Electronic Theses and Dissertations, 2004-2019

2011

Virtual Router Approach For Wireless Ad Hoc Networks

Ai Hua Ho
University of Central Florida

University of Central Florida

STARS
Showcase of Text, Archives, Research & Scholarship

VIRTUAL ROUTER APPROACH FOR WIRELESS AD HOC NETWORKS

by

AI HUA HO
B.S. University of Central Florida, 2001
M.S. University of Central Florida, 2005

A dissertation submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy
in the Department of Electrical Engineering and Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2011

Major Professor: Kien A. Hua

ABSTRACT

Wireless networks have become increasingly popular in recent years. There are two variations of mobile wireless networks: infrastructure mobile networks and infrastructureless mobile networks. The latter are also known as *mobile ad hoc network* (MANET). MANETs have no fixed routers. Instead, mobile nodes function as relay nodes or routers, which discover and maintain communication connections between source nodes and destination nodes for various data transmission sessions. In other words, an MANET is a self-organizing multi-hop wireless network in which all nodes within a given geographical area participate in the routing and data forwarding process. Such networks are scalable and self-healing. They support mobile applications where an infrastructure is either not available (e.g., rescue operations and underground networks) or not desirable (e.g., harsh industrial environments).

In many ad hoc networks such as vehicular networks, links among nodes change constantly and rapidly due to high node speed. Maintaining communication links of an established communication path that extends between source and destination nodes is a significant challenge in mobile ad hoc networks due to movement of the mobile nodes. In particular, such communication links are often broken under a high mobility environment. Communication links can also be broken by obstacles such as buildings in a street environment that block radio signal. In a street environment, obstacles and fast moving nodes result in a very short window of communication between nodes on different streets. Although a new communication route can be established when a break in the communication path occurs, repeatedly reestablishing new routes incurs delay and substantial overhead. To address this

limitation, we introduce the *Virtual Router* abstraction in this dissertation. A virtual router is a dynamically-created logical router that is associated with a particular geographical area. Its routing functionality is provided by the physical nodes (i.e., mobile devices) currently within the geographical region served by the virtual router. These physical nodes take turns in forwarding data packets for the virtual router. In this environment, data packets are transmitted from a source node to a destination node over a series of virtual routers. Since virtual routers do not move, this scheme is much less susceptible to node mobility.

There can be two virtual router approaches: *Static Virtual Router* (SVR) and *Dynamic Virtual Router* (DVR). In SVR, the virtual routers are predetermined and shared by all communication sessions over time. This scheme requires each mobile node to have a map of the virtual routers, and use a *global positioning system* (GPS) to determine if the node is within the geographical region of a given router. DVR is different from SVR with the following distinctions: (1) virtual routers are dynamically created for each communication sessions as needed, and deprecated after their use; (2) mobile nodes do not need to have a GPS; and (3) mobile nodes do not need to know whereabouts of the virtual routers.

In this dissertation, we apply Virtual Router approach to address mobility challenges in routing data. We first propose a data routing protocol that uses SVR to overcome the extreme fast topology change in a street environment. We then propose a routing protocol that does not require node locations by adapting a DVR approach. We also explore how the Virtual Router Approach can reduce the overhead associated with initial route or location requests used by many existing routing protocols to find a destination. An initial request for a destination is expensive

because all the nodes need to be reached to locate the destination. We propose two broadcast protocols; one in an open terrain environment and the other in a street environment. Both broadcast protocols apply SVR. We provide simulation results to demonstrate the effectiveness of the proposed protocols in handling high mobility. They show Virtual Router approach can achieve several times better performance than traditional routing and broadcast approach based on physical routers (i.e., relay nodes).

ACKNOWLEDGMENTS

Getting a PhD degree is one of the most significant accomplishments in my life. I am grateful to have many wonderful people to help me through this journey. I take this opportunity to thank those who help me during the process of achieving this accomplishment.

As my advisor, Prof. Kien A. Hua receives my utmost gratitude. He provided me many opportunities from which I learned much about and discovered my passion for computer science. Through working on projects on research assistantships, I gained valuable experience working in group projects. Through lab sessions in teaching assistantships, I discovered my passion for sharing my knowledge. Through research, I discovered many interesting fields of computer science and have a greater appreciation of computer science. I am fortunate to have him as my advisor. Also, I like to thank all members of my committee for their helpful comments regard to my research and for being accommodating in schedule.

I thank members of the Data System Group. I learned much about web development working on software development projects with them. Their warm smiles and humorous words brighten my days and alleviate my frustration and confusion with administrative tasks. I enjoy discussions about challenges of pursuing the degree and matters beyond academic. Not being alone makes facing challenges throughout the journey all that much joyful.

Many thanks go to my older brother, Yao Hua. He always looked out for me and encouraged me to do better. He is a giant on which I stand on and see further both in life and in research. Also, I thank my parents Mei-Hui Lin and Kenneth who supported me financially and with their never ending love and care.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 <u>Motivation</u>

Wireless networks have become increasingly popular in recent years. There are two variations of mobile wireless network: infrastructure mobile networks and infrastructureless mobile networks. The latter are also known as *mobile ad hoc network* (MANET). MANETs have no fixed routers. Instead, mobile nodes function as relay nodes or routers, which discover and maintain communication connections between source nodes and destination nodes for various data transmission sessions. In other words, an MANET is a self-organizing multi-hop wireless network in which all nodes within a given geographical area participate in the routing and data forwarding process. Such networks are scalable and self-healing. They support mobile applications where an infrastructure is either not available (e.g., rescue operations and underground networks) or not desirable (e.g., harsh industrial environments).

In many ad hoc networks such as vehicular networks, links among nodes change constantly and rapidly due to high node speed. One approach is to update the links in response to topology change. Routing protocols such as DSR [43] and AODV [67][24] follow this approach. However, this approach results large overhead and delay due to congested medium by the large amount of control packet. Another approach is to use cluster [12] to minimize the nodes needed to maintain topology. However, the cluster structure itself is also susceptible to high mobility. More recent clustering techniques such as GRID [53] and VSA [62] cluster nodes based on their positions to make cluster structure more stable. However, high mobility means nodes enter and

leave the geographic cluster constantly and requires cluster states to be frequently communicated by nodes leaving the cluster to the nodes entering the cluster. Another approach is to let each intermediate node to select next hop based on its neighbors' positions and the trajectory or geographic points predefined in a data packet. The neighbor information is obtained through beacon packets from neighbors. This approach has no overhead of maintaining link or cluster states. However, the beacon packets need to be exchanged frequently in the face of high node speed.

In addition to routing data, frequent topology also affects the broadcast protocol used by a routing protocol to find the routes or locations to destinations. Most of the aforementioned routing protocols use a simple Plain Flooding technique [36] that can cause a broadcast storm, especially in a high density environment [60]. Broadcast protocols that relies on the information of 2-hop [58][65][76][83] or 1-hop [54][11] neighboring information suffers outdated neighboring information due to high mobility. 0-hop broadcast protocols such as *Counter Based* [60] and *Probabilistic Based* [71] offer a simple way of reducing overhead but do not select rebroadcasting nodes optimally because the nodes are selected probabilistically. Advanced 0-hop broadcast protocols 0-hop protocols such as *Angle Based* [77], *PANDA* [52], and *Border Aware* [86] use a distance delay algorithm to select nodes near the edge of a broadcast radio range to rebroadcast packets. However, their node selection criteria also result most of the nodes near the edge participate in rebroadcasting the packet. In a certain environments such as street environment, these protocols can degenerate to a plain flooding protocol.

1.2 **Main Contribution**

To reduce the number of nodes needed in data forwarding and still be able to handle a high mobility environment, we introduce the Virtual Router abstraction in this dissertation. A virtual router is a logical router that is associated with a particular geographic area. A virtual router comprises one or more physical mobile nodes currently within a geographical region served by the virtual router. Once within the geographical region of the virtual router, those physical nodes can take turns in forwarding data packets. In this environment, data packets are transmitted from a source node to a destination node over a series of virtual routers. There can be two virtual router approaches: *Static Virtual Router* (SVR) and *Dynamic Virtual Router* (DVR). In SVR, the virtual routers are predetermined and shared by all communication sessions over time. This scheme requires each mobile node to have a map of the virtual routers, and use a *global positioning system* (GPS) to determine if the node is within the geographical region of a given router. In DVR, the virtual routers are dynamically created for each communication sessions as needed and deprecated after their use, mobile nodes do not need to have a GPS, and mobile nodes do not need to know whereabouts of the virtual routers. We will present SVR in Chapter 2 and DVR in Chapter 3. In Chapters 4 and 5, we apply virtual router approach (in particular, SVR) to reduce overhead of the initial route or location request in open and street environments, respectively.

The advantages of the virtual router approach are as follows:

- Although the physical nodes may move, the virtual routers do not since they are defined by the geographical region. Due to that stability, a virtual connection

comprising the virtual routers is much more robust than a traditional physical

connection used in existing MANET designs.

- This strategy eliminates the overhead of maintaining clusters as in GRID [53] and

  VSA [62].

2. STATIC VIRTUAL ROUTER IN STREET ENVIRONMENTS

2.1 Introduction

There are many routing protocols designed to relay data in *mobile ad hoc networks* *(*MANETs). However, most of them are not designed for street environments. Earlier protocols such as *Dynamic Source Routing* (DSR) [43], *Ad-hoc On-Demand Distance Vector Routing* (AODV) [69], and *Location Aided Routing* (LAR) [47] require a source to use route request to establish a hop-by-hop route between itself and a destination before sending data. In the street environments, however, obstacles and fast moving nodes result in a very short window of communication between nodes on different streets. The established route expires quickly and the source needs to reissue another expensive network wide route request after sending only a few data packets via the previous route. These protocols, when applied in the street environments, will incur a high control overhead in terms of route request packets.

To overcome the fragility of multiple-hop routes, one-hop- based approaches, such as *Trajectory Based Forwarding* (TBF) [61], let each forwarder select the next forwarding node by comparing the positions of all its neighbors with the trajectory defined by a source. This position information is obtained through periodic broadcasts from neighboring nodes. The short window of communication in the street environments, however, means that the nodes need to broadcast more frequently in order to maintain up-to-date location information. This strategy incurs a high control overhead in terms of frequent beaconing packets that also congest the wireless medium. We identify the above protocols as a Connection-oriented approach because each link a packet

traverses must first be established through a network wide route request or location information exchange among all the nodes in the network.

Rather than using the expensive control overhead to pre-establish each link, the Connectionless approach allows a node to dynamically participate in a forwarding of data by comparing its current location with headers of the data, which contain location information of a source, a destination, and a previous relaying node. Existing Connectionless techniques such as *Contention-based Forwarding* (CBF) [22], *Beacon-Less Routing* (BLR) [27], and *Connectionless Approach to Mobile Ad Hoc Network* (CLA) [35] only allow nodes that have the shortest distance to the destination or are on the shortest geographic path (i.e. a straight line) between the source and destination to relay data. When applying these Connectionless approaches to the street environments, nodes that can relay data around obstacles often do not get to relay the data because they are farther from the destination than the previous relaying node or are not on the forwarding path of the data. Thus, these approaches cannot be applied directly to the street environments.

Recently, a new method [57] has modified the CBF technique to address the obstacle problems in a city environment. This scheme allows a source to specify a forwarding path as a list of junctions, and applies CBF between consecutive junctions. This solution requires at least one node at each turning junction, which is often difficult to achieve over an extended communication period. To overcome this drawback, we utilize multiple communication paths in the proposed technique.

The contribution of this chapter is to adapt a **Connectionless approach** [35] to a street environment (CLA-S) where mobile nodes are vehicles moving in high speed. We name our technique *Connectionless Approach for Streets* (CLA-S). This approach can quickly adapt to topology changes because it does not require a node to maintain its neighbors' locations or hop-by-hop routes. Any nodes in the source designated forwarding virtual cells can help relay data. To relay data around large obstacles typically found in the street environment, we revise the original algorithm for forwarding zone selection to leverage the on-board map database included in a vehicular navigation system. This new solution also achieves fault tolerance by providing multiple geographic paths by which data can be relayed from a source to a destination.

2.2 Proposed Solution: Connectionless Approach For Street Environments (CLA-S)

As with many routing protocols [22][35][45][47][61], we also assume that all nodes can obtain location information provided by technologies such as the *Global Positioning System* (GPS) [19][64]. This is a reasonable assumption because of the increasing availability of these devices and because the GPS service is provided without charge. If GPS is not available, one can conceive that nodes could calculate their positions with a local scheme - a research area that has recently been well studied [70]. In our presentation, we use *xy*-coordination. Devices such as GPS can, in fact, provide a 3-D location in terms of longitude, latitude, and altitude.

Though standalone GPS is not accurate to a precise degree, there are technologies that, when combined with GPS, improve its accuracy to centimeters [80], which is sufficient for our

7

protocol. Low radio frequency and onboard sensors facilitate GPS availability in urban areas, where there may not exist line of sight with GPS satellites [80][44].

2.2.1 Virtual Cell

Streets are divided into small "virtual cells." These cells are divided according to intersections and blocks (i.e. the street segment in between 2 adjacent intersections). For example, in Figure 1, Cell A and Cell C are intersections. Cell B is the entire block. In this chapter, we assume that all nodes have the same radio range. This is a reasonable requirement in a city environment because a car itself can generate its own power. We can set the radio range large enough that it can cover one block plus one intersection for most streets. For those blocks that exceed the radio range, we can fragment the blocks according to the radio range and identify the fragments separately like Cell B_1 and Cell B_2. In other words, the virtual cell is designed such that all the nodes in a cell (e.g., Cell B in Figure 1) can communicate directly with all the nodes in the adjoining cells (e.g., Cell A and Cell C in Figure 1).

		Building		
	Cell A	Cell B	Cell C	
		Building		
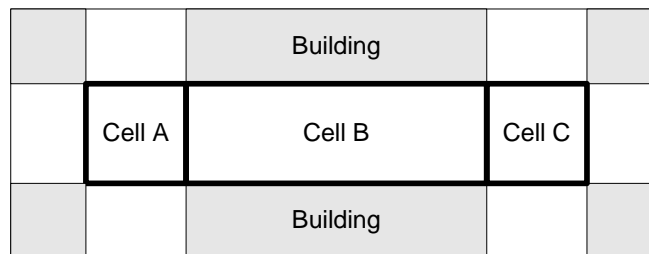
Figure 1. Virtual Cell

To uniquely identify each cell without the need to communicate all cell IDs among all the nodes, we identify a cell by its center point coordinates indicates as (x,y) where $x = \frac{(x_1 + x_2)}{2}$ and $y = \frac{(y_1 + y_2)}{2}$ with (x_1, y_1) and (x_2, y_2) denoting the 2 diagonal corners of the cell. The coordinates of the 2 diagonal corners can be calculated from the on-board map database. Thus, when a node moves to a new cell, it finds out the center points of the cell and any adjoining cells. To handle the inconsistency among different map databases of the nodes, 2 center points can be considered as the same if they are within a very short distance of each other (e.g., 1 or 2 meters apart).

2.2.2 Location Discovery

The location discovery phase of this technique is similar to that of CLA. A source node initiates a LOCATION DISCOVERY to find a destination node's location. Any node that hears the packet will send a LOCATION REPLY containing the destination node's location if it either is the destination or has fresh location information of the destination; otherwise, it will rebroadcast every unique LOCATION DISCOVERY once. However, since the routine used to send the reply back to the source needs to be modified in order to relay the reply (as well as subsequent data from the source and subsequent control packets to/from the source) around obstacles and to provide multiple paths, we will term the routine here as *Area Computation* in order to distinguish it from the *Path Computation* routine of the CLA approach. We will describe the routines of area computation and packet forwarding in Section 2.2.3 and Section 2.2.4, respectively.

9

2.2.3 Area Computation

In our approach, we do not need to maintain a hop-by-hop route between the source and destination nodes. Our technique selects a list of cells that form a "connecting" forwarding area between the source and destination. An example is illustrated in Figure 2; it shows that the lightly shaded cells are part of the area selected. Nodes within each of these cells alternate in forwarding data toward the destination node. We do not need to use every cell to forward data.
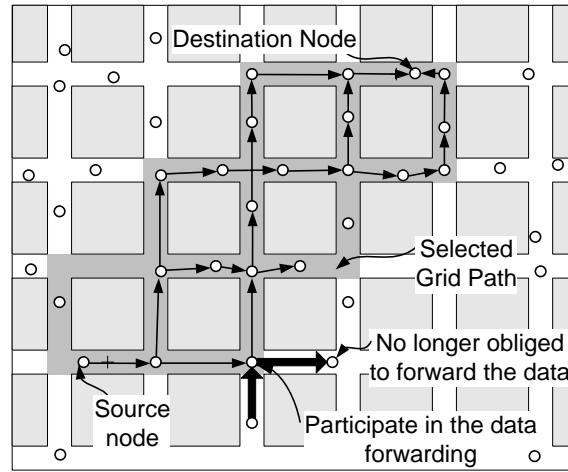
Figure 2. Forwarding Area

When a node leaves the selected cells, it is no longer obliged to forward data. Similarly, if a node enters the area, this node must participate in the data forwarding. A delay forwarding scheme, discussed in the Section 2.2.4, is used to coordinate the nodes within the area such that they may take turns forwarding the data. We observe that a forwarding area is much more robust than a traditional hop-by-hop route. The latter would fail if any one node along the route "fails." In contrast, a forwarding area is much more tolerant of "node failures" since a neighboring node

10

can dynamically substitute for the failed node with no overhead. This characteristic makes the Connectionless approach more suitable for ad-hoc networks.

