IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 5, Issue 4 (Mar. - Apr. 2013), PP 46-50
www.iosrjournals.org

Network Formation, Addressing And Routing Protocol for Path-Connected-Cluster Wireless Sensor Network

Sulakshana.V[1], Ramesh.D[2]

[1]*(Electronics and Communication Engineering, Sudharsan Engineering College/Anna University, Chennai)*
[2]*(Electronics and Communication Engineering, Sudharsan Engineering College/Anna University, Chennai)*

***Abstract**: Wireless sensor networks are currently being employed in a variety of applications such as medical, military, home and industry. Sensor node deployment plays a vital role in WSNs to optimize the network performance. Protocol Zigbee is used in such networks. But the zigbee protocol has its own limitations. This paper we propose to Path-Connected-Cluster (PCC) topology and Distributed algorithm. This PCC includes Formation, addressing and routing. It's used to mitigate the concerns in node deployment. The networks which adopted PCC are called as PCC-WSN (Path Connected Cluster Wireless Sensor network). The Distributed algorithm is used to optimize the throughput and time delay of the entire network. This also known as PPR (PUSH PULL RELIABLE) algorithm.*

***Keywords -** Address assignment, protocol designs, routing, wireless sensor networks, ZigBee*

I. Introduction

Wireless Sensor Network (WSN) usually needs to configure itself automatically and support ad hoc routing. A lot of research works have been dedicated to WSNs, including power management [1], routing [2], data gathering [3], [4], sensor deployment and coverage issues [5]. On the application side, habitat monitoring is explored in [6], healthcare system is proposed in [7]. To form a WSN, two most important issues are addressing and routing. 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 [8]. 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 and routing protocol for WSNs is very important. Recently, ZigBee 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 paper path-connected-cluster (PCC*)* topology, where regions requiring intensive sensing are deployed with clusters of sensor nodes and these clusters, which are physically separated, are connected by long paths for occasional communications. We call such topologies PCC-WSNs. Although ZigBee supports address-based routing through its distributed addressing scheme, it could incur a lot of orphans or result in waste of address space. The virtual coordinate addressing schemes try to provide stateless routings directly from nodes' addresses. However, additional GPS devices or localization mechanisms should be involved. Moreover, these schemes still need a lot of address spaces.

The goal of our work is to propose an address-light and routing-light protocol for PCC-WSNs. Our approach is based on the principle of ZigBee address assignment, but leads to much more compact address usage than the original ZigBee's design, thus significantly alleviating the orphan problem in PCC-WSNs. Furthermore, based on our addressing, routing still incurs low communication overheads. This work contributes in formally defining the PCC-WSN topology. Given a PCC-WSN, we present a formation scheme to automatically separate paths from clusters in a distributed manner. Then we propose a ZigBee-like address assignment scheme for a PCC-WSN. In particular, we design different addressing strategies for slim parts (paths) and fat parts (clusters) of a PCC-WSN. This design allows us to conduct light-weight, address-based routing. Although this requires slight modification to ZigBee specification, we find this leads to quite efficient communications.

II. Network Formation, Addressing, And Routing Protocols

Given a PCC-WSN, 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.

2.1 Network formation

To form a network, the coordinator determines the maximum number of children per router (*Cm*), the maximum number of child routers per router (*Rm*), and the maximum depth of the network (*Lm*). Note that children of a router include child routers and child end devices by using eq. (1):

