VAASAN AMMATTIKORKEAKOULU
VASA YRKESHÖGSKOLA
UNIVERSITY OF APPLIED SCIENCES

Qiang Wang

BREADTH-FIRST SEARCH FOR ZIGBEE TOPOLOGY

Technology and Communication

2014

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Information Technology

ABSTRACT

Author	Qiang Wang
Title	Breadth-First Search for Zigbee Topology
Year	2014
Language	English
Pages	40 + 4 References
Name of Supervisor	Chao Gao

In order to improve the performance of Zigbee tree topology, Breadth-First Search has been proposed in this paper. Zigbee is a wireless standard for Ad hoc networks based on the IEEE 802.15.4.It was originally designed for low data rate, low energy consumption and low cost Wireless Personal Data Networks (WPANs). Because of these characteristics, Zigbee is expanded to be used in Wireless Sensor Networks (WSNs) for industrial use. Because of the structure and characteristics of the Zigbee network, data packet might travel through unnecessary paths until it reaches the destination, which will consume extra energy.

In order to optimize the transmission route, the Breadth-First Search has been proposed for the Zigbee tree formation in this thesis. The implementation of Breadth First Search begins at the coordinator node and connecting the nodes, which are in transmission range. Then for each of the children nodes, respectively, it connects their children nodes, which have not been connected. The method used in this paper was using Breadth First Search to establish the network, which greatly reduces the network depth. Due to the reducing depth, the hop count compared to the original tree routing was also reduced.

This thesis also compares the orphan node problem of the original tree routing and Breadth-First Search tree routing. The result is that the number of orphan nodes in Breadth First Search was significantly decreased.

Keywords: Zigbee, Breadth First Search

ABBREVIATIONS

ART	Adaptive Robust Tree
BoNM	Based on the Node Movement
CRC	Checksum
CSMA-CA	Carrier Sense Multiple Access Collision Avoidance
DAAM	Distributed Address Assignment Mechanism
DFG-TR	Destination Family Group Tree Routing
EAMTR	Energy Aware Multi-Tree Routing
EHRA	Efficient Hybrid Routing Algorithm
ERD	Enable Route Discovery
ESTR	Energy-efficient Shortcut Tree Routing
FFD	Full Function Devices
FRD	Force Route Discovery
FSC	Frame Checksum
GMs	Group Members
IEEE	Institute of Electrical and Electronics Engineers
LQI	Link Quality Indication
MRT	Multicast Routing Table
NTR	Neighbor Tree Routing
RAAM	Random Address Assignment Mechanism
RFD	Reduced Function Devices

RREQ	Broadcast Routing Request Packet
SRD	Suppress Route Discovery
STRcost	Neighbor’s hop count to the destination based on the neighbor table
TR	Tree Routing
TRcost	Hop count of Zigbee tree routing
WPANs	Wireless Personal Data Networks
WSNs	Wireless Sensor Networks
ZT	Zigbee Tree

CONTENTS

1 INTRODUCTION

The aim of this thesis is to do a thorough investigation on the Zigbee protocol and make a method to improve the performance of the Zigbee protocol. Zigbee is a highly reliable wireless connection; Zigbee it uses Carrier Sense Multiple Access Collision Avoidance (CSMA-CA) to increase the reliability. Before transmitting, a node will listen to the channel. If it is clear the node begins to transmit. This will avoid the signal overlapping problem (corrupted data). Zigbee uses a 16-bit CRC on each packet called a Frame Checksum (FSC), which will guarantee that the received data is correct. The Zigbee network can be formed as star, mesh and tree topology and of these three topologies the tree topology is most widely used because of its reliability of data routing in the network. But the disadvantage of tree routing is also obvious. In the Zigbee routing, the router does not store the routing table, so in this case more hops will be used to reach the destination and more energy will be used, which will decrease the lifetime of the network. Furthermore the node in a higher depth will transmit more data than the node in a lower depth. Due to the characteristic of tree topology and its drawbacks the easiest way to improve the performance is to reduce the hop between the source and destination. There are researches made that mainly focus on the reducing hop count. These methods do greatly reduce the hop count between the source and the destination and improve the performance of the Zigbee protocol but none of these methods considered reducing the hop count while the tree is formatting. Some of the methods require memory to store the neighbor table; some of them need to send extra data which will consume unexpected energy. Other researches focus on reducing the energy consumption of the network and prolonging the network lifetime. There is also literature that mainly focuses on the Zigbee tree topology formatting problems. In this thesis the starting point was that by improving the tree formation the Zigbee performance could be improved. This thesis proposed Breadth-First Search method to format the tree. The result shows that the improvement is obvious. By reducing the depth of the network it can reduce the hop count between two nodes and it can cut down the number of the orphan nodes.

This paper is organized as follows: Chapter 2 is a detailed description about the Zigbee network. The improved methods in fixed node and mobile node condition are described in Chapter 3. In Chapter 4, the Breadth First Search method is presented; the result and the analysis also included in this chapter. Finally Chapter 5 summarizes the work.

2 ZIGBEE NETWORK DESCRIPTIONS

A Zigbee network consists of three types of node: the coordinator node, router and terminal equipment. The coordinator node is the root that identifies the whole network. The router has the ability to execute routing algorithms and forward messages to and from the other devices. The terminal equipment known as end device, has limited resources, it does not allow association and does not participate in routing /6/. Meanwhile, these nodes can be divided into full function devices (FFD) and reduced function devices (RFD). The FFD implements the full IEEE 802.15.4/Zigbee protocol track and it can communicate with both FFDs and RFDs. The RFD implements a subset of the protocol stack and it can only communicate with FFDs. So as a consequence the coordinator and router must be a FFD but terminal equipment can be not only a FFD node but also a RFD.

2.1 Network Address Assignment

Zigbee has two address assignment mechanisms: distributed address assignment mechanism (DAAM) /29/ and random address assignment mechanism (RAAM). A tree routing algorithm is put forward on the basis of distributed address assignment mechanism, which decides the next hop according to the computation of local nodes and destination nodes' network addresses /26/.

In DAAM, when a network is constructed, it needs three parameters: the maximum number of children of a router (C_m), the maximum number of child routers of a router (R_m), and the depth of the network (L_m). With the three parameters, a parent router can also determine the unique network addresses of its child devices. Each router of depth d calculates it's $C_{skip(d)}$ to determine the number of network addresses allocated to each router-capable child device. The function $C_{skip(d)}$ is calculated as follows:

$$\mathrm{Cskip(d)} = \begin{cases} 1 + \mathrm{C_m}(\mathrm{L_m} - \mathrm{d} - 1) & \text{if } \mathrm{R_m} = 1 \\ \dfrac{1 + \mathrm{C_m} - \mathrm{R_m} - \mathrm{C_m} * \mathrm{R_m}^{\mathrm{L_m} - \mathrm{d} - 1}}{1 - \mathrm{R_m}} & \text{other} \end{cases} \tag{1}$$

When the parent node's $C_{skip(d)}$ =0, this means there will be no more child node joined in the network. The addresses to router-capable child node are:

$$A_k = A_{parent} + \mathrm{Cskip(d)} * (\mathrm{k}-1) + 1 \ (1 \le \mathrm{k} \le \mathrm{R_m}). \quad (2)$$

And the equation for the addresses to end devices in a sequential manner with the nth address is:

$$A_n = A_{parent} + C_{skip(d)} * \mathrm{R_m} + \mathrm{n} \ (1 \le \mathrm{n} \le \mathrm{C_m} - \mathrm{R_m}) \quad (3)$$

For a Zigbee router with address *A* at depth *d*, if $A < D < A + C_{skip(d-1)}$ is true, then a destination device with address *D* is a descendant. Thus, the address *N* of the next hop device is given by:

$$N = \begin{cases} D, & \mathit{if\ end\ device} \\ A + 1 + \left\lfloor \frac{D-(A+1)}{\mathrm{Cskip(d)}} \right\rfloor * C_{skip(d)}, & \text{otherwise} \end{cases} \quad (4)$$

In general, the device sends the data to one of destination children if it is a descendant; otherwise, it sends data to its parent in the tree routing.

Figure 1 shows a case in the tree routing. The source node will follow Equation (4) to determine which the next hop device of current packet is. The next hop device could be a descendant of this router if $A < D < A + C_{skip(d-1)}$

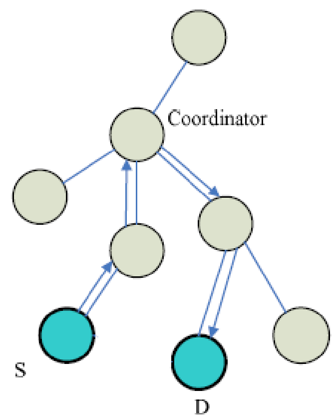

Fig.1 Tree routing in ZigBee

2.2 Neighbor Table

Each FFD node in the Zigbee network has a neighbor table which contains all neighbors' information in the single hop transmission range. It consists of network identifier; network address and the extended address; device type and the relationship with the node; network depths; link quality indication (LQI) and so on /22/.

2.3 Routing Protocols

Zigbee defines two routing protocols: tree routing protocol and AODVjr routing protocol. In the tree routing protocol it uses DAAM. The tree routing (TR) can find the next hop node without routing tables/24/ and AODVjr /7/ can find the optimal route to destinations through broadcast routing request packet (RREQ).

2.4 Node Movement in Zigbee Network

There are three coordinator movement strategies: random, predictable and controlled./21/

Random mobility: the coordinator randomly selects the length and direction of segments in its path.

Predictable Mobility: The predictable or fixed trajectory of a mobile coordinator is fully deterministic as the coordinator always follows the same path throughout the network.

Controlled Mobility: The path of the coordinator becomes a function in the current state of network flows and nodes' energy consumption, and it keeps adjusting itself to ensure optimal network performance at all times.

According to the standard in the network layer, the DiscoverRoute field of the data frame head is defined as the routing approach for data frames and they have three values: Suppress route discovery (SRD), Enable route discovery (ERD) and Force route discovery (FRD). In SRD the network only uses the routing tables that exist already. In ERD if there is the routing address in the routing table, the routing will follow this routing table. Otherwise, the router will initiate the routing discovery. In FRD the node has to initiate the routing discovery constrainedly no matter whether there is the corresponding routing table or not. /8/

3 IMPROVEMENT METHODS

In this chapter the improvement methods by various researchers to improve the performance of Zigbee tree topology are introduced.

