

Variable Length and Dynamic Addressing for Mobile Ad Hoc Networks

Som Chandra Neema Venkata Nishanth Lolla
{sneema,vlolla}@cs.ucr.edu
Computer Science Department
University of California, Riverside

Abstract

Most current ad hoc routing architectures use flat addressing and thus, need to keep track of each node individually, creating a massive overhead problem in the form of exchanging huge routing tables as the network grows. So, scalability of routing protocols for large-scale ad hoc networks is a concern. With fixed length addressing, the overhead in control packets of source routing is high. In this paper, we attempt to solve both the above-mentioned problems using a variable length and dynamic address allocation scheme, which aids routing in large-scale Mobile Ad Hoc Networks (MANETs). Each node has a permanent unique identifier, which is mapped to another routing address depending on the node to which it attaches to the network. The length of this routing address is variable but remains same for all the nodes in the network at one point of time. Our scheme allocates addresses in such a way that nearby nodes have same prefixes, thereby helping the routing protocols to make use of the location information of nodes from their addresses. Routing protocols that require source routes in the headers of every packet could benefit tremendously from the use of variable length addresses. Dynamic addressing can be leveraged by routing protocols to improve scalability.

1. Introduction

MANETs are composed of nodes, which form a peer-peer architecture. Due to lack of infrastructure, the peers should do all the operations like configuration of participating nodes and communication control by themselves. If MANET uses flat addressing scheme as in wired networks, it poses problems in routing for very large networks as the size of routing table at each node increases linearly.

A lot of research was done in developing routing protocols for ad hoc networks but none of them prove to be scalable for a network of more than a few hundred nodes. Scalability is a critical requirement in the use and deployment of ad hoc networks, if we want this technology to reach its full potential. The main reason behind the lack of scalability is that these protocols rely on flat and static addressing.

In [1], the authors propose an efficient dynamic addressing scheme. We identify that their addressing scheme does not work well for string topologies. They mention in the paper that string topologies of large length are extremely uncommon. But, consider a scenario

where a helicopter drops sensors in a battlefield. These sensors form an ad hoc network of string topology. Another example is a line of cars on a freeway forming a one-dimensional network. In this report, we present an addressing scheme, which works best in such scenarios along with random and uniformly distributed topologies. However, our scheme does not fare well in the case of a star topology of high degree, which is very uncommon in an ad hoc environment where nodes use omni directional antennas for transmission and reception of signals [1].

The remainder of this paper will discuss the related previous work in section 2 and some general requirements of the network layer in a MANET in section 3. We present our approach in section 4. Section 5 focuses on the efficiency of our algorithm. Finally, section 6 will summarize and provide some directions for future work in this area.

2. Related Previous Work

Popular IP-based ad hoc routing protocols like AODV [3] and DSR [4] use addresses as pure identifiers. These *reactive* protocols flood the network with route requests and route replies. The bigger the network, the larger the control overhead! A *proactive* routing protocol like DSDV [5] requires each node to maintain a routing table with entries corresponding to all other nodes in the network. So, both the alternatives are not scalable. LAR [6] uses geographic location information to assist in the routing and thereby tries to achieve scalability. However, this approach has serious limitations as information is not always available and can be misleading in nonplanar networks. Area Routing, as described by Kleinrock and Kamoun in [7], is the method most similar to the one used in today's Internet. Here, nodes that are close to each other in the network topology have similar addresses, without any explicit hierarchy of nodes.

In [1], the authors propose an efficient dynamic addressing scheme, which works well for random and distributed topologies for large-scale ad hoc networks. They use binary trees for address allocation consistent with the prefix sub graph constraint. We worked out the same scheme using K-ary trees with larger K values and quadtrees. In both the cases, the address space was used up quickly and the size of routing table at each node was of the order of $(K \log n)$ as opposed to $(\log n)$ using binary trees. Here n is the number of nodes in the network. However, their scheme does not work well for string and star topologies. In this paper, we propose a variable length and dynamic addressing scheme, which works well for string topology as well. The variable length addressing helps source routing by reducing the control overhead in the network.

