Sensors & Transducers, Vol. 170, Issue 5, May 2014, pp. 101-106

Sensors & Transducers

An Improved Energy-balance AODVjr Routing Algorithm

[1] Ren Li-Li, [2] Kang Bing, [1] Wang Chu-Hang
[1] Center of Monitoring, Changchun Normal University, Changchun 130032, PRC
[2] The Communication Engineering College, Jilin University, Changchun 130022, PRC
[1] Tel.: 15804305288, fax: 0431-8616386
E-mail:444251129@qq.com

Received: 20 March 2014 /Accepted: 30 April 2014 /Published: 31 May 2014

Abstract: In order to improve the performance and prolong the lifetime of the networks, this paper presents an improved algorithm called Energy-balance AODVjr Routing Algorithm (EARA) which can balance the network energy consumption by improving the routes of data transmission. During the stage of constructing the network, a table called neighbor table is built for every node to store the information of its neighbor nodes. Then, considered the residual energy of the nodes along the route as well as the rules in AODVjr, a route between the source and the destination nodes is established in order to balance the energy consumption and make the hops as few as possible. Both simulation and prototype tests results show that EARA can extend the lifetime of the network more effectively than AODVjr. *Copyright © 2014 IFSA Publishing, S. L.*

Keywords: ZigBee, Energy balance, Routing algorithm, Energy consumption, Lifetime.

1. Introduction

ZigBee has been the mainstream in warless sensor networks because of its characteristics such as low power, low cost, short delay, high capacity, high security and free of license. With the development of ZigBee technology, it will be widely used in the fields such as smart house, industry, medical equipment and oil exploiting [1].

The most widely adopted routing algorithms in ZigBee networks are Cluster-Tree algorithm [2] and AODVj (AODV Junior) algorithm [3], in which the former is a network protocol that is suitable for static or less dynamic condition and the latter uses an improved Ad hoc On-demand Distance Vector that selects the route containing the fewest jumps from resource to destination through comparing one route with the others [4]. However using these algorithms, the lifetime of the network is short because the algorithms ignore the energy consumption of the node when selecting the route to send the data. Most improved algorithms just focus on the network protocol of transmission of data [5] or the structure of network topology [6-8]. Even though there are some algorithms such as ZBR and F-ZBR [9] that consider energy of the network, they still lack full consideration of the management of energy of a single node and energy balance of the whole network.

In order to prolong the lifetime of ZigBee networks, this paper proposes an improved energy-balance AODVjr routing algorithm (EARA) based on improved ZigBee algorithm and AODVjr so as to improve the management of the single node and selecting route to send data.

2. Related Works

The improved algorithm proposed in this paper is based on the well-known AODVjr algorithm, and it related to Cluster-Tree algorithm in which address assignment mechanism of ZigBee network is adopted. Then, analysis of AODVjr and Cluster-tree algorithm is given as follow.

2.1. Address Assignment Mechanism of ZigBee

Every node in the ZigBee network has a short address of 16 bits, which is used to identify the node when ZigBee network transfers signal or packet [10]. In this mechanism, the address of the coordinator is 0 and its depth is also 0. If the address of one father is P_a and the address of its k^{th} child P_k is described as formula (1), where R_{max} is defined as the most routing nodes one child node has and $S_k(l)$ is the address offset between the father node with depth l and its child node.

$$\begin{gathered} P_k = P_a + S_k(l) \times (k-1) + 1 \\ 1 \le k \le R_{max} \end{gathered} \quad (1)$$

$S_k(l)$ can be obtained from the formula (2):

$$S_k(l) = \begin{cases} 1 + C_{max}(D_{max} - l - 1), R_{max} = 1 \\ \dfrac{1 + C_{max} - R_{max} - C_{max} R_{max}^{D_{max} - l - 1}}{1 - R_{max}}, R_{max} \ne 1 \end{cases} \quad (2)$$

In formula (2), C_{max} denotes the child nodes one father node can connect to and D_{max} is the maximum depth of the network that the formula to get the address of end device node is different from that of the route node, the address of the eth end node of the father node P_n is described as follow:

$$\begin{gathered} P_n = P_a + S_k(l) \times (k-1) + n \\ 1 \le k \le C_{max} - R_{max} \end{gathered} \quad (3)$$

2.2. Cluster-tree Routing Algorithm

Each node has only one father node, but for one node it can have several child nodes. By using this mechanism we can get a Cluster-Tree network. There is no process for searching a route since the address of the whole network is distributed in order, that is, there are principles in address distribution. If we know the address of one node, we can calculate the address of its father node and all addresses of its upper nodes. Cluster-Tree routing algorithm makes use of this mechanism to calculate the address of upper node of the destination node [2].

