Automatic Cluster Formation and Address Assignment for Wireless Sensor Network

[1]Prashant P.Rewagad, [2]Harshal K.Nemade
[1](HOD CSE GHRIEM, Jalgaon)
[2](M.E.CSE 2nd Year GHRIEM, Jalgaon)

Abstract - *Although wireless sensor networks (WSNs) have been extensively researched, their deployment is still a main concern. In this paper, we show that the address assignment scheme defined by ZigBee will perform poorly in terms of address utilization. This paper makes contribution is that we show that the Automatic Address Assignment. We thus propose Simple, Efficient. systematically formation of path connected cluster and Automatic Address Assignment for Wireless Sensor Network.*

Keywords - *Automatic Address assignment, Wireless Sensor Network, ZigBee.*

I. INTRODUCTION

To form a WSN, the most important issues is addressing. Strict per-node addressing is expensive in a dense network, because not only the address space would be large, but also these addresses would need to be allocated and managed according to the topology change. Allocation of addresses in a dense network is a problem which is often underestimated [11]. On the other hand, routing is to discover paths from source nodes to destination nodes based on their network addresses. Path discovery in a dense network could incur high communication overhands. Therefore, designing a light-weight addressing or Automatic Address Assignment for WSNs is very important.

Recently, ZigBee [12] has been proposed for addressing and routing on WSNs. It supports three kinds of network topologies, namely star, tree, and mesh networks. A *ZigBee coordinator* is responsible for initializing, maintaining, and controlling the network. Star networks can only cover small areas. For tree and mesh networks, communications can be conducted in a multi-hop fashion. The backbone of a tree/mesh network is formed by one ZigBee coordinator and multiple *ZigBee routers*. An *end device* must associate with the coordinator or a router. In a tree network, routing can be done in a stateless manner; a node can simply route packets based on nodes' 16-bit short addresses, which are assigned based on the tree structure. In fact, a mesh network also has a tree inside to serve as its backbone; routing can go directly along the tree without route discovery or go along better paths if a node is willing to conduct route discovery first.

In this work, we propose Automatic Address assignment for ZigBee-based WSN.

II. PRELIMINARIES AND PROBLEM DEFINITION

ZigBee Address Assignment

In ZigBee, network addresses are assigned to devices by a distributed address assignment scheme. Before forming a network, the coordinator determines the maximum number of children of a router (Cm), the maximum number of child routers of a router (Rm), and the depth of the network (Lm). Note that a child of a router can be a router or an end device, so $Cm \geq Rm$. The coordinator and routers can each have at most Rm child routers and at least $Cm - Rm$ child end devices. Devices' addresses are assigned in a top-down manner. For the coordinator, the whole address space is logically partitioned into $Rm + 1$ blocks. The first Rm blocks are to be assigned to the coordinator's child routers and the last block is reserved for the coordinator's own child end devices. From Cm, Rm, and Lm, each router computes a parameter called *Cskip* to derive the starting addresses of its children's address pools. The *Cskip* for the coordinator or a router in depth d is defined as:

$$Cskip(d) = \begin{cases} 1 + Cm \times (Lm - d - 1) & \text{if } Rm = 1 \\ \frac{1 + Cm - Rm - CmRm^{Lm-d-1}}{1 - Rm} & \text{otherwise.} \end{cases}$$
$$(1)$$

The coordinator is said to be at depth $d = 0$, and d is increased by one after each level. Address assignment begins from the ZigBee coordinator by assigning address 0 to itself. If a parent node at depth d has an address *Aparent*, the n-th child router is assigned to address $Aparent + (n − 1) \times Cskip(d) + 1$ and n-th child end device is assigned to address $Aparent + Rm \times Cskip(d)+n$. An example of the address assignment is shown in Fig. 1. The *Cskip* of the coordinator is obtained from Eq. (1) by setting $d = 0$, $Cm = 5$, $Rm = 4$, and $Lm = 2$.