3.1 Fixed Node Improvement Methods

There are two types of improving methods. The first is to reduce the routing cost by using a neighbor table. Another type is to focus on balancing the energy consumption of each node in order to extend the network lifetime.

3.1.1 Reduce Routing Cost (Hop Count)

Due to the characteristic of Zigbee tree routing, the transmitting packet may be passed through several hops towards the destination node, even if the destination node is located nearby. In this case, the neighbor table is widely used to reduce the hop count.

Kim *et al*, proposed the shortcut tree routing protocol to reduce the hop count by using neighbor table. In general, the theory of shortcut tree routing is to check the neighbor table while transmitting the packet. First the source node will calculate the hop count of Zigbee tree routing (TRcost) and calculate each neighbor’s hop count to the destination based on the neighbor table (STRcost). Then the smallest value of STRcost is chosen to compare the value of TRcost and pass the packet to the smallest value´s node. /15/

Sheng and Honglian /22/ suggest the following methodology: first, it is determined whether or not the destination node is a neighbor node of the source node by checking neighbor table. If it is true, it forwards the data to the neighbor node. Otherwise the source node will check its neighbor list. By checking the neighbor list, it firstly calculates total hops of each neighbor to get L_{kd} and also the residual energy ratio *E%* using $C_{skip(d_k-1)}$. d_k is the network depth of the *k-th* neighbor. After that, it gets the simplified evaluation function value of each neighbor using

$$f\,(k_i) = \frac{E_\Delta}{E_r} * E\% \tag{5}$$

where E_Δ is the saving energy that energy consumption forwarding along the original routing tree and minus it forwarding through the optimum node. The formula of E_Δ is

$$E_\Delta = (R_x * t_R + T_x * t_R) * (L_T - L_{kd}) \tag{6}$$

where L_T is the link hops forwarding packets along the tree, R_x is the receiving power of the node, T_x is the transmitting power, L_{kd} refers to hops to the destination node, t_R and t_T represent receiving and sending cycle respectively. E_r is the probable energy consumption that the neighbor node may transmit data for its children node. The formula of E_r is

$$RC = (d_d - d_{nd}) + (d_n - d_{nd}),\ d_d \tag{7}$$

Where λ is a parameter that obeys the Poisson distribution /30/; N is the total number of descendant nodes.

And in a specific Zigbee network if α is given N/λ is a constant. Then the evaluation function can be simplified as follows:

$$RC = (d_d - d_{nd}) + (d_n - d_{nd}),\ d_d \tag{8}$$

After getting each neighbor´s $f\,(k_i)$, it forwards data to one of the maximum. So it can balance the entire network energy savings and the life cycle of a single node. Moreover, Sheng and Honglian also made some improvement of this method: before processing this method a judgment should be made, whether the hops between source and destination node is less than 3. It means source and destination node share the same neighbor node.

Khatiri *et al*, propose energy-efficient shortcut tree routing (ESTR) to reduce hop counts and to balance energy in the network. /13/

In this research the key formula is the total cost (TC):

$$TC = \alpha * \widetilde{RC} + \beta * \widetilde{NC} + \gamma * \widetilde{LC} \qquad (9)$$

Coefficients α, β, γ should satisfied 0≤ α, β, γ≤1, α +β +γ=1

RC is the hop count, which defines by $RC = (d_d - d_{nd}) + (d_n - d_{nd})$, d_d and d_n are the neighbors' depth and destination's depth respectively and d_{nd} is the depth of the closer common ancestor of neighbor and destination.

NC is a counter used to count the number of packets sent or received by the neighbor node.

LC is link quality indicator, which indicates the quality of the received signal.

And $\widetilde{RC}$, $\widetilde{NC}$ and $\widetilde{LC}$ are locally normalized values of RC , NC and LC respectively.

$$\widetilde{RC}=\frac{RC}{\max_i RC_i},\ \widetilde{NC}=\frac{NC}{\max_i NC_i},\ \widetilde{LC}=\frac{LC}{\max_i LC_i}$$

Here are calculated by node n for its ith neighbor.

The ESTR will choose the minimum TC as next hop from neighbor nodes. In general, if node n is the destination node, the packet will deliver it directly; no ESTR method will be used. Otherwise, the ESTR method will be applied to choose the next hop among the neighbor nodes with the minimum TC and forward the packet to it.

Bidai *et al*, propose multipath routing, where multiple paths are used simultaneously to transfer data between a source and the coordinator. First, they explain the assumptions and definitions of multipath forwarding. /1/

In order to check parent-child relationships, if two paths are node disjoint or not, Bidai *et al* introduce the Zigbee Tree Path Information (ZT). ZT has two forms:

$$ZTP_c = \begin{cases} (C_1, C_2, \dots, C_{d_c}, 0, \dots 0) \; if \; internal \; node \\ (C_1, C_2, \dots C_{L_m}) \; if \; leaf \; node \end{cases} \quad (10)$$

Where the network depth $d_c \in [1, L_m]$, and Zigbee tree path information of node N_c noted ZTP_c is an integer sequence $C_1, C_2, \dots, C_{d_c}$ that defines the parent child path in the tree from the coordinator to N_c. /1/

The value of C_k can be represented by:

$$C_k = \left\lfloor \frac{A_c - k - \sum_{i=1}^{k-1} C_{skip}(i-1)(C_i - 1)}{C_{skip}(k-1)} \right\rfloor + 1 \quad (11)$$

When $C_k = 0$, it indicates termination of the path. And A_c is the network address of N_c.

After these definitions, the forwarding path decisions should be made. In this model, it is assumed that the three paths exist. The coordinator keeps track of the source adjacent nodes from which it receives data packets. Then, it uses the parent-child path towards the source to report these nodes identities. Based on that, the source can decide to stop transmitting via an adjacent node with no available path to the source. For each packet a new field is added in the header called a flag, which can take two values 0 or 1. When flag=1, it means TR is applied on this packet, otherwise new forwarding rules are used.

Yukun *et al*, use the beacon frame to finish the establishment of 2-hop neighbor table, which not only makes the broadcasting cost less (because beacon frame only contains the network address information), but also the node could use the beacon frame to relay the data packet after 2-hop neighbor table established. In the beacon frame it adds the 1-hop neighbor network address information, except the parent and child neighbor node network address itself. /26/

The specific algorithm is as follows:

1. If the destination node is the node´s neighbor, send directly to destination node.
2. If the goal node is the descendant node, according to the tree algorithm routing to send.
3. If it does not fit the above two kinds of circumstances, first use 2-hop neighbor algorithm; the native node to destination node of the hop counts is $N1$. And according to the tree routing algorithms the native node to destination node the hop count is $N2$. So if $N1 < (N2+2)$, it means that the 2-hop neighbor algorithm´s path has less hop count than tree routing´s path. And after this, 1-hop node according the Cluster tree routing algorithm to destination node of the hop count is $N3$, if $(N3-1) < N2$, it means from now on a jump forward data to the neighbors node destination node is superior to the node from tree routing algorithm according to forward packets. So, if 1-hop and 2-hop neighbors are better than native nodes according to the tree routing algorithm to forward packet, then compare these two methods. If $(N1-1) > N2$, then the packet will send to the neighbor and let neighbor send the packet. Otherwise, it sends to 2-hop neighbor node in the table.
4. If above situation does not appear, the node will send the packet to its parent.

Liu *et al*, propose the Neighbor Tree Routing Algorithm (NTR) by using the neighbor table. The deduction of NTR is when the data reaches the parent node, it will check the node whether it is the destination node. If it is the destination node it will receive the data, if it is not, the destination node, it will check if the destination node is in the child node. If it is in the child node, it will pass the data to the child node, if the destination node is not in the children nodes, it will use neighbor table to check the destination node neighbor nodes, and compare with the parent node’s neighbor nodes. If there is a duplicate neighbor node, the data will be passed to that neighbor node. If there is no duplicate neighbor node, the data will be passed to the best path node and start the process again. For this methodology Liu Dan *et al* propose a simulation experiment. /17/

Boujelben *et al*, made some improvement for the Z-Cast multicast routing mechanism. The details of this protocol are: /2/

1. Multicast routing table (MRT): the MRT that must be created inside each Zigbee router, which stores membership table status of the children. The MRT has two fields: Multicast group address, 16 bits short address that identifies a certain group. GMs address, contains the list of the network short addresses of nodes that are members of the group along the cluster tree network.

2. Routing table update: when every node join and leave operations in the network, the routing table must be updated. When a node joins the network, the Zigbee coordinator must add the multicast address of the group to the Multicast group address field and the address to the GMs address, so that Zigbee router will know all the membership information of the child routers of its tree.

3. Routing in Zigbee coordinator: Boujelben *et al* (2013) propose to add a flag to multicast message, so that the multicast message has already been treated by the coordinator and send the multicast message to the Zigbee coordinator before sending it to the group members. When a frame is received by the Zigbee Coordinator, it analyzes the frame and checks if the destination address is a multicast or a unicast address. If it is a multicast address, the ZC will add a flag to the frame and send it to all it is directly connected child Routers. If the destination address of the frame contains a unicast address, the default cluster-tree routing will be applied.

4. Routing in Zigbee routers: When a multicast packet reaches a Zigbee Router, there are different possibilities:

If the multicast group address is not found in the MRT, then the multicast packet will be discarded.

If the multicast group address is found in the MRT, two different cases may occur:

– If the GMs address field contains only one member address of the corresponding group, the packet will be transmitted by unicast to the group member by applying the default Zigbee cluster-tree routing algorithm. The unicast here is necessary because there is only one member in the leaf.

– If the GMs address field contains two or more addresses of the corresponding group members, the packet will be transmitted to all its direct child nodes (Zigbee Routers and Zigbee End-Devices).

Boujelben *et al* (2013) propose ameliorations in two parts of the multicast routing table.

