

A Wireless Sensor Network Clustering Algorithm based on Hypergraph

Xu Qian^{1,2}, Hu Ji-cheng^{*1,2}, Lin Hai³, Kong Ruo-shan³, Luo Yong-en^{1,2}, Zhu Li⁴
and Mao Hua-qing⁴

(1. State Key Lab of Software Engineering, Wuhan University, Wuhan, Hubei, China, 430072

2. School of Computer Science, Wuhan University, Wuhan, Hubei, China, 430072

3. International School of Software, Wuhan University, Wuhan, Hubei, China, 430072

4. Oujiang College, Wenzhou University, Chashan University Town, Wenzhou, Zhejiang, China, 325035)

*Corresponding author: HU Ji-cheng, E-mail: jicheng@whu.edu.cn

Abstract

The paper proposes a clustering algorithm for wireless sensor network based on hypergraph. Under the hypergraph model, a wireless sensor network is mapped to a hypergraph. Then a hierarchical iterative clustering algorithm is applied to the hypergraph, thus dividing the hypergraph into multiple parts. A cluster head is selected from each part. In order to improve the partitioning process, a new modularity function is proposed. Compared with the classic clustering algorithm LEACH, simulation shows that our algorithm performs better in energy consuming and network lifetime.

Keywords: wireless sensor network; hypergraph; modularity; clustering; LEACH

1. Introduction

Wireless sensor networks (WSNs) have been widely studied in recent years and have attracted extensive attention from military, health care and many other industries. A wireless sensor network contains many nodes of very small size, each of which contains four basic components, i.e. data sensing unit, data processor, data transceiver and battery. The WSNs are deployed in various environments to capture and measure specific information (e.g., temperature, pressure, humidity) and transfer the information to a base station (also called a sink node) in a wireless manner. Since sensor nodes have limited and non-rechargeable energy resources, the energy efficiency of the nodes and the life time of the network are the main objectives of research.

In order to increase energy efficiency and reduce the power consumption caused by wireless transmission, a network is divided into multiple groups called clusters. A node is selected for every cluster and is called the cluster head (CH), and the other nodes in the cluster are called member nodes. A CH aggregates the data collected from members and transmits it to the base station. There have been various clustering algorithms proposed so far. Clustering algorithms are divided into three categories on basis of cluster head selection: centralized clustering, distributed clustering and hybrid clustering. In centralized clustering, CHs are selected by a central control node which is usually the base station, typical examples of such algorithms are LEACH-C [3], APTEEN [4]. The advantage of this kind of algorithms is that the energy consumption and message passing for head acknowledgement over the network is unnecessary. Distributed algorithms are those in which cluster heads are chosen via message passing among sensors. This kind of algorithm is efficient and flexible, examples of such algorithms are well known LEACH

[5], HEED [6]. Hybrid clustering, however, selectively combines some features of both centralized and distributed clustering.

LEACH, as a distributed low energy adaptive clustering algorithm, defines the conception of "round" which consists of an initialization stage and a data transmission stage. The initialization stage is actually the formation stage of clusters. A number ranging from 0 to 1 is generated randomly by every node and if the number is less than a certain value calculated by a predefined formula, the node becomes a cluster head. All CHs have to broadcast a message, and when the other nodes receive the message, they decide which cluster to join based on the signal strength and inform that CH with another message. The nodes in every cluster use TDMA to communicate with each other. In the data transmission stage, the members transmit data to their cluster head only once and the CH aggregates all the data and sends them to the base station. After data transmission of all clusters, the network starts a new round and gets into the initialization stage and again to re-organize clusters. There are also some defects for LEACH in spite of its simplicity and good performance. LEACH generates CHs purely in a way of probability without taking residual energy of nodes into consideration, which might result in the fact that the nodes with low energy become cluster head thus deteriorate their energy situation. Besides, energy consumption of the whole network cannot be optimized [7] due to the ignorance of locations and the density of the nodes. What's more, a huge waste of energy is caused in the frequent re-organization of clusters. Aiming at the defects mentioned above, this paper proposes a new clustering algorithm based on the hypergraph theory which models and clusters considering the density and the residual energy of the sensor nodes.

Hypergraph [8] is the extension of the graph theory, in which an edge called hyperedge can connect more than two vertices. Hypergraph has wide applications in image retrieval [9] and video segmentation [10], however, few researches have been done in WSN field. Abdelmorhit *et al.* [11] proposed a centralized clustering method for wireless sensor networks based on network energy maps and Quality-of-Service (QoS) requirements. The method abstracted the clustering into a hypergraph partitioning and its resolution is based on a tabu search heuristic. Sun YuJing *et al.* [12] presented an algorithm aiming at the secure routing problem based on hypergraph theory. Li Xiaoqiang *et al.* [13] analyzed the transmission layer, net layer and link layer of wireless sensor networks and modelled the network using hypergraph, and a reliable communication protocol for wireless sensor networks, RCBHGT was put forward. Generally speaking, the researches concerning applying hypergraph theory in WSN are rare to see. This paper proposes a clustering algorithm based on hypergraph for the WSN and a new concept called modularity [19] is introduced. Taking density and residual energy of nodes into consideration, the proposed algorithm divides the network into clusters.

The remainder of this paper is organized as follows: Section 2 introduces the theory of hypergraph and the concept of modularity and illustrates the clustering algorithm based on hypergraph. Simulation in comparison with LEACH is performed and the results are reported in Section 3. Section 4 concludes this paper and delineates some points to improve in further work.

2. WSN Clustering Algorithm Based On Hypergraph

This paper models a WSN with the hypergraph theory so that the centralized clustering is abstracted into a hypergraph partitioning problem. This section will introduce the concept of hypergraph and modularity, and then present a hypergraph hierarchical iterative clustering algorithm to solve the partitioning problem.

2.1. Hypergraph and Modularity

Hypergraph is the extension of the graph theory which expands the definition of edge, and more than two vertices can be connected by a single edge. Formally, a hypergraph

$G=(V, E)$ contains a set of vertices V and a set of edges E , and a hyperedge is a subset of V . In addition, we assign weights to vertices and hyperedges so that the hypergraph becomes a weighted graph. It can be defined as $G=(V, E, w)$. Here are some related definitions about hypergraph:

Definition 1: Partition

For a hypergraph $G=(V, E)$, a partition (or k -section) of G is a mapping of hypergraph $P: V \rightarrow I_k = \{0, 1, \dots, k - 1\}$. If k equals 2 then P is a bisection.

