MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

A Structured Addressing Scheme for Wireless Multi-Hop Networks

Ghulam Bhatti, Gaofeng Yue

TR2005-149 June 2006

Abstract

The latest advances in wireless networking technologies and the growing popularity of these technologies have opened the door for exciting new applications. The deployment of such systems has attracted a wide variety of potential areas that were not feasible before. These areas include home and industrial automation, environment monitoring and sensing, security and surveillance, human and goods tracking, medical monitoring, and applications in defense relating scenarios. Obviously, the wireless systems are supposed to be robust, flexible, reliable, and secure ones.

Each node in a wireless network can communicate with nodes in its immediate neighborhood (that is within its transmission range). Obviously, each node must have a unique identity, called its network address, to facilitate peer-to-peer communication among the nodes. Due to the ad hoc, irregular, and spontaneous nature of the network topology, the address assignment becomes a non-trivial issue that needs to be addressed carefully. Several approaches for address assignment to the nodes in a wireless network have been reported. But most of these schemes suffer from one failure or the other to fully address the issue.

Here we propose a new address assignment scheme for wireless networks that may equally be applicable to wireless ad hoc networks, wireless sensors networks, wireless mesh network, and any other kind of wireless multi-hop networks. This scheme treats the address space as a regular shaped structure, i.e. an n-dimensional hyper-cube, which allows nodes to assign addresses to their child nodes in a systematic manner. In some addressing schemes such as one used by ZigBee, there is a restriction on the maximum number of levels that the addressing tree can have in the network. Our scheme removes such a restriction. Moreover, it brings in robustness against network partitioning when a few nodes fail. This is an improvement over ZigBee protocol, where failure of a single node partitions the underlying network tree.

MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

A Structured Addressing Scheme for Wireless Multi-Hop Networks

Ghulam Bhatti, Gaofeng Yue

TR2005-149 June 2006

Abstract

The latest advances in wireless networking technologies and the growing popularity of these technologies have opened the door for exciting new applications. The deployment of such systems has attracted a wide variety of potential areas that were not feasible before. These areas include home and industrial automation, environment monitoring and sensing, security and surveillance, human and goods tracking, medical monitoring, and applications in defense relating scenarios. Obviously, the wireless systems are supposed to be robust, flexible, reliable, and secure ones.

Each node in a wireless network can communicate with nodes in its immediate neighborhood (that is within its transmission range). Obviously, each node must have a unique identity, called its network address, to facilitate peer-to-peer communication among the nodes. Due to the ad hoc, irregular, and spontaneous nature of the network topology, the address assignment becomes a non-trivial issue that needs to be addressed carefully. Several approaches for address assignment to the nodes in a wireless network have been reported. But most of these schemes suffer from one failure or the other to fully address the issue.

Here we propose a new address assignment scheme for wireless networks that may equally be applicable to wireless ad hoc networks, wireless sensors networks, wireless mesh network, and any other kind of wireless multi-hop networks. This scheme treats the address space as a regular shaped structure, i.e. an n-dimensional hyper-cube, which allows nodes to assign addresses to their child nodes in a systematic manner. In some addressing schemes such as one used by ZigBee, there is a restriction on the maximum number of levels that the addressing tree can have in the network. Our scheme removes such a restriction. Moreover, it brings in robustness against network partitioning when a few nodes fail. This is an improvement over ZigBee protocol, where failure of a single node partitions the underlying network tree.

A Structured Addressing Scheme for Wireless Multi-Hop Networks

Ghulam Bhatti and Gaofeng Yue*
Mitsubishi Electric Research Labs
Cambridge, MA

1 Introduction

The latest advances in wireless networking technologies and the growing popularity of these technologies have opened the door for exciting new applications. The deployment of such systems has attracted a wide variety of potential areas that were not feasible before. These areas include home and industrial automation, environment monitoring and sensing, security and surveillance, human and goods tracking, medical monitoring, and applications in defense relating scenarios. Obviously, the wireless systems are supposed to be robust, flexible, reliable, and secure ones.