2.2.3.1 Establish Reference Line and Reference Points

We define the *destination cell* as the cell containing the destination node. Similarly, the *source cell* is the cell containing the source node. To determine a forwarding area between the source and destination nodes, we first establish a "reference line" between the source and destination cells. The *reference line* (RL) is the straight line that connects the center of the source cell (X_S, Y_S) with the center of the destination cell (X_D, Y_D). As illustrated in Figure 3, the *reference line* is between the two cells: one at the upper right and the other at lower left corners. Obviously, the coordinates of the two end points define the *reference line*.

Once the reference line has been established, we need to determine the reference points. The *reference points* (RP's) on a reference line are the interceptions of the reference line and the centerline of either a vertical street or a horizontal street (see Figure 3).
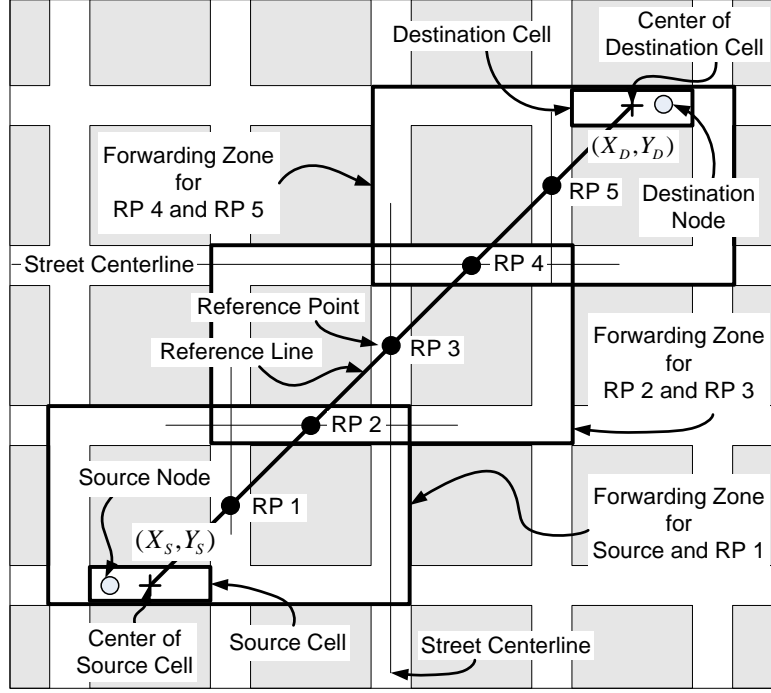
Figure 3. Reference Line, Reference Points, and Forwarding Zones

2.2.3.2 Determine the Forwarding Zone

Once all reference points of a reference line have been determined, we will use reference points to determine each *Forwarding Zone*. A *Forwarding Zone* is an area that is determined by a reference point or the center of a source cell. A reference point can be on a horizontal block, a vertical block, or an intersection (a block is considered as horizontal if the street it is on has a horizontal orientation; otherwise, it is vertical). We will describe how the *Forwarding Zone* for each case is determined for a vertical reference line in the following 3 sections. For a horizontal reference line, the width and height of each rectangular forwarding zone are swapped.

12

2.2.3.2.1 On a Horizontal Block

For any of the 3 cases, the *Forwarding Zone* of a reference point is a rectangular area that includes at least 2 horizontal streets and 3 vertical streets. For a reference point on a horizontal block (such as RP 2 in Figure 3), the 2 horizontal streets are the horizontal street where the reference point is on and the next horizontal street that is closer to the destination. The 3 vertical streets include one adjacent vertical street on each side of the reference point and one more vertical street on the side closer to the destination.

2.2.3.2.2 On a Vertical Block

For a reference point on a vertical block (such as RP 5 in Figure 3), the 2 horizontal streets are one horizontal street on the top of the reference point and one horizontal street on the bottom. The 3 vertical streets include one adjacent vertical street on each side of the reference point and one more vertical street where the reference point is on.

2.2.3.2.3 On an Intersection

For a reference point on an intersection (such as the RP *n* in Figure 4), the 2 horizontal streets are the horizontal street where the reference point is on and the next horizontal street that is closer to the destination. The 3 vertical streets include one adjacent vertical street on each side of the reference point and one more vertical street where the reference point is on.
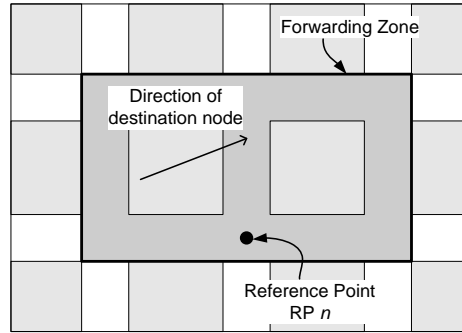
Figure 4. The forwarding zone for a reference point on an intersection.

For example in Figure 3, we can determine the Forwarding Zone for each reference point. Notice that the source cell and RP1 have the same forwarding zone. Similarly, RP2 and RP3, and RP4 and RP5 also have the same forwarding zones.

2.2.3.3 Select Cells for a Forwarding Area

After the forwarding zones of all reference points are determined, any nodes within the cells encapsulated by the zones are responsible for forwarding data. Notice that, the forwarding zones give us multiple paths that connect the source and destination (see Figure 2).
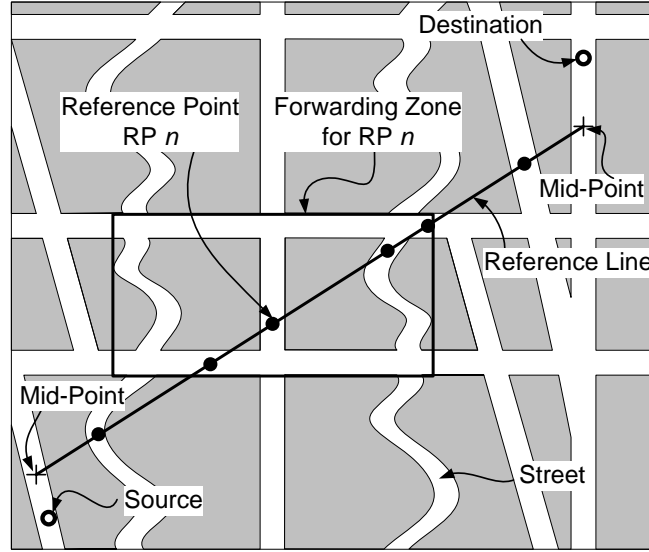
Figure 5. Reference Line and Forwarding Zone for irregular street pattern.

2.2.3.4 Irregular Street Pattern

Until now, in this chapter, we have assumed that all buildings are of the same size. We show in this section how to apply our technique to an environment where all streets are not necessarily parallel or perpendicular in relation to each other. For an irregular street pattern, the procedures for finding a reference line and reference points are the same as before (see the reference line in Figure 5). When determining a forwarding zone in an irregular street pattern, the size of the forwarding zone will still need to cover the same number of vertical streets and horizontal streets as discussed in previous section (see the forwarding zone for RP n in Figure 5).

Our initial application of the proposed protocol is for urban environments (e.g., Manhattan in New York City) where most streets have general orientations. Currently, our protocol does not use streets that lack of general direction (e.g., loop streets in subdivisions). Our

future work will try to take advantages of these streets when they are helpful to data forwarding (e.g., when either the source or destination is on a loop street).

2.2.4 Data Forwarding

To transmit a data packet in forwarding zones, the source node includes the following information in the data header: *Source Node ID*, *Source Cell ID*, *Destination Node ID*, *Destination Cell ID*, *Packet ID*, *Current Cell ID*, *Orientation*, *Direction*, and *FZone_Size*. The *Source Node ID* and *Destination Node ID* fields are the node IDs of the source and destination nodes, respectively. The *Source Cell ID* and *Destination Cell ID* are respectively the IDs of the cells that the source and destination are currently on. As mentioned in Section 2.2.1, cells can be identified by their center points. The *Current Cell ID* is the ID of the cell containing the node that is about to forward the data packet. Thus, each intermediate node updates this header field before relaying the data packet. The *Orientation* indicates whether the reference line is vertically or horizontally oriented. The *Direction* indicates whether the packet is from the source to the destination or vice versa since the same forwarding area and the same forwarding procedure are also used to transmit the control packets (except LOCATION DISCOVERY) mentioned in Sections 2.2-2.2.2, 2.2-2.2.5, and 2.2-2.2.6. The *FZone_Size* defines the size of the forwarding zone. Initially, the *FZone_Size* is set to one and will change according to the need. We will discuss more on increasing *FZone_Size* size in Section 2.2.6

When a node *n* receives a data packet from *m*, the data forwarding procedure is as follows:

1. If *n* is the destination, *n* does not forward the data.

2. If *n* is not in the forwarding zones, *n* does not forward.

3. If *n* or any other node in the cell containing *n* has forwarded, *n* does not forward.

4. If Steps 1, 2, and 3 fail (i.e. *n* might need to forward the data), *n* delays the forwarding.

5. During this delay period, *n* will cancel the forwarding if *n* either hears the same packet from a neighboring node on the same cell or if *n* is in a block cell and *n* hears the same packet from both adjacent intersections.

6. At the end of the delay period, if the forwarding decision has not been cancelled, *n* forwards the data.

When a node receives a packet with a new Forwarding Area (because of a new reference line), it will compute the Forwarding Zones and save the result as a list of streets and the ranges of the streets that are encompassed by the Forwarding Zones. This allows the node a quick and simple way to determine if it is in a Forwarding Zone for subsequent packets with the same Forwarding Area. Although the implementation of this routine is out of scope of this chapter, one can conceive a routine that queries the on-board map database to obtain the coordinates of the reference points, find the nearest streets to the reference points, and determine the intersections of these streets for the ranges [2][14][63].

In the above procedure, the delay of a node *n* is computed as follows:

$$DELAY_n = \left| \frac{\alpha}{2 \bullet D\_Dist_n} - \frac{\alpha}{2 \bullet Dist_n} \right| \tag{1}$$

where α is a maximum delay constant in μsec, D_Dist_n the distance between node n and the center of the cell denoted by the *Destination Cell ID* in the packet header, and $Dist_n$ the distance between node n and the center of the cell denoted by the *Current Cell ID* (cell of previous relaying node m) in the packet header (See Figure 6). The significance of this equation is to select a node farther away from m and closer to the destination node to forward the data packet.

If the node n is at an intersection of two streets, we will set a shorter delay period. In the simulation, the delay for an intersection node is set to one third of the normal *DELAY*. The reason for this is that, when at an intersection, a node's effective radio range can cover the 2 intersecting streets compared to the single street coverage of another node on a block

Notice we can always factor other parameters into the delay computation such as workload or network traffic. That is, we use a longer delay for a node with high workload to allow another neighboring node with lower workload to forward.
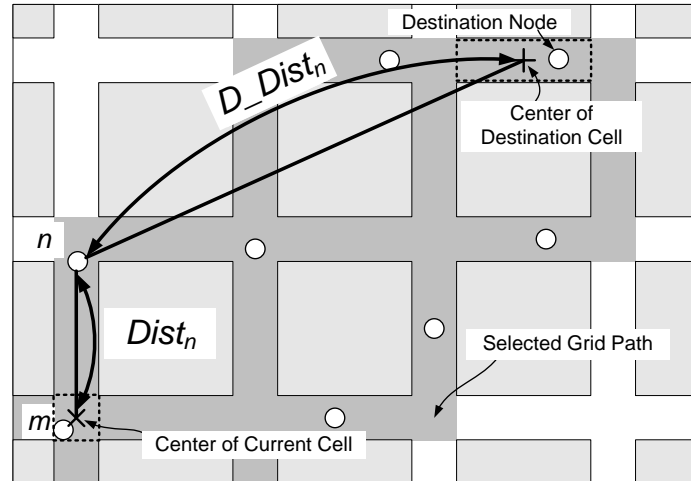
Figure 6. *DELAY* for node n.

18

2.2.5 Area Maintenance

In order to maintain an effective forwarding area for packet delivery after the initial location exchange, the source and destination nodes need to update their location information with each other as they move around. To update its location, the source node piggybacks its current location information to every data packet for the destination. To update its location, the destination node periodically checks its own location to see if it is out of the destination cell lastly informed the source. If it is, it sends a LOCATION UPDATE packet to update its location with the source node using the most current source node location information contained in recently received data. Both the source and the destination will call the *Area Computation* routine (see Section 2.2.3) to find a new area to send packets when they notice either of them has moved to a new cell.

2.2.6 Low Node Density Environment

We note that there may be a situation in which we cannot find a connecting path to forward data packets. This can happen in a low density area of a network. When a destination node does not receive any data from a source node for a period of time (*Tifz*) since last data packet or the location request received from the source, it will send an INCREASING FORWARDING ZONE packet. The packet will be relayed in a forwarding area with *FZone_Size* = 1+ *FZone_Size*. The destination will keep sending an INCREASING FORWARDING ZONE packet every *Tifz* seconds of not receiving data from the source. When the forwarding zone eventually gets large enough and the source receives one or more of the INCREASING FORWARDING ZONE

packets, the source sends data with forwarding zone size of *FZone_Size* found in the received

INCREASING FORWARDING ZONE packet or the smallest *FZone_Size* if multiple INCREASING

FORWARDING ZONE packets are received within a period of *Tifz*.

To increase the number of nodes participating relaying packets in a low density situation,

the *Forwarding Zone* will be enlarged according to the direction of the destination node. In

Figure 7, the new forwarding zone is obtained by including one more vertical street and one more

horizontal street closer to the destination because the differences between the *FZone_Size* of the

two forwarding zones is 1. When *Forwarding Zones* are enlarged, the area between the source

node and the destination node is also increased (see Figure 7). The size of a *Forwarding Zone*

can be increased more if the source receives more INCREASING FORWARDING ZONE packets from

the destination.
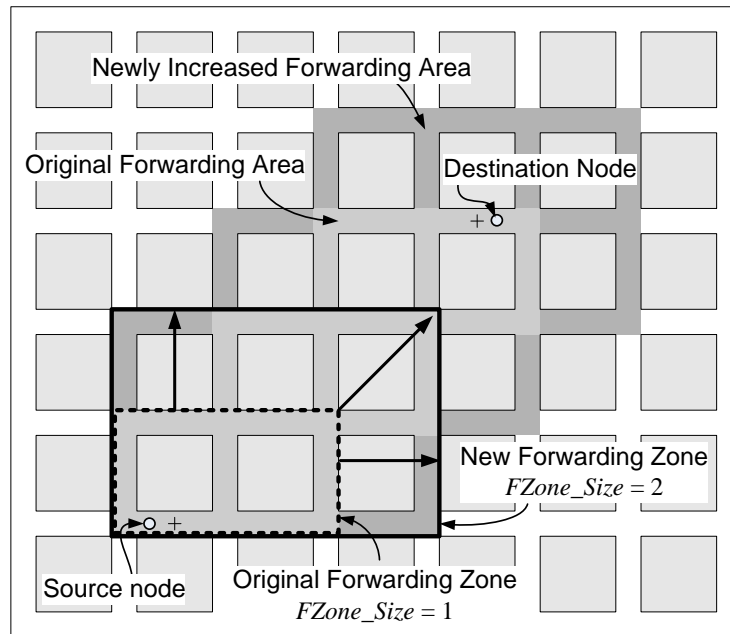
Figure 7. New Area with Incresased Forwarding Zones.

20

To indicate that there is no more data to send, the source will send a SESSION TERMINATE packet after not seeing any packets from its upper layer applications for a period of time (*Tse*; *Tse* < *Tifz*). Upon receiving the SESSION TERMINATE packet, the destination will not send INCREASING FORWARDING ZONE packet and LOCATION UPDATE packet. The source will have to initiate another location discovery if it still has more data to send after *Tse*. In this chapter, we set *Tse* to be larger than packet interval of Constant Bit Rate (CBR) applications.

When a source receives an INCREASE FORWARDING ZONE packet, it will increase the forwarding area of subsequent data and SESSION TERMINATE packets for the destination. When a destination receives data with increased forwarding zone information, it will adjust the *FZone_Size* of the forwarding zone used to send subsequent LOCATION UPDATE and INCREASE FORWARDING ZONE packets to the source.

2.3 Simulations Study

To evaluate our approach, we perform simulations using a network simulator called GloMoSim [85]. This simulator, developed at UCLA, is a packet-level simulator specifically designed for ad-hoc networks. It follows the OSI 5-layer network communication model.

We simulate and compare our CLA-S protocol with 3 protocols provided by GloMoSim: AODV, DSR, and LAR. We do not compare with CBF, BLR, and CBF in Street [57] because they do not provide a location update service. CBF and CBF in Street require an additional location service (e.g., [7][46][51]) whose ability to maintain up-to-date location information and control packets can affect the performance of the protocols. BLR uses reactive local routing to

21

establish a hop-by-hop route between the destination and a node that is within the radio range of the known destination location. There is no location update from the destination to the source and the protocol can degenerate into a Connection-oriented approach once the destination has moved out of the radio range of its previous location known to the source. Although CLA provides location update mechanism, we also do not compare with CLA because it is ineffective when the source and destination are not on the same street (i.e. buildings block the straight line path between the source and destination).

