A Wireless Sensor Network Topology Control Method Based on Hypergraph

Heng Shao[1], Ruoshan Kong[1*], Jicheng Hu[2], Li Zhu[3] and Huaqing Mao[3]

[1] International School of Software, Wuhan University, Wuhan, Hubei, China
[2] School of Computer, Wuhan University, Wuhan, Hubei, China
[3] Oujiang College, Wenzhou University, Chashan University Town, Wenzhou, Zhejiang, China
*Corresponding author: Ruoshan Kong, Email: kongruoshan@whu.edu.cn

Abstract

A Wireless Sensor Network (WSN) model is proposed. The model is based on the hypergraph theory which partitions the topology structure of the WSN. The partition is performed intermittently to optimize the structure of the WSN. Compared with HEED, the proposed model preforms better to extend the network lifetime. Although the model is centralized, it is applicable for most scenarios where a distributed model is needed.

Keywords: wireless sensor network; hypergraph; topology; lifetime

1. Introduction

1.1. Wireless Sensor Network

A Wireless Sensor Network (WSN) is a distributed sensor network. The sensors at the end of the network can perceive the surrounding environment and the collected data can be transferred to a sink node by the network. The sensor nodes in a WSN communicate with each other wirelessly, and the location of these sensor nodes is irregular. Each sensor node has four basic components: data collection, data processing, data transmission and power supply. Different from common networks, WSN is energy-conscious since their power is restricted. And the researches show shat the topology control has a huge impact on the performance of WSN.

All sensor nodes have their own power thresholds for communication. If the sensor node communicates under its own power threshold, its energy consumption will have a linear relationship with the communication distance, while if the communication power goes above the threshold, the energy consumption will increase exponentially with the increase of the communication distance.

1.2. Graph Theory and Topology Control

A Graph is one type of data structure which is composed of a collection of vertexes and a collection of edges. It can be described as $G = (V, E)$ in mathematical expression. Topology control can be applied to enhance the performance of the network. Topology control algorithms of the sensor networks can be divided into two main stages—topology construction and topology maintenance.

A hypergraph [6] is one kind of generalization of a graph, where the set of edges is replaced by a set of hyperedges. A hyperedge extends the definition of an edge by allowing more than two vertices to be connected by a single hyperedge, so a hypergraph can store more information than a graph. Formally, a hypergraph $H = (V,E)$ is defined as a set of vertices V and a set of hyperedges E, where each hyperedge is a subset of the vertex set V, and the size of a hyperedge is the cardinality of this subset.

Hypergraph partition is to partition the vertices of a hypergraph into k roughly equal parts, such that more hyperedges could be limited within a single part and the number of cross-part hyperedges is minimized.

The hypergraph partition applied in this paper is based on the HMETIS algorithm. The HMETIS algorithm consists of three phases--Coarsening Phase, Initial Partitioning Phase, and Uncoarsening and Refinement Phase. The HMETIS is based on multilevel hypergraph partitioning. Different from traditional graph partitioning algorithms that compute a partition of a graph by operating directly on the original graph, the HMETIS reduces the size of the hypergraph by order reduction in the coarsening phase and after it partitions the smaller, lower order hypergraph, it refines the hypergraph to construct an optimized partition for the original hypergraph.

During the coarsening phase, the HMETIS employs algorithms that make it easier to find a high-quality partition at the coarsest graph. During refinement, the HMETIS focuses primarily on the portion of the graph that is close to the partition boundary. Because of that, the HMETIS can quickly produce high-quality partitions for a large variety of hypergraph.

1.3. Related Works

Most researches are within two aspects—power control and sleep scheduling. Power control means controlling the transmission power of each sensor node. Sleep scheduling means scheduling sensor nodes between sleep state and work state. Clustering is the key technique of sleep scheduling and the LEACH algorithm proposed by Heinzelman [3] is one of the representative technologies. Topology control aims to make as many sensor nodes as possible works under their own power threshold. Kirousis studied the complexity of the topology control problem [1], and Narayanaswamy puts forward one way of combining the topology control and the network protocol [2]. One major challenge of sleep scheduling or power control is constructing and controlling the topology of the network.

Our research in this paper is based on sleep scheduling, and it focuses on topology control to choose appropriate sensor nodes as cluster heads which will perform sleep scheduling in their own clusters.

Topology control can be centralized or distributed. "Distributed" here means the topology control is performed by multiple sensor nodes in a distributed manner, so a central node that oversees the global topology information is not required. On the contrary, the centralized topology control requires collecting information of all the sensor nodes, and the algorithm should be executed in a certain sensor node or the base station.