Definition 2: Cut Edge

For a hypergraph $G=(V, E)$, and a partition P on G , if there exist two nodes v_1, v_2 belonging to a hyperedge of G which satisfy $P(v_1) \neq P(v_2)$, then e is called a cut edge.

Definition 3: Boundary Edges, Boundary Vertices

For a hypergraph $G=(V, E)$, C is a subset of V . The boundary edges $\partial_e C$ of C is a subset of E , in which every edge contains at least one vertex belonging to C and at least one vertex not belonging to C . The boundary vertices $\partial_v C$ of C is the set of vertices each of which belongs to boundary edge of C .

Definition 4: Edge Cut

For a hypergraph $G=(V, E)$, there exists a partition P on G . The edge cut of P is the total number of cut edges caused by P .

Figure 1 is an example of a hypergraph and its partition: (a) presents the relations between the 7 vertices (v_1, v_2, \dots, v_7), in which v_1, v_2 and v_3 constitute one hyperedge and so do v_5, v_6 and v_7 ; in (b), the line across e_2 indicates a bisection (see definition 1) so that e_2 is a cut edge as well as a boundary edge for both the left and the right parts (see definition 2 and 3).

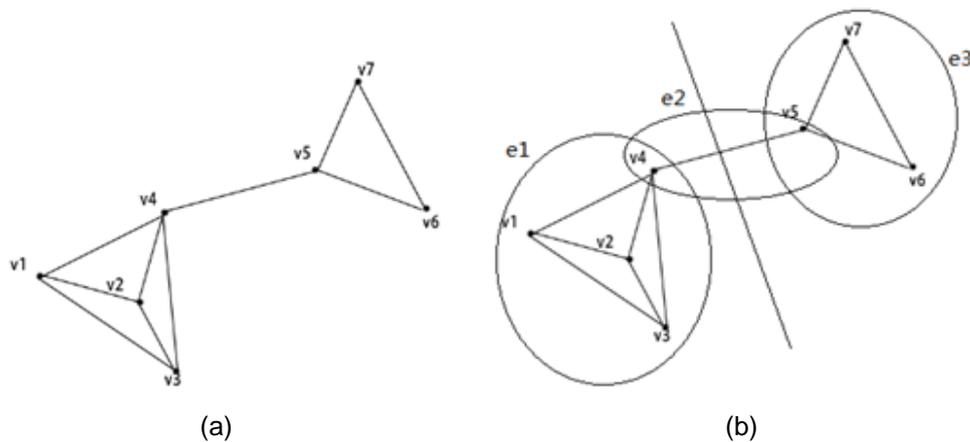


Figure 1. An Instance of Hypergraph and Its Partition

The advantage of hypergraph is that its ability to describe multi-dimensional relationships between objects, and we can build a sparse model for intensive relationship with the help of hypergraph. This feature reduces the complexity of computation and makes it possible to describe or solve some complicated and multi-dimensional problems without any loss of information.

Since the WSN clustering is modeled as a hypergraph partition problem, we need to find an optimized solution to hypergraph partition. Traditional hypergraph partition algorithms [17] are mostly based on Minimum Cut principles which usually require some priori knowledge such as the number of partition and the size of every part. However, it's impossible to define the number of clusters of WSN and the size of every cluster in advance, so the traditional algorithms cannot be applied. A basic standard for a good partition of a hypergraph is that the connections between partitions should be as few as possible and the compactness of internal connections should be greater than before partitioning. This kind of requirement is quite similar to that of the software engineering,

where the software modules should be of high cohesion and low coupling. Inspired by [19], this paper introduces the concept of modularity to improve the partitioning process and seek a better result.

Considering a hypergraph $G=(V, E)$, for a node set $C \subset V$ and a node v in C , we define the internal edges set as $E_C^o(v) = \{e \in E_C^o, v \in e\}$, where E_C^o is a hyperedge set in which each node in each hyperedge is in C . $H(C)$ indicates the cohesion of C and $P(C)$ represents the coupling of C . The definition of modularity is as follows:

$$Q_g(C) = H(C) - P(C) \quad (1)$$

The cohesion $H(C)$ is defined as follows:

$$H(C) = \frac{2}{\#C} \sum_{e \in E_C^o} weight(e) \cdot \varepsilon(e) \quad (2)$$

where $\varepsilon(e)$ is the correction factor designed to avoid huge edges that contain too many nodes, $\%e$ is the degree of a weighted hyperedge and $\#C$ is the number of nodes in C .

$$\varepsilon(e) = 1 - \frac{1}{\%e} \quad (3)$$

$$\%e = \sum_{v \in e} weight(v) \quad (4)$$

$$P(C) = \frac{1}{\#C} \sum_{e \in \partial_E C} weight(e) \quad (5)$$

$\partial_E C$ represents the set of boundary edges (see definition 3).

$$\partial_E C = \{e | e \cap C \neq \emptyset \text{ and } e \cap S_C \neq \emptyset\} \quad (6)$$

S_C means the complement set of C .

$$S_C = \{v | v \in V \text{ and } v \notin C\} \quad (7)$$

A method of iterative bisection is applied to divide a hypergraph with the modularity function defined above. In the process of bisection, two sub-hypergraphs are generated and the modularity of these two sub-hypergraphs is calculated. Then we move the nodes in boundary edges to the other side and recalculate the values of modularity between two parts. We choose the bisection scheme with the largest value of modularity as the best partition for this iteration. In this way the optimized result can be guaranteed.

Figure 2 shows an instance of how to get an optimized partition with the modularity based bisection. We take $h1$ as an example of partition, the left part of $h1$ is C , in which $e1$ and $e2$ are the internal edges of C . and $e3$ is a boundary hyperedge of C . According to formula (1), the modularity for C and for the complement set of C are calculated. The expectation value is 0.750. When we move the node 2 and node 4 to the other side of $h1$ in boundary edge $e3$, alternative bisection scheme $h2$ or $h3$ may be formed, and their values of modularity are calculated to be 0.533 and 0.611. Obviously $h1$ is the best partition at present iteration.

