

Adaptive Clustering with Virtual Subnets Support in Ad Hoc Networks

Tzu-Chiang Chiang^{1,2}, Ming-Hui Tsai³, and Yueh-Min Huang¹

¹ Department of Engineering Science, National Cheng-Kung University, Taiwan, ROC

² Department of Information Management, Hisng-Kuo University of Management, Taiwan

³ National Tainan First Senior High School, Taiwan, R.O.C

n9892105@mail.ncku.edu.tw, colux@tnfsh.tn.edu.tw,
huang@mail.ncku.edu.tw

Abstract. This paper concerns how the virtual subnet mechanism is engaged with the hierarchical architecture in ad hoc networks. The employment of virtual subnets performed at the Data Link Layer can prevent the reveal of broadcast storm and further improve the efficiency of traffic between members of specific group in ad hoc networks. A convenient method is proposed to select crucial backbone nodes which will furnish the ability of filtering frames in virtual subnet against the unnecessary flooding, termed CVA (Clustering with Virtual subnets support Algorithm). The chosen nodes include two different types, station nodes and port nodes, which act functionally as the switch and the ports of the switch to filter frames, respectively. We evaluate CVA with the metrics of average nodes in the backbone and duration of the protocols. The simulation result manifests the proper performance while supporting the virtual subnets in ad hoc networks.

1 Introduction

A self-organizing and self-configuring multihop wireless network is called a mobile ad hoc network (MANET) where the network structure changes dynamically due to the mobility of nodes. Nevertheless, a flat ad hoc network applied in such circumstance has poor scalability. Clustering is regarded as the familiar solution to provide the flat network with a hierarchical structure [5]. There are at least three benefits [2]. First, it can increase the system capacity by facilitating the spatial reuse of resources. Second, it can use bandwidth efficiently by spreading restrictedly the routing information in the set of nodes (clusterheads / clustergateways). Last, it can make an ad hoc network appear smaller and more stable in this hierarchical view. Oppositely, it also must suffer the cost of clustering, for instance, extra control message for clustering, computation load of nodes, re-clustering due to nodal movements, and so on.

Most of the backbone formation algorithms have the following two phases: the first phase (clustering) is to find a set of nodes as the *dominators* (or clusterheads) and the second phase is to find a set of nodes, called *connector*, to connect these dominators to form the connected dominating set. Such the connected set, also referred to *spine*, plays a very important role in routing, broadcasting and connectivity

management at layer 3 in wireless ad hoc networks [6, 7, and 9]. It integrates the clustering hierarchy to form a communication hierarchical structure that is practical in providing desirable properties such as minimizing communication overhead and increasing the probability of aggregating redundant data [1].

We aim to select the adaptive nodes in this hierarchical structure to form the virtual backbone and integrate virtual subnet filtering mechanism into an ad hoc network. Virtual subnets (IEEE 802.1Q) [17] can be used to effectively control broadcast traffic at layer 2 in wired circumstance. Broadcast traffic from servers and clients in a particular virtual subnet is propagated only on those switch ports connected to clients belonging to the same virtual subnet. Hence, we exercise the virtual backbone to establish virtual subnets as switches do in wireless circumstance. Some nodes of virtual backbone act functionally as the physical ports of switches for filtering frames.

The paper is organized as follows. In Section 2 we give a quite intensive overview of distributed clustering algorithm, backbone formation and virtual subnet architecture. Section 3 presents our convenient backbone formation algorithm with the ease of virtual subnet membership maintenance. In Section 4 we describe in details our simulation environment and we evaluate and comment on the performance of the protocols with respect to the selected metrics of interest. Finally, Section 5 concludes the paper.

2 Related Works

2.1 Clustering

Clustering for ad hoc networks has been investigated for over twenty years. Many clustering algorithm had been proposed including [8, 10, and 13]. Clusterhead selection and backbone formation, although differently from the methods used in the mentioned solutions, are also the two fundamental steps in [11]. Moreover, regarding the backbone formation, most of the proposed protocols end up generating a clustering and a corresponding backbone whose nodes form a dominating set of nodes [1], [3]. Clusterheads are joined through gateway nodes to form a backbone that is a *connected dominating set* (CDS) [6, 7, 14, and 16]. For instance, [12] is a simple distributed procedure for creating the desired CDS efficiently but tends to create a very large CDS. Trade-off between message complexity and size of CDS exists evidently. A different approach to create a small CDS is to compute a *maximal independent set* (MIS) first and connect it up [4, 14, and 15].