3. Network Layer Requirements

The network layer in a MANET must provide the following capabilities as identified in [2]:

- 1) *Network Initiate*: allow a node or nodes to establish a network.
- 2) *Node Join*: allow nodes to join the network.
- 3) *Node Separate*: allow nodes to separate from or leave the network.
- 4) *Network Combine*: allow two previously independent MANETs to combine and form one MANET.
- 5) *Network Attach*: allow a MANET to attach to an existing wired network.

In our addressing scheme, we have considered all the above requirements except the last one. We discuss them in detail in the next section.

Dynamic Host Configuration Protocol (DHCP) [8] takes care of address assignment, leasing, aging, and reclamation in wired networks. A central server provides these services. In MANET, nodes may have a much more ad hoc and transitory nature that does not allow for set server assignment to provide services. A primary goal then is to provide services to MANETs that are realized by distributed and equally shared responsibilities for each node. This creates an environment where all nodes are truly peers.

4. Our Approach

In this section, we will explain our addressing scheme and routing using it. In a way, our design is similar to the one described in [1] but the address allocation is different. Hence, topological information used by the routing scheme also changes.

There are three major functions to be performed:

1. **Address allocation**- address indicates the node's network location.
2. **Routing**- delivers packets from a node to a given routing address.
3. **Node lookup**- is a distributed lookup table mapping every node identifier to its current network address.

Assumptions:

1. Nodes transmit and receive signals using omni-directional antennas. So, a high degree star topology is very uncommon.
2. The network environment is homogeneous, i.e., all the nodes have same power capabilities.