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

Given a PCC-WSN G = (*V*, *E*), the network formation process has three goals: (i) to partition G into clusters and paths, (ii) to assign a group ID (GID) to each cluster/path, which should be known to each member in that cluster/path, and (iii) to identify an entry node for each cluster/path, which is the one nearest to *t* in terms of the number of clusters/paths if we travel from *t* to the entry node (as a special case, *t* will serve as the entry node of its cluster). At this stage, addressing is based on devices' MAC addresses. It is assumed each node v has a unique MAC address MAC (v).The GID of a cluster/path will be the MAC address of its entry node. A network is to be created, First nodes are to be created and configured in fig 1. The routing is done in FOUR phases. INIT, CLF, PB, TERM. Initially, all nodes are in the INIT state.co- ordinator node is assigned for this state. Maximum number of children per router node is assigned as a Co- ordinator node co- co-ordinator node determines maximum number of child router per router and maximum depth of the network. The coordinator first enters the CLF state and starts the network formation process. Then the co-ordinator node tries to determine its degree (i.e., no. of neighbor nodes in total area) this can be achieved by nodes exchanging periodically HELLO packets. According to its neighbor node, node classifies its type 'cluster' or 'path'. If cluster degree is greater than or equal to 3, otherwise 'path' will set. If path is set, it needs to confirm its type If path is set, then we have to assign the clustering head (entry node) and edge node. And send a Transverse packet to calculate the length of its path.

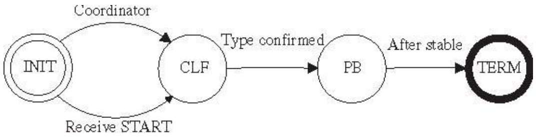

Fig 1. State transition diagram of a node

2.2 Address Assignment

A node once confirming its type as "cluster" or "path", will enter the PB state. To assign a group ID (GID) to each cluster/path, which should be known to each member in that cluster/path. To identify an entry node for each cluster/path. At this stage, addressing is based on MAC address. Each node has a unique MAC address.The GID of cluster/path will be MAC address of its entry node. Initially co- ordinator node assign 0 address value. 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 super node. 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 (*L*1(v), *L*2(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.

2.3 Routing

Based on our two-level addressing, given a source *x* and a destination *y* both in the same cluster/path, the distance $P^{(2)}(L2(x), L2(y))$ between them can be easily determined as follows:

$$P^{(2)}(L2(x), L2(y)) = D^{(2)}(L2(x)) + D^{(2)}(L2(y))$$

$$-2 \times D^{(2)} (LCA^{(2)}(\mathrm{L2(x), L2(y)})), \quad \text{--- (2)}$$

Where $D^{(2)}$ $(L2(x))$ is the depth of node x to its entry node, and $LCA^{(2)}$ $(L2(x), L2(y))$ is the least common ancestor's level-2 address of x and y. Also our addressing can further make our routing take shortcuts to improve the routing performance. Hence, after obtaining a network address, each node should periodically broadcast HELLO packets including its network address. In this work, we consider only symmetric links. A communication link (*u*, v) is established only if *u* receives v's HELLO including *u* as its neighbor and the HELLO's signal quality is above a threshold. Note that the signal quality should be the average of several packets. Then when routing a packet, each node will transmit the packet to a neighbor which has shortest tree path to the destination, which is determined by Eq. (2). Moreover, for making our routing more efficient, we further introduce Descendant Table (DT) table.

III. Proposed Algorithm

We propose a distributed algorithm to resolve the vertex-constraint maximum flow problem. The algorithm is named as PPR algorithm (PULL PUSH RELABEL algorithm). This algorithm contains three types of solutions to balance the traffic in the network. In this method, if the node u is overflowing with high traffic rate, then it looks for the possible successor node to handle the excess traffic. If the successor is found, then its height is determined. (The direction of the successor node from current node)If the current node's height is lower than the successor node, then the current node initiates PULL the traffic into successor. If the height of the current node is higher than the successor node, then the PUSH method is initiated to push the traffic to the successor. If the successor and current node are in the same height, then the RELABEL is initiated and either successor or current node increases its height and perform PULL or PUSH based on the height. This PULL PUSH RELABEL method considerably handle the excess traffic occurred in the network. So the throughput of the network is considerably increased and communication could not be interrupted.