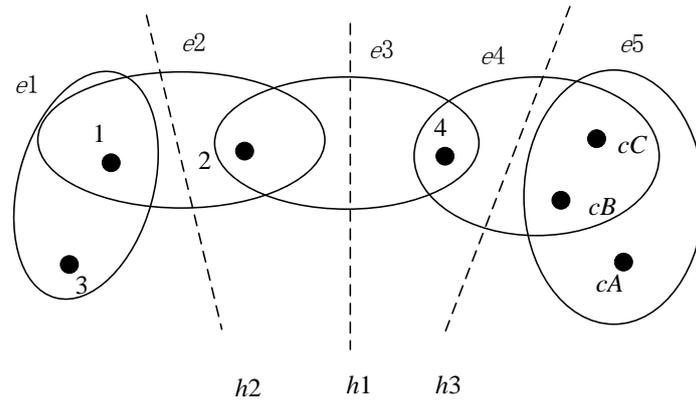


Figure 2. The Bisection Instance of Hypergraph Based On Modularity

What's more, modularity is not only a standard for evaluating partition results, but also a key to the termination of the iterative bisection. In the process of iterative bisection, if a value of modularity after the bisection is less than a predefined threshold which is 0.00 in this paper, we should rollback the bisection and end this iterative operation. Otherwise the bisection takes effect and two sub-hypergraphs will be generated.

Figure 3 shows an instance of bisection termination based on modularity. We complete the first bisection via $H1$ and get two sub-hypergraphs $M0$ and $M1$. Since the value of modularity is 0.600 which is greater than 0.00, the bisection process goes on. Then we divide $M0$ and $M1$, and get two values -0.000 and 0.600 respectively. Hence $M1$ is divided successfully and we get $m00$ and $m01$, while the division of $M0$ is cancelled. If we continue to divide $m00$ and $m01$, we get -0.000 in both cases. Thus, the partition of fig. 3 terminates at the bisection $H2$ of $M1$.

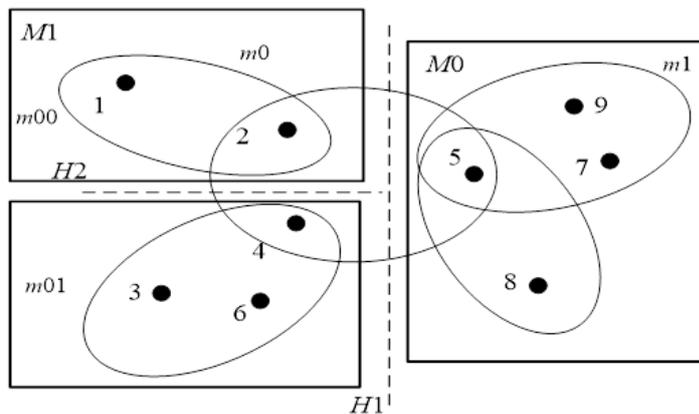


Figure 3. An Instance of Bisection Termination Based On Modularity

2.2. Clustering Algorithm

2.1.1. Hypergraph modeling: The circumstance we consider in this paper is that sensor nodes distribute randomly in a specific square district with predefined size. Sensor nodes collect data and send them to their cluster heads. Cluster heads aggregate the data and send them to the base station directly. The base station is located at the center of the

district. This paper focuses on isomorphic WSN. Our work is based on the following assumptions:

- All nodes in the network are equal and are able to connect with each other directly, they are power limited and have the same initial energy.
- The locations of sensor nodes and base station are known in advance and fixed in the whole network lifetime.
- Sensor nodes are capable of sensing their own residual energy.
- Sensor nodes' initial energy is enough for sending data to the base station and the transmission power can be adapted to different distances.

Based on the assumptions mentioned above, we model the wireless sensor network with a hypergraph in compliance with the following rules: Each sensor node corresponds to a vertex in the hypergraph; a hyperedge is generated for each sensor node, which connects all the other nodes covered by a circle that takes that sensor node as the center and sets the radius to a predefined threshold value. Thus, the hypergraph can be defined as follows:

$$G=(V,E,W(v),W(e))$$

where $V=\{v_1, v_2, \dots, v_n\}$ is the set of sensor nodes, $E=\{e_1, e_2, \dots, e_n\}$ is the set of hyperedges, $W(v)=\{w(v_1), w(v_2) \dots w(v_n)\}$ is the weights set of sensor nodes, and $W(E)=\{w(e_1), w(e_2) \dots w(e_n)\}$ is the set of hyperedge weights. The weights for a sensor node and for a hyperedge are defined as follows:

$$\text{weight}(v) = \frac{E_{\text{current}}}{E_{\text{init}}} \cdot \text{Neighbor_num} \quad (8)$$

$$\text{weight}(e) = \text{AREA_X} / \text{averDistance}(\text{central}, \text{all}) \cdot \frac{E(\text{central})_{\text{current}}}{E(\text{central})_{\text{init}}} \quad (9)$$

where E_{current} is the residual energy of node v , E_{init} is the initial energy of node v , and Neighbor_num presents the number of neighbor nodes covered by the above mentioned circle of the node v . $\text{averDistance}(\text{central}, \text{all})$ means the average distance between the center node and all the other nodes. AREA_X is the length of whole district. $E(\text{central})_{\text{current}}$ is the residual energy of the center node and $E(\text{central})_{\text{init}}$ is the initial energy of the center node.

2.2.2. Hypergraph partition: The fundamentals of hypergraph partition in this paper are iterative bisections. On the basis of the modularity standard and the comparison of energy consumption before and after partition, the bisection which meets the standard will iteratively generate two sub-hypergraphs till the end.

The first step of hypergraph partition is seed selection. A node is selected randomly as the seed. Considering the possibility that a huge hyperedge containing many vertices exists, which will lead to duplicate selection, we seek a dimension reduced selection strategy shown in fig. 4. Firstly nodes in each hyperedge are coarsened into a single vertex, and then a seed is selected randomly from the set of vertices generated by coarsening. At last the seed is mapped from the coarsened graph to the original hypergraph. This strategy guarantees the uniform distribution of seeds.

