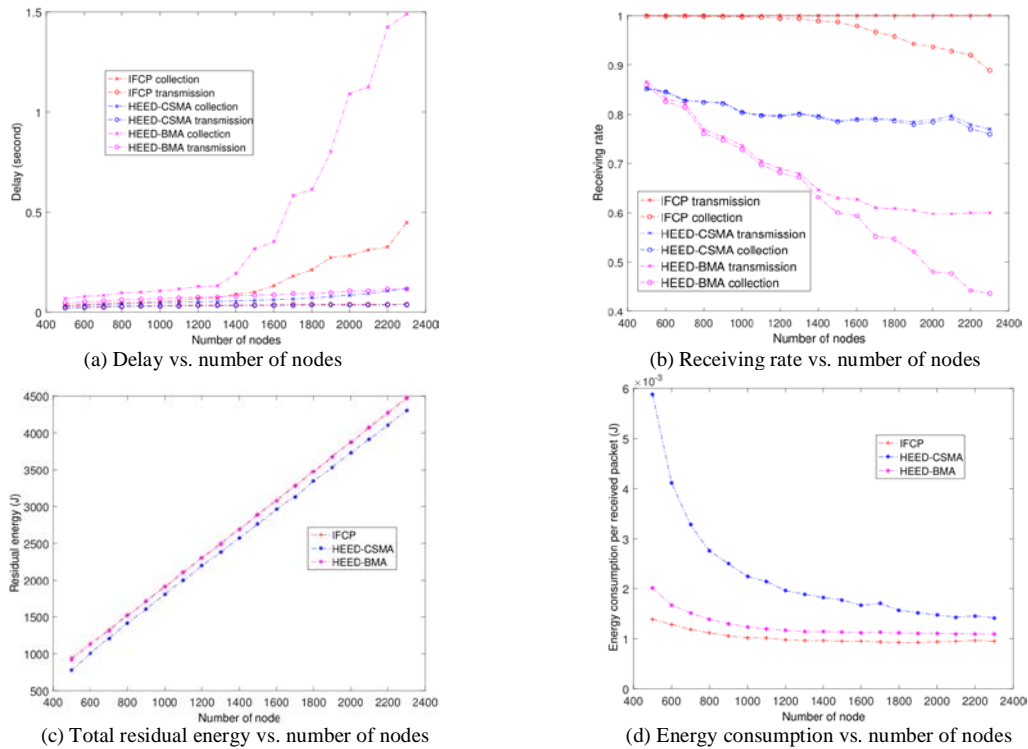


Fig. 10. Performance analysis by varying the number of nodes

residual energy (**Fig. 9.c**) and energy efficiency (**Fig. 9.d**). As expected, since less data will be accumulated, we observe in **Fig. 9.b** that the collection rate of IFCP increases with the coverage radius. As for residual energy, since more nodes act as isolated nodes, less energy is consumed for IFCP, which brings more residual energy. However, we observe that, in **Fig. 9.d**, the energy efficiency for IFCP degrades quickly, since a cluster has fewer members and so less data is aggregated.

Finally, we fix the coverage radius to 30m and vary the number of total nodes to analyze the performance. The results are shown in **Fig. 10**. When the node number increases, each cluster has more members, which increases the aggregation time. Hence, from the **Fig. 10.a**, we observe that all delay increases with the node number. The increase of the aggregation time, together with the increase of the accumulation time, makes the collection delay increased markedly. In **Fig. 10.b**, we observe that IFCP outperforms both HEED-CSMA and HEED-BMA in terms of receiving rate. When the number is less than 1000, the collection rate of IFCP is almost as the transmission rate which is 100 percent. When this number increases, a cluster has more members, which brings accumulation of data on a node. In consequence, the collection rate begins to decrease. But even when the network consists of 2300 nodes, IFCP is still has the rate more than 90%. In **Fig. 10.c**, IFCP almost consumes the same energies as HEED-BMA and both outperform HEED-CSMA. Although IFCP and HEED-BMA have the similar performance in term of the residual energy, we can observe, from **Fig. 10.d**, that IFCP has better energy efficiency than HEED-BMA. This implies that IFCP receives more packets than two others, even though fewer nodes participate in data collection.

