

# Efficient Routing in Zig-Bee Wireless Network using Shortcut Tree Routing

Anand Kumar. N  
PG Scholar  
Department of CSE (PG)  
Sri Ramakrishna Engineering College  
Coimbatore

A. Grace Selvarani, Ph.D  
Professor & Head  
Department of CSE (PG)  
Sri Ramakrishna Engineering College  
Coimbatore

## ABSTRACT:

Zigbee Tree Routing, which doesn't need any routing table/route discovery overhead is used in several resource limited devices and applications. ZTR has a basic limitation regarding providing of optimal routing path as it follows tree topology, hence an optimal routing path can't be achieved. In this paper, we proposed a protocol stated as Shortcut Tree Routing (STR) similar to ZTR's entities, such as low memory consumption, no route discovery overhead, providing nearest optimal routing path using hierarchical addressing scheme and calculating the remaining hops from source to destination. The specifications are unaltered, as STR uses just the addressing scheme and neighbor table in association with the Zigbee standards. The research process illustrates the 1-Hop neighbor communication representation upgrades the overall network performance execution by splitting up of the traffic load concentrated on the tree links. The performance evaluation indicates, STR accomplishes the performances of AODV and ZTR in certain conditions of it, such as network density, configurations and network traffic patterns.

## Index Terms:

ZTR- Zigbee Tree Routing, STR- Shortcut Tree Routing, neighbor table, MANET, WSN, and IEEE 802.15.4.

## 1. INTRODUCTION

Zigbee, a low power and cost effective radio standard accepted widely in association with Personal Area Networks can connect up to 1000 devices through wireless mesh network patterns. Zigbee find its usage in home automation [1], MANETS, vehicle tracking services etc. Zigbee network layer [4] facilitates with routing network formation specifies and allots a 16-bit short address, dynamically for each node connected. AODVjr [5] finds its presence in the reactive protocol of Zigbee, which depicts MANETS throughout the on-demand route discovery. Communication between the source and destination nodes increases the route discovery overhead, traffic and memory consumption in ordinary communication protocol. Whereas ZTR reduces aforementioned through distributed block addressing scheme [4]. The main factor that distinguishes ZTR over other protocols is its capacity to transfer packets from the source to destination via intermediary nodes; which doesn't require a route discovery overhead, as other nodes are issued with hierarchical addresses. This promising factor of ZTR find its application over IOT, smart grid services, etc. Even though ZTR uses the tree

topology pattern to communicate or transfer packets from one node to another; optimal routing path is yet to be achieved.

In order to preserve the advantages of ZTR such as no route discovery overhead, lesser memory bandwidth consumption and to avoid the tree link communication to nearer nodes, a concept of 1-Hop is introduced in STR. 1-Hop mechanism uses the nearby nodes' information and shortcuts the tree routing in mesh topology. STR makes use of the smallest remaining tree hops to destination while communication and transferring the packets between the nodes, thus enhancing the speed of the transaction and limiting the usage of time effectiveness. STR finds its process attractive in the field of mesh topology and Zigbee standards, as STR doesn't need any extra offering in mechanism standards but just adding upon the 1-Hop information. This paper furnishes the objectives as, first ZTR has certain issues regarding the network performances, such as detour path problem and traffic concentrated problem as they are rectified by proposed STR. Second, the traffic concentration problem of ZTR is minimized to a great extent by introducing the 1-Hop mechanism by STR. Third, performance analysis of ZTR, STR and AODV is carried forward with criterias like traffic types, network constraints, network density.

## 2. RELATED WORKS

MANET [11] routing protocol is said to be proactive and reactive. Proactive routing protocol, as the name itself illustrates has an up-to-date tracking of all the transmission process and will be always active. Topology status and required fields of processing are frequently updated. OLSR [6], DSDV [7] are some of the examples of it. Meanwhile reactive protocols updates the fields when only a transmission happens and not periodically. Thus the route discovery overhead is used only when a transmission takes place, leading to a later waiting time. Examples are ODV[8], DSR[9], TORA[10]. Nevertheless of its kind such as proactive or reactive, MANETS provide optimal routing path from a source node to destination node. Hence causing the routing table size to be in a bigger larger manner. To find the routing path, MANETs need to put equivalent control packets in their places of one another and sending of packets may experience a low rate and shorter bandwidth channels.