One widely known algorithm of sleep scheduling and distributed topology control is the LEACH algorithm, in which the base station communicates with cluster heads in single hop. Because of the single hop communication, the LEACH algorithm is more appropriate for wireless sensor networks in small scale. Another representative algorithm is the HEED algorithm. It selects cluster heads by the factor of average minimum reachability power (AMRP) which means the average power of all nodes to communicate with a specific node in the cluster. Different from the LEACH algorithm, the HEED algorithm can be applied for wireless sensor networks of all scales because of its multi-hop communication.

The HEED[5] (Hybrid Energy-Efficient Distributed clustering) algorithm, which we will compare with in this paper, periodically selects cluster heads according to a hybrid of the nodes' residual energy and one secondary parameter, and the rest nodes join clusters such that communication cost is minimized. The secondary parameter for selecting cluster heads is the intra-cluster communication cost. With a constant number of iterations, the HEED selects all cluster heads ensuring the connectivity of the network.

Although the HEED achieves a connected multi-hop [4] inter-cluster network when a specified density model and a specified relation between cluster range and transmission

range hold, it also requires each node to participate in the clustering process and hence will cost much energy for topology control, so we propose a new methods that can solve this problem in this paper.

In this paper, a scheme of topology control is put forward based on hypergraph theory. Since the base station communicates with all cluster heads in a single hop manner, our algorithm performs better when the network scale is not very large.

1.4. The Organization of this Paper

This paper is organized as follows: section two describes the way of collecting basic information to from the WSN topology, and it also analyzes the characteristics of the WSN topology. We simulate the design scheme proposed in this paper and compare its performance with the HEED algorithm in section three. Section four makes a summary for this paper.

2. WSN Topology Construction

2.1. Basic Assumptions

The wireless sensor network that can use the proposed topology control method in this paper has characteristics as follows:
(1) The location information of all the sensor nodes is unknown, and the network is quasi-stationary which means although the position of the sensor nodes is dynamic, each sensor node's location is static for the entire network;
(2) The connection between each pair of sensor nodes is bidirectional, which means each pair of connected sensor nodes can communicate with each other at the same power;
(3) Every sensor node has the ability to act as the cluster head. In the beginning, each node has different amount of energy, and the node with less energy should not take more work than the others;
(4) The location of the base station is not predefined, but its location has big effect on the performance of the network. One duty of the base station is to store and process the data collected from the sensor nodes.

With our topology control method, a wireless sensor network will be divided into many clusters. In each cluster, a cluster head will be selected by the base station. The cluster head collects data from each sensor node and performs sleep scheduling for sensor nodes in the cluster. Following some rules that will be introduced later, the cluster heads are undertaken by different sensor nodes from time to time, so the lifetime of the network can be extended.

2.2. Basic Information Needed to Build Topology

Most centralized topology control methods require getting accurate location information of each sensor node, which limits their applicability. Although the topology control method proposed in this paper is centralized, it does not require each sensor node's location information. Similar to those distributed methods, it requires the base station to collect the neighbor table of each sensor node, and the hypergraph can be constructed based on these tables without requiring the accurate location information of all sensor nodes. The neighbor table of a certain node is generated at the initial stage of the WSN and is used to indicate which nodes in this network can be directly connected to this node under its own communication power threshold.

2.3. Construct Topology of WSN

The construction of the topology consists of three steps—generating neighbor table, clustering, and selecting cluster heads.

2.3.1. Neighbor table generation: Each node in a WSN is represented with its ID number, so a neighbor table of one sensor node is composed of neighbor nodes' ID numbers.

As mentioned in the previous section, each sensor node has its own communication power threshold and transmits within this threshold so as to save energy.

At the beginning of generating neighbor tables, each sensor node broadcasts its ID number and then sends its ID number again by unicast to those nodes from which it receives their ID numbers. When one node receives an ID number by unicast, it can add this ID number into its neighbor table. In this way, a node can make sure that each node in its neighbor table can communicate with itself bi-directionally.

2.3.2.Clustering: Based on the neighbor tables collected from each sensor node, the base station maps these data to a hypergraph, in which a hyperedge is generated for every sensor node and consists of this sensor node and all its neighbor nodes. Obviously there will be a lot of overlapping hyperedges, and we will partition the hypergraph according to the overlapping situation.

We use the third party program that implements the HMETIS algorithm to compute a hypergraph partitioning in the base station. As mentioned in section one, the HMETIS contains three phases--Coarsening Phase, Initial Partitioning Phase, and Uncoarsening and Refinement Phase. With HMETIS, we need to decide how the nodes should be grouped together during the Coarsening Phase, and how the quality of the bisection generated in the Initial Partitioning Phase is refined during the Uncoarsening and Refinement Phase.