We perform sensitivity analysis with respect to mobile speed, node density, and application load. The field configuration is a 1000m × 1000m field with a street width of 10 meters and a building block size of 100 meters by 100 meters. Each node has a radio range of about 375 meters. Initially, nodes are placed uniformly with 2 nodes per intersection and 8 nodes per block. Then the nodes move in the directions permitted in the streets. Upon arriving at an intersection, a node probabilistically changes its direction of movement (e.g., turn left, turn right, or continue in the same direction). Traffic applications are constant-bit-rate sessions involving 1/10 of all the nodes. Each data packet is 512 bytes and sent at 2.5 sec interval. In this study, our CAL-S does not cache the overheard location information. This additional feature would only enhance our technique further.

The protocols are compared under three performance metrics: (i) **fraction of packets delivered** – measures the ratio of the data packets delivered to the destinations and the data packets generated by the CBR sources; (ii) **end-to-end delay** – measured in ms, includes processing, route discovery latency, queuing delays, retransmission delay at the MAC, and

propagation and transmission times; (iii) **normalized routing load** – measures the number of control packets (e.g., route discovery packets in the Connection-oriented techniques or LOCATION DISCOVERY packets in the Connectionless technique) transmitted per data packet delivered to the destinations; this metric is also referred to as control overhead.

2.3.1 Simulation Results

We present the simulation results in the following subsections. We study the effect of mobile speed and the effect of node density.

To understand the **effect of mobile speed** on performance, we varied the speed of the mobile nodes between 10 and 25 (m/s). The simulation results are presented in Figure 8, Figure 9, and Figure 10. It demonstrates that the Connection-oriented approach is not suitable to be used in street environments. For example, it only takes ½ of a second to traverse an intersection of 10 meters wide at a speed of 20m/s. This means a route involving a connection between 2 nodes on 2 different streets last only ½ second, long enough to transmit one data packet for CBR sessions that have a 2.5 second interval between each data packet. This means that the Connection-oriented approaches either drop a large amount of data packets (in the case of DSR in Figure 8) or require a large amount of control overhead to keep routes from the sources to the destinations up to date (in the case of AODV and LAR1 in Figure 10). Maintaining routes can also induce long delay (in the case of LAR1 in Figure 9). However, the CLA-S allows any node in the geographical forwarding zones to relay packets. This flexibility allows CLA-S to perform well under all three metrics.
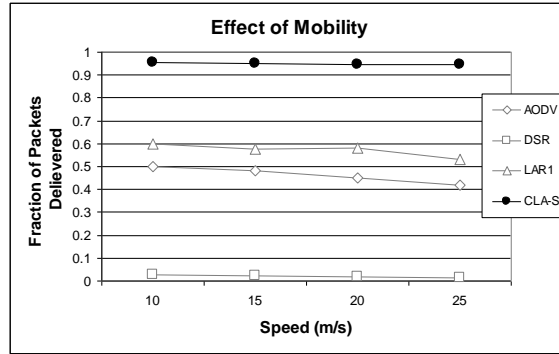
Figure 8. Effect of Mobile Speed: Fraction of packet delivered
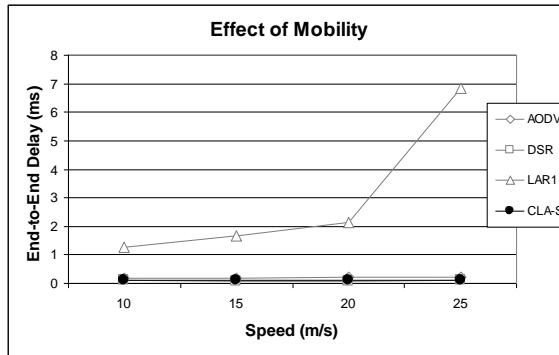
Figure 9. Effect of Mobile Speed: End-to-end delay
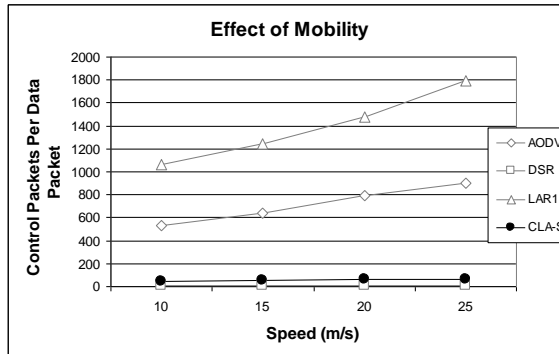
Figure 10. Effect of Mobile Speed: Normalize routing load

To understand the **effect of node density**, we varied the number of nodes per intersection

from 1 to 8 and number of nodes per block from 4 to 32. The simulation results are presented in

Figure 11, Figure 12, and Figure 13. These figures show that the high density increases the overhead without improving delivered rate for the Connection-oriented approaches. This is because every time a source issues a route request to maintain route, either almost all nodes need to relay the request at least once (in the case of AODV and LAR1) or intermediate nodes reply to the source with outdated routes (in the case of DSR). On the other hand, CLA-S has no route to maintain, thus does not suffer either drawback of the Connection-oriented approaches.
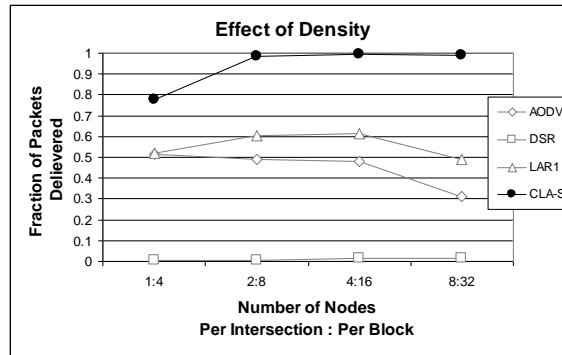
Figure 11. Effect of Node Density: Fraction of packet delivered
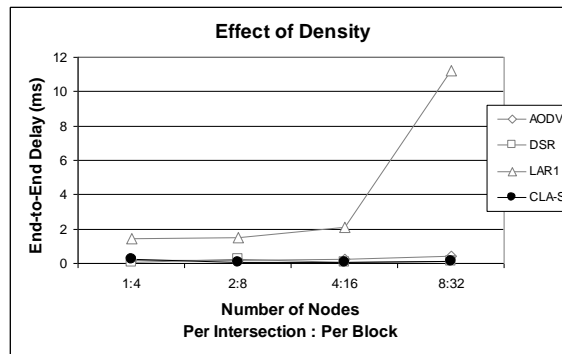
Figure 12. Effect of Node Density: End-to-end delay

25
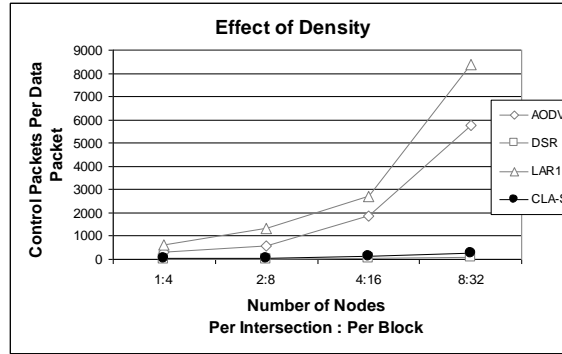
Figure 13. Effect of Node Density: Normalize routing load

To understand the **effect of application loads**, we varied the portion of nodes involved in communication from 1/20 to 1/5. The simulation results are presented in Figure 14, Figure 15, and Figure 16. Again, they show that other Connection-oriented approaches make trade off among fractions of packets delivered, end-to-end delay, and control overhead. Only our connectionless approach can robustly adapt to the changes in network load.
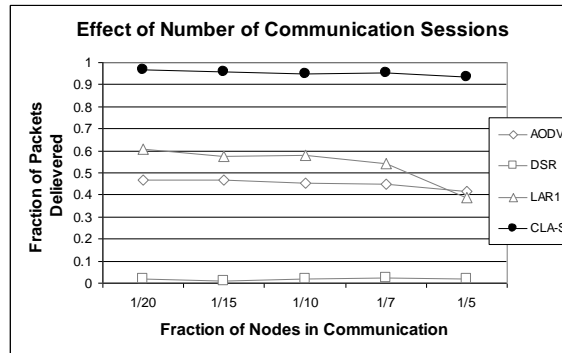
Figure 14. Effect of Application Loads: Fraction of packet delivered
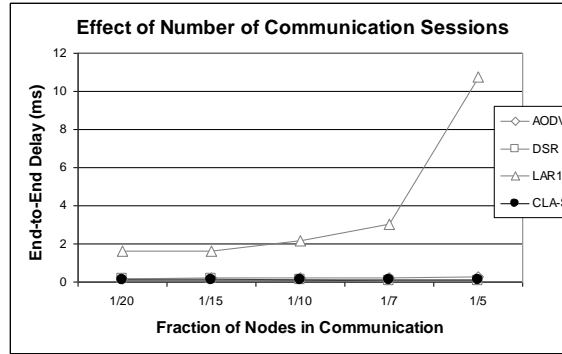
Figure 15. Effect of Application Loads: End-to-end delay
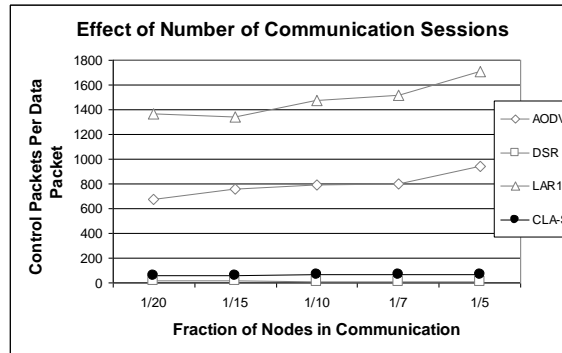
Figure 16. Effect of Application Loads: Normalize routing load

2.4 **Summary**

This chapter introduces a Connectionless approach to wireless mobile ad hoc networks. Its performance comparison with three conventional Connection-oriented techniques is summarized in the following list.

The key advantages of the CLA-S approach are as follows:

- Low control overhead: It has no communication connections to break or to maintain.

- Low packet loss: When a node moves away from a forwarding area, a nearby node can take over the data-forwarding task without delay such as reestablishing a new

route or updating location information in the caches of near-by nodes as in the case of the Connection-oriented approach.

- Short delay: Transmission delay is optimized for each packet transmission with minimum overhead of control packets.

3. DYNAMIC VIRTUAL ROUTER IN OPEN SPACE

3.1 Introduction

Wireless networks have become increasingly popular in recent years. There are two variations of mobile wireless network: infrastructure mobile networks and infrastructureless mobile networks. The latter are also known as *mobile ad hoc network* (MANET). MANETs have no fixed routers. Instead, mobile nodes function as relay nodes or routers, which discover and maintain communication connections between source nodes and destination nodes for various data transmission sessions. In other words, an MANET is a self-organizing multi-hop wireless network in which all nodes within a given geographical area participate in the routing and data forwarding process. Such networks are scalable and self-healing. They support mobile applications where an infrastructure is either not available (e.g., rescue operations and underground networks) or not desirable (e.g., harsh industrial environments).

Next-generation ad hoc networks need to be able to handle high mobility in order to support a wide range of emerging applications such vehicular networks and mobile sensor networks. Maintaining communication links of an established communication path that extends between source and destination nodes is a significant challenge in MANETs. In particular, such communication links are often broken with rapid movement of the mobile nodes. Although a new communication route can be established when a break in the communication path occurs as in DSR [43] and AODV [67][24], repeatedly reestablishing new routes incurs delay and substantial overhead.

Protocols such as DSDV [66][56], WRP [59], Fisheye [41], and ExOR [9][8] proactively maintain link states to every node in the network. This approach has been shown to offer excellent performance for mesh network or roof-top network [1]. It can provide instant route to any node in a relatively static network. These designs, however, are not intended for a high mobility environment where link states become outdated quickly.

Protocols such as CGSR [12], GRID [53], and VSA [62] maintain a cluster structure with cluster heads and gateway nodes responsible for relaying data. This structure, maintained by nodes in the network, allows nodes that are not cluster heads and gateways to sleep and thus save energy. This property is desirable for applications such as wireless sensor networks. This approach, however, is not suited for a high mobility environment where nodes move in and out of the clusters constantly. This results in significant overhead due to control packets necessary for joining new clusters and selecting new cluster heads and gateways.

Protocols such as ZRP [25] and ARAMA [40] are a hybrid of proactive and reactive approaches in that they proactively maintain local neighborhood information and reactively find routes to remote nodes. The local neighborhood information allows an intermediate node to select a better next hop from the one the source node decided. This approach requires each node to exchange beacon packets with its neighboring nodes in order to maintain the local neighborhood information. The frequency of such information exchange is proportional to the node mobility. For a high mobility environment, the exchange must occur very frequently to ensure the accuracy of local neighborhood information and thus can congest the wireless medium.

To address the instability caused by high node mobility, some protocols use node locations as part of the data forwarding process. Protocols such as GRID [53] and VSA [62] use locations as the criteria for clustering nodes. While a node can easily determine if it belongs to a cluster based on its location and the location of the cluster, these protocols still have the overhead associated with maintaining cluster states that need to be actively communicated among nodes. Other protocols such as TBF [61] and TRR [10] use a geographic path defined by a trajectory [61] or a list of geographic positions [10] between the source node and destination node. Base on the positions of its neighbors and information in the header, an intermediate node can pick a next hop most suitable to forward data. These protocols relay data packets toward a destination without the overhead of maintaining the link states between nodes. However, like the hybrid protocols discussed in the last paragraph, these protocols use beacon packets that need to be exchanged frequently in the face of a high mobility environment. Protocols such as CBF [22] eliminate the beacon packets by letting the neighbors of the current relaying node to compete to become the next hop. The competing nodes set a delay based on their respective distances to the previous hop and the destination node. A competing node forwards the data packet after the delay if it still has not heard the packet forwarded by another node closer to the destination. A potential weakness of this strategy is as follows. Since there could be two neighboring nodes, both having a similar distance from the previous hop and the destination node, but still out of each other's range, this protocol involves more nodes than necessary to relay data. This fan-out effect is illustrated in Figure 17, which shows many nodes are involved when only a few of them are needed to forward the data to the destination.
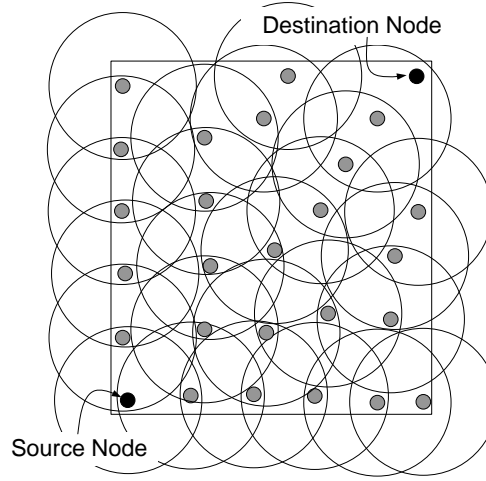
Figure 17. Fan-out effect of CBF

To reduce the number of nodes needed in data forwarding and still be able to handle a high mobility environment, we introduce the Virtual Router abstraction in this chapter. A virtual router is a logical router that is associated with a particular geographic area. A virtual router comprises one or more physical mobile nodes currently within a geographical region served by the virtual router. Once within the geographical region of the virtual router, those physical nodes can take turns in forwarding data packets. In this environment, data packets are transmitted from a source node to a destination node over a series of virtual routers. There can be two virtual router approaches: *Static Virtual Router* (SVR) and *Dynamic Virtual Router* (DVR). In SVR, the virtual routers are predetermined and shared by all communication sessions over time. This scheme requires each mobile node to have a map of the virtual routers, and use a *global positioning system* (GPS) to determine if the node is within the geographical region of a given router. We present SVR in [34][35]. In this chapter, we introduce the DVR approach with the following distinctions: (1) virtual routers are dynamically created for each communication

32

sessions as needed, and deprecated after their use; (2) mobile nodes do not need to have a GPS; and (3) mobile nodes do not need to know whereabouts of the virtual routers.

The advantages of the virtual router approach are as follows:

- Although the physical nodes may move, the virtual routers do not since they are defined by the geographical region. Due to that stability, a virtual connection comprising the virtual routers is much more robust than a traditional physical connection used in existing MANET designs.

- Since the forwarding of each data packet is confined to the virtual routers along the communication path, the virtual router approach does not suffer the fan-out effect as in CBF [22].

- This strategy eliminates the overhead of maintaining clusters as in GRID [53] and VSA [62].