2.2 Virtual Subnet

In wired networks, a switch is a device responsible for multiple functions such as flooding, sending frames, and filtering. Layer 2 switching is *hardware* based, which means it uses the MAC address from the host's NIC cards to filter the frames. The layer 2 switches check the frames' hardware addresses instead of the Network layer (layer 3) header information before deciding to either forward or drop them. There are two basic functions at layer 2 switch.

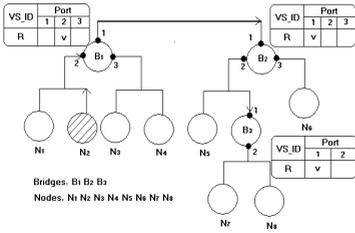


Fig. 1. (left Fig) N2 declares its initiation with virtual subnet R

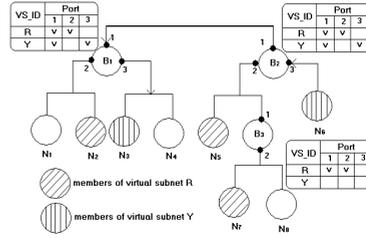


Fig. 2. (right Fig) N6 sends frames to its members of virtual subnet Y

- (1) Address learning: they can remember the source hardware address of each frame received on an interface and enter this information into a MAC database.
- (2) Forward/filter decisions: when a frame is received on an interface, they look at the destination hardware address and find the exit interface in the MAC database.

The detailed functionality of the virtual subnet refer to the standard of IEEE 802.1Q. Only frames that are destined for addresses outside the virtual subnet need to proceed to switch ports for forwarding. We illustrate the concept of virtual subnets with the both Figures 1 and 2.

Whiles N2 declares its initiation into virtual subnet R, the B1 (a switch) updates its filtering database at port 2 to maintain the virtual subnet membership. This declaration is then forwarded and excites the update of the filtering database in B2 and B3 as the same way. See Figure 1. Figure 2 also illustrated, N2, N5, N7 and N3, N6 are the members of virtual subnet R and virtual subnet Y, respectively. If N6 sends a packet to its members of virtual subnet Y, both port 2 of B2 and B1 do not relay this packet according to the filtering database.

3 The Virtual Subnet Cluster Architecture

Our concerns are how to automatically segment these mobile nodes into the correct virtual subnets and how to prevent the unnecessary broadcasts from interfering with the other subnets. In wired networks, however, virtual subnet membership is typically defined by the physical Layer 2 switch port to which a node/device is connected. In wireless mobile networks, all nodes act as Layer 3 router but are not lightly tied to a physical port of router.

In order to implement the filtering functionality of virtual subnets, some nodes chosen by the underlying backbone formation algorithm must be able to maintain the virtual subnet filtering tables and act as the switch ports. The chosen nodes include two different types, station nodes and port nodes, which act functionally as the switch and the ports of the switch, respectively. Therefore, we first present the clustering and backbone formation for supporting virtual subnets (Clustering with Virtual subnets support Algorithm, termed as CVA), that is to say, the selection of station and port nodes after clustering. As for the implementation of virtual subnet functionality within the hierarchical structure and the maintenance of virtual subnet memberships, they are then presented as the following.

3.1 Clustering and Backbone Formation

We aim to select the adaptive nodes to form the virtual backbone and act as a virtual subnet filtering devices no matter which clustering algorithm is adopted. The main reason that we adopt DCA here to partition networks is its advantages in terms of low communication complexity. During clustering and consequent backbone formation, there are three types of backbone nodes:

- (1) Participant nodes are candidates for station/port nodes whether their *order* is greater than two. (Order, i.e. how many clusters a node can access one hop away)
- (2) Port nodes that filter the subnet frames depending on its filtering database.
- (3) Station nodes that just forward the virtual subnets frames without filtering functionality or act as port nodes in the specific condition described later.

The chosen nodes would be station nodes or port nodes depending on their physical connection status. For avoiding the performance degradation by the excess filtering verification occurring at an oversized backbone, how many nodes should be determined as station/port nodes in this hierarchical clustering architecture will affect significantly the performance of the whole networks especially throughput.