2.3. AODVjr Algorithm

The AODVjr is an improved AODV algorithm. It is not allowed to generate a Route Reply (RREP) among all intermediate nodes whether they have a current route to the destination or not, and only the destination gives back a RREP signal to set a backward route which can reduce the energy consumption.

Fig. 1 is the comparison of AODVjr algorithm and AODV algorithms. The packet will be sent from node 1 to node 5, and we image that node 3 has a current route to node 5. In the AODV algorithm, both node 2 and node 3 will generate a RREP to node 1. However, only node 5 will generate RREP in AODVjr algorithm. From the above description, we can know that the source in the AODVjr algorithm deals with fewer RREPs than that of AODV, so the AODVjr algorithm simplifies the process of routing discovery and reduces the consumption to some degree.

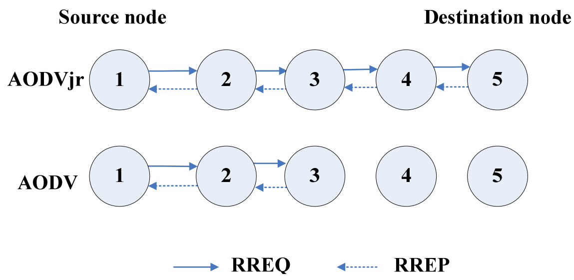

Fig. 1. Comparison of route discovery between AODVjr and AODV.

Because the AODVjr algorithm uses the route that contains the fewest jumps from the source to the destination, so there are routes more frequently used than the others, which leads to the result that the energy consumption of some nodes in those frequently used route is too high. High energy consumption will accelerate the death of the node and make some routes be failure, as well as reduce the lifetime of the network in the end.

Therefore, this paper presents an improved algorithm to solve the problem described above that the consumption of the single node is too high and the lifetime of the network is short, its main objective is to prolong the lifetime of the network by managing the energy of every node and the whole energy of the network.

3. Design of EARA

EARA can improve the method to find route sending packets. Neighbor table is built in the beginning to evaluate the initial energy of the node and the warning energy threshold which will be changing with the average energy of the network. One route (from A to B) is restored for the nth time, and the route is not refreshed during nth - $(n + k)^{th}$ time, where k is an integrity and k>0. If for the $(n + k + 1)^{th}$ time the source is A and the destination is B, the stored route will be used. Moreover, the network will check the energy before the source sends its packet. And if there is a low energy node, it will drop this existed route and discover a new route. Through this way the network can balance its energy and prolong its lifetime.

The network built by EARA is still a cluster-tree network, but the EARA network will discover the neighbor nodes after the network has informed. It defines all the nodes in one jump except its father node as its neighbor node and the removing the father node will save the storage space for the node. In Fig. 2, the node f has neighbor nodes: node c, node 2, node I and node e, in which node c, node I and node e are FFDs (Full function device) and node 2 is RFD (Refined function device). Node b is not a neighbor node of node f since the former is the father node of the latter. On the condition that the storage of RFD is limited, it will not build a neighbor table and its role in EARA algorithm is to receive and send packet.

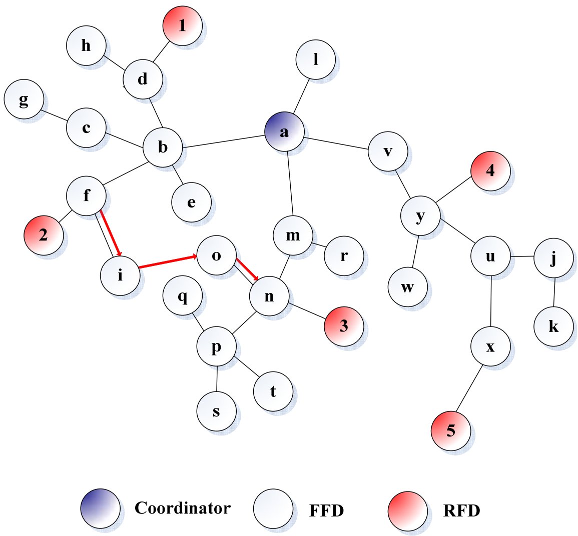

Fig. 2. The topology of the network.