The virtualization concept has been used in Computer Science for a long time including virtual memory, virtual computer, etc. However, the *Virtual Mobile Node* abstraction, proposed in [18], is more related to our work. In [18], the virtual mobile node abstraction is introduced to simplify the task of designing algorithms for mobile networks. While movement of physical nodes is unpredictable, virtual mobile nodes can be programmed to move in a predictable and useful manner. In this framework, algorithms can be designed to take advantage of the virtual mobile nodes to simply and efficiently perform complicated tasks in a highly dynamic and unpredictable MANET. Another work [48] applies the virtual node concept to handle selfish nodes on a route. Neighboring nodes can take over forwarding when they perceive a node on a

route do not relay a data packet after a period of time. A taking-over node also sends a notification informing the source about the selfish node on the route. Since the protocol does not distinguish failure due to a malicious node from mobility, the protocol incurs excessive notifications sent to the source and a long delay when this technique is applied to a high mobility environment. Another related work is called *Virtual Routing* [13]. This idea is similar to virtual computers, in which multiple TCP/IP stacks are built into one physical router. These stacks look and behave like independent stacks, yet the only things that really need to be independent are the routing tables. Such a router can replace potentially many conventional routers to save costs. To the best of our knowledge, the Virtual Router approach, presented in the this chapter, is the first to use virtualization in designing routing techniques for MANETs to address problems associated with high mobility.

The remainder of this chapter is organized as follows. We discuss the proposed Dynamic Virtual Router approach in Section 3.2. The simulation results are presented in Section 3.3. Finally, we conclude this chapter and discuss future work in Section 3.4.

3.2 Proposed Solution:
Dynamic Virtual Router (DVR)

Existing rerouting techniques, some presented in Section 3.1, suffer constant link breaks in high mobility environments. In this section, we introduce a reactive technique using virtual routers to address this problem. Contrary to prior-art methods, focus is placed upon preventing breakage of communication links rather than rapid re-establishment of a broken communication path.

3.2.1 Route Request

When a source node needs to send a data packet to a destination node and has not yet established a communication route, the source node initiates a route request. More particularly, the source node broadcasts a route request message packet that contains the source node's *identification* (ID). When the route request message packet is broadcasted, nearby mobile nodes receive the packet. Although each mobile node that receives the route request message packet could simply forward (i.e., broadcast) the packet, a probabilistic delay technique [29][60][74][81] can instead be employed to avoid flooding the network with route request messages. To that end, each mobile node that receives the route request message packet delays forwarding and monitors forwarding of the packet by another node. The mobile node can delay the packet for a predetermined or random time interval. During this period, if the mobile node detects forwarding of the route request message packet by another node, then the mobile node does not forward the message, given that such forwarding is not necessary. If, on the other hand, forwarding of the route request message packet is not detected within the time interval, the mobile node appends its own ID to the packet and forwards (i.e., broadcasts) the packet. The above process is performed by each mobile node in the wireless network that receives the route request packet so that the route request message packet traverses the network and ultimately arrives at the destination node. In this traversal process, the route request message packet being forwarded by certain nodes, referred to herein as *relay nodes*. In a rare occasion where a probabilistic technique does not find a route to a destination, a source will send another route request if no reply is received from the destination within a given time period.

35

3.2.2 Route Reply

Once the destination node has received the route request, it initiates a route reply to establish a communication route. More particularly, the destination node broadcasts a route reply message packet. That packet can contain various information including the destination node's ID, the source node's ID, a list of the relay nodes the route request message packet traversed, and a route ID. The route ID comprises the destination node's ID concatenated with a locally-generated unique number to ensure the uniqueness of the route ID in the network. After the route reply message packet has been broadcasted, it is routed by the relay nodes identified in the packet. Some optimization of route choice is possible. A destination waits for a given time period and picks the shortest route from the received requests to send a route reply.

Significantly, any neighboring node that "overhears" the packet can potentially join the communication route and form part of one or more virtual routers of the route. Therefore, like the relay nodes, the *overhearing nodes* will be available for forwarding data packets along the route. For any mobile node in the route, including both the relay nodes and the overhearing nodes, the determination as to whether to forward a data packet is made relative to the *distance-to-destination* (DTD) of the node to the destination node. The DTD of a node is expressed in terms of the number of virtual hops to the destination node. Figure 18 illustrates an example of how the DTD is determined. In Figure 18, each relay node has been assigned a label in the range of *r1-r7*. In this example, the DTD of relay node *r1* is 1, the DTD of the relay node *r2* is 2, and so forth. Each relay node uses its DTD value to update the virtual hop count field in the route reply message packet it forwards.

36
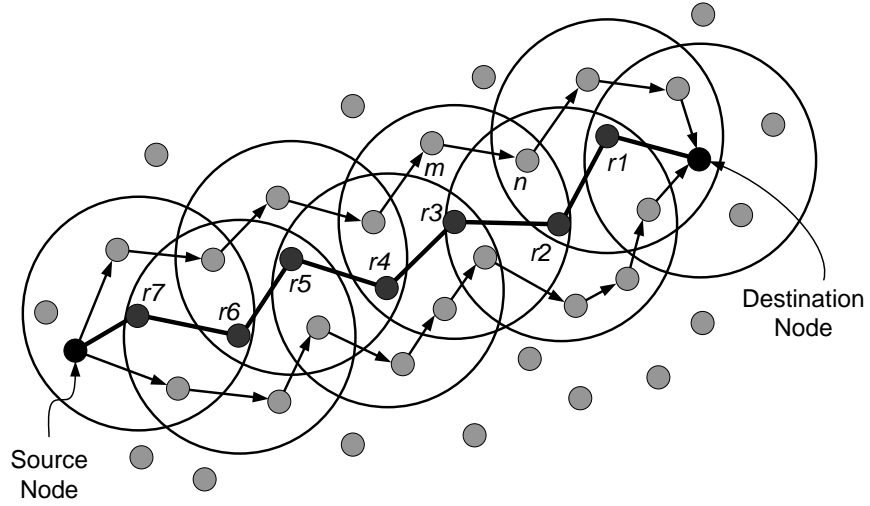
Figure 18. Virtual routers and the route from source to destination

With further reference to Figure 18, the circles identify the broadcast range of the respective relay nodes, and therefore the geographical extent of the virtual routers. The neighboring nodes within those circles can overhear transmissions of one or more of the relay nodes and therefore comprise the overhearing nodes. Each overhearing node can determine its own DTD as the minimum virtual hop count it has overheard for a given route reply. As an example, node "n" can overhear a route reply message packet broadcast by relay nodes *r1*, *r2*, and *r3*, which are 1, 2, and 3 hops away from the destination node *D*, respectively. In such a case, the DTD of node *n* is 1, the lowest overheard virtual hop count.

3.2.3 Data Forwarding

Once the communication route has been established in the manner described above, it can be used to transmit data packets between the source and the destination nodes. The source node includes the following information in the header of the data packets it transmits over the

37

communication route: *Source Node ID*, *Destination Node ID, Packet ID, Route ID, Virtual Hop Count*, and a list of traversed nodes. The *Packet ID* refers to the ID the source node assigns to this data packet. The *Virtual Hop Count* field is updated with the DTD of each node before the node forwards the data packet along the route. The list of traversed nodes refers to the list of nodes that has forwarded the data packet. The list is generated by each intermediate node appending its node ID to this list before forwarding the packet. We will explain the purpose of this list in Section 3.2.4.

When a node *n* receives a data packet from *m*, the data forwarding procedure is as follows:

1. If *n* is the destination, n does not forward the data.

2. If *n* has seen the data packet, n does not forward the data.

3. If *n* is not in the route, *n* does not forward the data.

4. If *n* is not downstream of *m*, n does not forward the data.

5. If all previous steps fail (i.e. *n* might need to forward the data), *n* delays the forwarding.

6. During this delay period, *n* cancels the forwarding if *n* hears the same packet again.

7. At the end of the delay period, if the forwarding decision has not been cancelled, *n* forwards the data.

In Step 3 of the above procedure, *n* determines if it is part of the route by comparing the *Route ID* field of the data packet with all the route ID's the node has identified from the overheard Route Reply packets. If the packet route ID matches the route ID of one of the overheard route reply message packets saved in a cache, the node *n* is in the communication

38

route. Periodically, each node can remove from its cache all the route IDs that have not been heard for a predetermined period of time.

In Step 4, *n* determines if it is downstream of *m* by checking if its DTD (as determined during the Route Reply procedure) is less than the *Virtual Hop Count* field of the data packet. If this condition is true, *n* can forward the data packet, i.e., proceeds to Step 6. As illustrated in Figure 18, since *m* has a DTD of 3, it sets the *Virtual Hop Count* field to 3 in the data packet. Since *n* has a DTD of 1, it belongs to a virtual router in the downstream and can help forward the data packet from *m*. We note that forwarding of a data packet might skip some of the virtual routers depending on which physical nodes are used in the forwarding.

In Step 5, node *n* sets its delay as *rand(n→seed)×t* seconds, where the function *rand(n→seed)* computes a random number using a predetermined seed at node *n*, and *t* is a time constant (e.g., 70 ms).

3.2.4 Route Update

If all the nodes move away from a virtual router, we may have a broken link in the virtual route. To minimize the occurrence of broken links, the destination node can periodically recruit new nodes to form replacement routers by sending out an unsolicited route reply, called *Route Update*, to the source node. A Route Update packet includes the ID of the destination node, the ID of the source node, the list of nodes traversed by the latest data packet (i.e., the nodes listed in the *Traversed Nodes* field of the data packet) received by the destination from the source, and the ID of this route. The ID of a route is generated by the destination node as discussed previously.

39
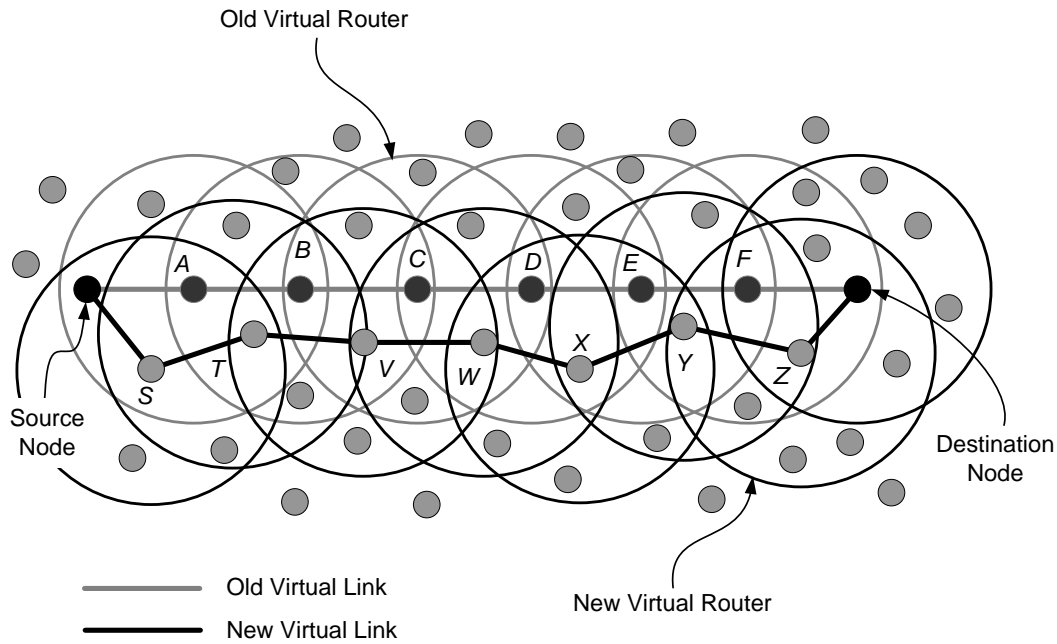
Figure 19. Old and New Virtual Routes.

The nodes listed in the Route Update packet relay the Route Update packet to the source node. Each of those nodes establishes a new virtual router identified by the corresponding node ID as in the case of processing a route reply packet; and the new virtual routers define a new route between the source and destination nodes. Once the source receives the Route Update, it discards the old route and old virtual routers, and includes the ID of the new route in the headers of the subsequent data packets. The route discovered in this manner is robust because it reflects the most recent topology between the source and destination. Operating in this manner, a communication session rarely experiences a virtual link break. Should it occurs, the source node can initiate route request to establish a new route.

Figure 19 gives an example of how a new virtual route is established. Suppose an existing virtual route centered at Nodes A, B, C, D, E, and F has been used for some time. The destination

node decides it is time to send a route update. It observes that the last data packet the destination receives is relayed by the following nodes in the virtual routers: nodes S, T, U, V, W, X, Y, and Z. Thus, the destination sends a Route Update through these nodes to reach the source node. As the Route Update packet traverses to the source node, the neighbors of these nodes overhear the update and form the new virtual routers (the darker circles in Figure 19). Once the source receives the new update, subsequent data packets will be relayed by these new virtual routers. The old virtual routers can simply be discarded after the nodes no longer see data traversing over the old virtual route for some time. We note that the proposed route update process does not require periodic beaconing from every node because the old virtual route is simply discarded. Since DVR is not a clustering technique such as [12], it does not need to maintain memberships and can easily be applicable in environments where nodes move independent of each other and in high speed, whereas a clustering protocol like CGSR is not suitable for such environments.

3.2.5 Route Recovery

In rare cases, a virtual link break can occur before the next route update. When this happens, route recovery is done as follows. Periodically, the source node expects a Route Update packet from the destination node. The source node detects a virtual link break if the next Route Update is late by a predetermined threshold. In this situation, the source node issues a Route Request (as described in Section 3.2.1) and waits for the Route Reply (as described in Section 3.2.2) about information on the new virtual route to the destination. This Route Reply packet contains the packet ID of the last data packet received by the destination node. Once the route

recovery is accomplished, the source node resumes the data transmission starting from the data packet succeeding the last data packet received by the destination node. These data packets are relayed along the new virtual routers. We note that the above route recovery procedure is a source-initiated technique. One can also design a destination-initiated scheme, in which the route recovery procedure is initiated by the destination node if it does not receive the next data packet by a predetermined maximum delay. In an application such as file transfer, a destination can use TCP in the transport layer to request a source to resend missing packets that may have lost in between route updates. In the simulation study later in this chapter, the applications are constant bit-rate sessions using UDP as the transport layer protocol, which does not attempt to resend the lost packets.

3.3 Simulation Study

To validate the proposed design and evaluate its performance, we implemented the *Dynamic Virtual Router* (DVR) approach in GloMoSim [85]. GloMoSim is a packet-level simulator specifically designed for ad-hoc networks. It follows the OSI 5-layer network communication model.

To evaluate the robustness of virtual routes (over virtual routers) relative to traditional routes (over physical nodes), we compare both the *Static Virtual Router* (SVR) approach and *Dynamic Virtual Router* (DVR) approach with the *Ad-hoc On-Demand Distance Vector Routing* (AODV) techniques. AODV is a relatively recent technique [67] and is a good reference for our study. Comparing SVR and DVR allows us to measure the benefit of using dynamically created

virtual routers over predefined static routers. For each technique, we performed sensitivity

analyses with respect to *mobile speed*, *node density*, and *terrain size* to investigate their effect on

performance.
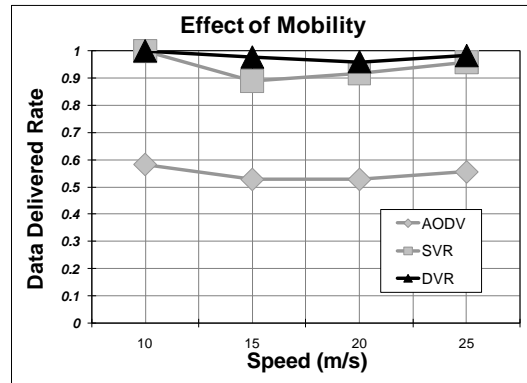
(a)
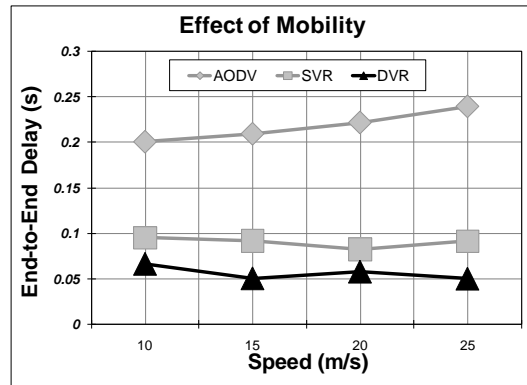
(b)
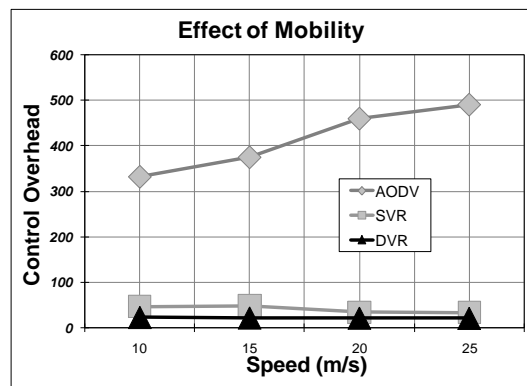
(c)

Figure 20. Effect of Speed: (a) Data Delivered Rate, (b) End-to-End Delay, and (c) Overhead.