A node will be set to “participant” node while it can connect more than two clusters. Note that the order of each node could be lightly and rapidly counted after clustering. After sending/receiving the broadcasts to/from neighbors, a node can claim itself as a station node only if it has the maximum order more than all its neighbors have or the same maximum order among its neighbors but with the minimum ID. And this station node then broadcast its claim to neighbors in order to set adjacent participant nodes to port nodes. It is worth to mention that if a port node really attached to a station node and received another station node claim, it would transform its state into station. In addition, some station nodes act as port nodes while they can connect not only station/port nodes but cluster members close to them. All parallel connected station nodes are regarded as a virtual switch/router to support virtual subnet by its joint port nodes.

Consider the topology snapshot and cluster architecture in Figure 3 and Figure 4, respectively. There are six clusters in an ad hoc network and all of the participant nodes are marked (which possibly but not certainly be selected as station/port nodes later, indicated as \diamond). Determination whether a node is a station/port node from those participant nodes is following our backbone formation algorithm. As shown in Figure 5, station nodes are indicated as \odot and port nodes as \square .

First, node 4, 6, and 20 have transformed as station nodes. Participant nodes close to them such as node 2, 3, 5, 9, 14, and 16 should transform their state into “port”. Note that the node 5 and node 16, both of the nodes receiving the dissemination from their adjacent station nodes (node 4, 20 and node 6, 20, respectively) transform their original state “port” into “station”. Despite that node 18 have been previously set to participant node, it have not ever transformed as station/port node. It is just a cluster member affiliated to its clusterhead (node 3). After the backbone formation, there are five station nodes and four port nodes in the system, which are {4, 5, 6, 16, 20} and {2, 3, 9, 14} respectively. All the parallel connected station nodes are regarded as a virtual switch to support virtual subnet by its joint port nodes, as shown in Figure 6.

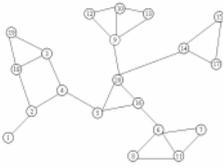


Fig. 3. Topology

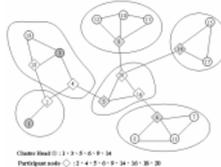


Fig. 4. DCA clustering

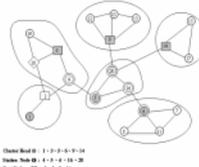


Fig. 5. Backbone formation

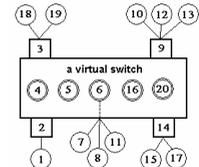


Fig. 6. Parallel connected station nodes are regarded as a virtual switch

3.2 Virtual Subnet Membership Maintenance

Virtual subnets configured by using Media Access Control addresses (or node ID) can recognize nodes even that has changed their affiliation with original virtual subnet. The virtual subnet membership maintenance should automatically reconfigure when a node transforms its affiliation into an appropriate virtual subnet without the need of changing the node's MAC address. The communication among virtual subnets in our system will be forwarded without any filtering functionality by some station node (for example, node 4, 5, 16, and 20 in Figure 6) whose virtual subnet filtering table (VSFT) is null. On the contrary, the others of station nodes (for example, node 6 in Figure 6) acting as port nodes are not always permeable for all virtual subnet frames depending on their VSFTs. In addition, it is necessary for all port nodes (for example, node 2, 3, 9, and 14 in Figure 6) to maintain the virtual subnets membership. There are two basic kind of virtual subnet management to be interpreted:

3.2.1 Join in a Virtual Subnet

While a node declares initiation into a virtual subnet or transforms its original affiliation into a new certain virtual subnet, it has to send the virtual subnet join message to it's accompany clusterhead first in consequence of the hierarchical organization. Its clusterhead forwards (broadcasts) the message to the adjacent backbone node to maintain the virtual subnet membership. If the clusterhead is station/port (backbone) node, the forwarding would not be processed. The backbone node updates subsequently its VSFT to enable the certain virtual subnet frames to pass through from outside in accordance with the received join message. The virtual subnet join message contains the following fields:

$\langle node_id, virtual_subnet_id, sequence_#\rangle$

For instance, while node 8 disseminates its initiation into a virtual subnet R (VS_R), this affiliation information will be send to its clusterhead node 6 first and this station node subsequently update its virtual subnet filtering table (VSFT) to enable the filtering function, refer to Figure 6 and Table 1. In addition, associated with virtual subnet ID entries is a timer, called the membership expiration timer. The purpose of this timer is to purge the stale virtual subnet membership. Each time a frame tagged to VS_R is forwarded through this node and the time out for the associated membership entry is reset to the current time plus a constant expiration time parameter τ (i.e., called as *ageing time* in IEEE 802.1Q).