Each node in a wireless network can communicate with nodes in its immediate neighborhood (that is within its transmission range). Obviously, each node must have a unique identity, called its network address, to facilitate peer-to-peer communication among the nodes. Due to the ad hoc, irregular, and spontaneous nature of the network topology, the address assignment becomes a non-trivial issue that needs to be addressed carefully. Several approaches for address assignment to the nodes in a wireless network have been reported. But most of these schemes suffer from one failure or the other to fully address the issue.

Here we propose a new address assignment scheme for wireless networks that may equally be applicable to wireless ad hoc networks, wireless sensors networks, wireless mesh network, and any other kind of wireless multi-hop networks. This scheme treats the address space as a regular shaped structure, i.e. an n-dimensional hyper-cube, which allows nodes to assign addresses to their child nodes in a systematic manner. In some addressing schemes such as one used by ZigBee, there is a restriction on the maximum number of levels that the addressing tree can have in the network. Our scheme removes such a restriction. Moreover, it brings in robustness against network partitioning when a few nodes fail. This is an improvement over ZigBee protocol, where failure of a single node partitions the underlying network tree.

2 Prior Art

* This study was conducted during Gaofeng Yue's stay at MERL as an intern in summer 2005.

Address assignment in wireless ad hoc networks has recently been investigated extensively but the issue has not yet been addressed in a satisfying manner. Here we briefly overview several of the approaches proposed in literature to address this issue.

MANETConf [1] proposes a distributed dynamic address assignment protocol based on a mutual exclusion algorithm, whereby the joining node selects one of its neighbors as its initiator node. This initiator node will choose an address and broadcast a message to notify all other nodes in the network of this address. It then waits for responses from other nodes in the network, and if the replies from all the nodes are positive (which means no other node is using this chosen address), the initiator node will assign this address to the joining node. Otherwise, the initiator node will choose another address and the same process will repeat. Each node that has already been assigned an address (that is, it has properly joined the network) maintains information about all allocated addresses in the network. That will require significant amount of memory as the network grows. In order to make information at each node consistent, a lot of broadcast messages are generated. This approach therefore introduces a very high overhead cost due to the flooding of network with transmission of management frames. Moreover, the initiator node can choose an address of its own choice for its assignment to the requesting node. So, the address assignment does not follow a regular or systematic pattern. Authors in [2] use the concept of binary split, one of different Buddy System approaches, to assign addresses to nodes. Under this scheme, each node maintains a pool of addresses. When a node wants to join the network, it asks the nearest configured node, a node that has already been assigned an address, for an address. The configured node will split the pool of its own addresses into two equal halves, keep one half for itself and give the other half to the requesting node. Every node maintains an address table to keep record of the address blocks of all nodes. This involves each node periodically broadcasting its pool of addresses throughout the network, enabling every other node to update its address table in a timely manner. When a node uses up all addresses from its address block and other nodes may still have some available addresses, it can borrow some addresses from those nodes with available addresses. However, the result of this kind of behavior makes it possible one node can own a block of addresses which are not contiguous. Thus the address assignment does not maintain a systematic structure, thus mixing up all different blocks of addresses making the tracking of these blocks virtually impossible. That means that routes in such a network must be discovered and stored in tables, which may not be very suitable for wireless sensor nodes that are normally scarce in resources. Moreover, this approach requires the address tables be maintained at each node, it may not be suitable for nodes with limited resources. Perkins et al. [3] proposed another solution to the assignment of a unique address by performing duplicate address detection. A joining node selects two random addresses for itself, each from a different pool of addresses. One of these two are intended for the final assignment as its network address pending to the verification that it does not conflict with any other node. The other one is temporarily used as source address only during the verification process. The node broadcasts address request (AREQ) message. It waits for address reply (AREP) messages, which are sent back by only those nodes that are using this particular address. If no AREP is received, the node will claim this address. Also this approach employs a lot of broadcast messages and the address is randomly generated. The approaches above mentioned either introduce too much communication overhead by flooding the whole network or do not maintain a systematic structure for addresses.