5. Conclusion

Different from previous works in which interference-free for clustered WSNs is achieved by multiple access technologies or complex timeslot-allocation method, the proposed clustering method IFCP partitions a WSN into clusters which are inherently interference-free. The interference-free feature makes IFCP suitable for large-scale networks. Moreover, its feature of density control makes it suitable for dense networks. The system is then modeled as a multi-objective optimization problem and we use genetic algorithm to jointly optimize the energy consumption and the coverage. The performance analysis demonstrates that IFCP works well for large-scale and dense WSNs. This paper achieves interference-free resorting on a centralized mode. This will be extended to a distributed model in our future works.

Acknowledgments

The work presented in this paper is supported by Natural Science Foundation of HuBei Province, China under grant No. 2016CFB453 and by Applied Basic Research Program of WuHan City, China under grand No. 2017010201010117.

References

- [1] O. Boyinbode, H. Le and M. Takizawa, "A survey on clustering algorithms for wireless sensor networks," *International Journal of Space-Based and Situated Computing*, vol. 1, no. 2-3, pp. 130–136, 2011. [Article \(CrossRef Link\)](#).
- [2] G. Han, L. Liu, J. Jiang, L. Shu and G. Hancke, "Analysis of energy-efficient connected target coverage algorithms for industrial wireless sensor networks," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 1, pp. 135–143, Feb 2017. [Article \(CrossRef Link\)](#).
- [3] I.F.Akyildiz, W.Su, Y.Sankarasubramaniam and E.Cayirci, "A survey on sensor networks," *IEEE Communications magazine*, vol. 40, no. 8, pp. 102–114, 2002. [Article \(CrossRef Link\)](#).
- [4] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *IEEE wireless communications*, vol. 11, no. 6, pp. 6–28, 2004. [Article \(CrossRef Link\)](#).
- [5] W. B. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002. [Article \(CrossRef Link\)](#).
- [6] M. Tarhani, Y. S. Kaviani and S. Siavoshi, "Seech: Scalable energy efficient clustering hierarchy protocol in wireless sensor networks," *IEEE Sensors Journal*, vol. 14, no. 11, pp. 3944–3954, 2014. [Article \(CrossRef Link\)](#).
- [7] G. Shafiullah, S. A. Azad and A. S. Ali, "Energy-efficient wireless mac protocols for railway monitoring applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 649–659, 2013. [Article \(CrossRef Link\)](#).
- [8] J. Podpora, L. Reznik and G. Von Pless, "Intelligent real-time adaptation for power efficiency in sensor networks," *IEEE Sensors Journal*, vol. 8, no. 12, pp. 2066–2073, 2008. [Article \(CrossRef Link\)](#).
- [9] H. Zhang and J. C. Hou, "Maintaining sensing coverage and connectivity in large sensor networks," *Ad Hoc & Sensor Wireless Networks*, vol. 1, no. 1-2, pp. 89–124, 2005. [Article \(CrossRef Link\)](#).
- [10] S. Wang and Z. Chen, "LCM: A Link-Aware Clustering Mechanism for Energy-Efficient Routing in Wireless Sensor Networks," *IEEE Sensors Journal*, vol. 13, no. 2, pp. 728–736, Feb. 2013. [Article \(CrossRef Link\)](#).