Then the child routers of the coordinator will be assigned to addresses $0 + (1 − 1) \times 6 + 1 = 1$, $0 + (2 − 1) \times 6 + 1 = 7$, $0 + (3 − 1) \times 6 + 1 = 13$, and etc. The address of the only child end device of the coordinator is $0 + 4 \times 6 + 1 = 25$. Note that the length of a network address is 16 bits; thus, the maximum address capacity is $216 = 65536$. Obviously, the above assignment is much suitable for regular networks, but not for LT WSNs. For example, when setting $Cm = Rm = 2$, the depth of the network can only be 15. Also, when there is a LT backbone, the address space is not well utilized.
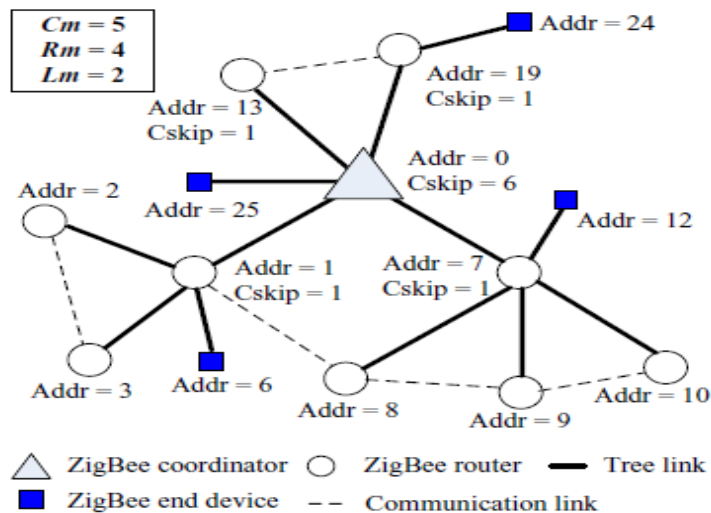

Figure.1 A ZigBee address assignment example

In a ZigBee network, the coordinator and routers can directly transmit packets along the tree without using any route discovery. When a device receives a packet, it first checks if it is the destination or one of its child end devices is the destination. If so, this device will accept the packet or forward this packet to the designated child. Otherwise, it forwards the packet to its parent. Assume that the depth of this device is d and its address is A. This packet is for one of its descendants if the destination address $Adest$ satisfies $A < Adest < A+ Cskip(d − 1)$, and this packet will be relayed to the child router with address

$$A_r = A + 1 + \left\lfloor \frac{A_{dest} - (A + 1)}{Cskip(d)} \right\rfloor \times Cskip(d).$$

If the destination is not a descendant of this device, this packet will be forwarded to its parent. In the ZigBee tree routing, each node can only choose its parent or child as the next node. This strategy may cause long delay in WSN networks.

III. AUTOMATIC ADDRESS ASSIGNMENT

We propose a low-cost, fully automated scheme to initialize it, assign addresses to nodes, and conduct ZigBee-like tree routing. First, a distributed network formation procedure will be launched by the coordinator t to divided nodes into two sets C and P. Then, a two-level address assignment scheme is conducted to assign a level-1 and a level-2 addresses to each node. A level-1 address is to uniquely identify a path or a cluster. A level-2 address is similar to ZigBee addressing but is confined within one cluster/path. For simplicity, we assume that all nodes are router-capable devices. Finally, we show how to conduct routing based on our two level addressing. Also, we address how our protocol can adapt to changeable topologies. We propose a two-level addressing. It has two purposes:

(i)   to reduce address space and
(ii)   to support ZigBee like stateless routing.