-MRT construction: When a node joins a multicast group, its parent will check if the node is the first one in the group. Then this parent node will inform its parent node that it has in its descendant members of a group. Thus, the parent nodes will only store the group address and not the group member address which can reduce the memory storage in sensor nodes.

-MRT update: If the routers have already members that belong to the same group, it will not inform the parent nodes of this information because it will be a redundancy. Thus, the number of messages is reduced.

With Liu *et al*, the procedure of the proposed method is as followed: when the packet starts to send, it checks if the target node address is satisfied with $A < D < A + C_{skip(d-1)}$, then it transmits the packet to the designated address node. Else, if the target address is the node’s neighboring node address or in the address ranges adjacent node’s offspring child node, then it sends the packet to the corresponding adjacent node. Else, it sends the packet to the parent node and checks, if the target node address is satisfied with $A < D < A + C_{skip(d-1)}$. /17/

Kasraoui *et al*. use the following principle of modified routing algorithm: The transmitting node checks if the destination is one of its descendants. If so, it sends it according to the basic hierarchical routing. If this is not the case, it sends requests to all of its one-hop neighbors of the same depth in the tree, after estimating the round-trip delay of the message and initiates the timestamp. Each neighbor receiving the message verifies even if the recipient is one of its descendants. If so, the neighbor sends an acknowledgment to the sender and takes care of routing the message. Otherwise, the neighbor drops the message. At the sending node, if the timer expires without receiving anything, the message is transmitted to the parent. /14/

Wanzhi Qiu *et al*. propose a hybrid routing; each node on the tree path tries to identify a short cut and if it exists uses it as the next hop. /23/

3.1.2 Energy Saving Methods

In /18/, the improved algorithm which can reduce the energy consumption and extend the lifetime of the whole network. The key formula for this research is:

$$E_{limit} = \frac{e^{numb}}{f(x)} (lgE_0) \frac{1}{D+1} \qquad (12)$$

E_{limit} is the energy threshold value of residual energy state of the sensor node, when node residual energy is greater than energy threshold value E_{limit}, it means the battery energy of this node is sufficient, it can transmit data. When node residual energy is smaller than energy threshold value E_{limit}, it means the battery energy of this node is not sufficient. In the process of data transmitting, this node should be avoided.

E_0 is node initial energy. D is node depth. e^{numb} is a specific coefficient (e=2.718), its role is to slow down the speed of E_{limit} decreasing.

The routing algorithm is as follows:

1. Initialize next hop node as next hop node n_k which is computed by traditional tree routing, and initialize the number of residual routing hops, every node computes current warning energy value.

2. If the residual energy of current node S is greater than warning energy E_{limit}, then still use traditional tree routing algorithm.

3. If the residual energy of current node S is smaller than warning energy E_{limit}, then switch to energy optimization algorithm and decide the destination nodes contained in neighbor list.

4. If next hop node is the destination node *D*, then send data directly.

5. Otherwise, in the neighbor list, find a node, whose value is greater than current E_{limit}, and *D*=1

6. If n_k is greater than current E_{limit}, then compute the residual routing hop number of selected nodes in (4), compare it to n_k, and choose a minimum as next hop node.

7. If n_k is greater than current E_{limit}, then compute the residual routing hop number of selected nodes in (4), choose a minimum as next hop node.

8. Attach the address of next hop node to the data packet, send data, and modify its residual energy value, send message to its neighbor node to modify the transmission information of the transmitting node in neighbor list.

Saeyoung *et al*. propose an improved algorithm of destination family group tree routing (DFG-TR). The DFG-TR algorithm requires more calculations to set fewer paths between source and destination than Zigbee routing. Thus the energy consumption is larger. /20/

So while making a neighbor node table, the depth and index information are also included. In original DFG-TR, it finds all of the families of destination, but only families that have routing ability are required. Therefore it is not necessary to calculate the process for finding a family or storing in the family array (DFG-TR1). Another way is to reduce the calculation rate and memory space by creating a family with a limited hop count which is logically far from the destination node (DFG-TR2). The third method is to consider family to be the only ancestor of destination (DFG-TR3). This method needs only the first two steps of DFG-TR so that the calculation rates and the memory resource usage are reduced.

Jianpo *et al*. propose a method to reduce the RREQ transmission. First it will set a transmission range m for RREQ by using this method: /10/

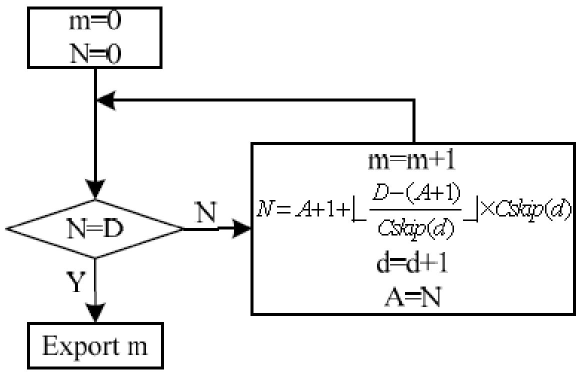

Figure 2 the flow of *m* calculation

N is the next jump address, *A* is the node address, *D* is the destination address, *d* is the network depth.

For further reducing control overhead all nodes in Zigbee network can use formula $A<D<A+C_{skip(d)}$ to judge whether the target node is its child node after receiving RREQ packet. The detailed method is to add a flag bit in RREQ. The

flag bit records the relationship of the node which should not transmit RREQ and the current node.

Fariborzi *et al*. propose energy aware multi-tree routing (EAMTR) protocol to balance the workload of nodes. In EAMTER, there are four phases: /5/

Initialization phase: creating adaptive robust tree (ART)

Tree selection phase: It is performed in a one-by-one manner; starting from the first-level nodes (hop count to coordinator = 1), all of the nodes gradually select their respective minimum cost tree as their main routing tree.

Normal phase: passing the packet.

Recovery phase: replaces its old main tree with its first alternative tree in the lowest-cost queue as obtained by the tree selection algorithm.

Zhi *et al*. propose an efficient hybrid routing algorithm (EHRA). EHRA implements after the network has used DAAM design addresses. The node broadcasts hello packet with 1 hop which contains node address. /32/

Then the nodes begin to delivery data, and the destinations are randomly set in the network.

Step 1. The source first determines if the destination is its neighbor or descendant, and if so, sends data to destination directly or to the descendant. If the destination is the ancestor or descendant of source’s neighbor, then sends data to neighbor. Otherwise, executes step 2.

Step 2. The source initiates a RREQ in multicast way; the set of hop threshold is the hops of TR-1.

Step 3. The intermediate router processes RREQ:

1. If the RREQ has been received or the number of past hops is more than threshold, then discards it. Otherwise, performs ii.

2. If the node is the ancestor or descendant of the destination, it calculates whether the experienced hops of RREQ will be beyond threshold along TR, if so, discards RREQ. Otherwise, forwards RREQ along TR.

3. If the node receives RREQ from its parent, then determines whether it and its descendants have neighbors, and if so, send RREQ to neighbor or child; otherwise, discards RREQ. If the node receives RREQ from its neighbors or children, processes the RREQ as in step2.

Step 4. The destination receives RREQ. If the number of experienced hops is less than the threshold, it responds RREP to the source along the reverse direction. The RREP contains the total depth of the intermediate node (initial value is 0), the minimum residual energy of nodes in path (initial value is residual energy of the destination), the RREQ past nodes information, etc.

Step 5. The intermediate node receives RREP, establishes the route to the destination, and updates the value of relevant field in RREP, then unicast RREP to the source.

Step 6. The source node receives RREP and establishes the route to the destination, and stores the total depth, the minimum residual energy and the hops information of the intermediate nodes.

3.1.3 Conclusion for the Methods

Consider the characteristic of Zigbee protocol, the most common way to reduce the hop count is to use the neighbor table to find an optimized route for the packet to transmit, under this condition the hop count can be reduced. However, other problem may occur, the hotspot problem, for instance. And with some of the proposed methods, they are only tested under the ideal circumstance and not in the real world, so the performance may differ from the simulation result. The disadvantage of the reducing hop count method, which only focuses on finding a shortcut, is that they have not considered the energy consumption problem. As a consequence, the packet transmission time may decrease, compared with original routing protocol, but the lifetime of the network may decrease.

For the proposed methods which focus on energy saving, the common way is to set a critical energy value for all the nodes which changes under a certain condition. The characteristic of these energy saving or energy balancing methods is that they pay more attention to the network lifetime and less for the packet transmission time.

In general, if the network requires not only less transmission time, but also longer network lifetime, a balance between reducing hop count and energy saving must be made. A balance must be made between transmission time and network lifetime.

3.2 Mobile Node Improvement Methods

In this chapter some previously proposed improvement methods for mobile node in Zigbee tree topology are introduced.

3.2.1 Previously Proposed Methods

Jiasong *et al*. propose a routing strategy based on the node movement (BoNM). In the BoNM routing strategy, all the nodes are set to SRD as initial. When a node has lost connection and failed the transmission for 3 times in total, it will automatically change the value of DiscoverRoute field into ERD. /8/

Yuan-Yao *et al*. consider the framework in a 2-D region. For router node deployment, a virtual grid that covers the whole region has been constructed. Each intersection of lines on the grid is a candidate location for a router node. The distance d between adjacent grid points is determined based on the particular scenario and application. In the framework, a router node can be placed at an arbitrary point in the region, not necessarily at a vertex on the grid. We assume that nodes are placed on the same *xy* plane. The communication range of the router node is represented by a polygon with an antenna gain profile ANTg that indicates the different gains at different angles of the antenna and a placement angle degree rant, $0^\circ \leq r_{ant} \leq 359^\circ$ that indicates the antenna direction of the router node. In particular, if a mobile sensor's nearest candidate location is *i* at time *t*, it is assumed that the sensor is in state *i* at time *t*. By counting the number of the events of sensors that move away from or toward each candidate location (state), Yuan-Yao shih *et al*. can then derive the transition probability matrix, M can be derived. The matrix is called the mobility profile, and the information that it contains is used in the router node deployment stage. After the router node deployment phase has been completed, the routing tree construction can be formulated as a graph problem, where a vertex represents an immobile node, and a directed edge represents a possible transmission link from an immobile node to another immobile node. The proposed Zigbee routing tree topology deployment and construction framework incorporates the mobility information, and algorithms are developed to implement the framework. Compared to existing approaches, this framework achieves higher data delivery ratios and longer path duration with much lower routing overhead in scenarios where the movements of mobile end devices are with regularity. /27/

3.2.2 Evaluation of the Method

The advantage of the proposed method is that BoNM was more efficient in a network, where the structure changed unpredictably, which was similar to the real applications. The disadvantage of the proposed method is that it requires a little more memory and calculation than traditional routing method.