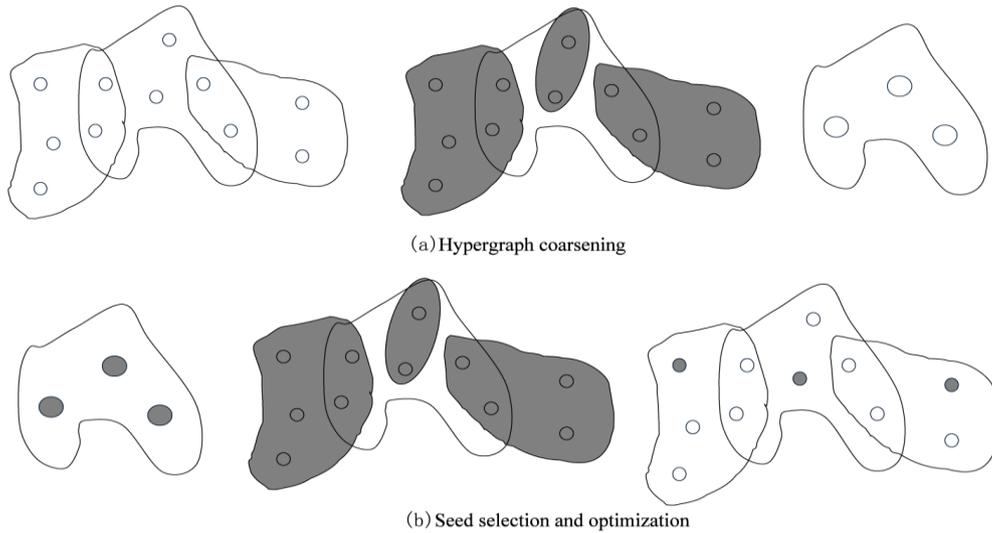


Figure 4. Hypergraph Coarsening and Seed Selection

After seed selection, the breadth-first search strategy(BFS)[18] is used to traverse the hyperedges centering on the selected seed. The hyperedge with heavy weight has the priority to be first traversed. When the number of nodes traversed reaches the half of the total number of nodes in these hyperedges, the traverse is terminated and the hyperedge which traverse stop at is the cut edge. In order to get an optimized partition, we repeat the bisection based on modularity with each vertex in the cut edge. The original hypergraph generates two sub-hypergraphs and the sub-hypergraphs generate more iteratively. The method above is used for every bisection. The process is described as follows and shown as fig. 5.

1. Define a *max_heap* and a *cut_heap*, and the nodes in cut edge are stored in *cut_heap*.
2. Apply the modularity based hypergraph bisection to the top node of *cut_heap*, which means that the top node generates a new bisection each time it is moved. We calculate expectation values of modularity for the two sub-hypergraphs and insert the max value into *max_heap*. Then we adjust the top element of *max_heap*.
3. Use breadth-first search strategy to traverse all nodes in *cut_heap* and implement step 2 to these nodes till *cut_heap* is empty. The bisection scheme with the max modularity value is the best partition of this hyperedge.

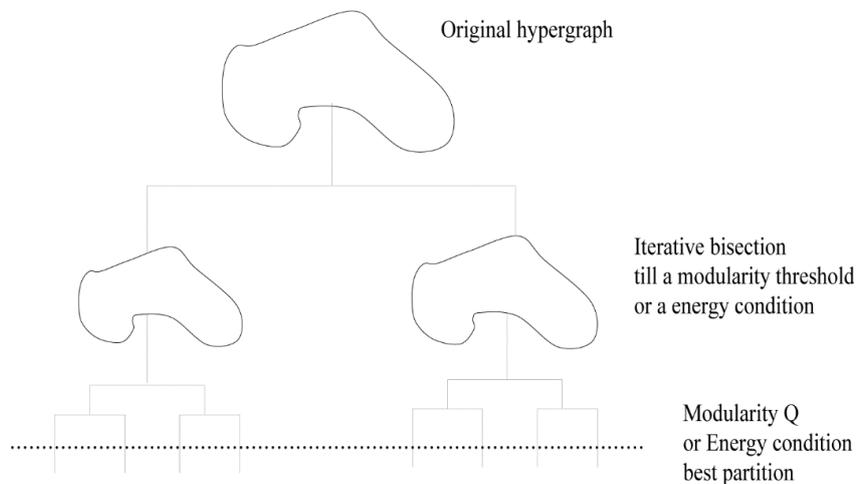


Figure 5. Hierarchical Iterative Clustering Algorithm

For each sub-hypergraph, we repeat the partition process till one of the two situations is satisfied: (1) The value of modularity is less than the threshold which is set to 0.00 in this paper; (2) For the two sub-hypergraphs after partition, we assume them as two clusters and select their cluster heads in a way which will be illustrated below. Then we calculate the value called E1 of the two clusters' energy consumption, i.e. the energy that members consume for sending data to their CH, and the energy consumed by the two CHs transmit data to the base station. For the original hypergraph before partition, we also assume it as a cluster and select a CH for it, and calculate its energy consumption in the same way which is called E2. If E1 is greater than E2, then this situation is satisfied.

The above steps finish the clustering preliminarily, but some isolated nodes may be produced which are far away from most of the nodes in its cluster and so must be adjusted. In order to remove an isolated node, we first traverse all the hyperedges containing this node and mark the clusters that have intersection with the hyperedges. Then the isolated node joins the cluster that intersects with the most hyperedges.

Now the wireless sensor network has been divided into clusters. The number of clusters and the size of each cluster are calculated automatically by the algorithm.

2.2.3. Cluster head selection: Cluster head is selected based on the residual energy of nodes and the distance between the node and the cluster center. We define a formula as follows to calculate the suitability of a node v to be the cluster head:

$$\text{Suitability}(v) = \alpha \cdot \frac{1}{\text{centerDistance}} + \beta \cdot \frac{E_{\text{current}}}{E_{\text{init}}} \quad (10)$$

where centerDistance is the distance between node v and the cluster center, E_{current} is the residual energy of v , E_{init} is the initial energy of v . α and β are weights of $\frac{1}{\text{centerDistance}}$ and $\frac{E_{\text{current}}}{E_{\text{init}}}$, which are both set to 1.

The greater the value of suitability is, the more appropriate is node v to be a cluster head. We take the node with the max value as the cluster head. When all CHs are selected, the clustering of wireless sensor network is completed.

2.3. Network Running Mode

The clustering algorithm in this paper is a centralized clustering algorithm, and the running mode of the WSN is similar to that of LEACH. Cluster reconstruction for network is continually executed in the whole lifetime, and the time between is called a period. A period contains two stages: clustering stage and data transfer stage. Clustering stage consists of the work that we have discussed in section 2.2, which is executed at the base station, and then the clustering results are sent to all the sensor nodes. This stage will not consume the nodes' energy so the lifetime can be prolonged effectively.

The data transfer stage starts after the clustering stage. This stage is made up of multiple rounds which are somewhat similar to the "round" in LEACH. During each round the member nodes transmit data to their CHs, and the CHs aggregate the data and transmit them to the base station in a single hop manner. Each cluster takes advantage of different TDMA to communicate with other clusters preventing from signal interference.