In level-1 addressing, we regard each cluster/path as a super node and use ZigBee-like addressing to assign an *m*-bit address to each supernode. In level-2 addressing, we again apply the ZigBee-like addressing on each individual cluster/path to assign an *n*-bit address to each node. The concatenation of the level-1 and the level-2 addresses forms a node *v*'s network address, denoted by *(L1(v), L2(v))*. During this process, we will also construct a *Descendant Table (DT)*, which allows an entry node to reach the entry nodes of its child super nodes.
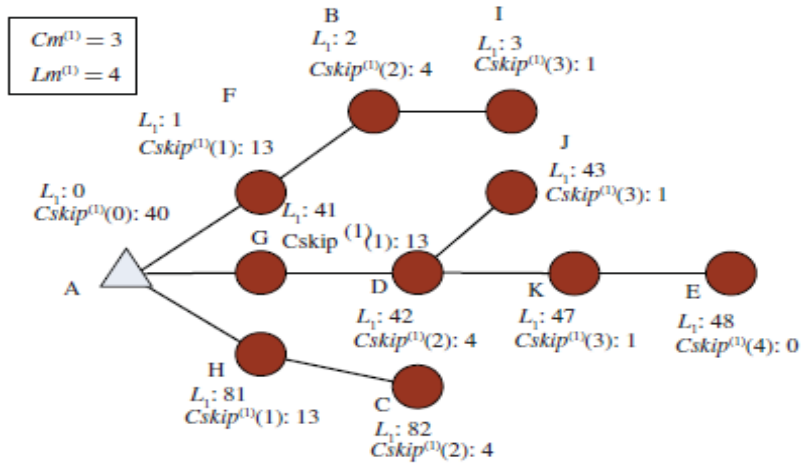

Figure 2 Logical Network

Before executing our two-level addressing, we need to determine *m* and *n* first. It relies on forming a *logical tree* from all supernodes and forming a BFS tree from all nodes of each super node. We summarize the steps as follows.

Step 1:) For each node *v* entering the TERM state and serving as an entry node, it reports the pair (*PAR(v)*, *GID(v)*) to *t*. From these information, *t* constructs a *logical tree TL* of super nodes. (For example, Fig. 5 is the logical network of Fig. 3.) With *TL* , *t* determines its tree parameters as $Cm(1) = Rm(1) =$ (max child degree of *TL*) and $Lm(1) =$ (height of *TL*). This would require $m = \log_2 M$ bits for addressing, where $M = \frac{Cm(1)^{Lm(1)+1}-1}{Cm(1)-1}$ is the maximal number of nodes in *TL*. (In the example of Fig. 2, $Cm(1) = Rm(1) = 3$ and $Lm(1) = 4$.)

Step 2:) For each node *v* entering the TERM state and serving as an entry node, it forms a BFS tree *T* among its members. With *T* , *v* determines its tree parameters as $Cm(2)(v) = Rm(2)(v) =$ (max child degree of *T*) and $Lm(2)(v) =$ (height of *T*). This would require $n(v) = \log_2 N$ bits for addressing, where $N = \frac{Cm(2)(v)^{Lm(2)(v)+1}-1}{Cm(2)(v)-1}$ is the maximal number of nodes in *T* . Each entry node *v* reports its $n(v)$ to *t*. Then *t* chooses the maximum one from all $n(v)$s as the value of *n*. The above process has determined *m*, *n*, and the tree formation parameters for level-1 and level-2. addressing. Next, *t* can start our two-level address assignment. It sets $L1(t) = L2(t) = 0$ and periodically broadcasts beacons containing *m*, *n*, *L1(t)*, *L2(t)*, the global level-1 parameters (*Cm(1)*, *Rm(1)*, *Lm(1)*), and its level-2 parameters (*Cm(2)(t)*, *Rm(2)(t)*, *Lm(2)(t)*). Our address assignment follows the ZigBee style recursively, but in a two-level manner. When a node *u* without a network address receives a beacon, it will send an *Association_Request* to the beacon sender. If it receives multiple beacons, the node with the strongest signal strength will be selected. When the beacon sender, say *v*, receives the association request, there are two cases:

Case 1: *u* and *v* belonging to different super nodes. If *u* is an entry node, *v* will process the request. v first unicasts the request packet to its entry node, say *e*, to retrieve a level-1 *m*bit address for *u*.If this is the *i* -th request received by *e*, it will respond the following level-1 address to *v*.

$$L1(u) = L1(v) + (i - 1) \times Cskip(1)(d) + 1, \quad\quad\quad\quad\quad (2)$$

where $Cskip(1)(d) = \frac{1-Cm(1)^{Lm(1)-d}}{1-Cm(1)}$. Then *v* replay an *Association_ Response* with an address (*L1(u)*, 0) to *u*. Note that for making our routing simple and efficient, the entry node *e* will also memorize this information (*L1(u)*, *L2(v)*) into its DT. The detailed functionality of the DT table will be addressed in Section III-C. On receipt of *v*'s response, *u* will start sending beacons containing *m*, *n*, *L1(u)*, *L2(u)*, the global level-1 parameters, and its own level-2 parameters (*Cm(2)(u)*, *Rm(2)(u)*, *Lm(2)(u)*).

Case 2: *u* and *v* belonging to the same supernode. If this is the *j* -th request received by *v*, it will respond the following level-2 address to *u*.

$L2(u) = L2(v) + (j − 1) \times Cskip(2)(d) + 1$, (3)

where $Cskip(2)(d) = 1−Cm(2)(e) \ Lm(2)(e)−d \ 1−Cm(2)(e)$

On receipt of *v*'s response, *u* will update its $L2(u)$ and set $L1(u) = L1(v)$. Then *u* will start sending beacons similar to case 1. In Fig. 2, we have shown a concrete subgraph of Fig. 1 with some assignment results, where each address is expressed in Hex and the first two symbols represent the *m*-bit address and the last two represent the *n*-bit address. Through Fig. 5, $Cm(1) = Rm(1)$ and $Lm(1)$ can be determined as 3 and 4, respectively. Hence, $m = 7$ bits and the $L1(x2) = 0 + 1−34 \ 1−3 \times 1 + 1 = 41$, where *x2* is the second child of *t*. Also, *n* is determined according to $n(x10)$, whose $Cm(2) = Rm(2) = 5$ and $Lm(2) = 3$, which is conducting the largest level-2 address space. Therefore, $n = 8$ bits and the level-2 address of the second child of *x10* is $0 + 1−53 \ 1−5 \times 1 + 1 = 32$. Other more addressing results will be given in Fig.2. Note that our two level addressing has better address space utilization than the pure ZigBee address assignment because each cluster/path can have its own network parameters.

IV. CONCLUSION

In this paper, we contribute in formally defining the Automatic Address Assignment Scheme. The Automatic address assignment scheme assigns each node both level-1 and level-2 addresses as its network address. We also show how to allow nodes to utilize shortcuts. With our design, not only network addresses can be efficiently utilized and the spaces required for the network addresses can be significantly reduced, but also the network scale can be enlarged to cover wider areas without suffering from address shortage. In the future, it deserves to consider applying this work to real cases such as environmental monitoring, military, ecology, agriculture, inventory control, robotics and health care.

REFERENCES

[1] F. Salvadori, M. de Campos, P. S. Sausen, R. F. de Camargo, C. Gehrke,C. Rech, M. A. Spohn, and A. C. Oliveira, "Monitoring in industrial systems using wireless sensor network with dynamic power management,"*IEEE Trans. Instrum. Meas.*, vol. 58, no. 9, pp. 3104–3111, Sep. 2009.

[2] C.-T. Cheng, C. K. Tse, and F. C. M. Lau, "A clustering algorithm for wireless sensor networks based on social insect colonies," *IEEE Sensors J.*, vol. 11, no. 3, pp. 711–721, Mar. 2011.

[4] C.-T. Cheng, C. K. Tse, and F. C. M. Lau, "A delay-aware data collection network structure for wireless sensor networks," *IEEE Sensors J.*, vol. 11, no. 3, pp. 699–710, Mar. 2011.