The goal of neighbor node discovery is to get all neighbor nodes, their information and to store them. In this algorithm the information stored in the neighbor table in Fig. 3 consists of two parts: one is the address of the neighbor node (Address), the other is the energy of the remainder energy of the neighbor node. We must know that the address is managed by the coordinator. The remainder energy is got by AD (Analog convert to Digital) in the physical platform experiment while is got by evaluating the times one node transmits packet. The initial energy is C_0 ideally.

Naddress	Cenergy

Fig. 3. Information of a neighbor table.

The network gets the value of the warning energy threshold $C_{warning}$ as soon as the neighbor table is built. The value of the warning energy is a relate value from the formula (4), which is defined as follows: the value of warning energy is half the average energy (C_{ave}) and the first value comes out before the building of neighbor table--that is, there is no packet transmission before the first vale of $C_{warning}$ comes out.

$$C_{warning} = C_{ave} / 2 \tag{4}$$

In formula (4), $C_{warning}$ is not a static value, it is the value of the nth energy warning. Therefore, the first value of $C_{warning}$= (1/2) C_0.

After finishing the transmission, Node whose energy is lower than $C_{warning}$ is defined as a temporarily dead node. The number of dead nodes is defined as A_1, the whole number of nodes A_0 and the rate of temporarily dead nodes P_t, there is:

$$P_t = \left(A_1 / A_0\right) * 100\% \tag{5}$$

If there is a new temporarily dead node, P_t will be updated by formula (5). If P_t > 20 %, both the C_{ave} and the $C_{warning}$ will be updated. In fact, C_{ave} is not the real average energy of the network, it is the value that all the energy of nodes whose remainder energy is less than $C_{warning}$ pluses the value of $C_{warning}$ * (A_0-A_1), then divided by A_0:

$$C_{ave} = \frac{\sum_{i=1}^{A_1} C_{energy}(i) + C_{warning} \bullet (A_0 - A_1)}{A_0} \tag{6}$$

Thus the value of $C_{warning}$ can be updated constantly with the continuous calculation of the network.

Actually, $C_{warning}$ is continuously decreasing. The temporary node whose energy is bigger than $C_{warning}$ can be restored to normal node which is utilized to transmit data. Generally, almost all of the nodes can become normal nodes by C_{ave} updating.

When $C_{warning}$ is updated to an order of magnitude, the nodes will be defined as dead nodes which meet $C_{energy} < C_{warning}$. N_{die} denotes the number of dead nodes. The life of the network is determined by the number of dead nodes and corresponding time. Compared with other networks, a network has a long life if its dead node appears late and after that it has less dead nodes at a certain point.

The flow chart of EARA algorithm is shown in Fig. 4. The source node R produces a signal and it is transmitted to the destination node D. At the beginning of the algorithm, it is processed as an ordinary node. The node has the ability to determine whether it is a FFD or RFD. If RFD, the node will detect the energy of its father node and transmit the detection signal to it, if the energy of its father node is greater than $C_{warning}$. Thus, the father node will become a new node which can receive detection signal. And if the energy of this father node is less than $C_{warning}$, it will be processed in subsequently.

Then, the algorithm judges whether the node has last stored path to the destination node, and sends a detection signal to detect whether the path is still available. This includes two aspects: whether the path still remains unimpeded and the energy of each node meets $C_{energy} > C_{warning}$. If the path is available, build path and transmit data, and process it as a node without the path to the destination node. If there doesn't have the path to destination node, the node will judge whether the destination node is its neighbor node, and if it is, the node will judge itself whether it is the source node, and send data to destination node directly. If not, it sends a TTEP group to the source node build path and transmit data. If the destination node is not the neighbor node, the node will query its neighbor table as well as the destination node.