44
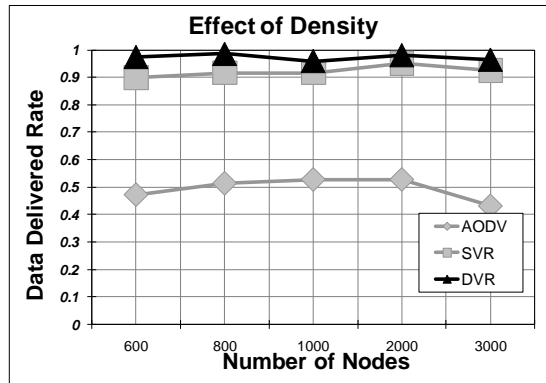
(a)
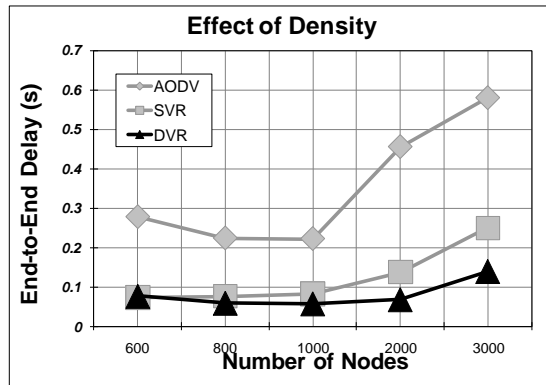
(b)
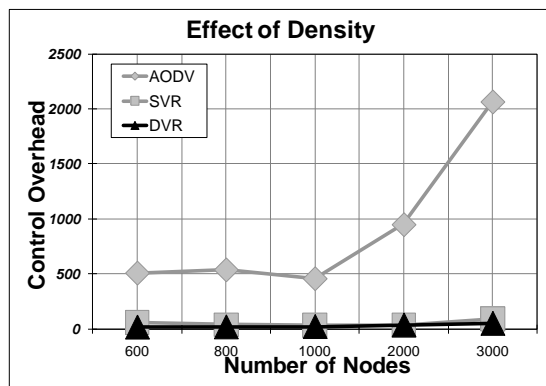
(c)

Figure 21. Effect of Density: (a) Data Delivered Rate, (b) End-to-End Delay, and (c) Overhead.

45
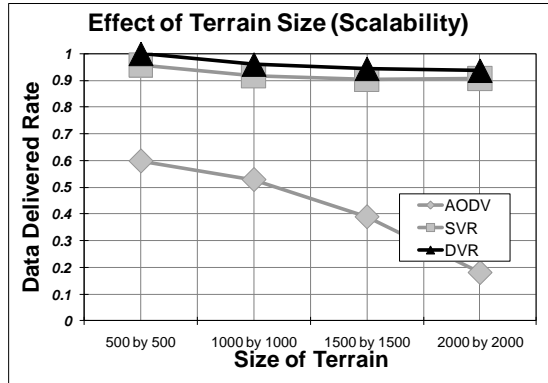
(a)
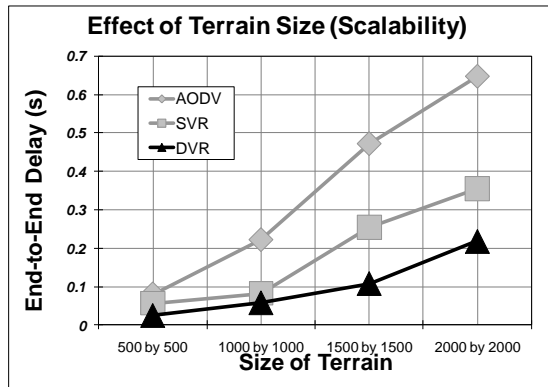
(b)
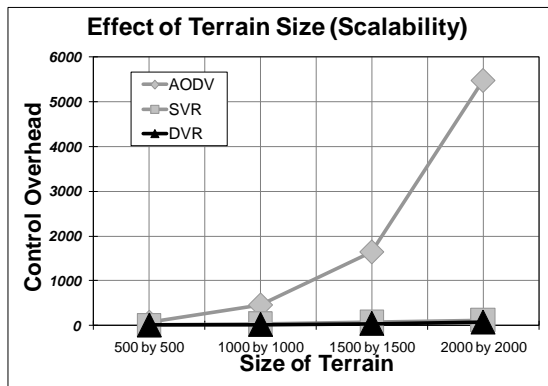
(c)

Figure 22. Effect of Terrain Size: (a) Data Deliver Rate, (b) End-to-End Delay, and (c)

Overhead.

46

3.3.1 Simulation Parameters

To test the robustness of the proposed DVR technique, we study this protocol in a terrain that is relatively large compared to the radio range of the nodes. The large setting allows us to include many routes that are several hops long in the simulations. The field configuration is a $1000m \times 1000m$ field with the radio range of nodes set to 133 meters.

Initially, nodes are uniformly distributed over the terrain. Existing mobility traces focus on either a small number of nodes [6] or nodes moving at pedestrian speed [23][73]. To show the robustness of the virtual router approach, we considered 1000 nodes moving at 20 meters per second (m/s) with zero pause time. The node behavior was modeled according to the Random Waypoint Mobility model which is also used in many other studies [10][35][68]. In this model, each node randomly selects a destination point. When the node reaches this destination point, it pauses for a period of time, and then selects another destination point.

The traffic applications are constant-bit-rate sessions involving 1/20 of all the nodes (i.e., 25 concurrent sessions) transmitting 512-byte data packets. The length of each simulation is 15 minutes with sessions randomly start throughout each simulation run. Each session lasts 3 minutes.

3.3.2 Performance Metrics

The protocols are compared under three performance metrics, which were suggested by the IETF MANET working group for routing protocol evaluation [15]. These metrics are: (1) *Fraction of Packets Delivered* − measures the ratio of the data packets delivered to the

destinations and the data packets generated by the sources. This number indicates the effectiveness of a protocol. (2) *End-to-End Delay* – measured in seconds (s), includes processing, route discovery latency, queuing delays, and retransmission delay at the MAC, and propagation and transmission times. This number measures the total delay time from a source to a destination. (3) *Normalized Routing Load* or *Overhead* – measures the number of control packets transmitted per data packet delivered to the destination. The routing overhead is an important metric for comparing these protocols as it measures the scalability and efficiency of a protocol in terms of throughput and power consumption.

3.3.3 Simulation Results

We present the simulation results in the following subsections.

3.3.3.1 Effect of Mobile Speed

To understand the effect of mobile speed on performance, we varied the speed of the mobile nodes between 10 and 25 meters per second (*m/s*). The simulation results are given in Figure 20.

The results demonstrate that physical routes break frequently under high node speeds. To maintain good delivered rate, AODV issues frequent route requests. This, however, incurs significant control overhead (see Figure 20(c)), and results in long delay for many data packets waiting at the source (see Figure 20(b)). In contrast, DVR is much more tolerant of node mobility. It allows any nodes of a virtual route to participate in data forwarding whereas AODV

restricts data to be relayed only by nodes chosen for a specific route. Furthermore, efficient route update is performed periodically in DVR to prevent virtual link break. Consequently, DVR rarely experiences link break and has a higher data delivered rate (see Figure 20(a)) with short delay (see Figure 20(b)) compared to AODV. This is achieved using only a fraction of control packets (see Figure 20(c)). Figure 20(b) and Figure 20(c) show that DVR has 4 times and 21 times improvement in delay and control overhead, respectively.

Figure 20(a) indicates that DVR can achieve similar data delivered rate as SVR without using GPS. The use of virtual routers allows both protocols to be tolerant of node mobility; and both can achieve high data delivered rates. However, DVR has a significantly lower end-to-end delay as shown in Figure 20(b). This can be explained as follows. Communication routes in SVR are constructed on predefined virtual routers, and generally involve a greater number of hop counts compared to communication routes in DVR, which are based on virtual routers created dynamically to better leverage the current underlying network topology. In terms of control overhead, DVR is also slightly better due to the following reason. In SVR, the destination node updates its new location with the source node through the virtual routers currently in the connection. This is different from DVR, in which a route update packet is sent back to the source node over a set of predetermined relay nodes. While the former case has an impact on all the physical nodes in the affected virtual routers, the latter only impacts the relay node. This phenomenon results in more control overhead for SVR as observed in Figure 20(c).

49

3.3.3.2 Effect of Node Density

In this study, we examine the effect of network density by varying the number of nodes between 600 and 3000 nodes. The simulation results are presented in Figure 21.

The results show that the high density increases the overhead without improving the delivered rate for AODV. This is due to the fact that every time a source node issues a route request to repair a route, almost all the nodes need to relay the request at least once. The increase in the routing messages incurs more overhead (Figure 21(c)) and has a negative impact on the data deliver rate (Figure 21(a)). In Figure 21(b), we observe a worse end-to-end delay for a higher density of nodes. This is due to the fact that more nodes are responding to every route request causing congestion in the wireless medium and therefore increasing the end-to-end delay for data packets.

Figure 21(a) and Figure 21(c) show that, as node density increases, AODV uses much more control packets to repair routes but the delivered rate still does not improve. On the contrary, DVR only requires slightly more control packets (due to more nodes responding to initial requests) for a higher density to maintain a delivered rate significantly higher than that of AODV. Figure 21(b) and Figure 21(c) show that, at the highest density, DVR provides 6 times reduction in delay and 20 times reduction in overhead, respectively, compared to AODV.

Again, we observe that SVR performs similar to DVR in terms of data delivered rate and control overhead. However, DVR is significantly better in term of end-to-end delay. The explanation for the latter is the same as in the previous subsection.

3.3.3.3 Effect of Terrain Size

To determine if the techniques under consideration are sufficiently scalable to allow for a large-area deployment, we increased both the network area and the number of nodes together as follows, maintaining a constant node density:

- 500m × 500m area with 250 nodes

- 1,000m × 1,000m area with 1000 nodes

- 1,500m × 1,500m area with 2250 nodes

- 2,000m × 2,000m area with 4000 nodes

The results are presented in Figure 22. They indicate that AODV is not efficient in maintaining long routes. As the terrain gets larger, the length of routes and the likelihood of route breaks also increase. By not using fixed hop-by-hop routes, DVR experiences fewer packet drops than AODV (see Figure 22(a)) with only minimal increase in route update overhead (see Figure 22(c)). The slight increase in the delay for DVR with the increases in the terrain size, shown in Figure 22(b), is due to longer routes the data packets need to traverse before arriving at the destination nodes.

The increase in the delay and control overhead for AODV is significantly more as seen in Figure 22(b) and Figure 22(c)), respectively. This is due to the fact that the routes become longer for a larger terrain; and they break more frequently causing an increase in the number of route request messages. Such control messages cause more overhead and congestion in the wireless medium, which increases the end-to-end delay for data packets.

In comparison, we observe in Figure 22(b) and Figure 22(c) that DVR is 3 times better in delay and 73 times better in overhead, respectively, compared to AODV when the terrain is 2,000m × 2,000m. We also note that use of virtual routers also provide good performance to SVR, however, its end-to-end delay is still more than that of DVR.

3.3.3.4 Effect of Very Low Node Density

In this study, we examine the effect of very low network density by varying the number of nodes between 100 and 500 nodes. Figure 23 presents the simulation result on data delivered rates of the three protocols.
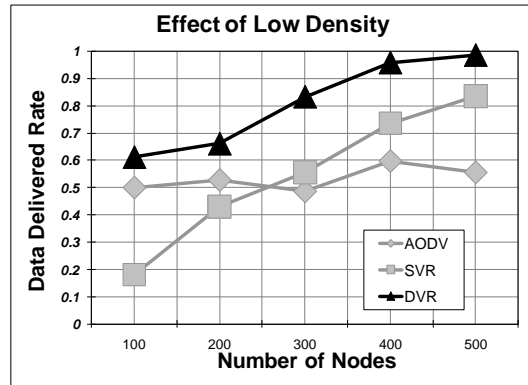
Figure 23. Effect of Low Density: Data Delivered Rate

The result shows that none of the protocols performs well when the density is too low, i.e., a disconnected network. However, the virtual router techniques start to deliver better performance as soon as the number of nodes is increased to 300 nodes. Between the two virtual router techniques, DVR provides better data delivered rate because its communication routers can better adapt to the underlying physical network topology. Routing is less flexible in SVR.

Its communication routes must use the virtual routers selected from a set of predefined virtual routers.

3.4 <u>Summary</u>

In this chapter, we examined the impact of node mobility on existing MANET designs; and proposed a technique, called *Dynamic Virtual Router* (DVR), to better handle the high mobility of mobile nodes. A virtual router is defined as a particular geographical area, with its routing functionality realized by physical nodes currently in the geographical region served by the virtual router. In this framework, data transmission between two nodes is done over a sequence of virtual routers. Since these virtual routers do not move, the communication connection is much less susceptible to node mobility. Although virtualization is not new in solving computing problems, the proposed Virtual Router approach is the first to use virtualization in designing routing techniques for MANETs. To assess its performance, we performed simulation studies to compare both the *Static Virtual Router* (SVR) approach and DVR approach with the *Ad-hoc On-Demand Distance Vector* (AODV) routing technique. The simulation results indicate that both virtual router techniques can handle very high mobility, and achieve several times better performance than that of AODV. Between the two virtual router approaches, DVR, with the flexibility of dynamically creating virtual routers on demand, shows significantly better end-to-end delay. DVR also has the benefit of not requiring each mobile node to equip with a global positioning system. Nodes also do not need to store information about whereabouts of the virtual routers.

4. STATIC VIRTUAL ROUTER FOR BROADCAST

4.1 Introduction

A mobile ad hoc network (MANET) consists of a set of nodes like laptops or PDAs that autonomously establish communication in a peer-to-peer fashion without needing predeployed infrastructure such as a central router or base station. Broadcasting in an ad hoc network is an important process by which routing protocols request information like routes or the location about a destination. Most of these routing protocols use a simple Plain Flooding technique [36] that can cause a broadcast storm, especially in a high density environment [60].

A recent paper by Ying Cai, et al. [11] divides current broadcasting techniques into 0-hop-, 1-hop-, and 2-hop-based approaches. In the 2-hop based approaches ([58][65][76][83], etc.), nodes either proactively select neighbors to be rebroadcast hosts or reactively determine whether to rebroadcast based upon location information about their 2-hop neighborhood. However, it is difficult to maintain this information in a high mobility environment. In the 1-hop based approaches like [54], a node only needs to know about its 1-hop neighbors through simple periodical beacon signals from other nodes. *Edge Forwarding* [11] is another 1-hop based technique. Although *Edge Forwarding* provides 100% reachability with minimal redundancy, it adds extra overhead (that is a beacon signal) for routing protocols such as Dynamic Source Routing [43], Ad-hoc On-Demand Distance Vector Routing [69], and ConnectionLess Approach to MANET [35] that do not use periodic beaconing. Also, in highly dense and mobile environments (such as ad hoc networks in battle grounds), nodes need to broadcast their

locations more frequently, congesting the wireless medium. To eliminate additional overhead of location beacons, 0-hop approaches can be used by the aforementioned routing protocols.

Earlier 0-hop protocols such as *Counter Based* [60] and *Probabilistic Based* [71] use random delay and simple rebroadcasting conditions. The rebroadcasting conditions require a node to rebroadcast a received packet unless it has seen the same packet more than a predefined threshold of times or its randomly chosen probability is less than the predefined probability threshold. These rebroadcasting conditions provide a simple way to significantly reduce control overhead compared to Plain Flooding. However, their random delay mechanism does not guarantee nodes whose transmission range cover more nodes (that have not received the packet) get to rebroadcast. Often, a node like this is inhibited from rebroadcasting either because of its chosen probability or because it has seen several duplicate messages, possibly from nodes close to the broadcasting node. As result, this random delay mechanism uses more nodes than necessary to broadcast in order to cover the entire terrain.

To provide a better delay mechanism, 0-hop protocols such as *Angle Based* [77], *PANDA* [52], and *Border Aware* [86] use a distance delay algorithm. The distance delay allows nodes near the edge of a transmission range to rebroadcast first. This delay mechanism allows nodes close to the broadcasting node to cancel their rebroadcasts because their transmission ranges are completely covered by the transmission ranges of the faraway nodes. In order for a node to have its transmission range covered completely, other rebroadcasting nodes must encircle that node. However, this condition is difficult to happen for nodes near the edge of a transmission range. As result, most of the nodes near the edge of a transmission range still need to rebroadcast.

In this chapter we propose *Cell Broadcast*, a 0-hop broadcast protocol that significantly reduces redundancy without the use of beaconing, while maintaining complete reachability in a high density environment. This technique divides a terrain into cells which assist a node in determining its geographic relationship with a broadcasting node. This geographic relationship can eliminate rebroadcasts not only from nodes close to a broadcasting node but also from a majority of the nodes near the transmission edge of the broadcasting node. The effect is that, in a high density environment, only a few nodes located near the 4 diagonal corners of a transmission range need to rebroadcast to maintain 100% reachability. To the best of our knowledge, this effect is not present in any of the existing techniques that do not use location beaconing.

The remainder of this chapter is organized as follows. In Section 4.2 we present our Cell Broadcast technique. We show simulation results in Section 4.3 to demonstrate the benefits of the proposed technique, and present our conclusions in Section 4.4.