Regarding communication traffic pattern, they can be segmented into any-to-any, many-to-one and one-to-many [13]. In any-to-any pattern, any node can act as a source

and destination. In many-to-one, some nodes i.e. greater than one node will act as a source node and a single node will be acting as a destination node. In one-to-many, a single node acts as the source information node and many nodes will be acting as destination nodes. Many-to-one and one-to-many traffic pattern can be stated through Collection Tree Protocol [CTP] and Routing Protocol for Low Power and Lossy Networks [RPL][15]. CTP deals with a base station, serving as the root which has some nodes connected to it, forming a group of bunch of sensor nodes. The metric through which CTP operates can be explained with Expected Transmission Count [ETX]. With the root nodes, ETX remains zero. The other nodes calculate their ETX through summing up of its link and parent nodes in order to transmit the information fast as the node with low ETX is chosen. CTP is deeply associated with TinyOs[16].

In order to achieve maximum efficiency on the many-to-one pattern RPL invokes DODAG - Destination Oriented Directed Acyclic Graph. Each node in DODAG achieves an optimal routing path on their communication using RPL through a single route request from destination area. This protocol avoids unwanted and over evaluated route discoveries which will be given by each and every destination node and with that information an optimal route head can be calculated. This causes an increases route discovery overhead. In order to avoid these chaos RPL can be invoked. Even though CPT and RPL has the above mentioned advantages over many-to-one and one-to-many, still there is no optimal solution for any-to-any, as it uses the tree topology as like ZTR causing detour path problem and traffic concentration problems. There has been researches going on the Zigbee standard for path efficiency improvement in ZTR. Initial version of this paper [17] indicates the use of 1-Hop neighbor table. This concerns with reduced routing cost of ZTR and STR's proposed algorithm just projects and determines the lowest routing path reducing up to 30 hop counts. With a lesser optimal routing method ZTR also faces issues in performance [17]. Tree link concentration problem, led by route discovery overhead are terminated and it has proven that STR has a higher overall performance compared to ZTR in the fields of packet delivery ratio, latency etc.

### 3. ZIGBEE TREE ROUTING

ZTR operates under a circumstance, that the Zigbee devices use multi-hops to transmit information from a node-to-node without any route discovery procedure and based on hierarchical block addressing scheme indicated in (1) and (2). The Following expression illustrates the addressing scheme of Zigbee with  $C_m$ ,  $R_m$  and  $L_n$  with their hierarchy expression. As  $C_m$  illustrates maximum number of children a parent can have and  $R_m$  illustrates maximum number of routers a parent can have as a child and  $L_m$  represents maximum tree level of the network.

$$C_{skip}(d) = \begin{cases} 1 + C_m \cdot (L_m - d - 1), & \text{if } R_m = 1, \\ \frac{1 + C_m - R_m - C_m \cdot R_m^{L_m - d - 1}}{1 - R_m}, & \text{otherwise,} \end{cases} \quad (1)$$

$$A_k = A_{parent} + C_{skip}(d) \cdot (k - 1) + 1 \quad (1 \leq k \leq R_m),$$

$$A_n = A_{parent} + C_{skip}(d) \cdot R_m + n \quad (1 \leq n \leq C_m - R_m). \quad (2)$$

$$A < D < A + C_{skip}(d - 1) \quad (3)$$

The  $C_{skip}(d)$  in (1) represents the address spacing size of each router node at the level 'd'. Following the above illustration, the assignment scheme of network address can be stated as for each Kth router, capable child and Nth end device is given by the parent at tree level d. In this mode of addressing, the available network address space is pre-allocated and divided recursively into spaces as there is an increase in tree categories. The  $C_{skip}(d)$  is said as the size of the address space in a tree level 'd' and  $C_{skip}(d+1)$  is the size of address space with respect to router capable children in definite addressing. A destination can be easily identified as an immediate or a descendant of each source with this hierarchical addressing scheme. If (3) is met with the resultant, then the destination having the addresses 'd' is said as descendant of a node with address 'A'. ZTR transmits the information to one of the child nodes if the destination node is a descendent, else it is stated as parent.