ZigBee address assignment scheme, for example, assumes a hierarchical addressing tree structure. Under this scheme, a parent node allocates a segment of its own address space to each of its child node while the child node joins network. The first address is for the joining child node itself while the addresses in the remaining address space are reserved for its own child nodes. A special node, called ZigBee coordinator (ZC), which starts network, initially owns the whole address space. As new nodes join the network through ZC, it allocates chunks of address space to these nodes. Other nodes in the network follow the same procedure while allocating the address space to their own child nodes. Since the number of child nodes for ZC, or any other node for that matter, is fixed as one of the configuration parameters, it is possible to systematically determine the segment of address space that will be allocated to a joining node. Please note that a joining node selects its potential parent node during the join process. As a result of that, it is possible for a parent node (including ZC) that it may have grandchild nodes before it can have maximum number of child nodes of its own. In other words, the underlying network tree may not necessarily be a symmetric one. That, in fact, causes a problem. While ZigBee address assignment scheme is nice in the sense it has a distributed reliable mechanism that imposes a very low overhead cost, it is a static scheme in its nature and thus is not very flexible. First, it allows a maximum depth of 0xF for the underlying network tree. Secondly, it can waste chunks of address space if, for example, the physical location of the nodes is not compatible with the tree structure. In other words, a node may not be allowed to have a new child node (because it already has its max number of permitted child nodes and all of its address space already be allocated to these child nodes) even though a chunk of free addresses is available to one of the peer nodes that is out of transmission range of the requesting node. That causes waste of a precious network resource (i.e. the address space) that may become a serious problem under certain situations.

Here we propose a new address assignment scheme for wireless networks that may equally be applicable to wireless ad hoc networks, wireless sensors networks, wireless mesh network, and any kind of wireless multi-hop networks. This scheme has the following properties.

- Our scheme is distributed with low overhead cost. A node is required to keep information on small number of its neighbor nodes and needs to communicate with only a limited number of nodes while assigning an address to its new child.
- The addressing scheme is not restricted by the max number of hops (or length of the largest chain of parent-child node relationship starting at coordinator) in the network.
- It tries to limit the waste of address space if the physical location of the nodes is not uniformly distributed.
- It is robust enough to resist network partitioning caused by the failure of one or a few nodes.

- A node becomes orphan if its parent node breaks down. Orphan nodes cause problems to normal functioning of a network. In our addressing scheme, if the parent node is lost, the child node can possibly chose another neighbor node as its parent node while keeping its existing network address.

Although described in the context of ZigBee networks, the proposed addressing scheme can work for a general wireless ad hoc network.

3 Description

An address space under this scheme is not just one value as in ZigBee. Rather, the addresses are n-tuple values, where n indicates that n-dimensional hyper-cube is used. A node in the network can have a max of n child nodes. This pre-fixed value is, in fact, one of the configuration parameters defined before a network is actually launched. If logically viewed, the address space is organized along an n-dimensional hyper-cube. A network starts at the origin point (for example, (0, 0) when n is 2) and can grow in any direction or dimension as suited to the physical distribution of the nodes.

Although the approach can be generalized to any value of n, we will consider only 2 dimensions just for the sake of simplicity of illustration.

A 2 dimensional hyper-cube takes the form of a grid as shown in Figure 1. Each intersection or point in that grid has Cartesian coordinates. These coordinates make up the addresses of nodes. So, each node has an address that consists of two values, named as address components. The range of these values need not be the same and depends on how many bits are allocated to each address component. If the node addresses in a given network, for example, consist if 16 bits, the two components may be allocated 8 bits each thus making the range for each component 0 through 255 (i.e. 2^8 values). Similarly, the two address components can be allocated 10 and 6 bits, respectively. In that case the first address component has a range of 0 through 1023 while the second one has that of 0 through 63. In all cases, the total number of distinct addresses is 2^{16}. The address assignment to nodes works as explained below.

1. The first node, say N_1, that starts network gets an address at origin (that is, (0x0, 0x0)). In fact, the first addressed to be assigned could be any of the four corner addresses. Considering that each address component has been allocated 8 bits, the four corner addresses are (0x0, 0x0), (0x0, 0xFF), (0xFF, 0x0), and (0xFF, 0xFF). We have selected (0x0, 0x0) as the start address just as a matter of convenience.
2. An existing node can assign an address to its new child node as the following
    a. The address must be adjacent to its own address.
    b. The address being assigned must have a value for each address component that is either greater or equal to that for the corresponding component in its own address.