4.2 Proposed Solution: Cell Broadcast

As do many of the aforementioned broadcasting protocols [11][52][77][86], we assume that all nodes can obtain location information provided by technologies such as the *Global Positioning System* (GPS) [19]. This is a reasonable assumption because of the increasing availability and pervasiveness of these devices and because the GPS service is provided without charge. In the case that GPS is not available, it is plausible that nodes may calculate their positions with a localization scheme—a research area that has recently received a lot of attention [72]. Positioning or GPS devices can, in fact, provide 3-D location information in terms of

longitude, latitude, and altitude. In this chapter, for simplicity, we use an *xy*-coordinate system in place of longitude and latitude. Although standalone GPS is not accurate to a precise degree, there are technologies available that can, when incorporated with GPS, improve its accuracy to within centimeters [80], which is more than sufficient for our protocol.

The idea of dividing a network area into smaller "virtual cell areas" is not new. Cells are also used by other protocols, e.g., for routing data [35], location service management [42], and power management [84]. In this chapter, we exploit the cell concept to reduce broadcast packets.

4.2.1 Virtual Cell

The network area is divided into small "virtual cells". These cells, defined by the coordinates of their upper right and lower left corners, are each assigned a unique cell ID. Each virtual cell has eight *neighboring* cells (see Figure 24. (A)). Note that the network area can also be divided into "cellular-like" cells, but for simplicity, we use square cells in the discussion of this chapter and our simulation study.

(A)　　　　　　　(B)
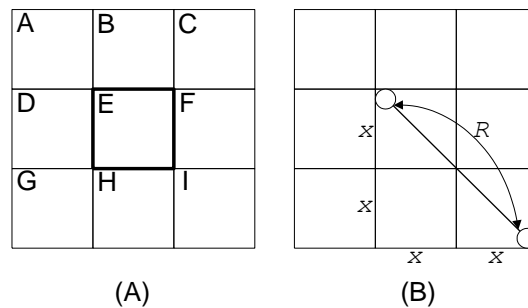
Figure 24. (A) Cell E has eight neighboring cells A, B, C, D, F, G, H, and I. (B) R is the farthest distance between any two nodes in two neighboring cells.

57

We construct our virtual cells based on the nominal radio range R as follows. Assume each virtual cell is square with x units on each side. The distance between any two, possible farthest, nodes in any two neighboring cells must not be larger than R (see Figure 24. (B)). Therefore, we have:

$$\sqrt{2}(2x) \leq R \quad \text{or} \quad x \leq \frac{R}{2\sqrt{2}} \tag{2}$$

In other words, the virtual cell is designed such that, for any two neighboring cells, all nodes in one cell can communicate with all nodes in its neighboring cells.
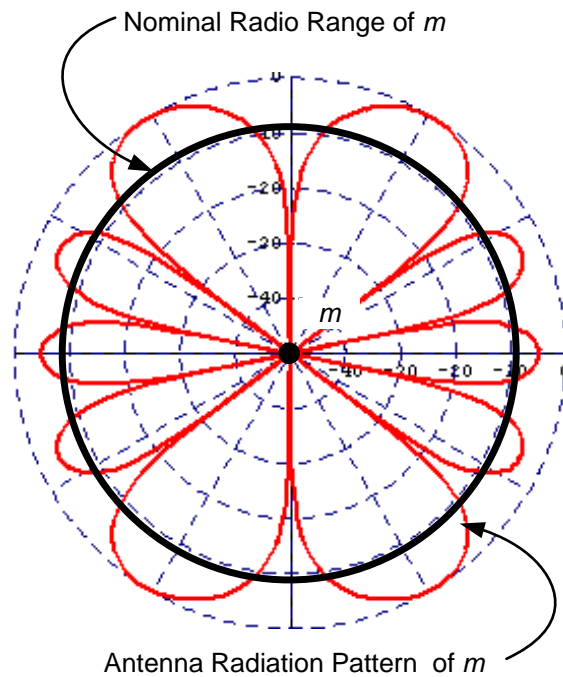
Figure 25. Radio signal strength is adjusted to allow the radiation pattern to cover the nominal radio range.

58

Because the radiation pattern of an antenna is not circular [3], a node may need to broadcast a stronger signal in order to for most of the nodes within distance R to be able to receive its broadcast. In this chapter, we refer to R as the nominal radio range of a node. A node can determine the desired signal strength by looking at its H-plane radiation pattern provided by a priori analysis on the antenna [21]. A signal is strong enough if the resulted radiation pattern can encompass α% of a circle of a radius equals to R as shown in Figure 25. In this chapter, we set α = 90. The radiation pattern in Figure 25 is from http://www-antenna.pe.titech.ac.jp/~hira/hobby/edu/em/dipole/. Also, many recent research works are developing antennas that have near circular H-plane radiation patterns [4][28].

4.2.2 Guaranteed Flooding Region

When a node n broadcasts a packet, nodes in its 8 neighboring cells definitely can hear this packet as they are within the nominal radio (or transmission) range of n. We refer to the 8 neighboring cells as the Guaranteed Flooding Region (they are also the 1-hop Neighboring Cells) of n. We refer to cells that are adjacent to the 1-hop cells as the 2-hop Neighboring Cells of n. In Figure 26, the cells inside the thick rectangle compose the Guaranteed Flooding Region of n, and those on the outside are the 2-hop Neighboring Cells of n.

2-hop Neighboring Cells of *n*

n

Guaranteed Flooding Region of
n's broadcast
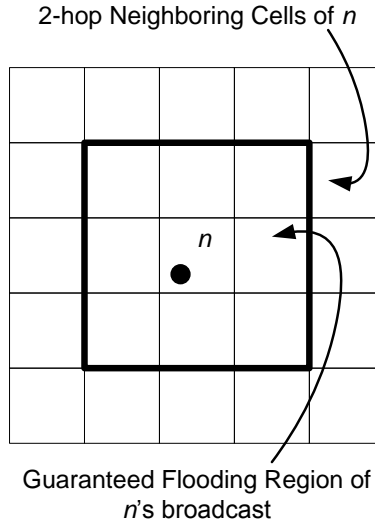
Figure 26. 1-hop and 2-hop Neighboring Cells of *n*

4.2.3 Initialization Phase

When a new node enters the network area, it first contacts any nearby node to obtain partition information. The partition information given out by a node *n* includes the size of the terrain area, the size of the virtual cell, and the cell ID and coordinates (of the two diagonal corners) of the virtual cell currently containing *n*. With this partition information the new node can easily compute the location and cell ID of other cells in the entire terrain area, as every virtual cell is relative to its neighboring cells in terms of its coordinates and cell ID. The new node can now determine which virtual cell it is contained in by comparing its own location with the coordinates of the virtual cells. The algorithm used to efficiently match a node to a cell is beyond the scope of this chapter and will not be discussed.

4.2.4 Broadcast Procedure

To broadcast a request packet for a destination, a source node includes the following information in the packet: *Source Node ID*, *Destination Node ID*, *Packet ID*, and *Current Node Position*. The *Source Node ID* and *Destination Node ID* fields are the node IDs of the source and destination nodes, respectively. The *Current Node Position* is the position of the node about to broadcast this packet. (Thus, each intermediate node updates the *Current Node Position* header field before rebroadcasting the packet.)

When a node *n* receives a packet from another node *m*, it will first find the cell containing *m*'s position, as indicated in the packet header. It then determines if it needs to rebroadcast the packet according to the following Cell Broadcast algorithm.

1. If *n* has rebroadcasted this packet before, *n* drops the packet.

2. If *n*'s cell is not a 1-hop or 2-hop Neighboring Cell in relation to *m*'s cell, *n* drops the packet.

3. If *n* has heard the same packet broadcasted from its cell, *n* drops the packet.

4. If all of *n*'s 8 1-hop Neighboring Cells are covered by the existing broadcasts heard by *n*, *n* drops the packet. A cell is said to be covered by a broadcast if all of its four corners are within the *R* distance from the broadcasting node.

5. If all previous conditions fail (i.e. *n* might need to broadcast the data), *n* sets the delay.

6. If at the end of the delay, all 8 of *n*'s 1-hop Neighboring Cells have been covered by existing broadcasts, *n* drops the packet.

7. Otherwise, *n* rebroadcasts the packet.

In Step 5 of the above algorithm, *n* sets its delay as follows:

1. If *n* is on a 2-hop cell that is a diagonal cell of *m*'s cell, *n* will set its delay as $rand(node \rightarrow seed) \times t$ seconds.

2. If *n* is on a 2-hop cell that is not a diagonal cell of *m*'s cell, *n* will set its delay as $rand(node \rightarrow seed) \times t + t$ seconds.

3. If *n* is on a 1-hop cell that is a diagonal cell of *m*'s cell, *n* will set its delay as $rand(node \rightarrow seed) \times t + 2 \times t$ seconds.

4. If *n* is on a 1-hop cell that is not a diagonal cell of *m*'s cell, *n* will set its delay as $rand(node \rightarrow seed) \times t + 3 \times t$ seconds.

We say that *n*'s cell a diagonal cell of *m*'s cell if the centers of the two cells have equal absolute differences (greater than zero) in both the x and y dimensions. We allow the nodes in the cells diagonal to *m*'s cell forward first because we can cover the maximum area with the fewest nodes. As shown in Figure 27, only 4 rebroadcasts are needed to cover the Guaranteed Flooding Region of a rebroadcast from any of the 2-hop or 1-hop Neighboring Cells of *m*. Additional rebroadcasts from any of *m*'s 1-hop or 2-hop Neighboring Cells do not provide much additional coverage. When combined with the broadcasting condition, the proposed delay technique can significantly reduce control overhead compared with existing broadcasting techniques while maintaining 100% reachability in a high density environment. Note that this angle based delay technique has not been used by any existing broadcasting techniques.
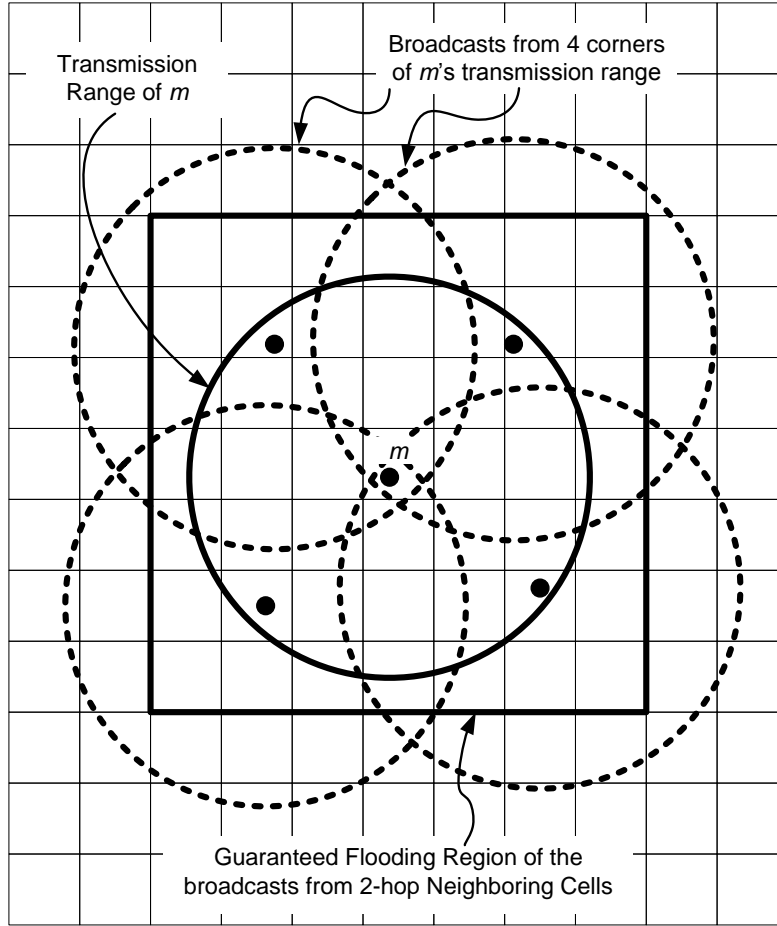
Figure 27. Existing broadcasts cover the Guaranteed Flooding Region of the 2-hop Neighboring

Cells of *m*.

Although the Cell Broadcast algorithm is designed to handle requests without a priori

knowledge of a destination's location, Cell Broadcast can also be applied, with some

modifications, to a geographic request region that is specified by protocols such as Location-

Aided Routing (LAR) [47] based on the known location of a destination. For example, a node

only needs to consider the 1-hop Neighboring Cells within the region when deciding whether or

not to rebroadcast a packet. Similarly, Cell Broadcast can also leverage a map database (such

database is often provided by the GPS device installed on a PDA) to find obstacles and ignore those cells that are occupied by the obstacles. Also, if a node n in a 2-hop Neighboring Cell of a broadcasting node perceives that the closest 2-hop diagonal cell is occupied by an obstacle or outside of a request region and n's cell is adjacent to the obstacle or the region border, n shortens its delay to $rand(node \rightarrow seed) \times t$. If n is in a 1-hop Neighboring Cell and under a similar situation with its closest 1-hop diagonal cell, n sets its delay as $rand(node \rightarrow seed) \times t + 2 \times t$.

4.3 Simulation

To evaluate our approach, we perform simulations using the network simulator called GloMoSim [85]. This simulator, developed at UCLA, is a packet-level simulator specifically designed for ad hoc networks. It follows the OSI 5-layer network communication model and provides comprehensive simulation for each of the 5 layers.

We use 802.11 for the MAC layer protocol, and for the radio layer, we use two-way propagation path loss, signal to noise reception, and noisy radio medium models. The two-way propagation model uses Friss free space path loss for near sight communications and plane earth path loss for far sight.

We simulate and compare the following broadcasting protocols: Plain Flooding, Counter Based, Angle Based, and our Cell Broadcast. All the protocols use 60 milliseconds for each delay period. For Counter Based, we set the value of the counter threshold to C = 3 in accordance with [60]. For Angle Based, we set the value of $\varepsilon = 2$ according to [77].

We perform the simulation study with respect to node density, terrain size, and application load. The field configuration is a 1000m × 1000m field where each cell is 100m × 100m. There are 300 nodes in the terrain and each node has a nominal radio range of about 300 meters. The nominal radio range is used by the Angle Based and Cell Broadcast protocols and set with respect to radio transmission power according to Section 4.2.1. Initially, each node starts at a random location uniformly distributed across the terrain. We employ the *random waypoint* mobility model. That is, each node randomly selects a destination point. When the node reaches this destination point, it pauses for a period of time, and then selects another destination point. In simulations, each node moves at 10 meters/second with 0 seconds of pause time at a waypoint. During each 15 minute simulation run, 1/10 of the nodes are randomly picked at random times to initiate request packets, 500 bytes each.

The protocols are compared under three performance metrics: (i) **average control overhead** – measures the average ratio of the nodes *transmitting* each request packet (80% means that, in average, 80% of nodes are involved to transmit each request packet); (ii) **average reachability** – measures the average ratio of the nodes *receiving* each request packet (100% means every node receives every request packet); (iii) **average delay** – is measured in seconds, the average interval from the time each request packet is initiated to the time the last node receives the packet.
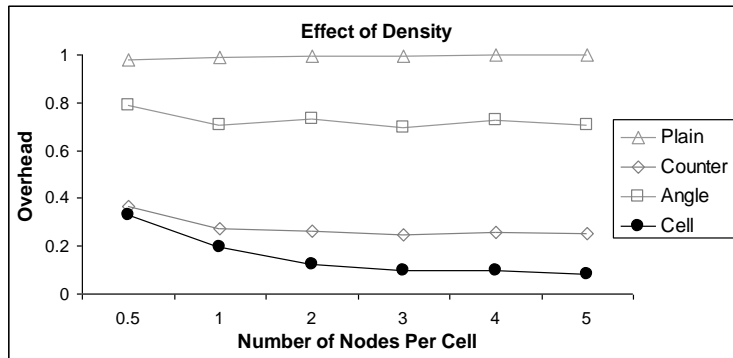
(a)
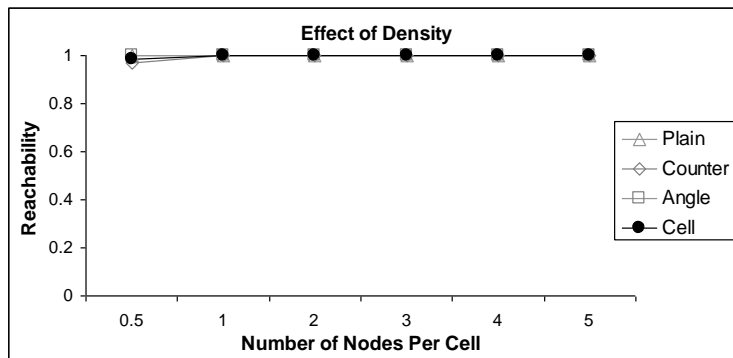
(b)
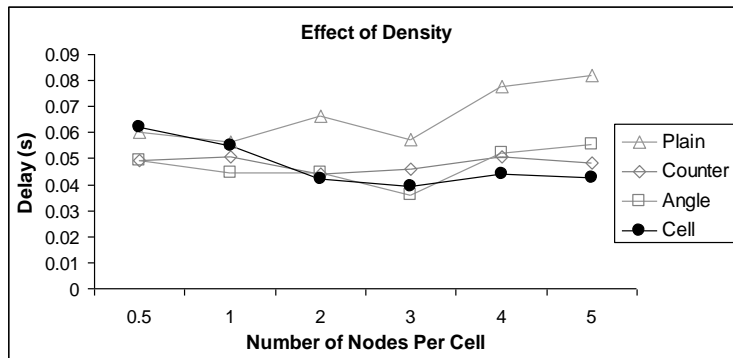
(c)