After a certain number of rounds of data transfer, once a cluster head is going to run out of its energy, it informs the base station and then all nodes will send information about their residual energy to the base station through their cluster heads. Thus the data transfer stage is over, a new period will start and the network will get into the clustering stage again to reconstruct clusters. We know that one disadvantage of centralized clustering algorithm is that all nodes have to inform the base station of their residual energy or some other information, which obviously results in much energy dissipation, and the method in this paper avoids such dissipation.

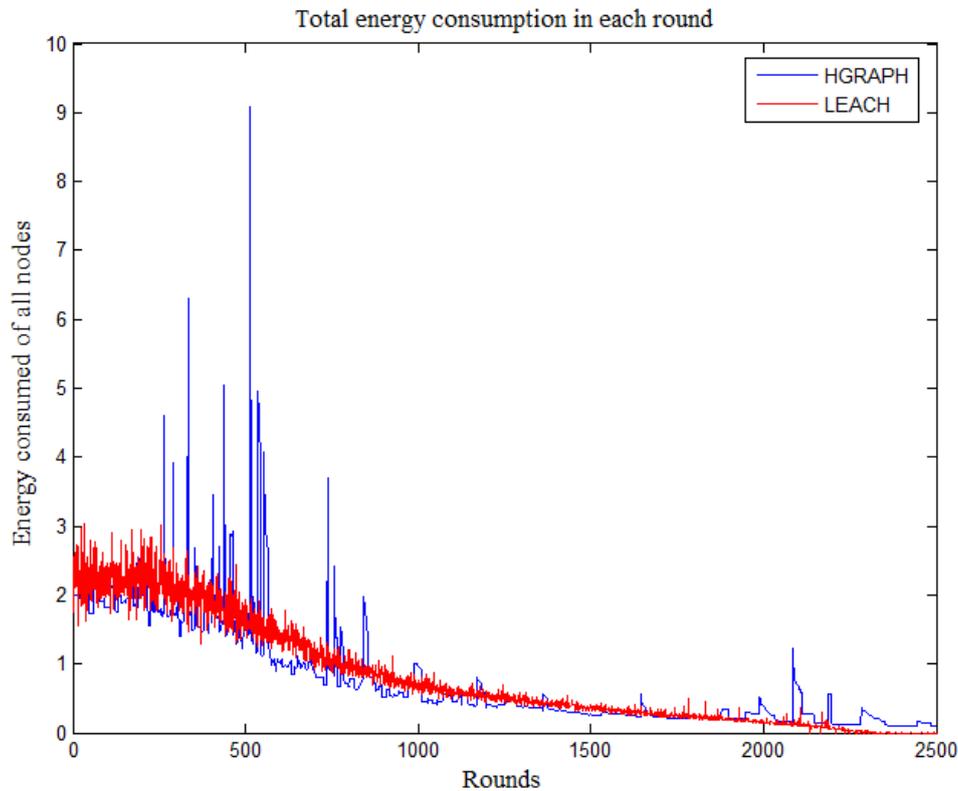
3. Simulation Experiments

In order to test the performance of the proposed algorithm, this paper simulates the wireless sensor network with Visual Studio2010 and MATLAB on Windows. The clustering stage is implemented with C++, which performs hypergraph modeling and partition and CH selection. MATLAB is used to simulate the data transfer stage. Finally we analyze and compare the results between our algorithm and LEACH. Some basic simulation parameters are listed in table 1.

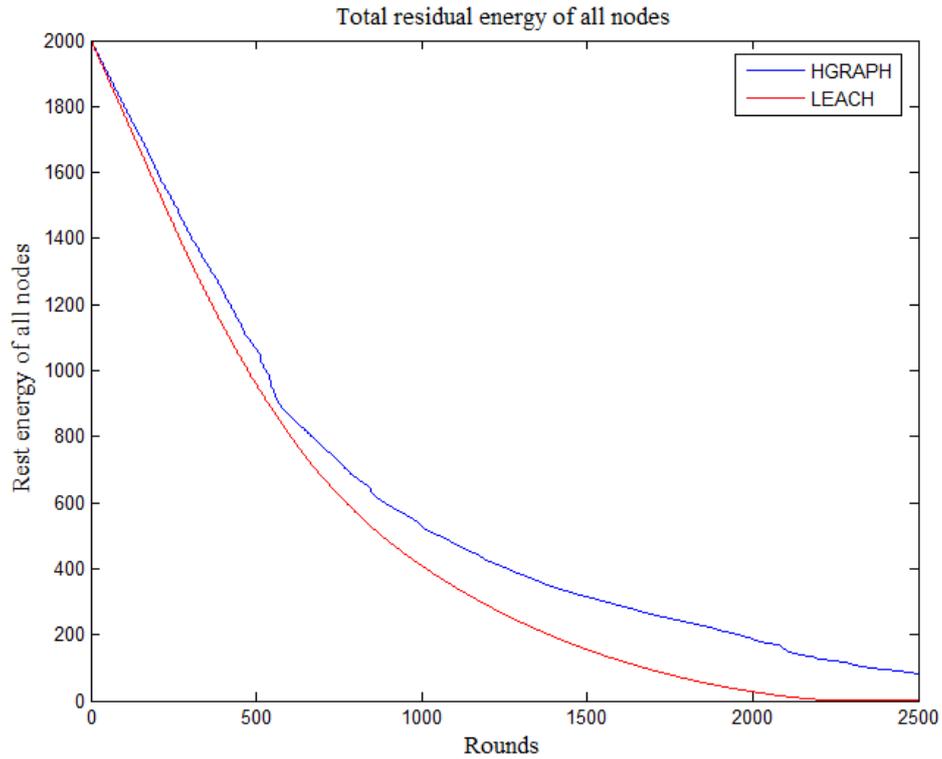
Table 1. Simulation Parameters

Parameters	Value	Parameters	Value
Distribution Area	[600,600]	Length of data packet	1000bit
Node Number	1000,2000	Power consumption of reception	50nJ/bit
Initial Energy	2J	Power consumption of transmission	50nJ/bit/m ²
Base station location	(300,300)	Power consumption of aggregation	5nJ/bit

Figure 6 describes the energy situation of the network in each round when nodes number is 1000. (a) depicts the total energy consumption in each round and our algorithm exhibits lower consumption in general, yet we can see some peak values which indicate the partition results are not good in certain periods and lead to the co-existence of some very large clusters and some very small clusters. As (b) shows, the total residual energy of our algorithm is always higher than that of LEACH. Totally, as far as energy is concerned, our algorithm performs better than LEACH.