3.3 Other Improvement Method

Yukai *et al*. present a vertex-constraint flow network which can be formulated as a directed graph $G = (V,E)$, where V represents the routers in the network and E represents the possible communication links between pairs of routers. Each vertex $u \in V$ is associated with a non-negative capacity, denoted by $\hat{c}(v) \geq 0$, which represents the GTS capacity of the router. /25/

A flow in a vertex-constraint flow network G with respect to a source s and a coordinator t is a real-value function $f: V * V \rightarrow R$ that satisfies the following three properties:

$$\sum_{u \in V} \{f(u,v) \mid f(u,v) > 0\} \leq \hat{c}(v), \forall v \in V \quad (13)$$

Skew symmetry:

$$f(u,v) = -f(u,v), \forall v, u \in V \quad (14)$$

Flow conservation:

$$\sum_{u \in V} f(u,v) = 0, \forall v \in V - \{s,t\} \quad (15)$$

Residual capacity of v:

$$\hat{c}_f(v) = \hat{c}(v) - \sum_{u \in V} \{f(u,v) \mid f(u,v) > 0\} \quad (16)$$

The residual capacity of (u,v) is

$$c_f(u,v) = c(u,v) - f(u,v) \quad (17)$$

that (u,v) is either ∞ or 0 depending on whether (u,v) is in E

The residual capacity of G induced by f is $G_f = (V, E_f)$, where E_f the residual edge set is and it is defined as follows:

$$E_f = \left\{ (u,v) \middle| \begin{array}{ll} \hat{c}_f(v) > 0, & if\ c(u,v) = \infty \\ c_f(u,v) > 0, & otherelse \end{array} \right\} \quad (18)$$

The rationale behind the algorithm is explained next. Throughout the algorithm, the height of the source s is fixed at $|V|$, and the height of the coordinator t is fixed at 0. The height of every other vertex starts at 0 and increases over time.

The height of a vertex determines the direction of force imposed on the flow, that is, a flow moves downward from a higher vertex to a lower vertex. In a PULL (u,v) operation, a lower vertex u pulls the flow of a higher vertex v downward to itself.

u pulls $\delta = \min\ (e(v), \hat{c}_f(u))$ unit of flow from v

In a PUSH (u, v) operation, a higher vertex u pushes the over pulled flow back to a lower vertex v along the edge $(v, u) \in G$.

A RELABEL (u) operation enables a vertex u to increase its height.

PULL-PUSH-RELABEL (u) is a compound operation in which a vertex u performs the three basic operations consecutively.

INIT (u) is a subroutine, whereby every vertex $u \in V$ initializes itself so as to create an initial preflow in G.

4 BREADTH -FIRST SEARCH

The problem with methods mentioned above is that none of them have noticed the tree formation problem. If the tree has a smaller network depth compared to the original Zigbee tree, then the hop count will decrease. Also the energy consumption will decrease. In this chapter, the Breadth First Search with a smaller network depth is compared to the original Zigbee tree network depth. The simulation result also shows that the orphan node problem has been solved by this method.

4.1 Definitions and Simulation

Breadth First Search begins at the coordinator node and connecting the nodes, which are in transmission range. Then for each of the children nodes, respectively, it connects their children nodes, which have not been connected. Due to this characteristic, the depth of the network can be reduced, compared to the DAAM. The simulation tool used in this research was Matlab. In the simulation the performance was tested under the ideal situation and the worst situation. In the ideal situation as many nodes as possible were connected to the network. So in order to do that, the *Cm* and *Rm* were set to be 4, the network depth to be 30. In this case, the performance between original Zigbee tree and improved Zigbee tree can be determined. In the worst situation, the purpose was to find out how the orphan node problem influences the network performance. So the *Cm* and *Rm* were set to be 2 and network depth to be 5. In this case the original Zigbee tree will suffer from orphan node problem but the improved Zigbee tree has solved the orphan node problem.

4.2 Simulation Studies and Result Analysis

The algorithm starts spanning from the coordinator by broadcasting a beacon message. Upon receiving this beacon, a router node may send an “association” message back to the coordinator. The first *Rm* associations are granted and each router gets an address and $C_{skip(d)}$ according to Equation (1) and (2). After this, each router will broadcast beacon to span their own children in the same fashion. Until the network depth limit *Lm* is reached.

Here is the algorithm for Breadth First Search mechanism which shows the tree formats:

1. First, find the coordinator neighbor table, randomly pick up *Cm* neighbor nodes and mark the chosen nodes as spanned nodes; also mark the network depth to be 1 and mark that the coordinator has children.
2. Then, find out the entire node, which has been spanned but no children, and for each of the chosen node find their neighbor nodes and mark them as spanned node; also mark the network depth plus 1 and mark that the chosen node has children.
3. Repeat step 2 until the network depth reaches *Lm* or all the nodes have been spanned and have children.

The Matlab Code is shown in follows:

```
while sum(spanned)<N
    spancandidate = find(spanned==0&parentImproved~=0);
    % Find out the node which has been spanned but no children
    if isempty(spancandidate)
        break;
    end
    while ~isempty(spancandidate)
    % Find the neighbor nodes with the same depth respectively
        curNode = spancandidate(1);
% Randomly choose one node from the spancandidate
        nd=ntable{curNode};
% Find out the chosen node's neighbor table
        unspanned=find(spanned==0&parentImproved==0);
% Find out node which has not been spanned and no parentImproved
        temp=intersect(unspanned,nd);
% Find out the intersection nodes
        if length(temp)>Rm
            randlist=temp(randperm(length(temp),Rm));
% Randomly pick up Rm intersection nodes
        else
            randlist=temp;
        end
        parentImproved(randlist)=curNode;
 % Mark the chosen node's parentImproved
        spanned(spancandidate(1))=1;
% Mark the chosen node's has been spanned
        spancandidate(1)=[];
% Remove the chosen node from the spancandidate
        depthImproved(randlist)=k;
```

```
% Mark the chosen node depthImproved
    end
    k=k+1;
    Lm=Lm-1;
    % Check the depth
    if Lm==0
        break;
    end
end
```

In Figure 3 and 4 a general idea is given about how the Zigbee tree and Breadth First Search tree looks like. The simulation area is 300meters*300meters and ratio is 50 meters. 100 nodes are randomly distributed in the simulation area. The *Cm* and *Rm* are both set to be 4. In order to get all the nodes connected and get a clear view about the difference between Zigbee tree and Breadth First Search tree the depth is limitless.

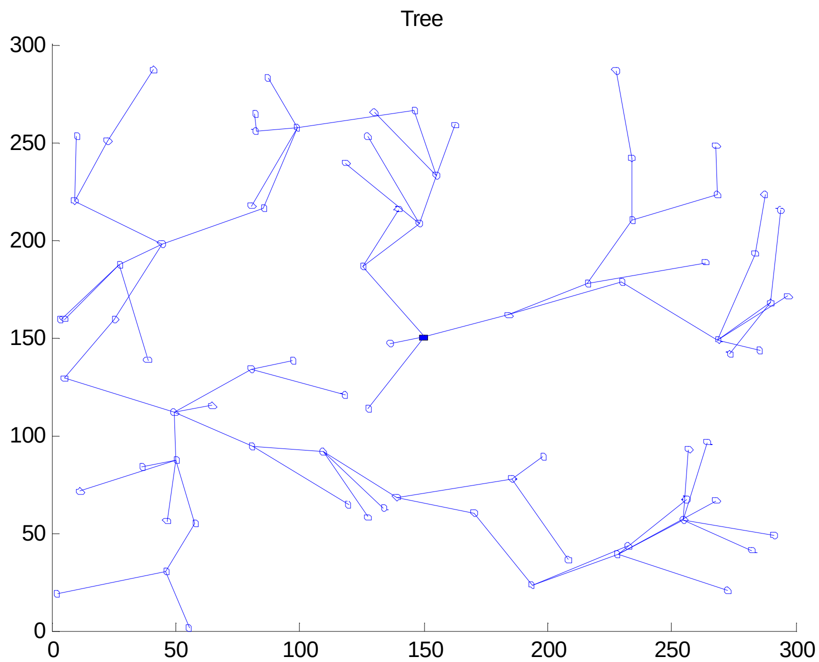

Figure 3 Zigbee Tree

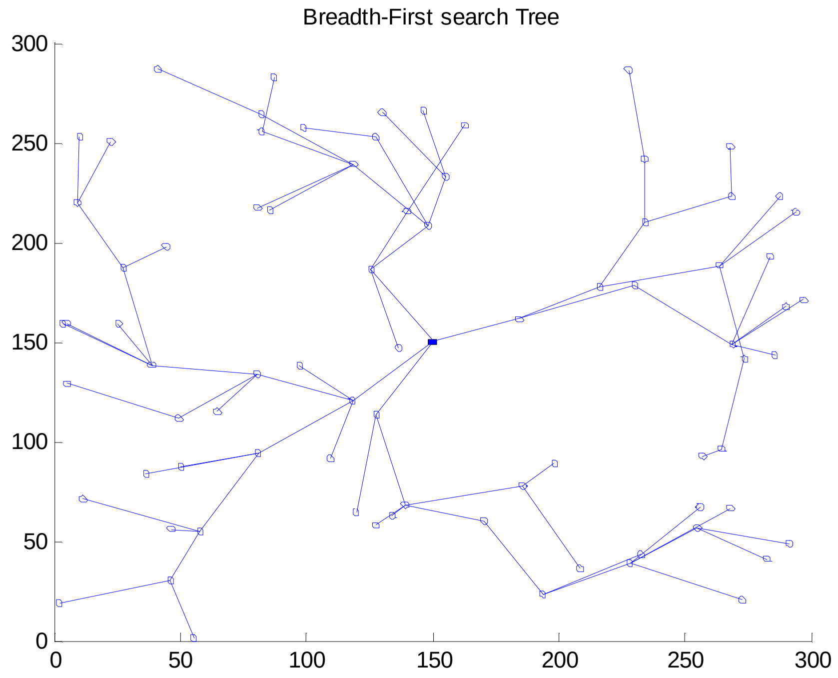

Figure 4 Breadth First Search Zigbee Tree

As it shows the network depth is much higher in Figure 3 than Figure 4. Thus the Zigbee tree hop count between two nodes will be higher than the hop count in Breadth First Search Tree. Detailed simulations followed. All the nodes were displayed randomly in 300meters*300meters area. The node number starts from 100 until 200 with 10 nodes interval. Each simulation was run for 50 times.