- [11] H. Lin, L. Wang, and R. Kong, "Energy efficient clustering protocol for large-scale sensor networks," *IEEE Sensors Journal*, vol. 15, no. 12, pp. 7150–7160, 2015. [Article \(CrossRef Link\)](#).
- [12] O. Younis and S. Fahmy, "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, 2004. [Article \(CrossRef Link\)](#).
- [13] G. Chen, C. Li, M. Ye and J. Wu, "An unequal cluster-based routing protocol in wireless sensor networks," *Wireless Networks*, vol. 15, no. 2, pp. 193–207, 2009. [Article \(CrossRef Link\)](#).
- [14] A. Verma and P. C. Vashist, "Enhanced clustering ant colony routing algorithm based on swarm intelligence in wireless sensor network," in *Proc. of IEEE international Conference on Advances in Computer Engineering and Applications (ICACEA)*, pp. 150–154, 2015. [Article \(CrossRef Link\)](#).
- [15] J. Zhang, X. Feng and Z. Liu, "A Grid-Based Clustering Algorithm via Load Analysis for Industrial Internet of Things," *IEEE Access*, vol. 6, pp. 13117-13128, 2018. [Article \(CrossRef Link\)](#).
- [16] S. Gherairi, S. Ouni, and F. Kamoun, "Optimized tdma multi-frequency scheduling access protocols for sensor networks," in *Proc. Of 2011 International Conference on Communications, Computing and Control Applications (CCCA)*, pp. 1–6, March 3-5, 2011. [Article \(CrossRef Link\)](#).
- [17] J. Li and G. Y. Lazarou, "A bit-map-assisted energy-efficient mac scheme for wireless sensor networks," in *Proc. of the 3rd international symposium on Information processing in sensor networks. ACM*, pp. 55–60, April 26-27, 2004. [Article \(CrossRef Link\)](#).
- [18] Z. Hanzalek and P. Jurcik, "Energy efficient scheduling for cluster-tree wireless sensor networks with time-bounded data flows: Application to ieee 802.15. 4/zigbee," *IEEE Transactions on Industrial Informatics*, vol. 6, no. 3, pp. 438–450, 2010. [Article \(CrossRef Link\)](#).
- [19] W. Li, F. C. Delicato and A. Y. Zomaya, "Adaptive energy-efficient scheduling for hierarchical wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 9, no. 3, p. 33, 2013. [Article \(CrossRef Link\)](#).
- [20] G. Haigang, L. Ming, W. Xiaomin, C. Lijun and X. Li, "An interference free cluster-based tdma protocol for wireless sensor networks," in *Proc. Of International Conference on Wireless Algorithms, Systems, and Applications, Springer*, pp. 217–227, February 11-12, 2006. [Article \(CrossRef Link\)](#).
- [21] F. Avril, T. Bernard, A. Bui and D. Sohier, "Clustering and communications scheduling in wsns using mixed integer linear programming," *Journal of Communications and Networks*, vol. 16, no. 4, pp. 421–429, 2014. [Article \(CrossRef Link\)](#).
- [22] L. Xu, R. Collier and G. M. P. O'Hare, "A Survey of Clustering Techniques in WSNs and Consideration of the Challenges of Applying Such to 5G IoT Scenarios," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1229-1249, Oct. 2017. [Article \(CrossRef Link\)](#).
- [23] M. ElGammal and M. Eltoweissy, "Location-aware affinity propagation clustering in wireless sensor networks," in *Proc. of Wireless and Mobile Computing, Networking and Communications, WIMOB*, pp. 471–475, Oct. 12-14, 2009. [Article \(CrossRef Link\)](#).
- [24] M. N. Halgamuge, S. M. Guru and A. Jennings, "Centralised strategies for cluster formation in sensor networks," *Classification and Clustering for Knowledge Discovery Applications, Springer*, vol.4, pp. 315–331, 2005. [Article \(CrossRef Link\)](#).
- [25] P. Nayak and B. Vathasavai, "Energy Efficient Clustering Algorithm for Multi-Hop Wireless Sensor Network Using Type-2 Fuzzy Logic," *IEEE Sensors Journal*, vol. 17, no. 14, pp. 4492-4499, July 15, 2017. [Article \(CrossRef Link\)](#).
- [26] A. E. Rhazi and S. Pierre, "A tabu search algorithm for cluster building in wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 4, pp. 433–444, April 2009. [Article \(CrossRef Link\)](#).
- [27] H. Uster and H. Lin, "Integrated topology control and routing in wireless sensor networks for prolonged network lifetime," *Ad Hoc Networks*, vol. 9, no. 5, pp. 835–851, 2011. [Article \(CrossRef Link\)](#).