3.1. Algorithm Steps:

step 1: for all u Є V
step 2: INIT (u)
step 3: While there exists any overflowing vertex
step 4: for all u Є V
step 5: PULL-PUSH-RELABEL (u)

For ease of presentation, Algorithm PPR is presented in a centralized manner. We explain how the algorithm can be implemented in a distributed manner by utilizing the innate mechanisms of ZigBee networks. In addition, to speed up the algorithm's convergence, we introduce the concept of parallel subsets. Vertices in the same parallel subset have neither direct links nor common neighbors. Consequently, the parallel subsets execute the algorithm sequentially, but the vertices in each parallel subset operate simultaneously so that the algorithm's correctness is ensured. Since the ZigBee architecture implicitly enables the formation of parallel subsets, parallel executions can be achieved seamlessly, and all of the algorithm's operations can be completed without any extra message exchange.

3.2 Properties of PPR Algorithm

The following properties are considered in the PPR Algorithm

1. During the execution of Algorithm PPR, if a vertex u Є V is overflowing performs a push or a relabel operation; otherwise, a pull operation is performed on u.
2. Whenever a vertex u performs a relabel operation on itself, its height h (u) increase by at least 1.
3. If h is initialized as a height function, then it remains a height function throughout Algorithm PPR.
4. Let f be a preflow in G and let h be a height function of f. Then, there will not be a path from the source s to the sink t in the residual network G_f.
5. The vertex-constraint maximum flow problem can be reduced to the traditional maximum flow problem.
6. For any overflowing vertex u, there must exist a path from u to the source s in the residual network G_f
7. During the execution of Algorithm PPR, $h(u) \le 2|V|$ always holds for any vertex u Є V.
8. During the execution of Algorithm PPR, any vertex performs at most $2|V|$ -1 relabel operations.
9. During the execution of Algorithm PPR, at least one relabel opration is performed during each pass.

When Algorithm PPR terminates, the preflow f is a maximum flow from the source s to the sink t in G. Next, we prove that the algorithm always terminates within some finite time. To normalize the algorithm's running time, we devise a specific time unit, called a pass, which represents iteration where every vertex performs the PULL-PUSH-RELABEL operation once.

IV. Performance Evaluations

We simulate some PCC-WSNs that are generated by a systematical method which has been developed by C language. In order to form a PCC-WSN in a systematic way, we impose a fail probability of *Pf* on each grid. If a grid is determined to fail, all sensor nodes inside the grid fail. This would partition the network into multiple sub networks when *Pf* is sufficiently large. The successful and adjacent grids will be grouped into the same cluster by our simulator. Then we apply a minimum spanning tree algorithm to build paths between clusters. Our simulator generates sensor nodes at every distance of *d* on each path. The coordinator *t* is at the left-top corner. Hence, we make the left-top grid always be not failed. Fig.2 shows an example of a random generated PCC-WSN. As Fig.2 shows, our protocol can partition the network nodes into 2 sets accurately. Also, each node can successfully connect to the network by our addressing assignment.

Based on the same determination principle of Cm, Rm, and Lm, ZigBee addressing can still make all nodes connect to the network as well. However, the address space conducted by ZigBee is excessively larger than that conducted by our protocol.

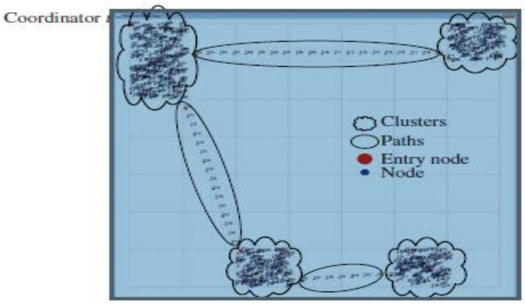

Fig. 2. Random generated PCC-WSN

The address space is mainly influenced by the values of Cm and Lm. Below; we will limit the address space. Also, we vary some factors such as the number of sensor nodes and the transmission ranges to investigate the performance.

This is because larger transmission range could increase the probability of no routing capacity for a router. Moreover, Fig. 3 shows address utility of both address assignment schemes.