For the first set of the simulation, in order to get a clear view about the difference between original Zigbee tree and Breadth-First Search Zigbee tree. *Cm* and *Rm* were chosen to be 4. According to the characteristic of Breadth-First Search and the setting of C_m and R_m, the coordinator can connect 4 children and each child can connect their own 4 children. So by calculation, to connect all 200 nodes (the maximum number of node in the simulation), in an ideal way, the maximum depth for Breadth-First Search tree should be 5. But the depth of the original Zigbee tree is unpredictable. So in order to get an ideal situation (all the nodes are connected) the *Lm* was set to be 30 and radio was set to be 100 meters. The results are shown in Figure 5 and Figure 6. In Figure 6 the average orphan node number is shown, almost all the nodes are connected so the result from Figure 5 shows a clear view about the difference between the original tree routing and improved tree routing (Breadth-First Search tree routing). In Figure 5 two pairs of results are compared: the average network depth and average hop count between two random nodes. As it is shown, the red lines stand for improved tree routing and blue lines stand for original tree routing. The improved tree routing average depth is around 4, and average hop count is around 7. The original tree routing the depth is about 6 in 100 nodes simulation but it increased to 10 in 200 nodes simulation, thus the average hop count has increased from 14 to 22.

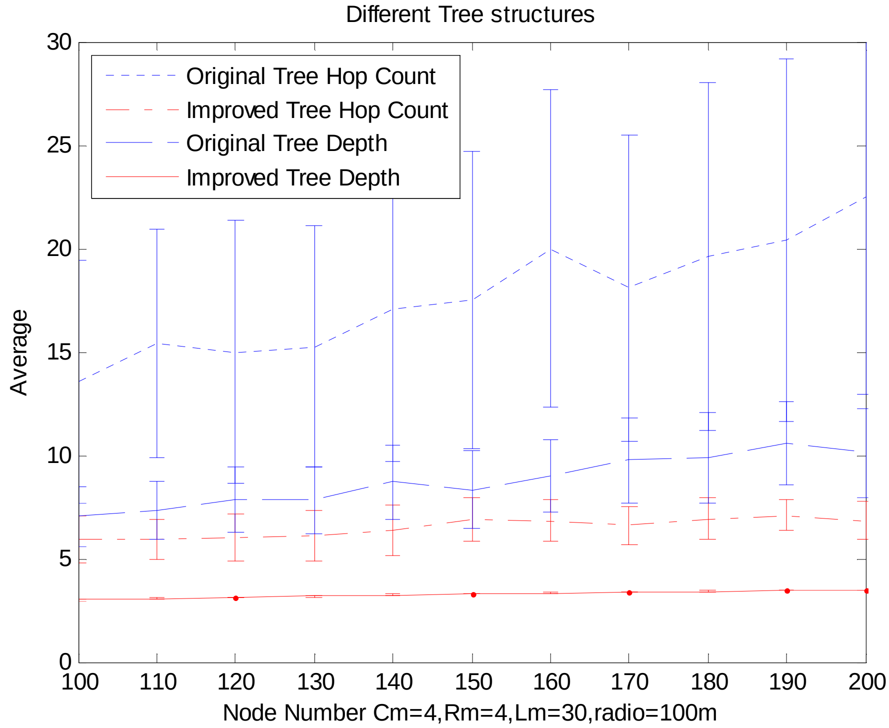

Figure 5 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=4, Rm=4, Lm=30, R=100m*)

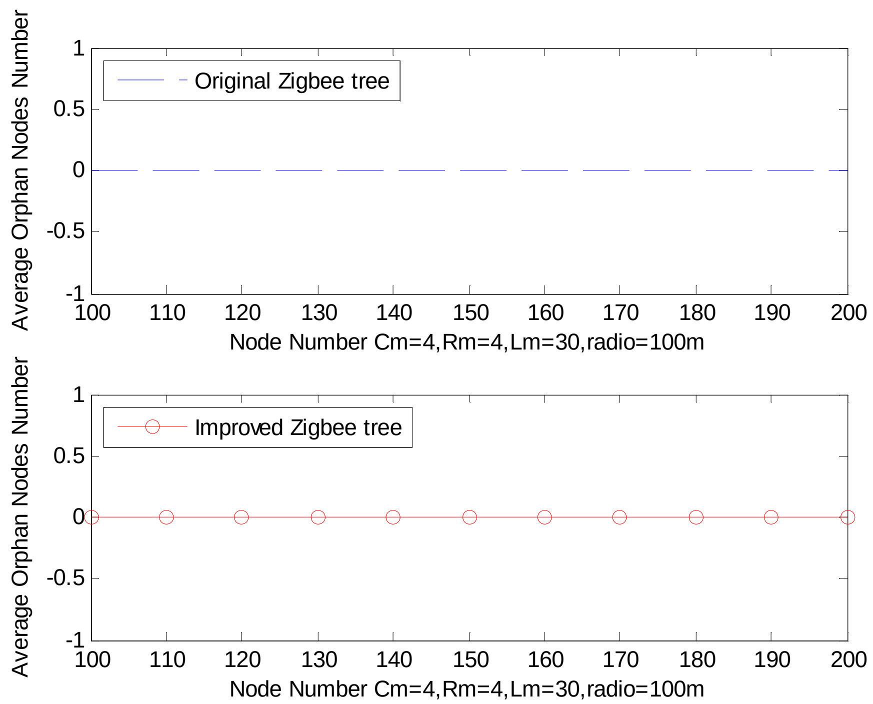

Figure 6 Orphan Node (*Cm=4, Rm=4, Lm=30, R=100m*)

For the second set of the simulation, due to the result shown in the first simulation the deepest depth of original tree can be reached more than 30, so in this simulation the value of *Lm* was decreased to 20 and the radio was decreased from 100 meters to 50 meters. The results shown in Figure 8 indicate that the improved tree routing also has an orphan node, but compared to the original tree routing, the orphan node number is much less. Moreover, as it shown, the orphan node number decreased when the node number increased. The reason is the transmission range. Because in this simulation the transmition range is set to be 50 meters the transmition range will affect the orphan node number. In Figure 7 the depth of the improved tree routing is maintained in 4 and the average hop count is around 8. In the original tree routing the average depth is about 6.2 in 100 nodes simulation and increased to 9 in 200 nodes simulation. The average hop count also increased from 13 in 100 nodes simulation to 18 in 200 nodes simulation. The performance of the improved tree routing is still much better than the original tree routing. So in order to increase the number of the orphan node and observe how orphan node problem influences the performance of both tree routings, the value of *Lm* is set to be 10 for the next set of the simulation.

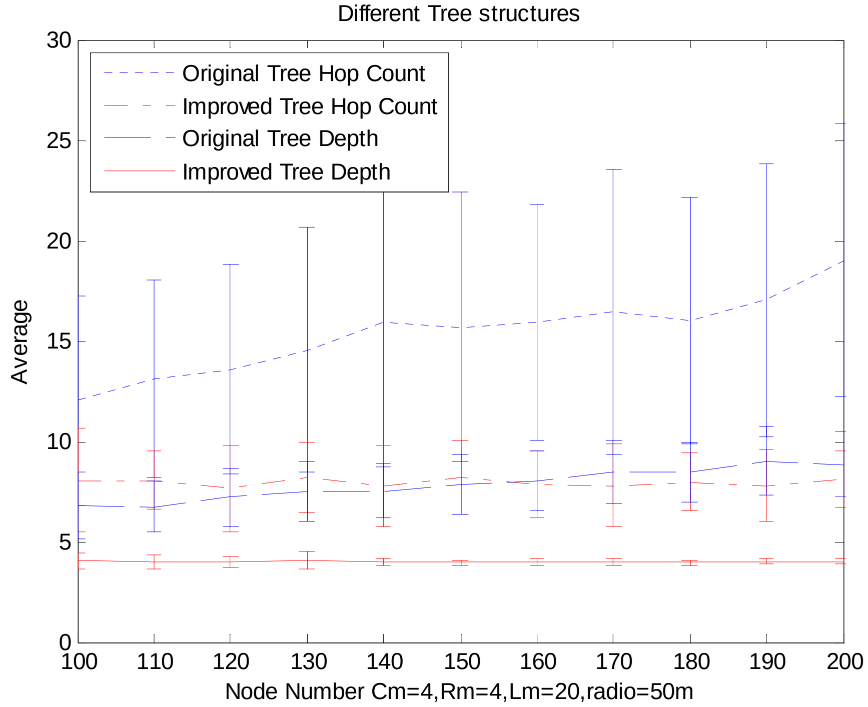

Figure 7 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=4, Rm=4, Lm=20, R=50m*)