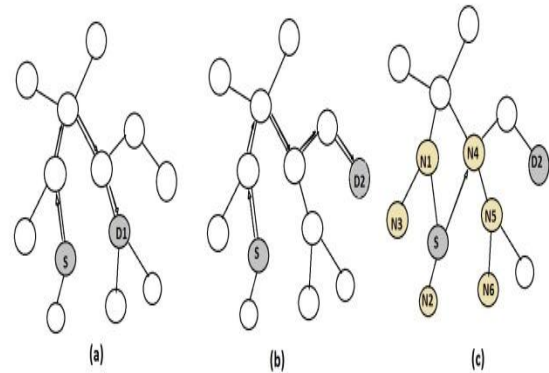


Fig.1 Zigbee Tree Routing and Shortcut Tree Routing

Fig 1a and 1b deals with the detour path problem of ZTR, which illustrates that the packet is sent through distant nodes even though the destination is available nearby and within a range of 2-Hop transmission. If the corresponding destination is in the neighbor table then the router can send the packet directly to the destination node without the router protocol, through a rule stated as direct transmission rule [6]. Fig 1b illustrates, if the destination node is beyond 2-Hop range, the transmission causes the direct transmission rule to fail and causing traffic concentration problem. Traffic concentration problem is caused due to a single node facing a series of packets passing through the same tree link. This causes collision of the packets leading to packet delivery ratio degradation, network performance degradation etc.

### 4. SHORTCUT TREE ROUTING

ZTR faces the above mentioned problem and is rectified in this following algorithm, said as Shortcut Tree Routing algorithm (STR). STR follows ZTR but utilizes the neighbor node as its next destination node using 1-Hop. In fig 2c using the above mentioned methods such as calculating remaining tree hops and

**Table 1: Algorithm to find Ancestors at Each Three level**

---

**Find\_Ancestors(devAddr)**

---

**Input:** devAddr-device's network address

**Output:** level(devAddr)-tree level of devAddr,  
A(devAddr)-network addresses of the devAddr's  
Ancestors at each tree level

```

1: A(devAddr, 0) ← 0
2: For i=0 to Lm-1
3: If(A(devAddr, i) ≡ devAddr)
4: Return i, A(devAddr)
5: End if
6: rIndex ← (devAddr - A(devAddr, i) - 1) / Cskip[i]
7: if(rIndex < Rm)
8: A(devAddr, i+1) ← A(devAddr, i) + Cskip[i] * rIndex + 1
9: Else if(rIndex > Rm)
10: A(devAddr, i+1) ← devAddr
11: End if
12: End for

```

---

**Table 2: Shortcut Tree Algorithm**

---

**Find\_NextHopAddr(dstAddr)**

---

**Input:** dstAddr- network address of the destination

**Output:** nextHopAddr – next hop address for the destination

---

```

1: Initialize minRouteCost with
2: Level(dstAddr), A(dstAddr) ← Find_Ancestors(dstAddr)
3: For each (neighbor's address Nk in neighbor table)
4: Level(Nk), A(Nk) ← Find_Ancestors(Nk)
5: Level(LCA) = 0
6: While (level(LCA) < min(level(dstAddr), level(Nk)) and
   A(dstAddr, level(LCA)) = A(Nk, level(LCA)))
7: ++level(LCA)
8: End while
9: nbrRouteCost < level(dstAddr) + level(Nk) - 2*level(LCA)
10: if (nbrRouteCost < minRouteCost)