(a) Total energy consumption in each round



(b) Total residual energy of all nodes

Figure 6. Energy Situation of the Network

Figure 7 describes the change of the number of living nodes. Due to the mechanism of cluster reconstruction in each round, all nodes share the energy consumption in LEACH. But our algorithm will not reconstruct the clusters until one of the cluster heads is going to die. Therefore the first dead node appears in the 22th round in LEACH, while in the 4th round in our algorithm. Generally speaking, LEACH performs better in early stage, but in the later phase, the nodes die faster than in our algorithm. As a result, our algorithm brings better energy efficiency and longer lifetime of the network.

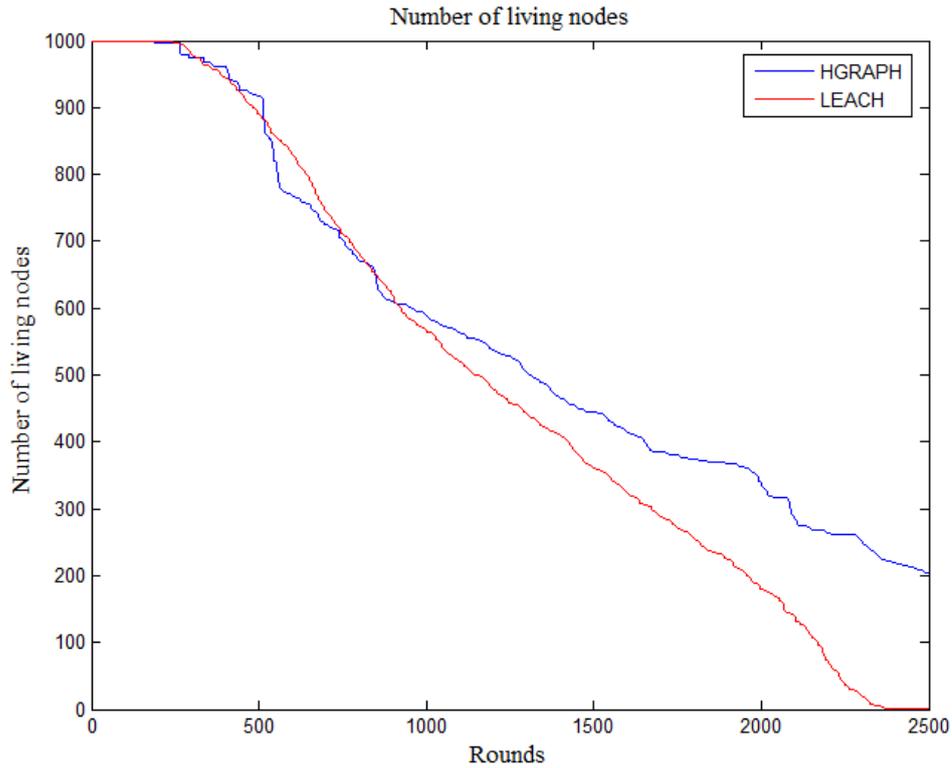
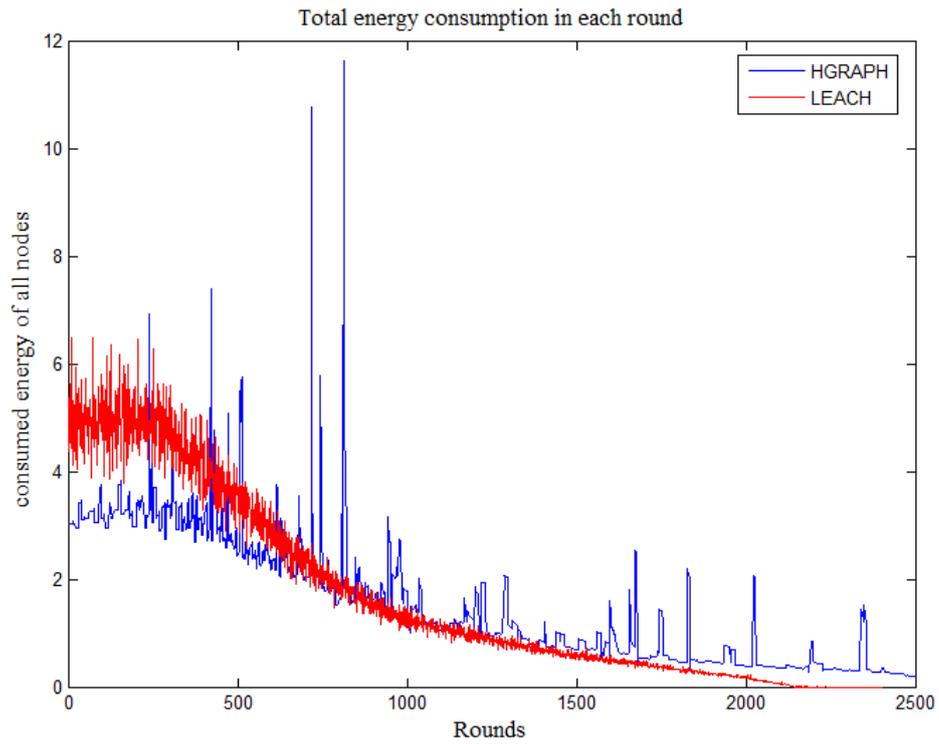