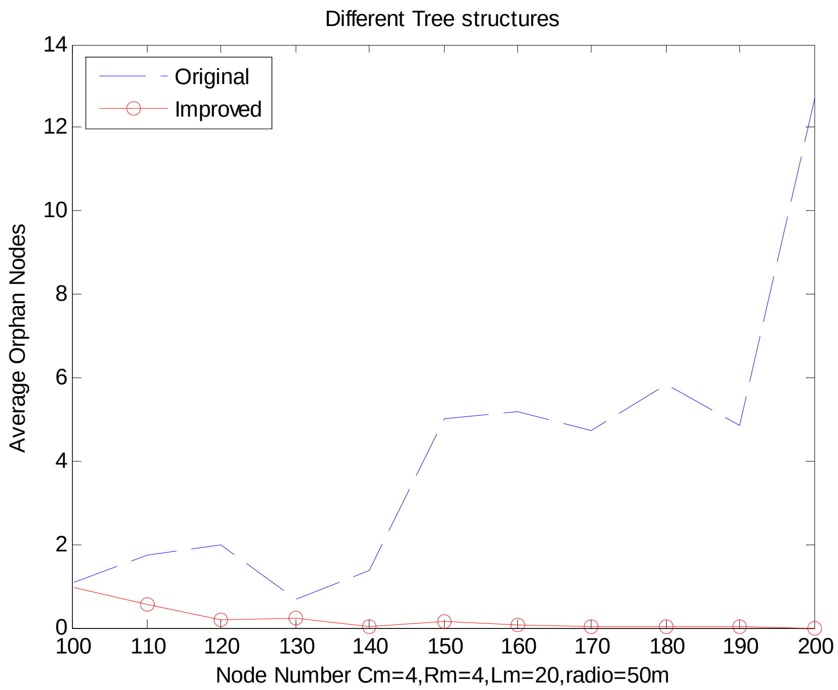

Figure 8 Orphan Node (*Cm=4, Rm=4, Lm=20, R=50m*)

The results shown in Figure 10 indicate that the orphan node in the original tree routing has increased significantly. Due to the value of *Lm* the average orphan node in the original tree routing has reached to 70 in 200 nodes simulation, 35% of the nodes did not connected. In Figure 9, as it is shown, the average depth of the improved tree routing is still around 4 and average hop count is 8. The depth of the original tree routing is stabilizing maintained around 6. Thus the average hop count of the original tree routing is around 12. .

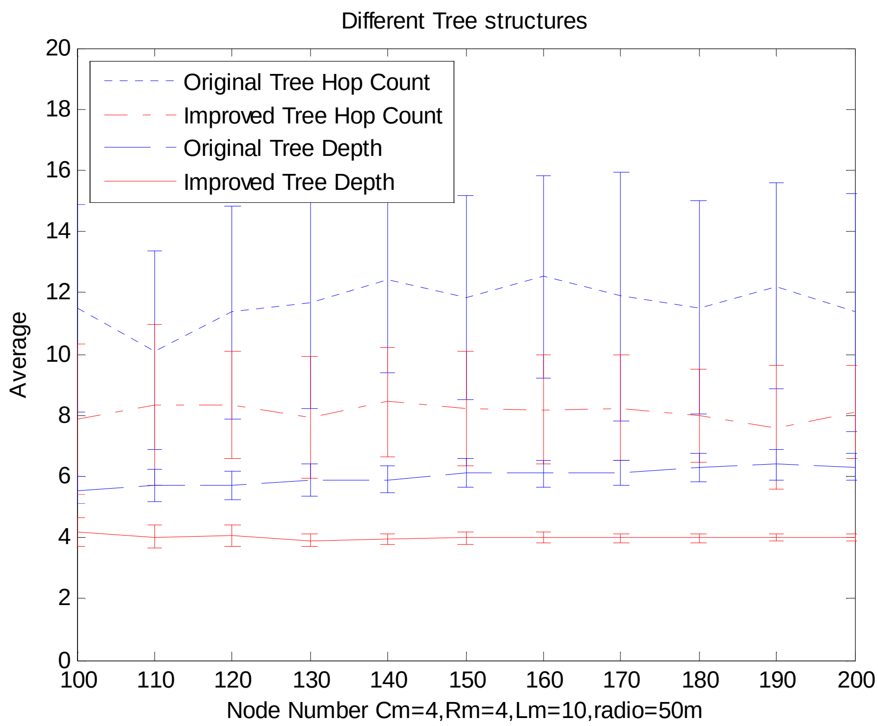

Figure 9 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=4, Rm=4, Lm=10, R=50m*)

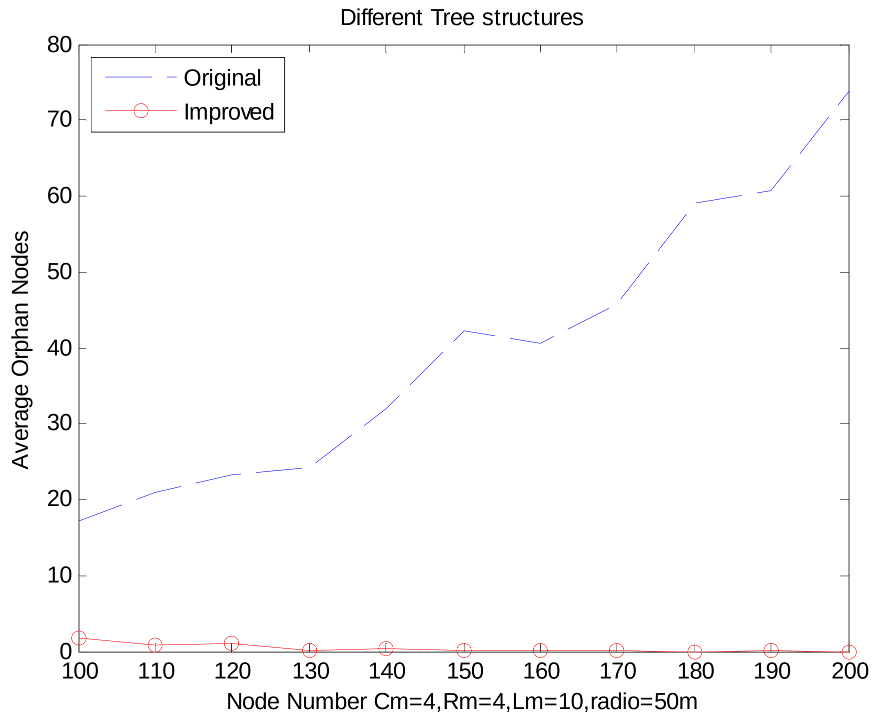

Figure 10 Orphan Node (*Cm=4, Rm=4, Lm=10, R=50m*)

For this set of the simulation, C_m and R_m were changed to be 2 and the radio was increased to 100 meters. In this simulation, it was expected to see how the performance of these two tree routings is in the worst case. Figure 12 shows that the orphan node in the original tree routing has increased significantly, almost 50% of the nodes did not connect in 100 nodes simulation and almost 75% nodes did not connect in 200 nodes simulation. The orphan node problem has greatly influenced the network structure. In Figure 13, due to the value of *Cm* and *Rm* the average depth of the improved tree routing has increased to 5. The average hop count increased to 10, but with the original tree routing the average depth is 6 and hop count is around 12 because of the orphan node problem. In general, the performance of the improved tree routing is still better than the original tree routing.

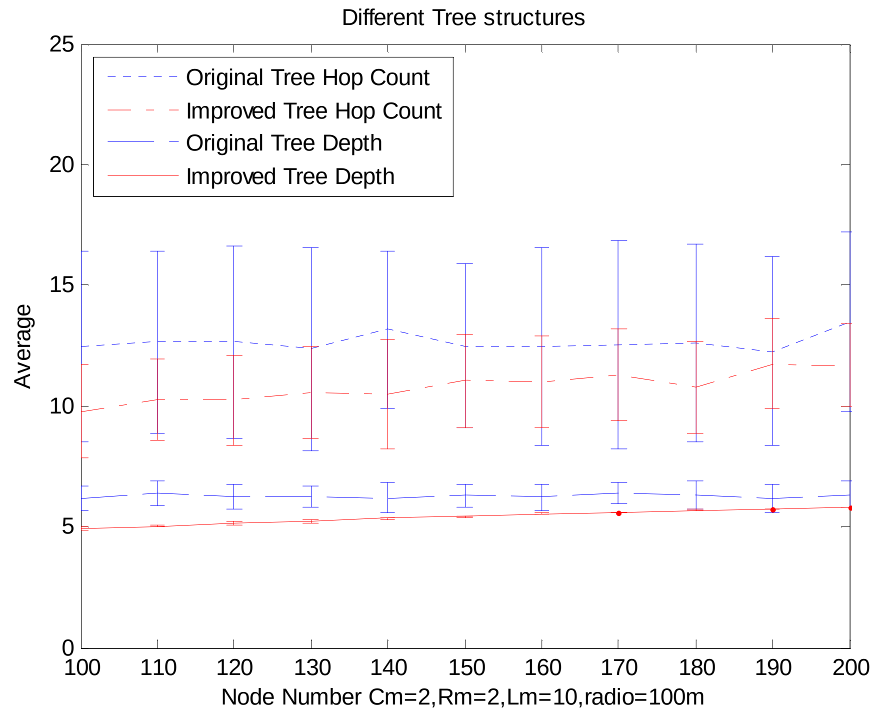

Figure 11 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=2, Rm=2, Lm=10, R=100m*)

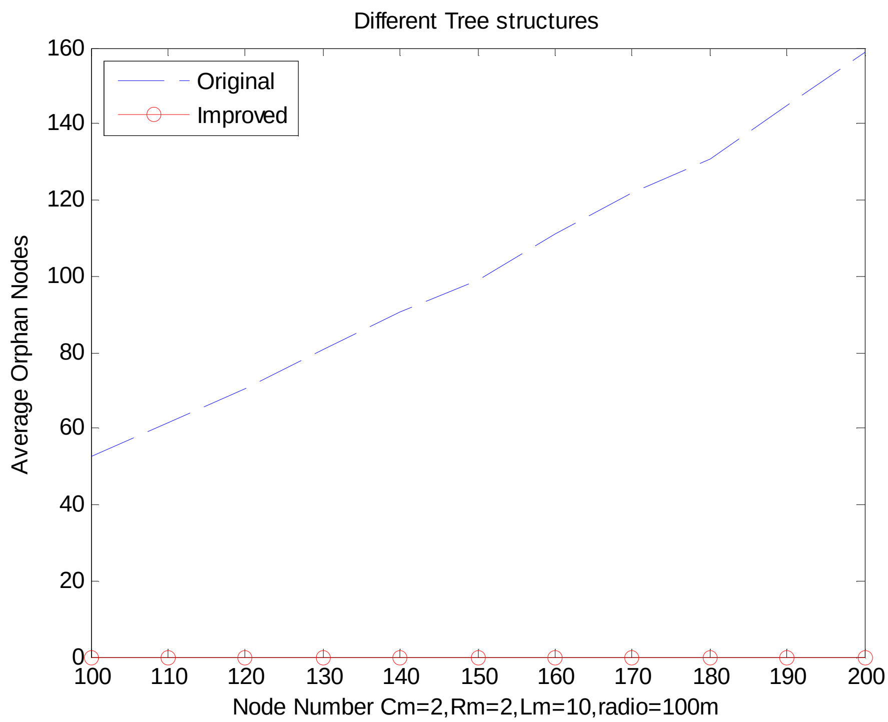

Figure12 Orphan Node (*Cm=2, Rm=2, Lm=10, R=100m*)

For this set of simulation, the radio was decreased from 100 meters to 50 meters. As it can be seen in Figure 14, the improved tree routing has some orphan nodes but compared to the original tree routing, the number of orphan nodes is not so big. Still, the orphan nodes in original tree routing are almost 50% in 100 nodes simulation and almost 75% in 200 nodes simulation. As shown in Figure 13, the average depth between the original tree routing and the improved tree routing has decreased. The average hop count in the original tree routing has dropped to 14 because of the orphan node problem. But the average hop count for improved tree routing did not change too much because of the network depth is still 5.