```

11: nextHopAddr ← Nk

12: minRouteCost ← nbrRouteCost

13: end if

14: end for each

15: Transmit packet to nextHopAddr

Zigbee address hierarchy, STR calculates the next hop node as  $N_4$  from source  $S$  to the destination  $D_2$ . This transmission can be illustrated as the levels of tree links when a packet is sent from source, its common ancestor node plays a vital role in transmitting that packet to the nearer or down by node and then to destination  $D_2$ . Through STR we can compute remaining tree hops from an arbitrary source to a destination using ZigBee address hierarchy and tree structure. Remaining tree hops can be computed using tree levels of source node, destination, and their common ancestor node, because the packet from the source node goes up to the common ancestor, which contains the address of destination, and goes down to destination in ZTR.

Tables 1 and 2 illustrates the algorithm and definitions used by STR. Let  $level(u)$  represents tree level of node  $u$  and  $A(u)$  be  $\{A(u, i) \mid A(u, i) \text{ is the network address of } u\text{'s ancestor at tree level } i, i < level(u)\}$ .  $LCA(s, d)$  [18] can be stated as lowest common ancestor between source node  $s$  and destination  $d$ .

Table 1 describes the algorithm to find ancestors' network address at each tree level together with tree level of given devAddr. Since the network address of device is contained in its ancestors' address space in lower tree levels, we can find the rIndex. rIndex is stated as the router-capable child order  $k$  in (2) by dividing the size of address space from  $A(devAddr, i)$  to the devAddr by the  $Cskip(i)$  [21]. If rIndex is less than  $R_m$ , then the  $A(devAddr, i+1)$  is router device, so the address is derived from the addressing scheme for  $A_k$  in (2). If rIndex is greater than or equal to  $R_m$ , it states that the  $A(devAddr, i+1)$  is network address of the end device and it is same as the devAddr [21].

Finding  $A(devAddr)$  process starts with the root node having its network address as 0 and incrementing its value with the significant devAddr which is close to the ancestors' address value. A common ancestor address can be found by comparing the source and destination value between the ancestor's addresses in each tree level. The common ancestors of the device is found in order to compute tree routing cost between a source and destination. Considering source node as  $S$  and Destination node as  $D$ , then the tree routing cost between  $S$  and  $D$  can be calculated with tree levels of  $S, D$  stated as  $LCA(S, D)$ . The packet from the source node  $S$  always goes up to the lowest common ancestor  $LCA(S, D)$  through parent node. From the  $LCA(S, D)$ , the packet directs to the subtree level of node and goes down through the child nodes to the destination. Since the routing hops from  $S$  to  $LCA(S, D)$  and from  $LCA(S, D)$  to  $D$  can be calculated using difference of tree levels, the tree routing cost from  $S$  to  $D$  can be calculated by the equation " $level(D) + level(D) - 2 * level(LCA(S, D))$ " [21].

Table 2 illustrates the proposed STR algorithm stating a source or an intermediate node to determine its next hop node that has a minimum remaining tree hop or hops for the given destination. In Table 2, level(dstAddr) and A(dstAddr) for the given dstAddr is computed. Then, for each neighbor entry nk, the remaining tree hops from the nk to the dstAddr, a nbrRouteCost, is calculated, by finding the level(nk) and level(LCA)(nk, dstAddr). Finally, a source or an intermediate node selects the neighbor nk as the next hop node, which has the minimum remaining tree hops to the given destination, and transmits a packet to the next hop node [21].

If there is zero or no neighbor node to reduce remaining tree hops comparing with ZTR, STR chooses either parent or one of children node as next hop node like ZTR. Table 1 has no information regarding selection of the next hop node which is according to ZTR, because all the parent and children are already included in the neighbor table. Therefore, the upper bound of minRouteCost in Table 2 is the same as in ZTR's routing cost which is decided when the next hop node is selected.