In HMETIS, there are five available schemes for choice to group nodes, which are described as follows according to the HMETIS manual document.

(1)   Hybrid First-Choice scheme (HFC) is a combination of the first-choice and the greedy first-choice scheme.

(2)   First-Choice scheme (FC) is a scheme in which nodes are grouped together if they are present in multiple hyperedges. Groups of arbitrary size are allowed to be collapsed together.

(3)   Greedy First-Choice scheme (GFC) is a scheme in which nodes are grouped based on the FC scheme, but the grouping is biased in favor of faster reduction in the number of the hyperedges that remain in the coarse hypergraph.

(4)   Hyperedge scheme is a scheme in which nodes are grouped together that correspond to entire hyperedges. Preference is given to hyperedges that have large weight.

(5)   Edge scheme is a scheme in which pairs of nodes are grouped together if they are connected by multiple hyperedges.

Also, there are two choices of refinement policy to refine the quality of the bisection. They are Fiduccia-Mattheyses (FM) refinement scheme and one-way Fiduccia-Mattheyses refinement scheme.

Since the hypergraph mapped from the WSN can be different in density and size, it is hard to find one best choice of the group scheme and the refinement policy for all. With experiments of each scheme, we give preference to the GFC scheme and FM policy in clustering.

2.3.3. Cluster heads selection: When selecting cluster heads, two factors must be considered—remaining energy and position. Since the cluster head will undertake the data collection from the common sensor nodes and the communication with the base station, the node with more remaining energy is more likely to be selected as the cluster head. The position factor affects the power consumption of a cluster too. It will be beneficial to select a sensor node as a cluster head from the middle area of the cluster because the average distance between the common nodes and the cluster head could be shorter and thus the transmission consumes less power. The influence of position factor is stronger if

a cluster occupies a larger scale, and conversely, it is weak when the scale is small. Considering this uncertainty, we choose the remaining energy as the primary factor to select cluster heads.

Firstly the base station traverses all nodes in a certain cluster to find out two nodes with the most remaining energy. If the difference of these two nodes' remaining energy is more than a preset threshold value, the node with more remaining energy will be selected as the cluster head; otherwise, we will compare the position of these two sensor nodes by calculating the intersection of their own neighbor node table and the set of this cluster. The node with the greater intersection will be selected as the cluster head. In particular case, if the intersections of these two nodes have the same size, we choose the one with more remaining energy.

Different from those distributed methods, the cluster heads selection is performed by the base station, so the energy consumption of the computation will be saved and the lifetime of the WSN can be extended.

After the cluster heads selection, the base station will send all the sensor nodes' ID numbers of a cluster to its cluster head, so that the cluster heads could know which nodes are under its control and broadcast their ID numbers to the corresponding sensor nodes. Inside each cluster, the communication is carried out in a single hop manner, since the HMETIS algorithm guarantees a small size for every cluster.

3. Simulation

3.1. Configurations

A simulation is divided into multiple rounds each of which consists of two phases. The proposed topology control method is executed in the first phase, and in the second phase, the cluster heads send the collected data from the sensor nodes under their control to the base station and perform sleep scheduling for those common sensor nodes to save their energy. In detail, the second phase of each round includes multiple periods of data collection. In this paper, each round of simulation includes 5 times of data collection.

The simulation is executed in MATLAB, through which we generate a wireless sensor network. Each node's energy is randomly assigned at a value between 1J and 4J. The upper limit of the transmission distance for each node is randomly set between 20 meters and 50 meters.

Although the HEED is distributed, it has some common assumptions on the sensor network as our method, as described in section 2.1. Some basic parameters for simulating the two algorithms are as follows:

(1) Position of the base station: the position will affect both algorithms, and so we simulate with the base station at two different position—(0,0), (1000,1000);

(2) Power consumption of reception: 50nJ/bit;

(3) Power consumption of transmission: 10pJ/bit/m2;

(4) Number of bits in one packet: 1000bit;