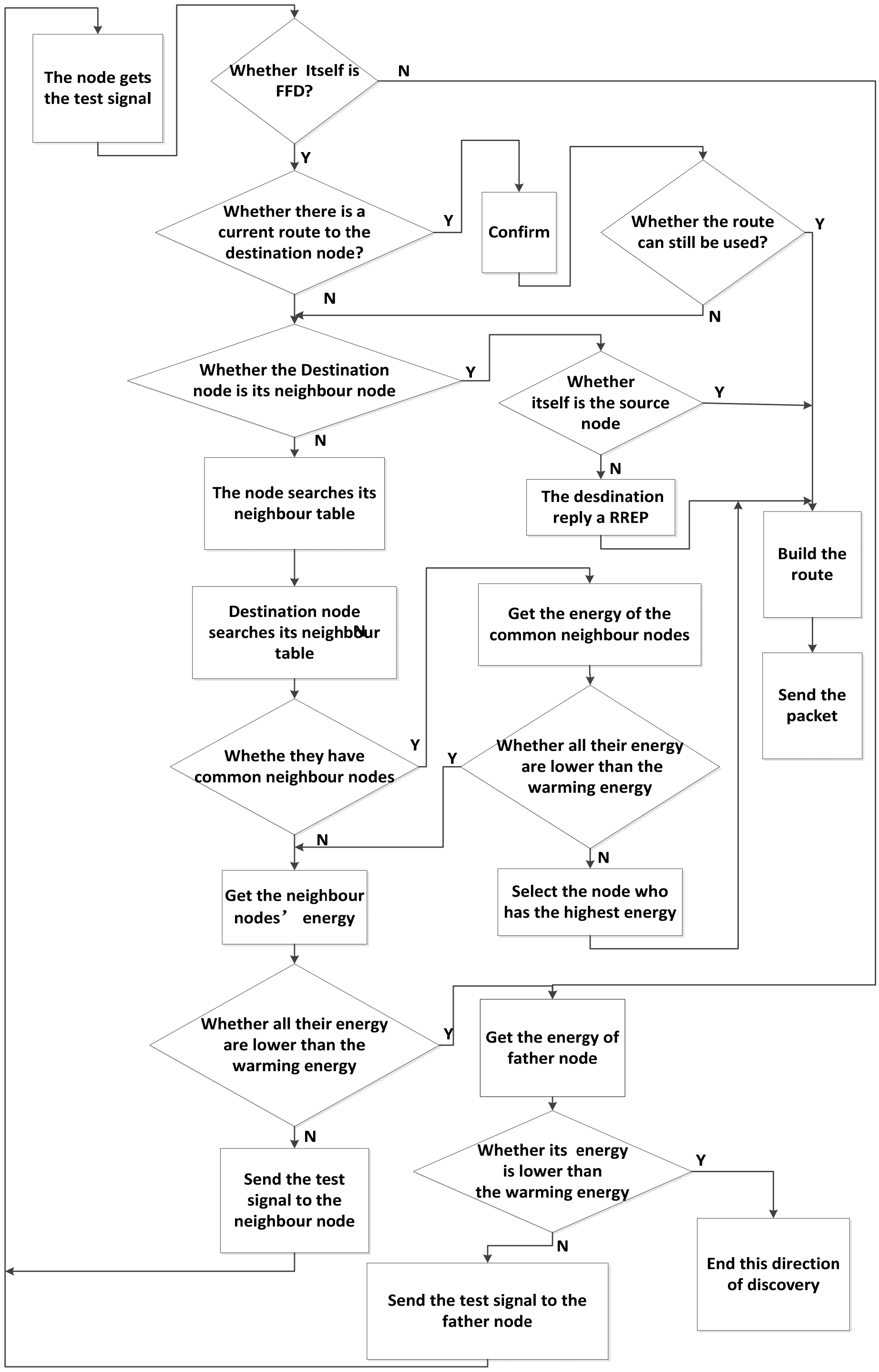

Fig. 4. Flow chart of EARA algorithm.

Next, the algorithm judges the node and the destination node whether they have the common neighbor node or not, and the node will detect the energy of all the common neighbor nodes and judge whether they are all greater than the warning value $C_{warning}$, if they have, select one common neighbor node which has the highest energy among which meet $C_{energy} > C_{warning}$ as one node of this path. Otherwise, if the energy of all the common neighbor nodes is lower than $C_{warning}$, the algorithm processes this situation in accordance with no common node between this node and destination node.