STR's routing path may not be always an optimal one in an aspect of the end-to-end hop distance as next hop node is selected based on 1-hop neighbor table. Maintaining a 2-hop neighbor information requires high protocol overhead with high node density [19], [20]; therefore in order to provide a resource efficient routing protocol as per memory consumption and routing overhead STR is used.

## 5. PERFORMANCE EVALUATION

STR evaluation is done in account with metric of routing performance and overhead; with the evaluation includes hop count, end-to-end latency, packet delivery ratio etc. simulations are done based

**Table 3: Simulation Parameters**

<u>Simulation Parameters</u>	<u>value</u>
Network size	90mx90m
Number of nodes	75,150.350
Deployment type	Random
Position of coordinator	Center
Number of iterations	25
PHY/MAC protocol	IEEE 802.15.4
Propagation model	Two-ray
Max. Rx range	20m
Max. Carrier sensing range	30m
Interface queue/size	Priority queue
Network protocol	ZTR/STR/AODV
Cm/Lm	3,9/2,4/4,8
Simulation time	250sec
Association duration	0-30sec
Application session	

Packet type	CBR
Packet interval	1packet/sec
Start/end time	225-250sec
Number of sessions	10,20,40,.....,100

on analyzing network density, traffic pattern and network configuration. NS-2[14] simulator is used for evaluation with IEEE 802.15.4 PHY/MAC protocols to compare STR with ZTR and AODV. Settings and parameters are stated above in Table 3 and are used in following sections unless otherwise stated to be. Standard procedures including deploying the nodes and Zigbee address assignment is started from any time within 0s to 30s. Zigbee standards persuades some link status message with 1-Hop broadcast with "nwkLinkStatusPeriod" invoked in simulation up to 15 seconds. The 1-Hop neighbor information is associated with ZTR's status on receiving successful beacons while transmission happens.

### 5.1 Effect of Network Density

The basic performance evaluation strategy begins with the evaluation of network performance against the network density in order to determine the stability structure and scalability. The simulation settings will be followed as per the Table 3. In addition to network density comparing of the any-to-any and many-to-one traffic pattern is dealt in sections 5.1.1 and 5.1.2 in accordance with selection of the source and destination. Traffic session numbers are limited to 20 traffic patterns.