Figure 28. Effect of Node Density: (a) Average Overhead, (b) Average Reachability, and (c)
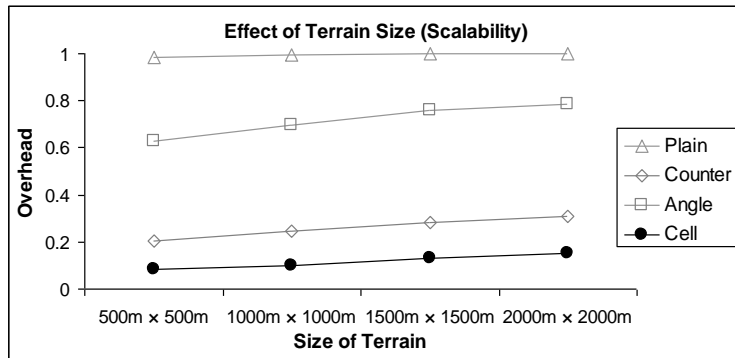
Average Delay

(a)
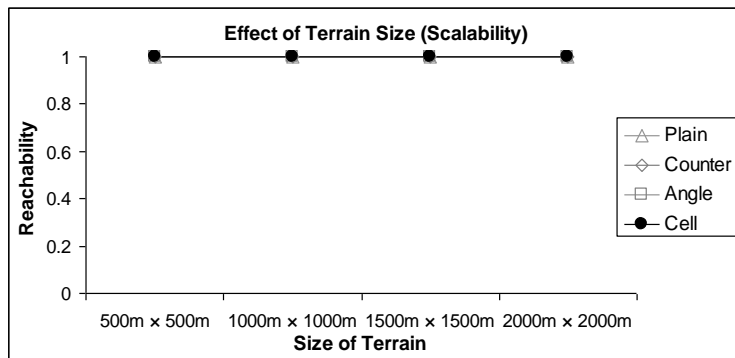
(b)
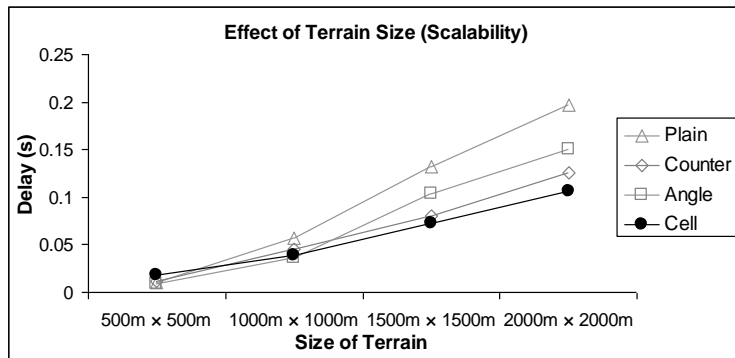
(c)

Figure 29. Effect of Terrain Size: (a) Average Overhead, (b) Average Reachability, and (c)

Average Delay

67
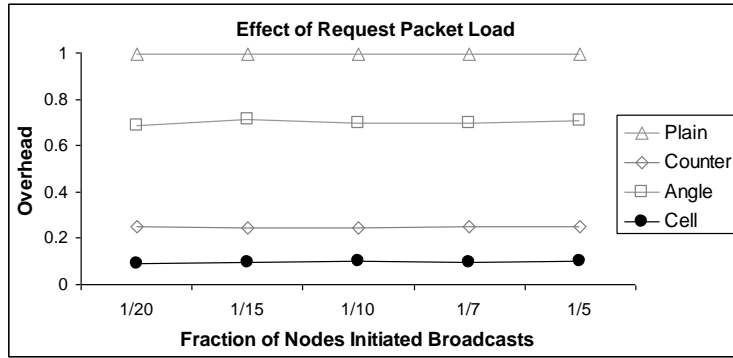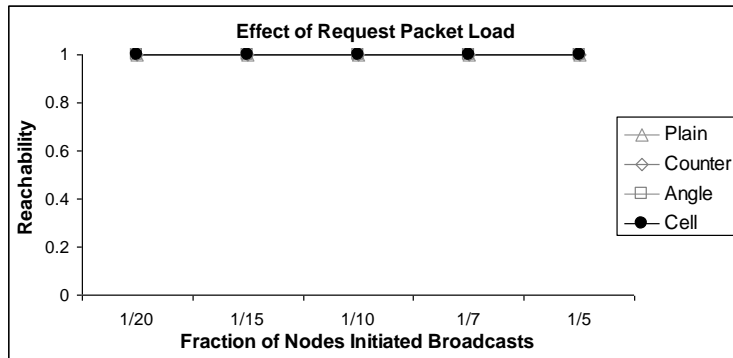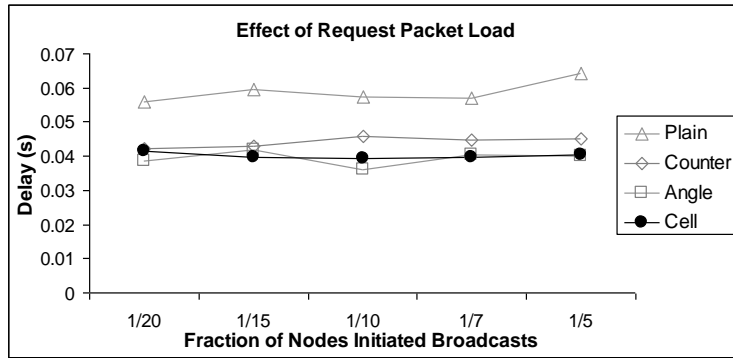
Figure 30. Effect of Request Packet Load: (a) Average Overhead, (b) Average Reachability,

and (c) Average Delay

4.3.1 Simulation Results

We present the simulation results in the following subsections. We study the effects of node density, terrain size, and application load.

To understand the **effect of node density**, we varied the initial node placement from 0.5 to 5 nodes per cell (or the number of nodes from 50 to 500 evenly distributed in the terrain). The simulation results are shown in Figure 28. As expected, Plain Flooding has the highest control overhead because every node needs to broadcast. Figure 28(a) shows that an intelligent delay function alone is not sufficient to reduce the control overhead. Although Angle Based has a more intelligent delay function than Counter Based, its forwarding condition causes most of the nodes near the edge of a transmission range to rebroadcast because the condition is difficult to be invalidated for these nodes. On the other hand, our Cell Broadcast takes advantage of the fact that additional rebroadcasts besides the 4 corner rebroadcasts do not provide much additional coverage (as described in Section 4.2.4). As result, Angle Based has a higher overhead than Counter Based while Cell Broadcast has a lower overhead. Figure 28(a) also shows that Cell Broadcast is more suitable in high density environments. In high density environments, Cell Broadcast can rely on nodes in the 4 corners of a nominal radio range to rebroadcast the packet. However, in a low density, there may not be nodes in the four corners, and nodes in other parts of the radio range will need to rebroadcast. As result, Cell Broadcast uses a higher fraction of nodes to broadcast request packets in a low density environment. Figure 28(a), together with Figure 28(b), show that in high density environments, an efficient broadcasting algorithm only needs few broadcasts to maintain 100% reachability. Cell Broadcast maintains 100% reachability

despite the fact that it requires only a fraction of the overhead consumed by other protocols. This shows that Cell Broadcast has a more efficient broadcasting algorithm than the other protocols. Figure 28(c) shows the delays of the protocols. Because Counter and Angle Based use fewer nodes than Plain Flooding to broadcast, they have shorter delays than Plain Flooding. In a low density environment, a protocol can use more nodes to propagate a packet to every node faster without significantly congesting the wireless medium. Thus, Counter and Angle Based have shorter delays than Cell Broadcast even though they have higher control overheads. However, as density increases, more nodes share the fixed amount of available bandwidth. As a result, Cell Broadcast has the shortest delay in high density terrains because it uses the fewest nodes to broadcast packets, as indicated in Figure 28(c).

To understand the **effect of terrain size**, we varied the size of the terrain from 500m × 500m to 2000m × 2000m. The simulation results in Figure 29 show that Cell Broadcast scales well to allow large-area deployment. Figure 29.a shows that the effective delay and broadcasting condition give Cell Broadcast a much lower overhead in a small terrain and a slower increasing trend as the terrain gets larger. In fact, Cell Broadcast involves a smaller portion of nodes to broadcast in the largest terrain than Counter Based does in the smallest terrain. Figure 29(c) shows that Cell Broadcast propagates packets faster than the other protocols in the 2 largest terrains. In a small terrain, a protocol can use more nodes to broadcast to reach every node faster since most of the nodes can be reached within one or two hops from the source. However, as the terrain gets larger, more hops are needed to reach nodes on the far side of the terrain. A protocol like Plain Flooding or Angle Based that uses many nodes to broadcast can cause congestion, and

70

a protocol like Counter Based that uses a naïve delay function can cause a packet to have to traverse more nodes (with each hop causing an additional delay) than is absolutely necessary to reach a faraway node. As a result, Cell Broadcast has a shorter delay compared to the other protocols as the terrain gets larger.

To understand the **effect of request packet load**, we varied the ratio of nodes initiating request packets from 1/20 to 1/5 of all the nodes. The simulation results are shown in Figure 30. The purpose of this study is to show that Cell Broadcast can support high request loads. Cell Broadcast has the lowest overhead and one of the lowest delays among all the protocols even when there are many nodes initiating network wide requests. In Figure 30(c), the distance based delay lets Angle Based to have a short delay, but our novel delay technique permits our protocol to have a delay that is both short and stable.

4.4 Summary

In this chapter, we propose a 0-hop broadcasting technique called Cell Broadcast to reduce broadcasts in a high density environment. Compared to the other simulated protocols, our novel delay mechanism and broadcasting condition significantly reduce broadcasts in high density environments, propagate a request packet much faster in large terrains, and maintains a short stable delay under different request loads. In all the scenarios, Cell Broadcast provides 100% reachability.

5. STATIC VIRTUAL ROUTER FOR BROADCAST IN STREET ENVIRONMENTS

5.1 Introduction

A vehicular network is a form of mobile ad hoc network. In a vehicular network, nodes are vehicles equipped with wireless communication devices. Nodes roam within the confines of a road network and communicate with each other wirelessly. Many vehicular applications can benefit from an efficient broadcast protocol. One example is message dissemination in a disastrous situation where the communication infrastructure may not be available. In [26], we propose an intelligent transportation system that utilizes specialized traffic signals to guide traffic away from an incident. To communicate without traffic signals, an ad hoc network can be used to inform nearby motorists about the incident and the associated evacuation plan. On-board intelligent navigation systems would be able to process the evacuation plan and guide the motorists away from the incident in a coordinated manner. Another example application is reducing control overhead in routing protocols [16][17][35][57]. In these routing protocols, establishing a connection with a node requires either querying a location service [51][75] or broadcasting a message to search the entire network for the node. The location service requires ongoing maintenance that is sensitive to the effect of high mobility and not needed when routing protocols such as [35][39][57] already provide a location update mechanism. For obtaining the location of a destination for the first time, these routing protocols can benefit from an efficient mechanism other than a simple Plain Flooding technique [36] that can cause a broadcast storm, especially in a high density environment [60].

A vehicular network in a street environment with relatively narrow streets surrounded by large buildings provides a very short window of time for communication between any two nodes moving at high speed on different streets. This short window means the network topology among nodes changes very fast when dealing with general protocols that are not sensitive to a street environment. For example, in the island of Manhattan in New York City, many streets are 20 meters wide, and measure in length from approximately 60 meters to 180 meters. Vehicles traveling in parallel in different streets that want to communicate only have a very short 20 meter window to do so every 60 to 180 meters, assuming that the cross street is sufficiently short to allow the communication to occur. Broadcast protocols [11][20][54][58][83] that rely on one or two hop neighborhood information either suffer from topology information that quickly becomes outdated, or need to exchange neighborhood information more frequently, which can cause congestion in the wireless medium.

In addition to changing topology, large buildings limit the coverage of a broadcast. A node's broadcast often cannot reach a node on another street. Earlier broadcast protocols such as Counter Based [60] and Probabilistic Based [71] provide simple ways to reduce overhead in a plain terrain environment without neighborhood information. These 0-hop protocols, however, usually fail to choose broadcasting nodes optimally in a street environment. The problem is that not all nodes should be treated equally in a street network due to obstacles. Nodes at intersections have better reachability to other nodes, compared to nodes in road segments. Techniques developed for an open space, not taking this factor into consideration, would not be able to achieve good performance in a street environment.

Another challenge present in a street environment is constraints in node mobility. In plain terrain, advanced protocols such as Angle Based [77], PANDA [52], Border Aware [86], and Cell Broadcast [29] use distance delay and geographic relationship as their rebroadcast criteria. With these strategies, many nodes can refrain from rebroadcasting because a message can be more efficiently rebroadcast by some neighboring node. As an example, a node in the Angle Based technique would not rebroadcast if the range of its rebroadcast is completely covered by some of the recent node broadcasts for the same message it overhears in the neighborhood. This strategy cannot be used for a street environment since most nodes would rebroadcast. This happens because the radio range of any node cannot be covered completely even if all its neighbors broadcast the message. According to the protocol, this node should rebroadcast even though its rebroadcast will not likely reach additional nodes.

MAC-layer protocols, such as [49][50][78], focus on reducing interference to improve the use of the wireless medium. These protocols modify the broadcast mechanism of the underlying MAC protocol that a routing protocol operates on. While this eliminates the need to modify the networking-layer protocol, these protocols have disadvantages (over a network-layer approach) such as relying on expensive repeaters [49], added delay due to sequential directional broadcast in intersections [50], or reduced spatial reuse of the wireless medium due to larger busy tone broadcast [78].

The broadcast protocol proposed in this paper is a network-layer approach that focuses on improving the use of network bandwidth through a reduction in the number of nodes needed in order to propagate a network wide message, such as a route or location request message used in

74
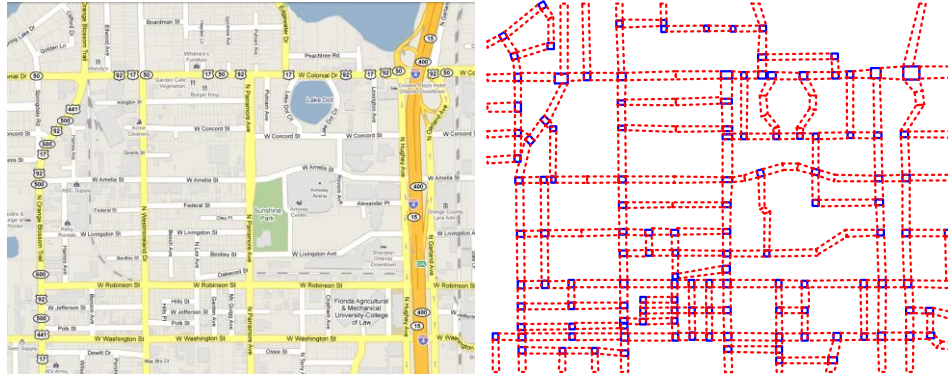
Figure 31. A street network and the cells specific to the street environment.

many routing protocols. We assume each node is equipped with a GPS (*Global Positioning System*) unit. The terrain is divided into cells to allow nodes to easily determine their geographic relationship with a sender and decide whether to rebroadcast the message. The design is an adaptation of a broadcast technique, called *Cell Broadcast* (CB) proposed in [29] for an open terrain environment. Although CB provides high reachability and involves fewer nodes in message dissemination, it does not work well in a street environment due to similar issues that arise in other wireless broadcast models not specifically adapt for the street model. We name the new technique *Cell Broadcast for Street Environment* (CB-S). The new design addresses a number of challenges found in this environment, including faster topology change, limited radio coverage, and constrained node mobility.

The remainder of this paper is organized as follows. We introduce the proposed Cell Broadcast for Street Environment (CB-S) in Section 5.2. The simulation setting is presented in Section 5.3, with the simulation results discussed in Section 5.4. Then we analyze the CB-S in Section 5.5 and finally, we present our conclusions in Section 5.6.

75

5.2 <u>Proposed Solution:</u>
<u>Cell Broadcast for Street Environments (CB-S)</u>

With increasing availability of on-board navigation systems, more and more vehicles are equipped with GPS and map systems. Many works such as driver assistance [79], routing [57], and traffic condition dissemination [82] leverage such systems to improve safety and communications in vehicular networks. The proposed CB-S technique focuses on improving vehicular communications, and can have many important applications. It can be used to reduce the high cost of request packets in routing protocols such as [35][57]. CB-S can also be used to reduce the overhead of disseminating network wide information, such as information about a particular event [26], to vehicles within a radius. The initiator of such messages can be one of the vehicles or a roadside unit connected to a venue such as an arena. A street terrain is first divided into intersections and street segments. An intersection is a cell and a street segment can be one cell or several cells depend on the length and other characteristics of the segment. Figure 31 shows an example street terrain and the cells on the local streets used by CB-S. When a street segment is longer than the nominal radio range of nodes or has a curvature that prevents a rebroadcast from covering the entire segment, the segment is divided into several cells. Compared to nodes on street segments, nodes on intersections are better candidates to relay packets as they can reach nodes on intersecting streets. Thus, nodes on intersections are used whenever possible to relay packets. In addition to reach nodes on more streets, rebroadcasts from intersections can also eliminate or reduce rebroadcasts from street segments in between intersections. In this section, we first describe our street environment and then present the

proposed technique. For the sake of clarity, we refer to the rebroadcast of a data packet from a single node as a *rebroadcast* or a *node broadcast*. The entire process of disseminating a data packet to all the target nodes in the network is referred to as a *broadcast*. Thus, a broadcast consists of many node broadcasts working together to disseminate a message to the target nodes in the wireless networks.