If the node has no common nodes with the destination node, it will detect the energy of all the neighbor nodes and send the detection signal to those whose energy is greater than $C_{warning}$. Thus the algorithm is recycled to the starting condition that all the neighbor nodes whose energy is greater than $C_{warning}$ have received the detection signal. After of that, each of them will conduct next detection until it can establish a path leading to the destination node. If the energy of the neighbor nodes is all lower than $C_{warning}$, this node will detect the energy of its father node because its father node does not belong to neighbor node according to the provision. If the energy of its father node is greater than $C_{warning}$, this node will send detection signal to its father node which will be recycled to be a new node that can receive detection signal, otherwise it will end the detection in this direction.

In the above procedure, if the node which produces data packet is RFD and the energy of its father node is lower than $C_{warning}$, the data will be sent to its father node directly, then to its father's father, until to the father node which is FFD and whose energy is greater than $C_{warning}$. Then the algorithm conducts the detection as the way of ordinary node in above.

4. Experiment Tests

In this section, experiments are carried out to test the performance of EARA comparison with AODVjr.

The experimental platform including 15 ZigBee nodes is used to test the performance of EARA, the length of data packet of every signal is increased to 100 bytes. In order to test the performance of the network quickly, the data packet transmission time is increased. Considering the relatively small scale of the network and the advent of RFD, which will reduce the possible propagation path so as to lead to the emergency of single path, all the nodes utilized in this time are FFDs. The network parameters are acquired by controlling several indicators. The comparison curves of network lifetime are shown in Fig. 5.

In the physical test period, the first dead node utilizing EARA to transmit data appears later than AODVjr. So EARA delays the emergency of the first dead node, which coincides with the simulation results.

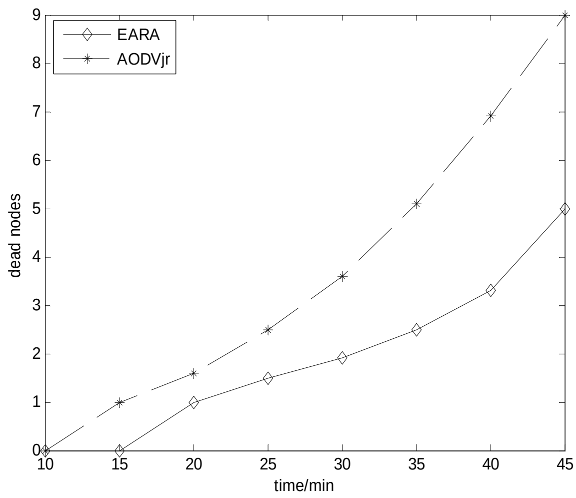

Fig. 5. Comparison of the number of dead nodes.

After the advent of the second dead node, the dead node appearance speed of AODVjr becomes faster, while the speed of EARA is slightly flat. So EARA has the ability to balance the energy of network and extend the network lifetime.

Next, Fig. 6 shows the comparison curves of the success rate of packet transmission, which are almost the same in early period. However, with the increase of the time, the rate of AODVjr declines sharply while the rate of EARA declines slowly. The reason is that EARA is aimed at balancing the energy of network. If energy of some nodes decline too quickly, their frequency of forwarding data will be reduced to slow down the rapid decline of the energy. So, the entire network will be running smoothly even if the advent of accidental death of some individual nodes.

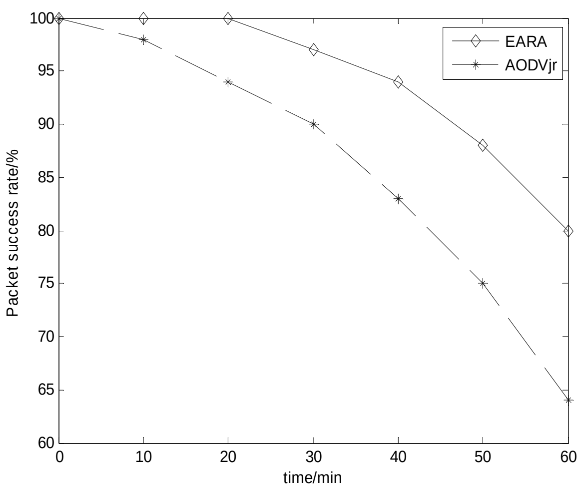

Fig. 6. Comparison of packet transmission success rate.