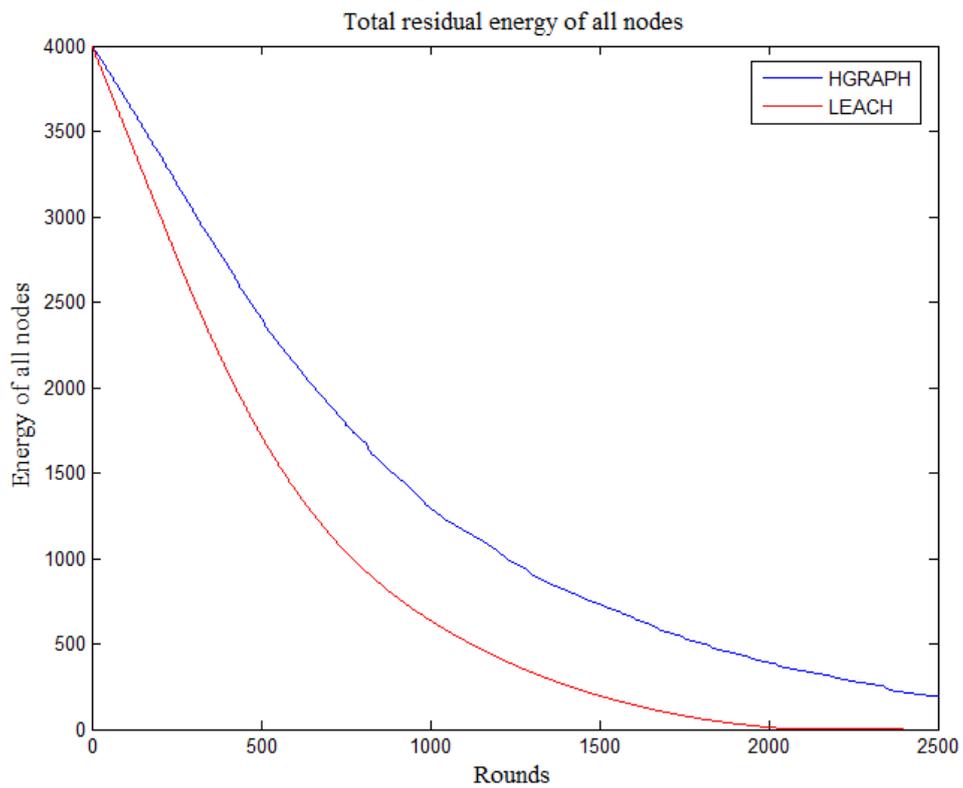
Figure 7. Number of Living Nodes

The reason for the above better results lies in two aspects, on one hand, the clustering stage of our algorithm is executed at the base station, which saves the energy of the sensor nodes consumed for cluster heads selection and acknowledgements, on the other hand, the modeling method and the key formulas about modularity are designed based on nodes' residual energy and node density, which makes the partition results more reasonable. LEACH, in contrast, selects cluster heads simply on the basis of probability without taking residual energy of nodes into consideration, which results in the possible uneven distribution of cluster heads.

Considering that hypergraph is suitable for large-scale topology, we do experiments with 2000 nodes and also have a comparison with LEACH. As figure 8 and 9 show compared with figure 6 and figure 7, the performance difference between our algorithm and LEACH is more significant as nodes number increases, which proves the advantage of our algorithm on larger scale networks.



(a) Total energy consumption in each round



(b) Total residual energy of all nodes

Figure 8. Energy Situation of the Network

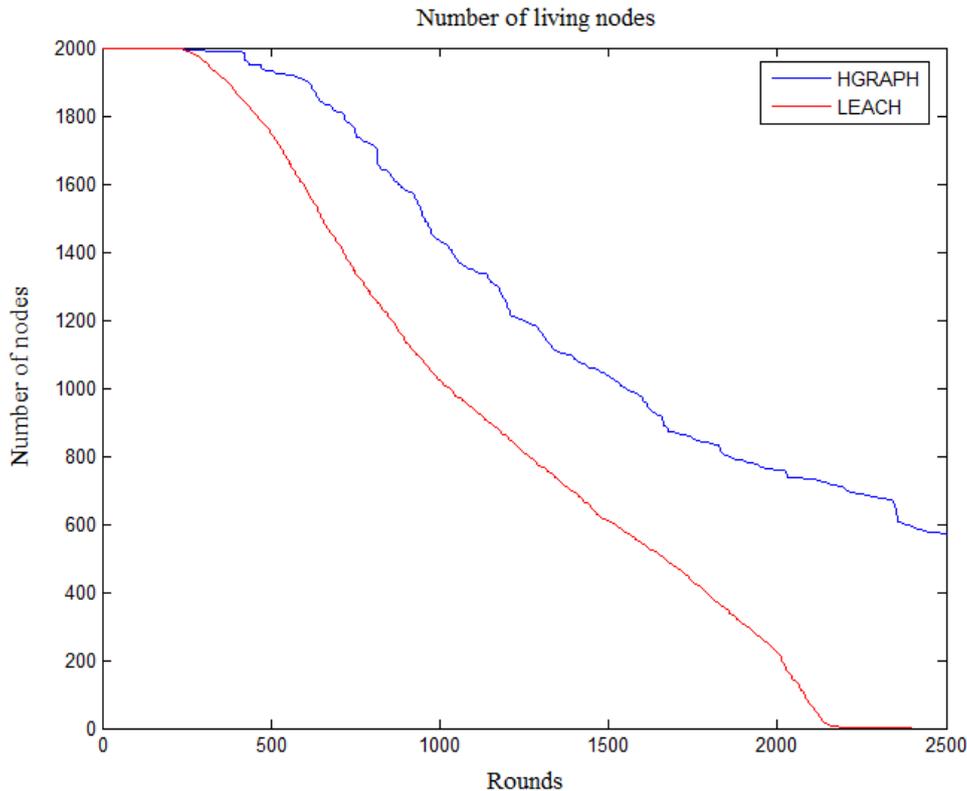


Figure 9. Number of Living Nodes

4. Conclusion

This paper introduces the concept of modularity and proposes a new clustering algorithm for wireless sensor networks based on hypergraph. The algorithm takes the residual energy of nodes and node density into consideration to select cluster heads, which makes clustering reasonable and optimized. Through simulations in comparison with LEACH, the algorithm is proved to be an effective method to prolong the network lifetime and increase the energy efficiency.

There is still something to be further improved in the future work: (1) The definition of modularity may be more suitable for the cases in which clusters communicate with each other by way of multihop, so changing the network mode to a multihop one deserves more research. (2) The partition results are not always ideal, hence we will consider about introducing a balance factor in the algorithm to improve the stability of the performance of the partition process.

Acknowledgements

The work presented in this paper was supported by the International Science & Technology Cooperation Program of China under grant No. 2013DFA12460, the Fundamental Research Funds for the Central Universities of China under grant No. 2042015kf0053, the Zhejiang Provincial Natural Science Foundation of China under grant No. LY16F010015 and No. LQ13D010001, and partially supported by the Department of Education of Zhejiang Province's Research Project (Y201430369).