[5] A. Chen, T. H. Lai, and D. Xuan, "Measuring and guaranteeing quality of barrier-coverage in wireless sensor networks," in *Proc. ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2008, pp. 421–430.

[6] H.-S. Ahn and K. H. Ko, "Simple pedestrian localization algorithms based on distributed wireless sensor networks," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 4296–4302, Oct. 2009.

[7] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson,"Wireless sensor networks for habitat monitoring," in *Proc. ACM Int. Workshop Wireless Sensor Netw. Appl.*, 2002, pp. 88–97.

[8] *Design and Construction of a Wildfire Instrumentation System Using Networked Sensors* [Online]. Available: http://firebug.sourceforge.net/

[9] R. Casas, A. Marco, I. Plaza, Y. Garrido, and J. Falco, "ZigBee-based alarm system for pervasive healthcare in rural areas," *IET Commun.*, vol. 2, no. 2, pp. 208–214, Feb. 2008.

[10] Z. Sun, P. Wang, M. C. Vuran, M. A. Al-Rodhaan, A. M. Al-Dhelaan, and I. F. Akyildiz, "MISE-PIPE: Magnetic induction-based wireless sensor networks for underground pipeline monitoring," *Ad Hoc Netw.*, vol. 9, no. 3, pp. 218–227, 2011.

[11] S. Yoon, W. Ye, J. Heidemann, B. Littlefield, and C. Shahabi, "SWATS: Wireless sensor networks for steamflood and waterflood pipeline monitoring,"*IEEE Netw.*, vol. 25, no. 1, pp. 50–56, Jan.–Feb. 2011.

[12] C. Jing, D. Shu, and D. Gu, "Design of streetlight monitoring and control system based on wireless sensor networks," in *Proc. IEEE Conf. Ind. Electron. Appl.*, May 2007, pp. 57–62.

[13] H.-C. Lee, C.-J. Liu, J. Yang, J.-T. Huang, Y.-M. Fang, B.-J. Lee, and C.-T. King, "Using mobile wireless sensors for in-situ tracking of debris flows," in *Proc. ACM Int. Conf. Embedded Netw. Sensor Syst.*, 2008, p. 1.

[14] A. Chen, S. Kumar, and T. H. Lai, "Designing localized algorithms for barrier coverage," in *Proc. ACM Int. Conf. Mobile Comput. Netw.*, 2007, pp. 1–12.

[15] J. Kim, J. S. Lim, J. Friedman, U. Lee, L. Vieira, D. Rosso, M. Gerla, and M. B. Srivastava, "SewerSnort: A drifting sensor for in-situ sewer gas monitoring," in *Proc. IEEE Sensor Ad Hoc Commun. Netw. Conf.*, Jun. 2009, pp. 1–9.

[16] M. Ali and Z. A. Uzmi, "An energy-efficient node address naming scheme for wireless sensor networks," in *Proc. IEEE Int. Netw. Commun. Conf.*, Jun. 2004, pp. 25–30.

[17] E. Ould-Ahmed-Vall, D. M. Blough, B. S. Heck, and G. F. Riley,"Distributed unique global ID assignment for sensor networks," in *Proc. IEEE Int. Conf. Mobile Adhoc Sensor Syst.*, Nov. 2005, pp. 580–587.

[18] C. Schurgers, G. Kulkarni, and M. B. Srivastava, "Distributed ondemand address assignment in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 13, no. 10, pp. 1056–1065, Oct. 2002.

[`19] H. Zhou, M. W. Mutka, and L. M. Ni, "Reactive ID assignment for sensor networks," in *Proc. IEEE Int. Conf. Mobile Adhoc Sensor Syst.*, Nov. 2005, pp. 572–577.

[20] Y.-C. Tseng and C.-C. Tan, "Termination detection protocols for mobile distributed systems," *IEEE Trans. Parallel Distrib. Syst.*, vol. 12, no. 6, pp. 558–566, Jun. 2001.