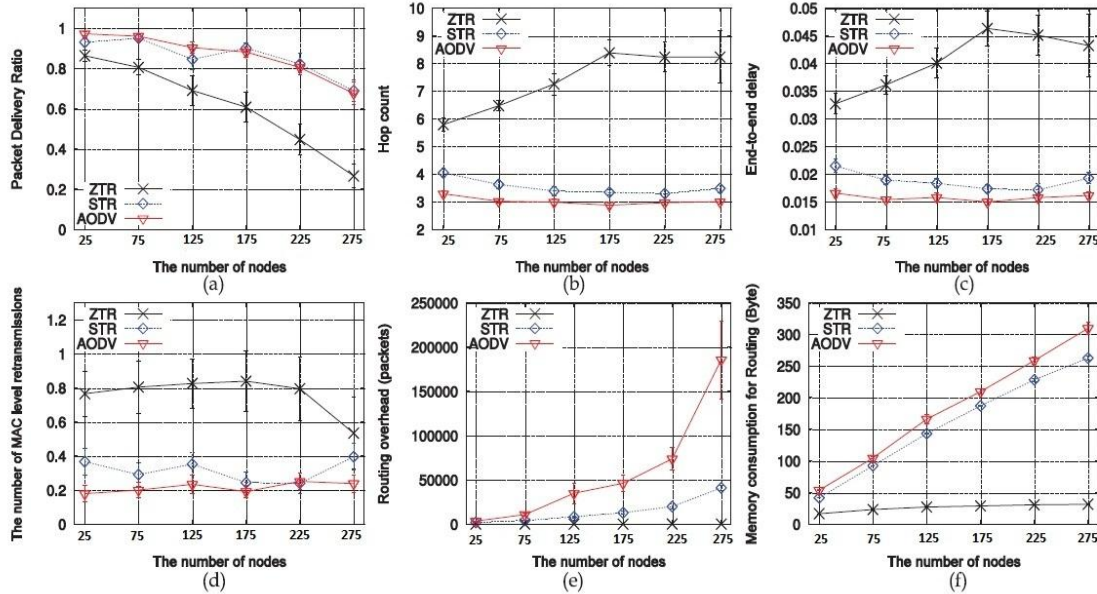
#### 5.1.1 Any-to-Any Traffic

The Fig. 2a clearly illustrates that there is a steep drop in packet delivery ratio of the ZTR up to 27 percent and is due to the increase in nodes. The reason behind indicates a large hop count and overlapped routing path of the packets as they are crowded and focused more on root of a tree causing collision and interferences among them. This causes a reduced packet delivery ratio. Meanwhile STR and AODV have their packet delivery ratio about 70 percent even in the 200 nodes. This is due to shorter routing paths. It is to be noted from the fig 2a that the performance of STR outshines over the AODV from the node 225 due to higher average hop count. As AODV has an overhead factor to be dealt with before data packet transmission; causing some interference which of whole are avoided by STR as it doesn't have an overhead that leads to network flooding and degradation.

Fig 2b illustrates the hop count of a packet from source to destination and shows a whooping increase of the ZTR hop count from 5.7 to 8.3 hops due to tree level escalation. There is a saturation level after 225 nodes and it is because of the dropping of packets after successful message deliveries by ZTR. Meanwhile STR and AODV has an average hop count of 3.2 to 4 and 2.9 to 3.2 hops respectively having a low level of concern over the density or topology of the network. As per the end-to-end latency i.e.; delaying is concerned, ZTR has a larger delay up to 47 ms due to the following of the tree level topology, as indicated in fig 2c. STR and Aodv lies far away from ZTR with about 18 and 15ms of end to end latency which is

clearly illustrated in fig 2d. Considering the average number of MAC level retransmissions that happens due to collision of packets, traffic, congestion etc. ZTR needs 0.8 times of retransmission which is a way longer than that of the STR and AODV which needs only 0.2 and 0.25 to 0.4 times in spite of the traffic patterns followed by network

topology. Fig 2e illustrates the routing overhead measurement wherein AODV's overhead was increased exponentially due to route request queue that floods the whole network. STR and ZTR acts according to the Zigbee standard protocol and has no or very less routing



**Fig 2 Routing performance and overhead for the network density (a) packet delivery ratio (b) hop count (c) end-to-end latency (d) number of mac level retransmission per session (e) number of routing packet overhead and (f) memory consumption for routing table**

overhead with the network density has least effect on them. Fig 2f shows the memory consumption graph stating ZTR having the least memory consumption as STR and AODV having certainly higher memory consumption. The reason behind ZTR's low consumption is that it does not need to store any 1-hop information, routing overhead, link state mechanisms. Meanwhile STR and AODV needs to store the aforementioned with AODV additionally stores route discovery table and routing table.

### 5.1.2 Many-to-One Traffic

Expecting many to one traffic patten to have lesser packet delivery ratio than any to any goes into vain. In this section, as the severe congestion among packets play a vital role affecting the delivery ratio many to one traffic remains 10 percent higher than that of any to any traffic due to many nodes communicating with a single destination node. Causing packets to collide and bottle neck situations to get happened.

## 6. CONCLUSION

In this paper, certain criterias affecting the overall network performances are identified, such as detour path problem and traffic concentration problems of ZTR. To overcome this, STR is proposed with the same Zigbee standards with a 1-Hop mechanism included to propagate the packet to its nearby nodes. The simulation results show a clear result of STR outperforming ZTR in major aspects. Therefore as discussed in section 4 we expect STR to be utilized in

many Zigbee oriented application for a small memory and high routing performances.