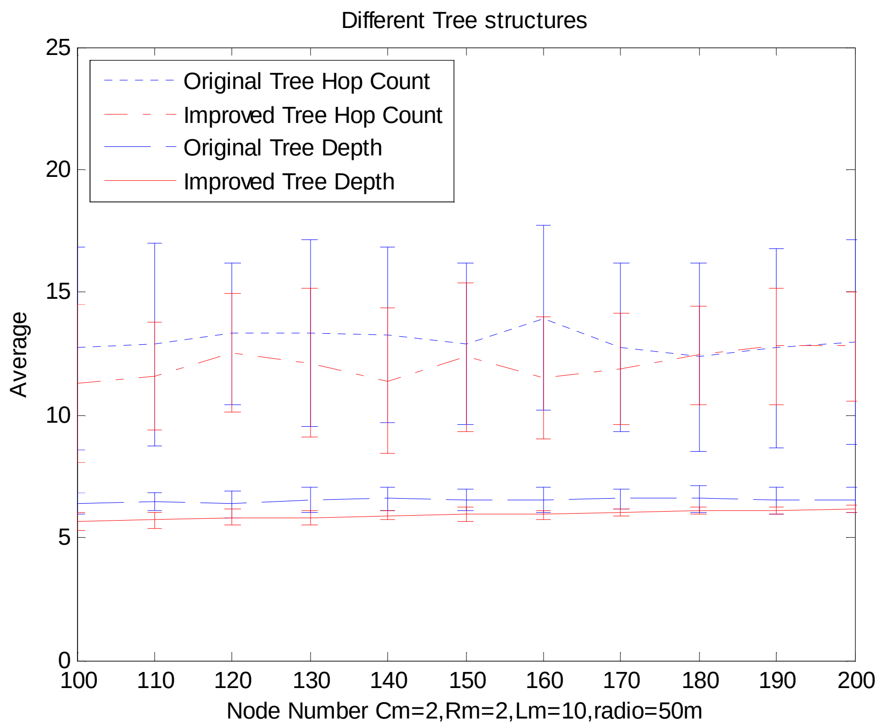

Figure 13 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=2*, *Rm=2*, *Lm=10*, *R=50m*)

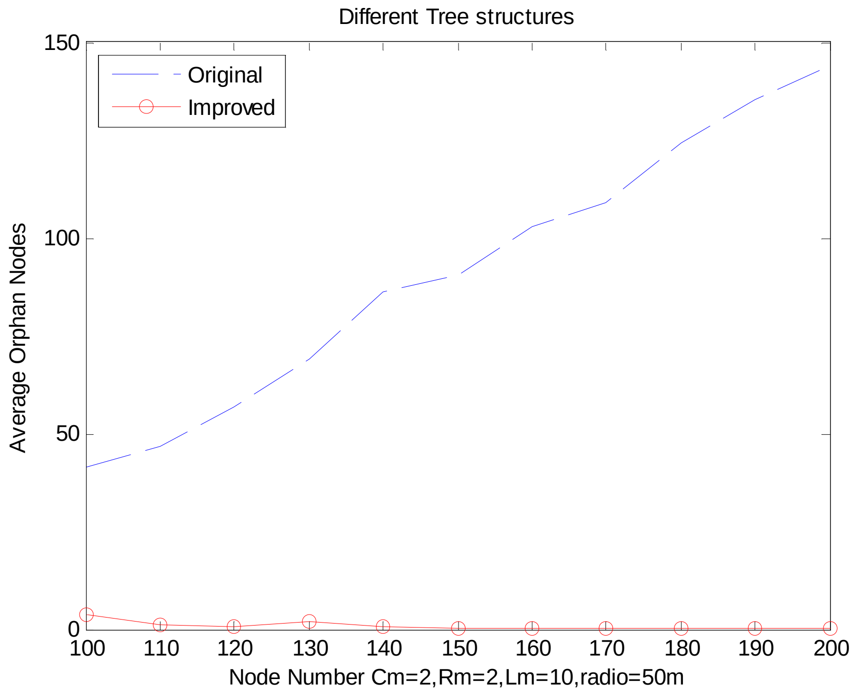

Figure 14 Orphan Node (*Cm=2, Rm=2, Lm=10, R=50m*)

In order to study the worst case, the results shown in the previous simulations indicate that the average depth for the improved tree routing is around 5 for C_m and R_m value 4. So in this simulation, *Cm* and *Rm* were set to be 2, *Lm* to be 5 and the radio to be 50 meters. The orphan node number is shown in Figure 16. As it can be seen, the orphan node in the improved tree routing has increased, 40% of the nodes are orphan nodes in 100 nodes simulation, and the orphan nodes grow to 45% in 200 nodes simulation. Needless to say in the original tree routing, 80% of the nodes are orphan node in 100 nodes simulation and 90% of the nodes are orphan nodes in 200 nodes simulation. In this case, as seen in Figure 15, the results are quite different compared to the previous simulations. Due to the orphan nodes, the average depth of the original tree routing is 4 and hop count is around 7, but the depth of improved tree routing is 5 and hop count is around 10.

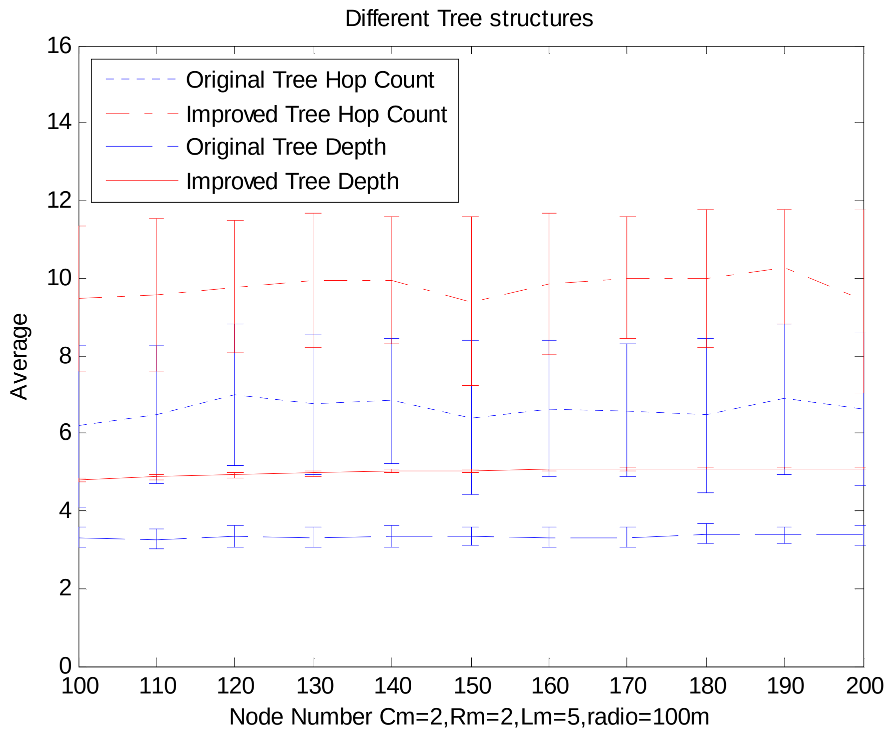

Figure 15 Tree Formation Result Comparison: Tree depth and Routing Hop Count (*Cm=2, Rm=2, Lm=5, R=100m*)

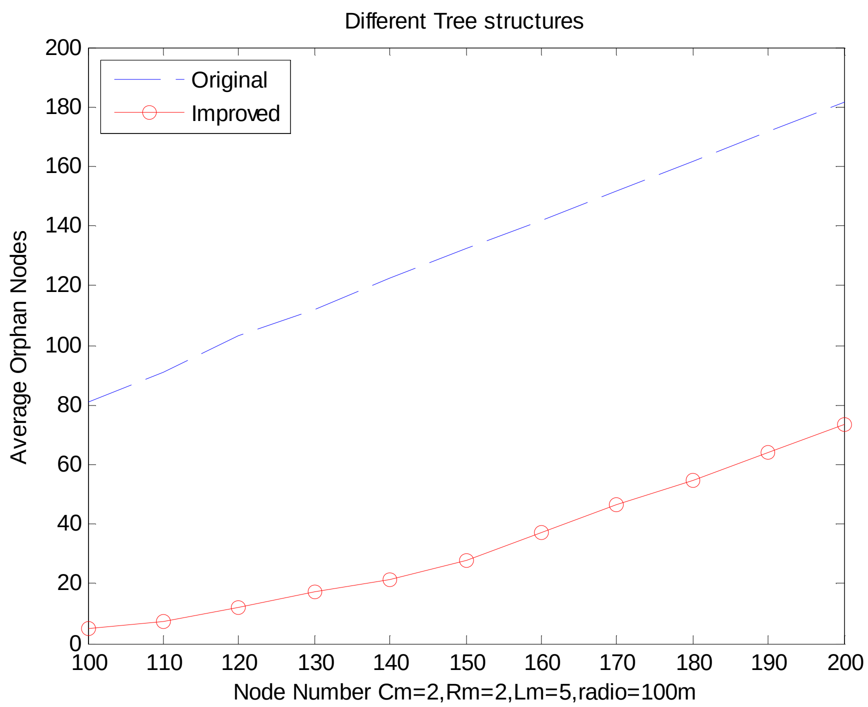

Figure 16 Orphan Node (*Cm=2, Rm=2, Lm=5, R=100m*)

5 CONCLUSIONS

After simulating the ideal situation and the worst situation, the performances of the improved tree routing demonstrate an outstanding result. The Breadth First Search is used to start the search from the root node and connect all the neighboring nodes. Then for each of the neighbor nodes find their children nodes in turn. So in this way the depth of the network has been reduced. And because of the reduced depth the hop count between two nodes also reduced. Furthermore, as the results show, the orphan node problem has also been solved by Breadth First Search. The Matlab simulation results show that when *Lm* is bigger than 5 there has only one or two orphan nodes in the Breadth First Search tree routing. For further research, the nodes join and leave problem has to be considered after the tree structure has been constructed. Because the depth of the Breadth First Search is much smaller than the original tree routing it can support more nodes in the same network layer.