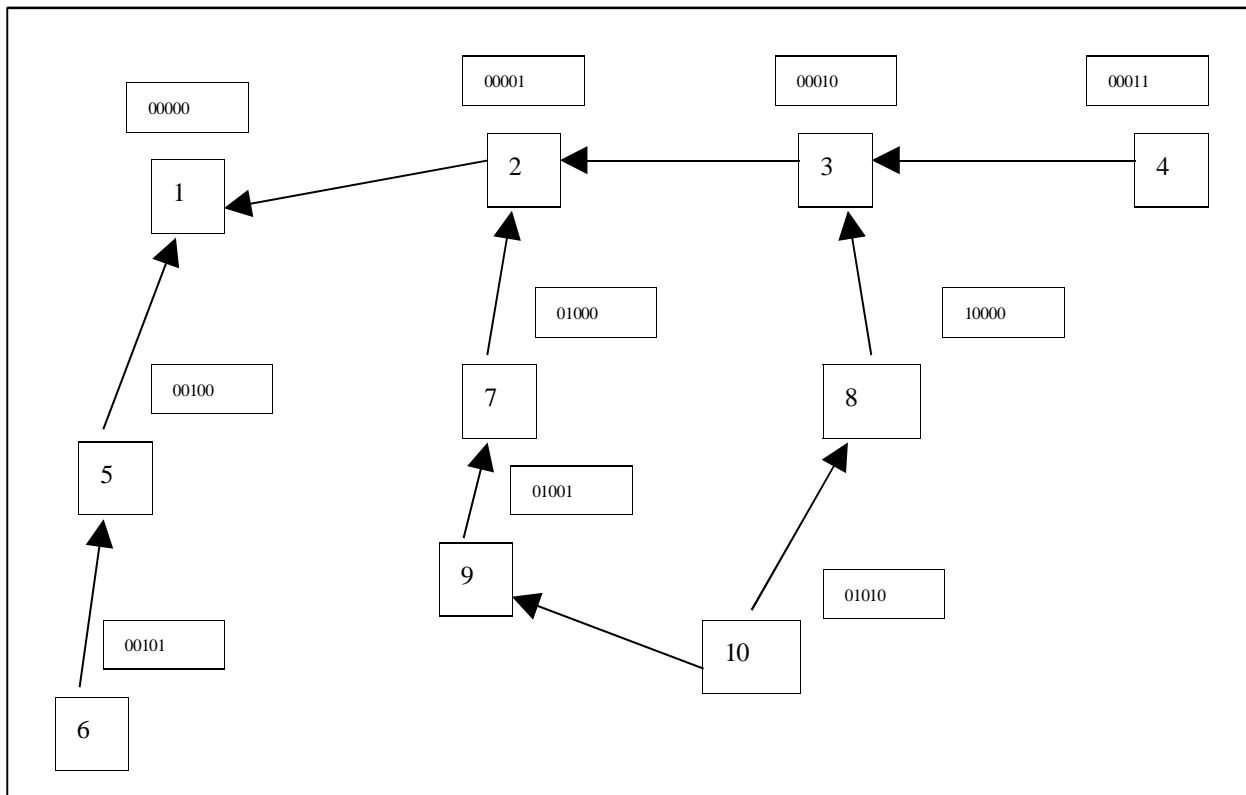


Figure 1 Assignment of addresses

Node	1	2	3	4	5	6	7	8	9	10
1	0	-	-	-	-	-	-	-	-	-
2	0	1	-	-	-	-	-	-	-	-
3	00	01	10	-	-	-	-	-	-	-
4	00	01	10	11	-	-	-	-	-	-
5	000	001	010	011	100	-	-	-	-	-
6	000	001	010	011	100	101	-	-	-	-
7	0000	0001	0010	0011	0100	0101	1000	-	-	-
8	00000	00001	00010	00011	00100	00101	01000	10000	-	-
9	00000	00001	00010	00011	00100	00101	01000	10000	01001	-
10	00000	00001	00010	00011	00100	00101	01000	10000	01001	01010

Table 1 Variable length addressing

Node Number	Free address Space	Number of free addresses
1	-	0
2	-	0
3	-	0
4	-	0
5	[00110-00111]	2
6	[00110-00111]	2
7	[01011-01111]	5
8	[10001-11111]	15
9	[01011-01111]	5
10	[01011-01111]	5

Table 2 Free addresses maintenance

We have considered a network of 10 nodes with random topology in figure 1. When a node joins the network, it listens to the periodic routing updates of its neighboring nodes, and uses these to identify an unoccupied address. The first node that comes in takes an address “0” and registers a free address “1” as available address. The second node that attaches to the network gets an address “1” looking at the routing updates from its neighbor. Here, we call the first node as the *creator* and the second node as *follower*. In general, any node that forces a bit increase in the address is called a creator. One may note that a node may be a creator or follower but not both. The third node will get to know that there are no free addresses available. So, it creates the address space by adding one more bit as a prefix for addressing. Now the previous two nodes get assigned “00” and “01” while the new node gets an address from the available free addresses (“10” or “11”). In our algorithm, the lowest address is assigned and the remaining addresses are maintained as free addresses. For example, in figure 1, node 10 hears the routing updates of nodes 8 and 9, but picks up the lowest address available. If there are no free addresses, the address space is expanded by adding a prefix “0” to the already existing addresses and the new node gets an address of the regular expression $[10^*]$ depending on the requirement of the address space. It is significant to note (from figure 1) that all the nodes in the network have the same address length at any point of time. This will lead to consistency in addressing.

The joining node registers its unique identifier and the newly obtained address in the distributed node lookup table. Due to mobility, the address may subsequently be changed and then the lookup table needs to be updated. When a node wants to send packets to a node known only by its identifier, it will use the lookup table to find its current address. Once the destination address is known the routing function takes care of the communication. The routing function should make use of the topological meaning that our routing addresses possess.

When a node leaves the network, it will relinquish the address and inform its followers of the free address spaces. If it happens to be the creator with no followers, it will inform all the nodes in the network to decrease the prefix by one bit, thereby, preventing wastage of addresses.