At last, the end-to-end delays of the two algorithms are performed, and the result is shown in Fig. 7. With the large quantity of data transmission and artificially increasing energy consumption, number of dead nodes increases. At the late period of network, EARA exemplifies its superiority in end-to-end delay, which is less than the delay of AODVjr. It can be seen from Fig. 7 that the end delay grows obviously because of the increasing number of dead nodes and the path changes. However, the end delay of EARA grows slowly. EARA reduces the end delay and improves the rapidity of ZigBee signal transmission on the basis of AODVjr.

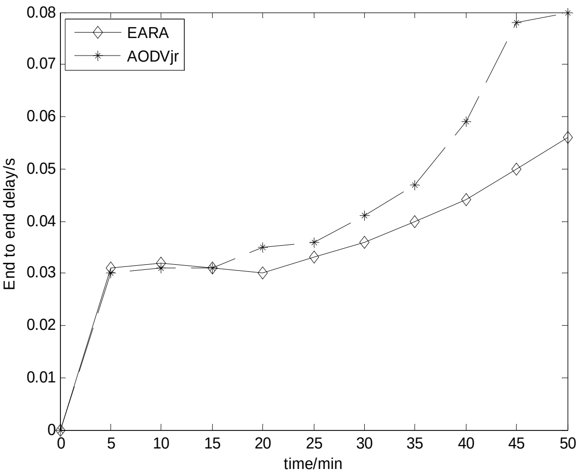

Fig. 7. Comparison of end to end delay.

5. Conclusions

EARA utilizes the method of using path dynamically to solve the energy balancing problem. When a path has nodes with too low energy, the algorithm will detect a new path to transmit data and the hops of the two paths have little difference. Furthermore, if the paths have too much difference, the Cluster-Tree algorithm will be utilized and the data will be transmitted by tree network directly to achieve the goal of balancing network energy.

Taking into account of the cost of the building large scale network, simulation and experiment are utilized to analyze the performance of AODVjr and EARA, the results show that compared with AODVjr, EARA can delay the emergency of the first dead node, slow down the speed of node death to extend the lifetime of the ZigBee wireless network, maintain the end-to-end delay, as well as improve the data transmission successful rate greatly, so as to balance the network energy and increase the stability of the network.

Acknowledgements

This article is supported by Jilin Province Science & Technology Department Youth Research Fund 2013 (20130522163JH), Jilin Province Department of Education 12th Five Year Science & Technology Plan (No. 411 [2012]), Changchun Normal University Natural Science Fundation (No. 011 [2011]).

References

[1]. Baker N., ZigBee and Bluetooth strengths and weaknesses for industrial applications, *Computing & Control Engineering Journal*, 16, 2, 2005, pp. 20-25.

[2]. Kim T., Kim D., Park N., Shortcut Tree Routing in ZigBee Network, [Online], Available: http://reslicuac.kr/~damrano/proc/iswpc2007_1 (10 December 2012)

[3]. Chakeres I. D., Klein Berndt, AODVjr AODV Simplified, *Mobile Computing and Communication Review*, 6, 3, 2002, pp. 100-101.

[4]. ZigBee specification, ZigBee Alliance for ZigBee specification 2008, 2008.

[5]. L. L. Ma, H. Li, J. Z. Wang, The research of audio and data simultaneous transmission technology based on ZigBee, *Beijing Ligong Daxue Xuebao (Transaction of Beijing Institute of Technology)*, 31, 3, 2011, pp. 299-302.

[6]. Y. Zhang, L. J. Chen, H. Y. Wei, An improved routing strategy for dynamic ZigBee network, *Advanced Materials Research*, 403, 48, 2012, pp. 2556-2559.

[7]. F. H. Zhang, H. L. Zhou, X. G. Zhou, A Routing Algorithm for ZigBee Network Based on Dynamic Energy Consumption Decisive Path, in *Proceedings of the IEEE International Conference on Computational Intelligence and Natural Computing*, Wuhan, China, 2009, pp. 429-432.

[8.]. J. S. Mu, K. H. Liu, W. G. Shi, Routing selection strategy based on node mobility in ZigBee network, *Journal of Tianjin University Science and Technology* , 45, 4, 2012, pp. 301-308.

[9]. Z. H. Qian, X. F. Zhang, Y. J. Wang, T. Y. Guan, Node characteristics based LR-WPAN routing algorithm research for network energy optimization, *Tongxin Xuebao (Journal on Communications)*, 31, 10, 2010, pp. 238-243.

[10]. ZigBee Specification, ZigBee Alliance for ZigBee Specification 2006, 2006.

___