- [28] H. Lin and H. Uster, "Exact and heuristic algorithms for data-gathering cluster-based wireless sensor network design problem," *IEEE/ACM transactions on networking*, vol. 22, no. 3, pp. 903–916, 2014. [Article \(CrossRef Link\)](#).
- [29] A. Ahmad and Z. Hanzálek, "An Energy Efficient Schedule for IEEE 802.15.4/ZigBee Cluster Tree WSN with Multiple Collision Domains and Period Crossing Constraint," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 1, pp. 12–23, January 2018. [Article \(CrossRef Link\)](#).
- [30] F. V. Martins, E. G. Carrano, E. F. Wanner, R. H. Takahashi and G. R. Mateus, "A hybrid multiobjective evolutionary approach for improving the performance of wireless sensor networks," *IEEE Sensors Journal*, vol. 11, no. 3, pp. 545–554, 2011. [Article \(CrossRef Link\)](#).
- [31] F. Nakamura, F. Quintão, G. Menezes and G. Mateus, "An optimal node scheduling for flat wireless sensor networks," in *Proc. of Networking-ICN 2005*, pp. 475–482, January 7–9, 2005. [Article \(CrossRef Link\)](#).
- [32] J. Podpora, L. Reznik and G. Von Pless, "Intelligent real-time adaptation for power efficiency in sensor networks," *IEEE Sensors Journal*, vol. 8, no. 12, pp. 2066–2073, 2008. [Article \(CrossRef Link\)](#).
- [33] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, April 2002. [Article \(CrossRef Link\)](#).
- [34] D. Yang, Y. Xu, H. Wang, T. Zheng, H. Zhang, H. Zhang and M. Gidlund, "Assignment of segmented slots enabling reliable real-time transmission in industrial wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3966–3977, June 2015. [Article \(CrossRef Link\)](#).



Zhihong Chen received the B.S. degree in information security from Wuhan University, Wuhan, in 2007 and the M.S. degree in information security from University of Electronic Science and Technology of China in 2010. She was a software engineer in ZTE from 2010 to 2012. She is currently pursuing the Ph.D degree in information security at Wuhan University. Her research interest includes the privacy of outsourced data and security of multiparty computation.



Hai Lin received his B.S. degree in the Department of Thermal Engineering from Huazhong Sciences and Technologies University, WuHan, China, in 1999 and his M.S. degree in Computer Science from University of Pierre & Marie Curie, Paris, France, in 2005. In 2008, he obtained his Ph.D. from Institute Telecom – Telecom ParisTech, Paris, France. After its Post-Doc research at France Telecom, He worked as a researcher at ZTE Europe. Since 2012, he joined Wuhan University, WuHan, China, where he is currently an associate professor in the School of Cyber Science and Engineering. His interests include Internet of Thing, Edge Computing, and Future networks



Lusheng Wang received his B.Sc. in Communications Engineering in 2004 from Beijing University of Posts and Telecommunications (BUPT), China and his Ph.D. in 2010 in Computer Science and Networks from Telecom ParisTech (ENST), France. He worked as a Post-Doc during 2010 at INSA-Lyon and a Post-doc during 2011–2012 at Eurecom, France. Currently, he is a research professor and the vice-dean of Communications Engineering Department at Hefei University of Technology (HFUT), China. His research interests are resource and interference management in hyper-dense and heterogeneous networks. He has published over 40 refereed international journals and conference papers.



Bo Zhao received the Ph.D. Degree in computer software and theory from Wuhan University, Wuhan, China, in 2006. He is the author of two books and has published more than 70 articles. He is the vice-president of the school of cyber science and engineering in Wuhan University. He is also the vice director of key laboratory of aerospace information security and trusted computing ministry of education in Wuhan University. He is a senior member of CCF. His research interests include information system security, trusted computing, embedded system and cloud computing security.