## 7. REFERENCES

- [1] D. Han and J. Lim, "Smart Home Energy Management System Using IEEE 802.15.4 and ZigBee," IEEE Trans. Consumer Electronics, vol. 56, no. 3, pp. 1403-1410, Oct. 2010.
- [2] S. Chen et al., "A Reliable Transmission Protocol for ZigBee-Based Wireless Patient Monitoring," IEEE Trans. Information Technology in Biomedicine, vol. 16, no. 1, pp. 6-16, Nov. 2012.
- [3] P. Yi, A. Iwayemi, and C. Zhou, "Developing ZigBee Deployment Guideline under WiFi Interference for Smart Grid Applications," IEEE Trans. Smart Grid, vol. 2, no. 1, pp. 110-120, Nov. 2011.
- [4] ZigBee Alliance, ZigBee Specification, 2009.
- [5] I. Chakeres, "AODVjr, AODV Simplified," ACM SIGMOBILE Mobile Computing and Comm. Rev., vol. 6, pp. 100-101, 2002.
- [6] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," Internet Request for Comments 3626, Oct. 2003.
- [7] C.E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing

- (DSDV) for Mobile Computers,” SIGCOMM Computer Comm. Rev., vol. 24, pp. 234-244, 1994.
- [8] C.E. Perkins and E.M. Royer, “Ad-Hoc On-Demand Distance Vector Routing,” Proc. IEEE Workshop Mobile Computer Systems and Applications, Feb. 1999.
- [9] D.B. Johnson and D.A. Maltz, “Dynamic Source Routing in Ad Hoc Wireless Networks,” Mobile Computing, vol. 353, pp. 153-181, 1996.
- [10] V.D. Park and M.S. Corson, “A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks,” Proc. IEEE INFOCOM '97, 1997.
- [11] W. Kiess and M. Mauve, “A Survey on Real-World Implementations of Mobile Ad-Hoc Networks,” Ad Hoc Networks, vol. 5, no. 3, pp. 324-339, Apr. 2007.
- [12] B.-R. Chen, M. Reddy, and M. Welsh, “Ad-Hoc Multicast Routing on Resource-Limited Sensor Nodes,” Proc. Second Int'l Workshop Multi-Hop Ad Hoc Networks from Theory to Reality, 2006.
- [13] A. Wheeler, “Commercial Applications of Wireless Sensor Networks Using ZigBee,” IEEE Comm. Magazine, vol. 45, pp. 70-77, Apr. 2007.
- [14] The Network Simulator, <http://www.isi.edu/nsnam/ns>, 2013.
- [15] J. Ko et al., “Connecting Low-Power and Lossy Networks to the Internet,” IEEE Comm. Magazine, vol. 49, no. 4, pp. 96-101, Apr. 2011.
- [16] P. Levis et al., “TinyOS: An Operating System for Wireless Sensor Networks,” Ambient Intelligence, Springer-verlag, 2005.
- [17] T. Kim, D. Kim, N. Park, S. Yoo, and T.S. Lopez, “Shortcut Tree Routing in ZigBee Networks,” Proc. Int'l Symp. Wireless Pervasive Computing (ISWPC), 2007.
- [18] A.V. Aho, J.E. Hopcroft, and J.D. Ullman, “On Finding Lowest Common Ancestors in Trees,” SIAM J. Computing, vol. 5, no. 1, pp. 115-132, 1976.
- [19] H. Lu et al., “A Distributed and Efficient Flooding Using 1-Hop Information” IEEE Trans. Parallel and Distributed Systems, vol. 18, no. 5, pp. 658-671, Apr. 2007.
- [20] Agbaria et al., “Efficient and Reliable Dissemination in Mobile Ad Hoc Networks by Location Extrapolation,” J. Computer Networks and Comm., 2011.
- [21] T. Kim, S. Kim, J. Yang and S. Yoo “Neighbor Table Based Shortcut Tree Routing in ZigBee Wireless Networks” IEEE Transactions on Parallel and Distributed systems, vol. 25, no. 3, March 2014.