Table 1. Virtual Subnet Filtering Table of node 6

Virtual Subnet Filtering Table	
VID	Membership Expiration Time (s)
R	$current\ time^* + \tau$

3.2.2 Depart from a Virtual Subnet

There is no specific message which is disseminated from a node to depart from a virtual subnet. The stale entries of VSFT in station/port nodes are automatically removed depending on their individual ageing time. As mentioned above, whenever a virtual subnet frame passes through the station/port nodes, the timer of those corresponding entries is extended. The dynamically ageing out of filtering entries ensures that nodes that have been affiliated with a different virtual subnet will not be permanently prevented from receiving frames.

For example, the VSFT of node 14 is shown as Table 2. It indicates that frames tagged for both of virtual subnet R and B can be forwarded bidirectionally through this node. Whenever node 14 receives a frame tagged for virtual subnet R, it checks the filtering entry first to determine whether the entry is expired. If not, the frame is successfully forwarded and the ageing time is reset. Oppositely if the entry is expired, the frame tagged for VS_B is filtering out and the entry is removed from the VSFT.

4 Simulation Analysis

The adaptive clustering with virtual subnets support architecture has been evaluated using the VINT project network simulator (ns2) [18]. We also advanced C++ Partition Toolkit (CCPT) to determine the nodes affiliation with virtual subnets in a clustering ad hoc network. Our simulations refer to static scenarios in which n wireless nodes with maximum transmission radius of 30 meters and the number of nodes n has been assigned the values between 20 and 200 with an interval of 20. All nodes are initially placed randomly within a predefined 200m x 200m grid area. In particular, we have considered the following metrics:

- (1) The average percentage of nodes in the backbone.
- (2) The average time (duration) needed by the protocol to complete the clustering and backbone formation.

Backbone Sizes

We choose the algorithms of Wan et al [11], Wu and Li [17], DCA [16], and our CVA to evaluate the backbone size. A smaller backbone improves network performance not only with respect to control overhead but also on account of a smaller number of nodes needed to involve in the data transmission. The average percentage of nodes in the backbones is shown in Figure 8. All the curves present that the percentage of nodes in the backbone decreases as n increases: as the higher density of network, the cluster size grows into larger scale and the selection of connectors become ampler. Our protocol curve is close to the DCA's as the result of clustering algorithm.

Protocol Duration

It show the result of the average time needed by the above four mentioned protocols to complete their operations in Figure 9. The more complex of protocol is, the longer duration of backbone formation is needed. This is due to the higher complexity, the raised number of information that each node needs to exchange with its neighbors, and the increasing possibility of communication collision. On the other side, the clusterheads selection of DCA is performed based on a generic weight associated to a node, so the operations require a node to wait for all its neighbors with maximum weight to communicate their role before it can decide its own. Our protocol show reasonably good performance, requiring a time which ranges from 1.44s when $n = 40$ to 5.53s when $n = 200$.

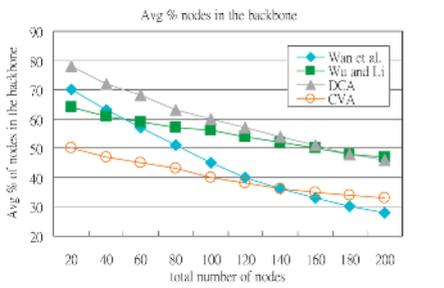


Fig. 7. Percentage of backbone nodes

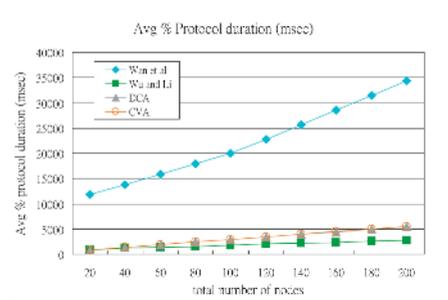


Fig. 8. Duration of the protocols

5 Conclusions

This paper represented an ingenious algorithm of backbone formation which is elaborately fitting for adopting the mechanism of virtual subnet membership maintenance, termed as CVA. Our protocol partitions an ad hoc network into physical clusters on its physical topology and partitions the network into logically virtual subnets on independent nodes' affiliation. The communication among the same virtual subnet will decrease the interference with irrelative clusters and can significantly improve the networks' performance. Topics for further research include the investigation of backbone nodes maintenance in presence of mobility.