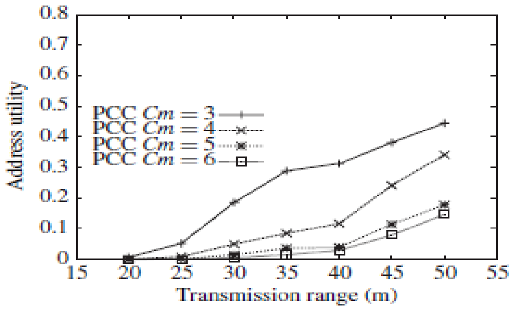

Fig 3. Addressing utility at fixed C_m values

Next, we limit the address pool of ZigBee as the one determined by our protocol. we vary Cm of ZigBee from 3 to 6. Fig. 4 shows the corresponding Lm of ZigBee on varying the Cm and fixing the address space. Obviously, ZigBee addressing incurs poor performance on the existence of paths. For reducing the influence of the paths on ZigBee addressing, we set Lm as the maximum depth of the network which is determined by processing a BFS scheme.

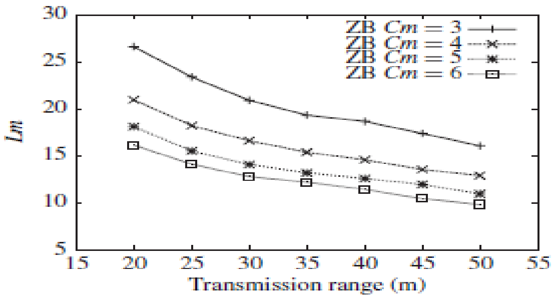

Fig. 4. Value of L_m of Zigbee addressing at various C_m values the same address space conducted by our addressing protocol

Here, we make the value of Cm at least as 2 even if the address space conducted by ZigBee may be larger than ours too much. Fig. 4 shows the results although this method makes ZigBee addressing have better performance than that used in Fig.4, ZigBee addressing still incurs many orphans.

Moreover, as Fig. 5 shows, the value of Lm will grow up to 80 when the transmission range is only 20 m. This means that the value of Cm only can be 2 and the address space will be up to 2^{80}.

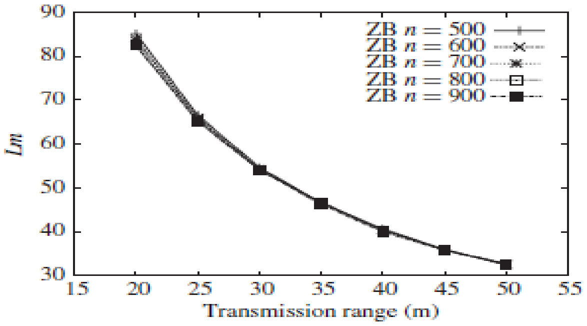

Fig. 5. Value of L_m of PCC addressing at max. depth of the network

Therefore, if we want to let all nodes connect to the network and increase the Cm of ZigBee, this will result in extremely larger address space and worse address utility.

V. Conclusion

In this paper, we contribute in formally defining the PCC-WSN topology. Also, we have proposed a formation scheme to divide nodes into several paths and clusters. Then a two-level ZigBee-like hierarchical address assignment and routing schemes for PCC-WSN are conducted. The PPR Algorithm also optimizes the throughput and time delay of the entire network. The proposed address assignment scheme assigns each node both level-1 and level-2 addresses as its network address. With such a hierarchical structure, routing can be easily done based on addresses of nodes. 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. We have also verified our schemes by simulation programs.

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. In strum. Meas., vol. 58, no. 9, pp. 3104–3111, Sep. 2009.

[2] H. Karkvandi, E. Pecht, and O. Yadid-Pecht, “Effective lifetime-aware routing in wireless sensor networks,” IEEE Sensors J., vol. 11, no. 12, pp. 3359–3367, Dec. 2011.

[3] 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] 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.

[7] 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.

[8] S. Kulkarni, A. Iyer, and C. Rosenberg, “An address-light, integrated MAC and routing protocol for wireless sensor networks,” *IEEE/ACM Trans. Netw.*, vol. 14, no. 4, pp. 793–806, Aug. 2006.