    So, for example, a new node, say N_2, joins the network as a child node of N_1. Then, N_2 will be assigned (0x1, 0x0) or (0x0, 0x1) as its address.

3. After a node gets its network address (x, y) from its parent node, other nodes that could use this address for a child node of their own must be informed that the address has already been used. The new node (x, y) will exactly do that right after it joins the network. So, only the nodes (x-1, y) and (x, y-1) are eligible to assign the address (x, y). One of these two nodes is the actual parent node of node (x, y). So, the other one must be informed by the joining node. In general, for an n-dimensional hyper-cube, only n-1 single hop frames will be transmitted.

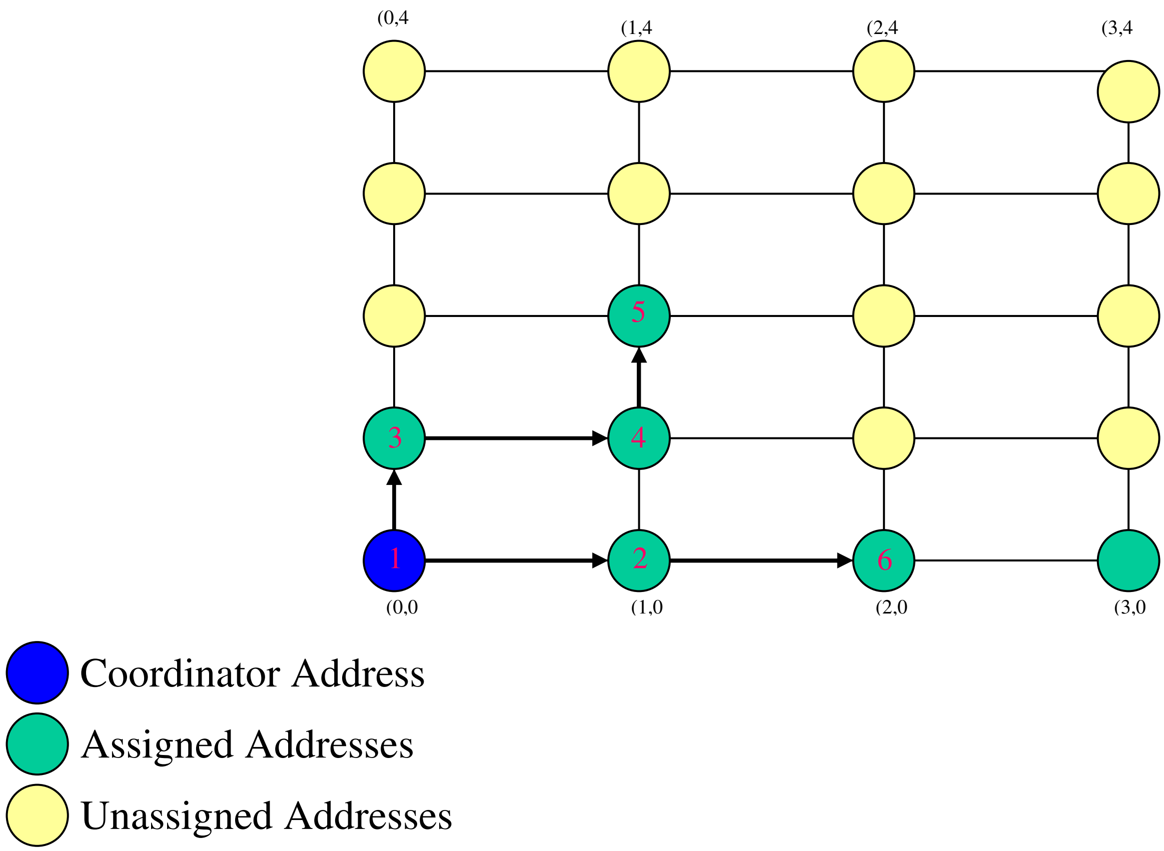

Figure 1

Going back to Figure 1, here we examine how a network grows when started by the first node that is normally called a network coordinator. As shown in the figure, the coordinator is labeled as the first node that assumes the address (0, 0). Now another node joins the network. It sends a join request to the coordinator, which assigns address (1, 0) to this node labeled as node 2 in the figure. It must be pointed out that the parent node (i.e. the coordinator in this case) had the choice of assigning either (1, 0), as it actually did, or (0, 1). It is possible for a system to actually specify that how a node chooses an address for assignment if it has multiple ones available at that time. In fact, we can force the address assignment patterns to facilitate or avoid straight chains of consecutive addresses being assigned along an axis. A simple rule, for example, may be that if both address-components are even or both are odd, the next address to be assigned should be along one axis (for example, along x-axis) if one is available. Otherwise assign the address along the other axis (e.g. along y-axis). This will force more uniform distribution of assigned addresses across the axes.