References

1. S. Basagni, M. Mastrogiovanni and C. Petrioli, "A Performance Comparison of Protocols for Clustering and Backbone Formation in Large Scale Ad Hoc Networks," the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems, October 24-27, 2004, Fort Lauderdale, Florida, USA.
2. Jane Y. Yu and Peter H.J. Chong, "A Survey of Clustering Schemes for Mobile Ad Hoc Networks", IEEE Communications Surveys & Tutorials First Quarter 2005, Volume 7, No.1, pp. 32-48.

3. B. Liang and Z. Haas, "Virtual backbone generation and maintenance in ad hoc network mobility management," in Proceedings of the 19th IEEE Infocom, vol. 3, Tel Aviv, Israel, March 26–30 2000, pp. 1293–1302.
4. I. Stojmenovic, M. Seddigh, and J. Zunic, "Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks," IEEE Transactions on Parallel and Distributed Systems, vol. 13, no. 1, pp. 14–25, 2002.
5. Y.M. Huang and Tzu-Chinag Chiang, "An Efficient Encoding Tree for Location-Aware Multicast Routing Protocol in Wireless Ad Hoc Networks", Journal of Internet Technology, Vol. 6, No. 1, 2005.
6. B. Das and V. Bhargavan, "Routing in ad-hoc networks using minimum connected dominating sets," IEEE International Conference on Communications (ICC '97), June 1997.
7. B. Dss, E. Sivakumar, and V. Bhargavan, "Routing in ad-hoc networks using a virtual backbone," Proceedings of the 6th International Conference on Computer Communications and Networks(IC3N '97), pages 1 - 20, Sept. 1997.
8. C. C. Chiang, H. K. Wu, W. Liu and M. Gerla, "Routing in Clustered Multihop, Mobile Wireless Networks with Fading Channel," Proc. IEEE Singapore international Conference on Networks (SICON97), 1997, pp. 197-211
9. Ming-Hui Tsai, Tzu-Chinag Chiang and Y.M. Huang, "On Scalability and Mobility Management of Hierarchical Large-Scale Ad Hoc Networks", Lecture Notes in Computer Science, Volume 3823 ,pp. 714 - 723,12/2005.
10. C. R. Lin, M. Gerla, "Adaptive Clustering for Mobile Wireless Networks," IEEE Jour. Selected Areas in Communications, Sept. 1997, pp. 1265-1275.
11. P.-J. Wan, K. M. Alzoubi, and O. Frieder, "Distributed construction of connected dominating sets in wireless ad hoc networks," in proceedings of INFOCOM 2002.
12. J. Wu and H. Li, "On calculating connected dominating sets for efficient routing in ad hoc wireless networks," Telecommunication Systems, Special Issue on Mobile Computing and Wireless Networks, vol. 18, no. 1/3, pp. 13–36, September 2001.
13. F. Garcia Nocetti, J. Solano Gonzales, and I. Stojmenovic, "Connectivity based k -hop clustering in wireless networks," TelecommunicationSystems, vol. 22, no. 1–4, pp. 205–220, 200
14. D. Dubhashi, A. Mei, A. Panconesi, J. Radhakrishnan, and A. Srinivasan, "Fast distributed algorithms for (weakly) connected dominating sets and linear-size skeletons," in Proceedings of the 14th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), Baltimore, MD, 12–14 January 2003, pp. 717–724.
15. S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," Algorithmica, vol. 20, no. 4, pp. 374–387, April 1998.
16. I. Chlamtac and A. Faragó, "A new approach to the design and analysis of peer-to-peer mobile networks," Wireless Networks, vol. 5, no. 3, pp. 149–156, May 1999.
17. IEEE draft standard for virtual bridged local area networks, 802.1Q/D11 July.30, 1998.
18. The VINT Project, The ns Manual. <http://www.isi.edu/nsnam/ns/>,2002.