3.2. Results of simulation

Firstly, we compare the performance of the HEED and our topology control method in networks with different density. All these three simulations are performed in an area of 2000m X 2000m, and the base station is in position (1000, 1000). Figure 1 shows two algorithms' performance when the number of nodes is 1000. Figure 2 and Figure 3 show the performance when the number of nodes is 2000 and 3000, respectively.
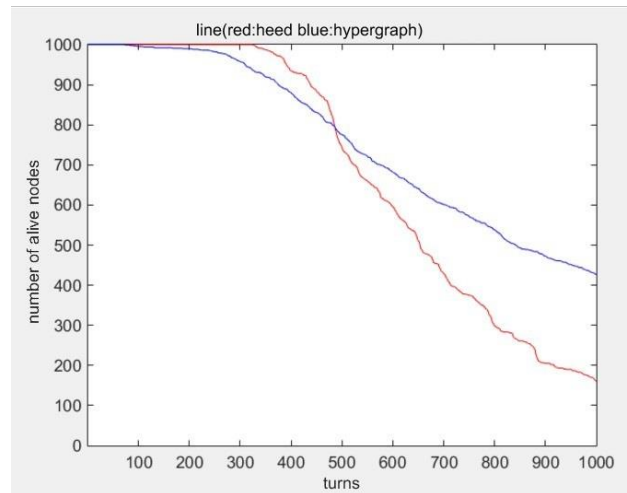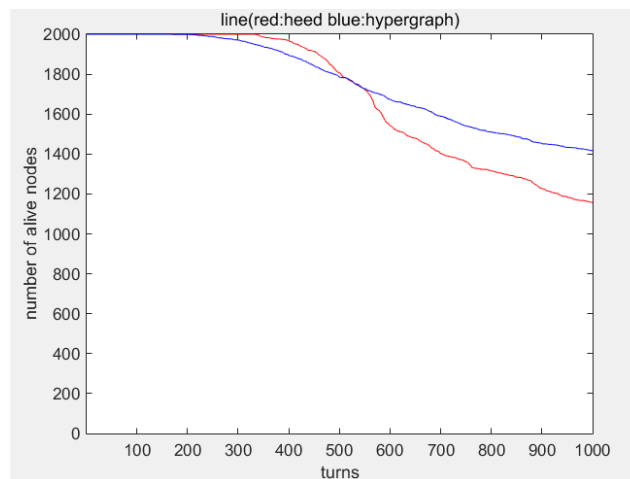
Figure 1. Performance Comparison with 1000 Nodes

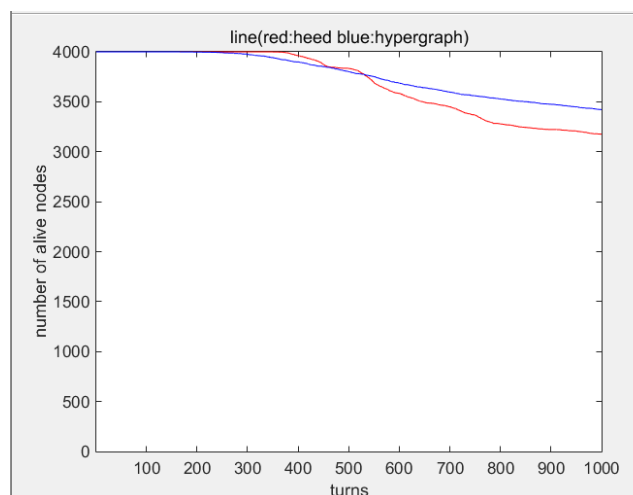**Figure 2. Performance Comparison with 2000 Nodes**

Figure 3. Performance Comparison with 3000 Nodes

As the three figures show, our topology control method exhibits weaker performance than the HEED at the beginning of these simulations, because our topology control method requires collecting neighbor tables from all sensor nodes to construct the

hypergraph in this stage, which consumes more energy than the HEED at the beginning. But after that, our topology control method performs better and consumes less energy due to its reasonable topology.

Secondly, in an area of 2000m X 2000m with 1000 nodes, we compared the performance of the HEED and our topology control method in networks with the base station at different positions. Figure 4 shows the base station at (1000,1000). Figure 5 shows the base station at (0,0).
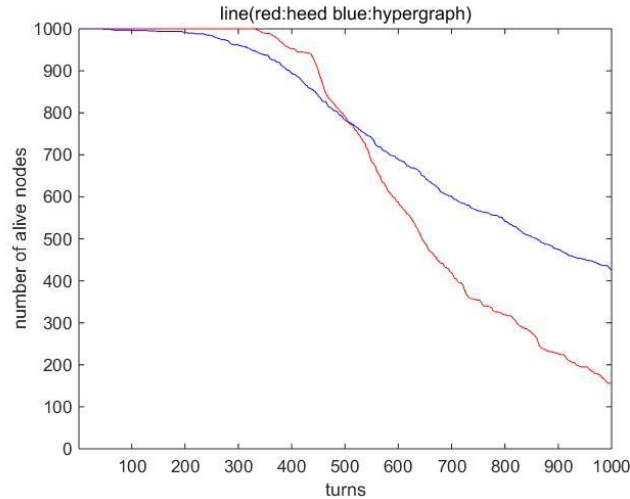