Now suppose another node wants to join the network as a child of coordinator. The coordinator will assign the only available address (i.e. (0, 1)) to this node. This was the third node that joined the network. As the next node joins the network, it creates an interesting situation. Suppose the new node sends a join request to Node 3 that assigns an address (1, 1) to this requesting node labeled as 4 in Figure 1. But this address could also be used by node 2 for its child and it, in fact, potentially could still try to assign that address (i.e. (1, 1)) to one of its child node. That scenario will definitely be problematic because two different nodes may be assigned the same address. In order to avoid that problem, node 2 must be informed that the address (1, 1) is no more available for it. That is where the item (3) as listed above applies. The parent of the new child node is responsible to inform the other nodes (that can potentially assign the same address to their own child node) that the address has been assigned and thus not available any more. Thus after a while node 2 gets its own child that is assigned address (2, 0).

4 Discussion

There are several proposed address assignment schemes as reported in literature. In most of those schemes, the route discovery is based on: either (i) each node is directly connected with all other nodes in the network, thus, allowing direct communication of packets between the nodes, or (ii) a node has to discover a route to the target remote node. The former approach is more common in ad hoc wireless networks that normally consist of resourceful nodes such as pocket PCs or PDAs. Moreover, this approach is not suitable for multi-hop wireless networks. The later approach, on the other hand, allows for the discovery of best possible route (based on, for example, hop count or link quality) between the nodes but it has significant overhead associated with it. This overhead is caused by (i) the extra network traffic that is generated while discovering a route, which is normally done by using broadcasting methods, (ii) transmission of extra frames is detrimental to one of the most precious resource in sensor nodes, that is, the battery power – transmission is very heavy on battery, and (iii) rather more importantly for sensors networks, the need for extra RAM that is required to store/maintain the routing tables, which can grow quite large in sizeable networks.

A better approach is to use the underlying logical structure of the network for routing purposes. In ZigBee networks, for example, underlying structure is a tree that is created by virtue of its addressing scheme that uses parent-child relationship. In such a network, a node can communicate with a remote node by sending frames along the tree, called tree routing. The basis of tree routing is that each node can determine if it needs to forward a packet, destined to a particular node, to its parent node or one of its child nodes and precisely which one. That determination of next hop node is done without transmitting a single frame in the air and without a need for a routing table. However, the drawback in

ZigBee tree routing is a broken tree. If a node breaks down, the network is essentially fragmented and only tree routing cannot simply be used.

Like ZigBee addressing scheme, our proposed approach facilitates a structured underlying topology, i.e. a hyper-cube, which can be used for routing purposes. If the nodes in the network have scarce resources and cannot afford to maintain routing tables, two remote nodes can still communicate by forwarding packets along the edges of the hyper-cube. Our approach is superior to tree routing because there may be multiple routes leading to the target node as opposed to only one path to each node in tree routing. But more importantly, our approach is much more robust and resistant to the possibility of network fragmentation. A network cannot be fragmented by the failure of a single node unless two parts of a network are connected with each other through only a single node.

References

[1] S. Nesargi, R. Prakash. “MANETconf: Configuration of Hosts in a Mobile Ad Hoc Network” in Proceedings of IEEE Infocom 2002.

[2] M. Mohsin, R. Prakash. “IP Address Assignment in a Mobile Ad Hoc Network” in Proceedings of Milicom 2002.

[3] Charles E. Perkins, J. T. Malinen, R. Wakikawa, E. M. Belding-Royer, and Y. Sun. “IP Address Autoconfiguration for Ad Hoc Networks”, IETF Internet Draft, draft-ietf-manet-autoconf-01.txt, November 2001