Table 1 demonstrates the way variable addressing works. The columns represent the addresses of all nodes in the network as each node enters the network in sequence as indicated by the rows. Table 2 represents the free address space available with each of the nodes.

Salient features of our algorithm:

- Variable length addressing can provide significant savings in network overhead. While not initially apparent, such savings can become substantial in light of source based routing protocols.
- Address reuse: Used up and available addresses are notified only to the local neighborhood (i.e., creator and its followers). If a node leaves the network and it happens to be the creator with no followers, it will inform all the nodes to decrease the prefix by one bit. This prevents wastage of unused addresses.
- When prefix is added or subtracted, it is done to all the nodes in the network. So all routing addresses in the distributed address lookup change and consistency is maintained.
- Address space is efficiently used:
 - Best-case: String topology
 - Average-case: Random and distributed topologies
 - Worst-case: Star topology
- When a node enters a network, it is treated as if two networks come in contact. Already existing nodes get the same prefix “0”, whereas the new node is treated as a different network. Hence, merging of different networks is just the special case of a node joining the network.
- Location based addressing is properly followed. So, the creator and its followers have the same prefix, which benefits hierarchical routing.
- The frequency of change in address length is high during the initial stages of ad hoc network configuration. However, as the network expands and stabilizes, it becomes less frequent because for every single bit increase in address length, the address space is doubled.

Scalable Routing using our addressing scheme:

In this work, we use a hierarchical form of proactive distance-vector routing. In its routing table, a node keeps one entry for each creator node with respect to the node's address. Intuitively, the routing entry for a creator indicates the next hop towards one of the followers of the creator. Clearly, the routing table will have at most k entries, where k is the length of the address.

In our example shown in figure 1, node 10 has routing entries for 1, 3, 5, 7 and 8, which are the creators. To route a packet to 6, node first determines the creator to which the destination address belongs to and then sends the packet to the neighbor closest to that creator. The process is repeated until the packet has reached the given destination address.

5. Efficiency of our algorithm

The size of routing table at each node is $O(\log n)$, where n is the number of nodes in the network. This can be derived from the fact that, with k bit addressing, there can be at most k creators and 2^k nodes in the network. As each node maintains the next hop to every creator in the network, it will have at most $\log n$ entries. This facilitates scalability for very large mobile ad hoc networks.

The frequency of change in address length is high during the initial stages of ad hoc network configuration. However, as the network expands and stabilizes, it becomes less frequent because for every single bit increase in address length, the address space is doubled.

6. Conclusions and future work

We have designed and implemented a variable length and dynamic addressing scheme for mobile ad hoc networks. It allocates addresses efficiently to different kinds of topology and helps routing in large-scale MANETs. In particular, it aids source routing due to variable length addressing. Simulations were not done to examine the performance of our scheme due to time constraints. Some areas needing work are specific to this addressing scheme and include the implementation of *routing* and *distributed node look up table*.

References

- [1] J. Eriksson, M. Faloutsos and S. Krishnamurthy. "Scalable Ad Hoc Routing: The Case for Dynamic Addressing".
- [2] J. Boleng. "Efficient Network Layer Addressing for Mobile Ad Hoc Networks".
- [3] C.E.Perkins and E.M.Royer, "Ad hoc on-demand distance vector routing," in *Proceedings of IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, 1999.
- [4] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad hoc Wireless Networks," in *Mobile Computing*, edited by T. Imielinski and H. Korth, pp 153-181, 1996.
- [5] C. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *ACM SIGCOMM*, 1994.
- [6] Y.B. Ko and N.H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," in *ACM/IEEE MobiCom*, 1998.
- [7] L. Kleinrock and F. Kamoun, "Hierarchical routing for large networks: Performance evaluation and optimization," *Computer Networks*, vol. 1, 1997.
- [8] Dynamic Host Configuration Protocol (DHCP): *RFC 2131*