Figure 4. Performance Comparison with the Base Station at (1000, 1000)
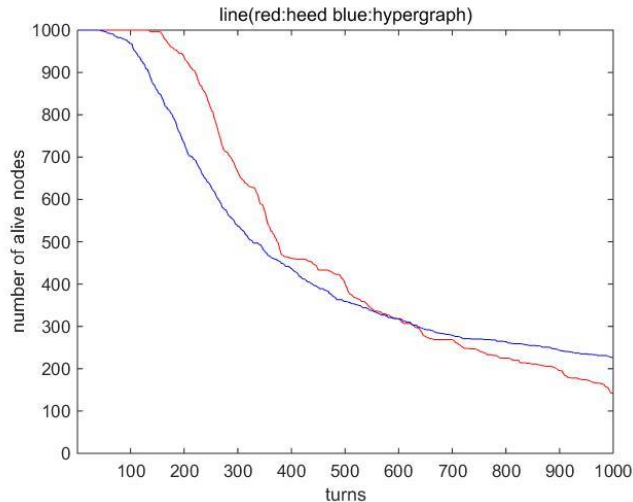
Figure 5. Performance Comparison with the Base Station at (0,0)

As the two figures show, the position of the base station has a significant effect on the performance of our topology control method. By contrast, the influence to the HEED is relatively slight.

4. Conclusion

Based on the HMETIS algorithm, the proposed method constructs the topology in the base station and sends topology messages to all the selected cluster heads. Since the topology message is sent in a single hop manner, the topology control method performs better when the network scale is not very large.

The disadvantage of the HEED is that it consumes much energy in executing its algorithm in all sensor nodes. Our topology control method is centralized, in which the core algorithm is executed in the base station, so sensor nodes will spend less energy in topology control. In comparison with the HEED, our proposed topology control method performs better on extending the lifetime of the WSN.

Acknowledgements

References

[1] L. M. Kirousis, E. Kranakis, D. Krizanc and A. Pelc, "Power consumption in packet radio networks", Theoretical Computer Science, vol. 243, no. 1-2, **(2000)**, pp. 289−305.

[2] S. Narayanaswamy, V. Kawadia, R. S. Sreenivas and P. R. Kumar, "Power control in ad-hoc networks: Theory, architecture, algorithm and implementation of the COMPOW protocol", Proceedings of the European Wireless Conf. Florence, **(2002)**, pp. 156−162.

[3] W. R. Heinzelman, A. P. Chandrakasan and H. Balakrishnan, "Energy-Efficient communication protocol for wireless microsensor networks", Nunamaker J, Sprague R, eds. Proc. of the Hawaaian Int'l Conf. on System Science (HICSS), Washington: IEEE Press, **(2000)**, pp**.** 3005−3014.

[4] F. Ye, G. Zhong, S. Lu and L. Zhang, "PEAS: A Robust Energy Conserving Protocol for Long-lived Sensor Networks", International Conferenece on Distributed Computing Systems (ICDCS), **(2003)**.

[5] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks", IEEE Transactions on Mobile Computing, **(2004)**, pp. 366–379.

[6] G. Karypis, R. Aggarwal, V. Kumar and S. Shekhar, "Multilevel hypergraph partitioning: Application in vlsi domain", IEEE Transactions on VLSI Systems, 1998 (to appear), A short version appears in the proceedings of DAC 1997.

Authors

Heng Shao, he received the B.S. degree in software engineering from Wuhan University, China in 2014. He is currently studying for master in software engineering, Wuhan University, China. His research interests include wireless sensor networks and hypergraph theory.
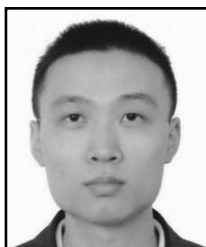

Ruoshan Kong, he received the B.S. and Ph.D. degrees in computer science from Wuhan University, China in 2002 and 2007, respectively. He is currently an associate professor in the International School of Software, Wuhan University, China. His research interests include mobile communication protocols,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

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 processor 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.

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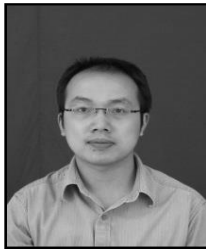
Huaqing Mao, he holds his PhD from Wuhan University, China, in 2010. He then joined the Information Security Laboratory at the Wenzhou University, China, and his main research interest is the tall building interior emergency response and evacuation. More recently, he has been focusing on the model construction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