References

- [1] Garcia-Hernández, C. F., Ibarguengoytia-González, P.H.,Hernández, J.G., and Perez-Díaz, J.A., Wireless Sensor Networks and Applications: a Survey in International Journal of Computer Science and Network Security, VOL.7 No.3, (2007).
- [2] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., and Cayirci, E.,Wireless Sensor Networks: a Survey , in Elseveir International Journal of Computer Networks(2001).
- [3] Heinzelman W B, CHANDRAKASAN A P, BALAKRISHNAN H. An application-specific protocol architecture for wireless microsensornetworks[J]. IEEE Transactions on Wireless Communications, 2002,1(4):660-670
- [4] MANJESHWAR A, AGRAWAL D P. APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks[C]//Proceedings of 16th International Parallel and Distributed Processing Symposium(IPDPS'02), Apr 15-19,2002,Fort Lauderdale, FL,USA.Los Alamitos, CA,USA:IEEE Computer Society, 2002:195-202.
- [5] Heinzelman WR, Chandrakasan AP, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensors networks. In: Nunamaker J, Sprague R, eds. Proc. of the Hawaaian Int'l Conf. on System Science (HICSS). Washington: IEEE Press, 2000.3005–3014.
- [6] YOUNIS O, FALMY S. HEED: A hybrid, energy-efficient, distributed clustering approach for Ad hoc sensor networks[J]. IEEE Transactions on Mobile Computing, 2004,3(4): 366-379.
- [7] Li Ya-qing, Li La-yuan. Improvement and Simulation of LEACH Routing Protocol in Wireless Sensor Networks.[J]-Computer Engineering.2009(10)
- [8] Karypis G, Aggarwal R, Kumar V, Shekhar S. Multilevel hypergraph partitioning: applications in VLSI domain[J]. Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, 1999, 7(1): 69-79.
- [9] Huang Y, Liu Q, Zhang S, Dimitris N.Metaxas. Image retrieval via probabilistic hypergraph ranking[C]//Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010: 3376-3383.
- [10] Huang Y, Liu Q, Metaxas D. Video object segmentation by hypergraph cut[C]//Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on. IEEE, 2009: 1738-1745.
- [11] A. El Rhazi and S. Pierre, "A Tabu Search Algorithm for Cluster Building in Wireless Sensor Networks," in IEEE Transactions on Mobile Computing, vol. 8, no. 4, pp. 433-444, April 2009.
- [12] SUN Yu-geng, Zhang Ju-wei, Ji Hao, Ding Ying-qiang. A Secure Routing Algorithm for Wireless Sensor Networks Based on Hypergraph Theory.[J].Journal of Tianjin University,2008,41(2):175-182.DOI:10.3969/j.issn.0493-2137.2008.02.010.
- [13] Li Xiao-qiang. Reliable communication protocol for wireless sensor networks based on hyper graph theory[J].Transducer and Microsystem Technologies,2010,29(6):46-49.DOI:10.3969/j.issn.1000-9787.2010.06.014.
- [14] Xiao Yong-kang, Shan Xiu-ming, Ren Yong.The Survey of the Fourth Generation of Mobile Communication[J].China Academic Journal Electronic Publishing House,2002,18(6):12-14.DOI:10.3969/j.issn.1000-0801.2002.06.003.
- [15] Shi Mei-lin, Ying Chun.Routing protocols for ad hoc networks: a survey[J].Journal of China Institute of Communications,2001,22(11):93-103.DOI:10.3321/j.issn:1000-436X.2001.11.013.
- [16] Tian Wen-feng, Liu Chen. Research on Clustering Routing Algorithms in Wireless Sensor Networks.[J].Computer Knowledge and Technology,2009,5(10):2729-2730,2733.DOI:10.3969/j.issn.1009-3044.2009.10.080.
- [17] Kak A C, Rama G M, Sarkar S. Api-Based And Information-Theoretic Metrics For Measuring The Quality Of Software Modularization[J]. Software Engineering IEEE Transactions on, 2007, 33(1):14-32.
- [18] Meyer, Ulrich. External memory BFS on undirected graphs with bounded degree; proceedings of the twelfth annual ACM-SIAM symposium on Discrete algorithms, 2001[C].
- [19] Wei Xiao-feng, Hu Ji-cheng, Luo Yong-en. Automatic Software Module Partition based on Hypergraph Partitioning Algorithm[J]. Computer Engineering,2016,01:71-76.

Authors



Qian Xu, she received the B.S. degree in software engineering from Wuhan University, China in 2015. She is currently a master student in computer science, Wuhan University, China. Her research interests include wireless sensor networks, hypergraph theory and software architecture.



Jicheng Hu, he received the B.S., M.S. and Ph.D degrees in computer science from Wuhan University, China in 1986, 1992 and 1994, respectively. He is currently a professor in computer science, Wuhan University, China. His research interests include embedded system, embedded hardware design and cloud computation. More recently, he has been focusing on hypergraph theory and its application.



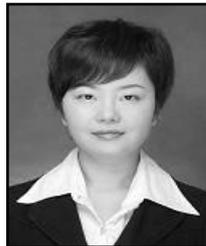
Hai Lin, he received the B.S. degree from the Department of Thermal Engineering, Huazhong Sciences and Technologies University, Wuhan, China, in 1999, the M.S. degree in computer science from the University of Pierre and Marie Curie, Paris, France, in 2005, and the Ph.D. degree from the Institute Telecom–Telecom ParisTech, Paris, in 2008. He held post-doc research with France Telecom. He was a Researcher with ZTE Europe. Since 2012, he has been with Wuhan University, Wuhan, where he is currently an Associate Professor with the International School of Software. His interests include Internet of Things, network mobility, sensor networks, and future networks.



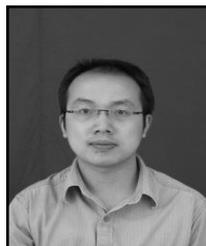
Ruoshan Kong, he received the B.S. and Ph.D. degrees in computer science from Wuhan University, China in 2002 and 2007, respectively. He is currently an associate professor in the International School of Software, Wuhan University, China. His research interests include mobile communication protocols, wireless network protocols, and cognitive network protocols.



Yong-en Luo, he received the B.S. degree in computer science from Wuhan University, China in 2013. He is currently a master student in computer science, Wuhan University, China. His research interests include hypergraph theory and data mining.



Li Zhu, she received his B.S., M.S., and Ph.D. degrees in 2004, 2006, and 2010, respectively, from Wuhan University, Wuhan, China; she has been working continuously in Wenzhou University as a lecturer until now. Her main research interests include cognitive network, wireless network, information security *etc.* She is coauthor of more than 30 articles, which mostly were published in high-profile journals.



Huaqing Mao, he holds his PhD from Wuhan University, China, in 2010. He then joined the Information Security Laboratory at the Wenzhou University, China, and his main research interest is the tall building interior emergency response and evacuation. More recently, he has been focusing on the model construction and optimization of tall building.