REFERENCES:

/1/ Bidai Z., Haffaf H, Maimour M. 2011. Node disjoint multi-path routing for Zigbee cluster-tree wireless sensor networks. Multimedia Computing and Systems (ICMCS). 2011 International Conference on pp. 1-6.

/2/ Boujelben Hichem, Gaddour Olfa, Abid Mohamed. 2013. Enhancement and performance evaluation of a multicast routing mechanism in Zigbee cluster-tree Wireless Sensor networks. Systems, Signals & Devices (SSD), 2013 10th International Multi-Conference on, pp.1-8.

/3/ Chau-Chung Song, Chen-Fu Feng, Chih-Hui Wang, Der-Cherng Liaw. "Simulation and experimental analysis of a Zigbee sensor network with fault detection and reconfiguration mechanism," Control Conference (ASCC), 2011 8th Asian , pp.659,664, 15-18 May 2011

/4/ Cuomo F, Della Luna S, Cipollone E, Todorova P, Suihko T. "Topology Formation in IEEE 802.15.4: Cluster-Tree Characterization," Pervasive Computing and Communications, 2008. PerCom 2008. Sixth Annual IEEE International Conference on , pp.276,281, 17-21 March 2008

/5/ Fariborzi H and Moghavvemi M. "EAMTR: energy aware multi-tree routing for wireless sensor networks," Communications, IET , vol.3, no.5, pp.733,739, May 2009

/6/ Gaddour O, Koubaa A, Cheikhrouhou O, Abid M. "Z-Cast: A Multicast Routing Mechanism in Zigbee Cluster-Tree Wireless Sensor Networks," Distributed Computing Systems Workshops (ICDCSW), 2010 IEEE 30th International Conference on , pp.171,179, 21-25 June 2010

/7/. I. D. ??? Chakeres, L. Klein-verndt. AODVjr, AODV Simplified[J]. Mobile Computing and Communication Review, 2002, 6(3): 100-101.

/8/ Jiasong Mu; Kaihua Liu, "A Routing strategy based on node movement in Zigbee networks," Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on , pp.583,587, 25-27 June 2010

/9/Jiasong Mu; Kaihua Liu, "Effect of Node Mobility and Network Dimension to the Zigbee Routing Method," Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on , pp.1,5, 23-25 Sept. 2010

/10/Jianpo Li; Xuning Zhu; Ning Tang; Jisheng Sui, "Study on Zigbee network architecture and routing algorithm," Signal Processing Systems (ICSPS), 2010 2nd International Conference on , vol.2, pp.V2-389,V2-393, 5-7 July 2010

/11/Jing Sun; Zhongxiao Wang; Hong Wang; Xiaofen Zhang, "Research on Routing Protocols Based on Zigbee Network," Intelligent Information Hiding and Multimedia Signal Processing, 2007. IIHMSP 2007. Third International Conference on , vol.1, pp.639,642, 26-28 Nov. 2007

/12/Jongwon Kwon; Gwanghoon Ahn; Seoklae Kim; Sanghyuk Kang; Jo Chun Kim, "A study on energy-efficient tree routing protocol based on link quality metrics for remote air environmental monitoring system," ICCAS-SICE, 2009 , pp.4863,4868, 18-21 Aug. 2009

/13/Khatiri, A.; Mirjalily, G.; Khademzadeh, A., "Energy-Efficient Shortcut Tree Routing in Zigbee Networks," Computational Intelligence, Communication Systems and Networks (CICSyN), 2012 Fourth International Conference on., pp.117,122, 24-26 July 2012

/14/Kasraoui, M.; Cabani, A.; Mouzna, J., "Improvement of Zigbee Routing Protocol," Green Computing and Communications (GreenCom), 2012 IEEE International Conference on , pp.788,793, 20-23 Nov. 2012

/15/Kim, T.; Kim, S.; Yang, J.; Yoo, S.; Kim, D., "Neighbor Table Based Shortcut Tree Routing in Zigbee Wireless Networks," Parallel and Distributed Systems, IEEE Transactions on , vol.PP, no.99, pp.1,1, 0

/16/Kuo-Hui Tsai; Weimin Hung; Shih-Jung Sun, "On the tree formation of Wireless Sensor Networks," Systems and Informatics (ICSAI), 2012 International Conference on , pp.891,894, 19-20 May 2012

/17/Liu Dan; Qian Zhihong; Zhang Xu; Li Yue, "Research on Tree Routing Improvement Algorithm in Zigbee Network," Multimedia and Information Technology (MMIT), 2010 Second International Conference on , vol.1., pp.89,92, 24-25 April 2010

/18/Li Tao; Chai Qiao-lin; Wang Deng-di; Ban Yan-li, "Zigbee Tree Routing Optimize Based on Energy-balanced," Distributed Computing and Applications to Business Engineering and Science (DCABES), 2010 Ninth International Symposium on ., pp.301,307, 10-12 Aug. 2010

/19/M. Sheng zhou, Xie and Pan Chengyi, Probability and Mathematical Statistics, Beijing, Higher Education Press.

/20/Saeyoung Ahn; Doohyun Ko; Bumjin Kim; Sangbin Lee; Sunshin An, "Energy-Efficient Tree Routing Algorithm-Based Destination Family Group in Zigbee Networks," Sensor Technologies and Applications (SENSORCOMM), 2010 Fourth International Conference on , pp.1,6, 18-25 July 2010

/21/Stevanovic, D.; Vlajic, N., "Performance of IEEE 802.15.4 in wireless sensor networks with a mobile coordinator implementing various mobility strategies," Local Computer Networks, 2008. LCN 2008. 33rd IEEE Conference on , pp.680,688, 14-17 Oct. 2008

/22/Sheng Zhang; Hongliang Zhou, "An Improvement of Zigbee Tree Routing in the Application of Intelligent Irrigation," Intelligent System Design and Engineering Application (ISDEA), 2010 International Conference on , vol.1., pp.255,260, 13-14 Oct. 2010

/23/Wanzhi Qiu; Skafidas, E.; Qi Cheng, "A hybrid routing protocol for wireless sensor networks," Communications and Information Technologies, 2007. ISCIT '07. International Symposium on , pp.1383,1388, 17-19 Oct. 2007

/24/X.-Q. Zhu, Z.-X. Chen. Design of Design of Home Wireless Network Using Routing Algorithm of Tree and AODVjr-PB[J]. Journal of Chongqing University of Posts and Telecommunications, 2011, 21(3): 343-348.

/25/Yu-Kai Huang; Ai-Chun Pang; Pi-Cheng Hsiu; Weihua Zhuang; Pangfeng Liu, "Distributed Throughput Optimization for Zigbee Cluster-Tree Networks," Parallel and Distributed Systems, IEEE Transactions on , vol.23, no.3, pp.513,520, March 2012

/26/Yukun Yao; Zilong Zhang; Zhi Ren; Sainan Shi, "Neighbor-Information-Assisted Tree Routing for Zigbee Networks," Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on , vol.1, no., pp.455,458, 23-25 March 2012

/27/Yuan-Yao Shih; Wei-Ho Chung; Pi-Cheng Hsiu; Ai-Chun Pang, "A Mobility-Aware Node Deployment and Tree Construction Framework for Zigbee Wireless Networks," Vehicular Technology, IEEE Transactions on , vol.62, no.6, pp.2763,2779, July 2013

/28/Yujie Zhang; Sale Hui, "The study of Zigbee network routing algorithm based on adjacent relation," Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on., pp.1484,1487, 21-23 April 2012

/29/Zigbee Alliance. Document 053474r17, Zigbee Specification, 2007: 370-372.

/30/26-Zigbee Alliance, Network Layer Specification 1.0, Dec. 2004.

/31/Zijing Lin; Meng, M.Q.-H.; Huawei Liang, "A route discovery method based on limited flooding in Zigbee networks," Automation and Logistics, 2008. ICAL 2008. IEEE International Conference on , pp.3039,3044, 1-3 Sept. 2008

/32/Zhi Ren; Lihua Tian; Jianling Cao; Jibi Li, "An efficient hybrid routing algorithm for Zigbee networks," Instrumentation & Measurement, Sensor Network and Automation (IMSNA), 2012 International Symposium on , vol.2, pp.415,418, 25-28 Aug. 2012

/33/Zhenli Zhang; Fazhu Zhou; Yuming Liang, "Zigbee routing algorithm based on energy awareness and energy balance," Consumer Electronics, Communications and Networks (CECNet), 2012 2nd International Conference on , pp.747,750, 21-23 April 2012

/34/Zhao Zhihong; Xia Hongsen; Zhou Hongjie; Guo Zhi, "Studying and implement of routing in wireless sensor network for the type of cluster tree," Control and Decision Conference, 2008. CCDC 2008. Chinese , pp.2012,2017, 2-4 July 2008