5.2.1 Cell in Street Environment

To support a street environment, we divide streets into cells according to intersections and blocks (i.e., the street segment in between two adjacent intersections). An example is given in Figure 32. Cell A and Cell C at the intersections are called *intersection cells*. Cell B is a street segment spanning an entire block and is called a *segment cell*. In this paper, we assume that all nodes or vehicles have a radio range sufficient to cover any three consecutive cells (e.g., Cells A, B, and C in Figure 32). With this requirement, any node in Cell A can communicate with any node in Cells B and C.
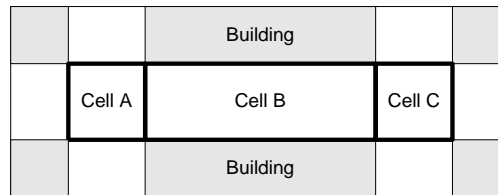
Figure 32. Intersection and segment cells.

The configuration, shown in Figure 32, is typical for most streets in cities. A street segment that exceeds the nominal radio range *R* can be further divided into multiple cells. This is illustrated in Figure 33, in which the road segment of length *sl* is divided into multiple cells of

length *cl*. When *sl* is not evenly divisible by *cl*, the last cell (the rightmost cell in Figure 33) can have a length less than *cl*. Let *sw* be the width of the street segment. To ensure that the nominal radio range *R* can cover any three consecutive cells, we have the following constraint:

$$\sqrt{sw^2 + (3 \times cl)^2} \le R \qquad \text{or} \qquad cl \le \frac{\sqrt{R^2 - sw^2}}{3}$$

We will show later how this cell configuration allows us to design a technique to allow a node in a farther downstream cell to rebroadcast the message and prevent rebroadcasts from any node in cells closer to the last broadcasting node. This way, the number of rebroadcasts for any message is approximately half the number of cells in a particular street regardless of the total number of nodes in the street. This characteristic provides two highly desirable properties: (1) the number of node rebroadcasts is very small, and (2) the performance is predictable and consistently good for all broadcasts. We will discuss these properties further when we present the simulation results in Section 5.4 and overhead analysis in Section 5.5.
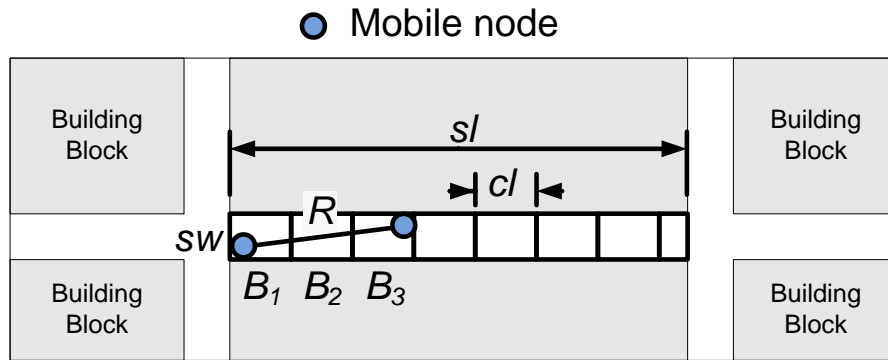
Figure 33. Multiple cells in a street segment.

We assume that each node has a map that includes information on all the cells including their identifier and the coordinates of the upper left and lower right corners. Note that

information about the streets is not needed, and that the cell map is very small in size. As an example, consider a large city with 5,000 cells. If 64 bytes are required to record information for each cell, the cell map is only 0.25 gigabytes. There are many ways to disseminate a map [55], such as downloading the map from a server in advance, from other nodes in the streets, or via mobile cellular network. This topic is beyond the scope of this paper but has many existing solutions can be used.

5.2.2 Broadcast Procedure

To broadcast a packet for a destination, a source node includes the following information in the packet: *Source Node ID*, *Destination Node ID*, *Packet ID*, *Current Node Position*, *Incident Location*, and *Dissemination Radius*. The *Source Node ID* and *Destination Node ID* fields are the node IDs of the source and destination nodes, respectively. The *Destination Node ID* can be set to a broadcast address if the message is intended for all the nodes, as in message dissemination. The *Current Node Position* is the position of the node about to rebroadcast this packet. Thus, each intermediate node updates the *Current Node Position* header field before rebroadcasting the packet to downstream nodes. The *Incident Location* and *Dissemination Radius* indicate the location of an incident and radius of the affected zone, respectively. These two fields can be used to disseminate information related to an incident such as an evacuation [26]. Such information often is of interest only to the vehicles within a certain radius of the incident. The *Incident Location* can be the location of the source or a nearby incident observed by the source.

When a node n receives a packet from another node m, it will first find the cell containing m's position as indicated in the packet header. It then determines if it needs to rebroadcast the packet according to the following algorithm:

Algorithm I:

---

1. If n is outside the dissemination area, it drops the packet.

2. If n has broadcast the packet before, it drops the packet.

3. If n is not on a downstream cell of m's cell, it drops the packet.

4. If n has heard the same packet broadcast from another node in its cell, it drops the packet.

5. If n has heard the same packet broadcast from its overtaking neighboring cells, it drops the packet.

6. If all previous conditions fail, n sets the delay.

7. If at the end of the delay, one of the previous conditions satisfies, n drops the packet.

8. Otherwise, n rebroadcasts the packet.

In Steps 5 and 6 of the above algorithm, n determines its overtaking neighboring cells and delay based on whether m's radio range can reach the next intersection cell. If m can reach at least one intersection cell in the downstream as in Figure 32, then n uses Algorithm II to determine its overtaking neighboring cells and delay.

Algorithm II:

1. If *n* is in an intersection, its overtaking neighboring cells are the nearest intersection at each incoming direction of its intersection cell and its delay is $rand_n \times t$ seconds.

2. If *n* is in a street segment, its overtaking neighboring cells are the adjoining intersection at each end of its segment cell and its delay is $rand_n \times t + t$ seconds.

In Step 1 of Algorithm II, if an incoming road is a dead-end street (i.e., there is no nearest intersection in the direction), then the overtaking neighboring cell for this particular direction is the street segment of the dead-end street. The $rand_n$ in Algorithm II is a pseudo-random number generator which produces a number from the range *[0, 1)* and is seeded uniquely for node *n*. The delay *t* can be adjusted based on node density derived from historical data collected by intelligent transportation systems such as [5]. When node density is high such as before a concert event or sports game, *t* can be set longer to reduce the radio contention. The value of *t* can be included in the message packet initiated by the event organizer.

The significance of Algorithm II is that nodes in intersections rebroadcast first and thus eliminate the need for rebroadcasts from the segment cells between the intersection cells. A rebroadcast from an intersection also has the advantage of propagating the packet to the intersecting street in addition to the street the packet is currently on and thus reaches more nodes than a rebroadcast from a segment cell.

In rare cases where *m*'s radio range is too small to reach the next intersection due to a long road segment (as in Figure 33), *n*'s overtaking neighboring cells are the adjoining cell at each end of its cell and *n* uses Algorithm III to determine its delay.

Algorithm III:

---

1. If *n* is in the downstream segment cell *s* adjoining the cell of *m* (see Figure 34(a)), *n* sets its delay as $rand_n \times t + t$ seconds.

2. If *n* is in the segment cell located immediately downstream of *s* (see Figure 34(b)), *n* sets its delay as $rand_n \times t$ seconds.

In Algorithm III, $rand_n$ is the same pseudo-random generator previously described.

When nodes on intersections cannot be utilized, Algorithm III takes advantage of information about node location to reduce delay and overhead. It allows far away downstream nodes to rebroadcast with little delay. It also eliminates unproductive rebroadcasts from nodes closer to the last broadcaster, as they will not reach additional downstream nodes.
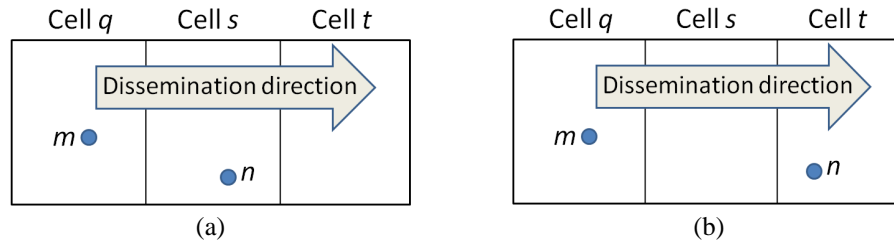
Figure 34. Two scenarios for delay computation in Algorithm III.

5.3 Simulation Setting and Performance Metrics

To evaluate our approach, we perform simulations using the network simulator called GloMoSim [85]. This simulator is a packet-level simulator specifically designed for ad hoc networks. It follows a layered network communication model and provides comprehensive simulation for each of the layers.

Since the proposed CB-S protocol is a network layer protocol that does not rely on easily outdated neighborhood knowledge, we simulated and compared CB-S and other network layer 0-hop broadcast protocols discussed in Section 5.1 including Plain Flooding, Counter Based, Angle Based, and Cell Broadcast (CB). These protocols do not assume the additional requirements by the MAC layer protocols [49][50][78] and thus can provide better insight to the performance result. All the protocols use 60 milliseconds for each delay period. For Counter Based, we set the value of the counter threshold to $C = 3$ in accordance with [60]. For Angle Based, we set the value of $\varepsilon = 2$ according to [77].

We consider 1,640 mobile nodes. The field configuration is a 1000m $\times$ 1000m space, with a street width of 10 meters and street block size of 100m $\times$ 100m. There are 81 street blocks in total. Each simulation emulates 15 minutes of time. Before the simulation begins, 10% of the nodes are randomly picked to be the set of nodes which initiates broadcasts. Each node initiates a broadcast one time during the simulation run, and this single broadcast occurs at a random time. A broadcast is 500 bytes long. We assume the broadcast messages are intended for all the nodes in the terrain, similar to request packets used by routing protocols [35][57]. Each node has a radio range of about 120 meters. Initially, nodes are placed uniformly with 2

nodes per intersection and 8 nodes per segment cell. The nodes then move in the directions permitted in the streets. Upon arriving at an intersection, a node probabilistically changes its direction of movement - turns left, turns right, or continues in the same direction.

We performed sensitivity studies for mobile node speed, openness of the terrain (in terms of signal obstruction), and dissemination radius. The protocols are compared under three performance metrics: (i) **average overhead** measures the average percentage of nodes participating in *relaying* each broadcast message in the entire terrain (a higher percentage indicates more rebroadcasts, and therefore more overhead); (ii) **average reachability** measures the average percentage of nodes *receiving* each broadcast message within a dissemination area (90% means, on average, 90% of the nodes receive the broadcast message); and (iii) **average delay**, measured in seconds, is the average interval from the time the first node initiates a message to the time the last node receives the message within a dissemination area. Unless otherwise noted, the dissemination area refers to entire area of the terrain.

5.4 Simulation Results

We present the simulation results in this section. We study the effects of mobile speed, openness of the terrain (in terms of signal obstruction), and dissemination radius on performance.

5.4.1 *Effect of Speed*

The simulation results for mobile node speed are shown in Figure 35, Figure 36, and Figure 37. We observe that all the protocols, except CB-S, make a tradeoff between reachability and overhead (see Figure 35 and Figure 36). The naïve Plain flooding offers good reachability by making every node rebroadcast. This results in very high overhead. The more advanced Angle Based protocol also has a high overhead because it is nearly impossible for any node to fail its rebroadcast criteria as we have discussed in Section 5.1. Protocol CB suffers from low reachability because the cells are not tailored to a street network. In this environment, the terrain is partitioned into grid cells regardless of the street network topology. As a result, nodes in intersections often do not rebroadcast because they have overheard another rebroadcast from the same cell but from a node in a street segment. Such rebroadcasts cannot forward the packet onto the intersecting street, therefore affecting the overall reachability performance. While the Counter Based protocol reduces overhead, it negatively affects reachability. This occurs because it fails to take into account the advantage of rebroadcasts at intersections. Consequently, the message does not always reach nodes near the edge of the terrain. In contrast to the aforementioned protocols, the proposed CB-S technique offers high reachability with low delay and low overhead. The high reachability can be attributed to the high percentage of rebroadcasts from the intersections of the streets. More nodes can therefore be reached in this environment. The low overhead is attributable to rebroadcasts occurring in every other cell in CB-S, thus usually skipping road segments in-between intersections. The low overhead contributes to the very low delay in CB-S (Figure 37).

We also observe that mobility does not have a significant impact on the protocol performance since they are all 0-hop protocols that do not rely on 1- or 2- hop neighborhood information. For Plain and Angle based approaches, the mobility has completely no effect on overhead since every node rebroadcasts as explained above. With no change in overhead, the protocols' delay and reachability remain the same.   For other protocols, the delay between the time a node initiates a packet to the time last node in the terrain receives the packet is less than half a second, and thus, the topology during this time period remains fairly static.   During this time, a node moves at most 9 meters with the fastest mobility speed in Figure 35, Figure 36, and Figure 37; this distance is only enough for a node to traverse from one intersection to a street segment or vice versa without making turns. For Counter based protocol, a node decides whether or not to rebroadcast base only on number of packets it overhears. This simple mechanism does not require high accuracy of nearby topology. As long neighbors do not move too far away, the perceived neighborhood serves its purpose. For CB protocol, its performance fluctuation is due to interaction between node distribution and its cells. A long narrow street allows CB to drop a packet only if there is already a rebroadcast from the same cell. This makes CB very sensitive to node distribution. If more nodes on a street segment are included in the CB cell overlaying an adjacent intersection, the lower probability a node on the intersection broadcasts. As result, the performance of CB fluctuates.   For CB-S, a node on a street segment drops a packet if it overhears a packet has been rebroadcast from all its adjacent intersections. With a short interval of less than 0.16 second (a node can move at most four meters with the fastest mobility setting), a node deciding to rebroadcast can safely assume the topology does not change drastically over this

86

period of time. When a node decides to drop a packet based on the overheard locations of rebroadcasting neighbors, the node can assume that the rebroadcasts cover all its neighboring nodes.

5.4.2 Effect of Building Obstruction

In this section, we examine how the presence or lack of buildings affects the performance of the protocols. We simulated this effect by specifying how far away a node can be from an intersection and still be able to communicate with another node on the intersecting street of the intersection. We call this parameter *Communication Distance from Intersection* (CDI). A low CDI value characterizes an urban environment where many tall buildings block the radio between two nodes on two different streets. A high CDI value represents a suburban environment where smaller buildings are farther apart and away from the streets, allowing for nodes on two different streets to be able to communicate. The simulation results are shown in Figure 38, Figure 39, and Figure 40.

The highest CDI in the figures represents a street environment without any buildings. This setting is used to exclusively show the effect of the constraint on node mobility (i.e., mobility is constrained to the road network) alone on the protocols. Again, we observe that only the proposed CB-S technique performs well under all three performance metrics. The performance of the other techniques is similar to the results shown in Figure 35, Figure 36, and Figure 37 with the following differences. CB improves as the building obstruction decreases. This is due to the fact that CB is very effective for an open terrain environment. When the

obstruction is minimal, it has the same reachability as that of CB-S. In terms of delay (Figure 40), both the Plain and Angle Based techniques experience longer delays with decreases in building obstruction (i.e., increases in CDI). This is due to the high overhead in these two techniques (Figure 38). Almost every node rebroadcasts, resulting in severe contention on the wireless medium. This contention increases with corresponding decreases in building obstruction because radio signals from more nodes on different streets can interfere with each other. As a consequence, nodes must wait longer before rebroadcasting each packet.

5.4.3 Effect of Message Dissemination Radius

In this section, we compare the performance of the protocols in disseminating messages to nodes in the proximity of the source node of the broadcast. A message is initiated near the center of the terrain. The *dissemination radius* (DR) is defined as a multiple of the nominal radio range and defines the dissemination area. This smaller area is only applicable to CB-S, whose *reachability* and *delay* metrics involve only the nodes within the area.

The simulation results are shown in Figure 41, Figure 42, and Figure 43. We observe that only CB-S responds to changes in DR. While other protocols utilize the nodes within the dissemination zone, they do not limit the broadcast to the dissemination region. Only CB-S utilizes less rebroadcast nodes for a smaller dissemination zone. Hence it incurs less overhead (Figure 41). Since only nodes in the dissemination area are reached, the delay is also reduced under CB-S. When the dissemination radius becomes very large (e.g., 5 or 6 times the radio range), almost covering the entire terrain, CB-S performance levels off.
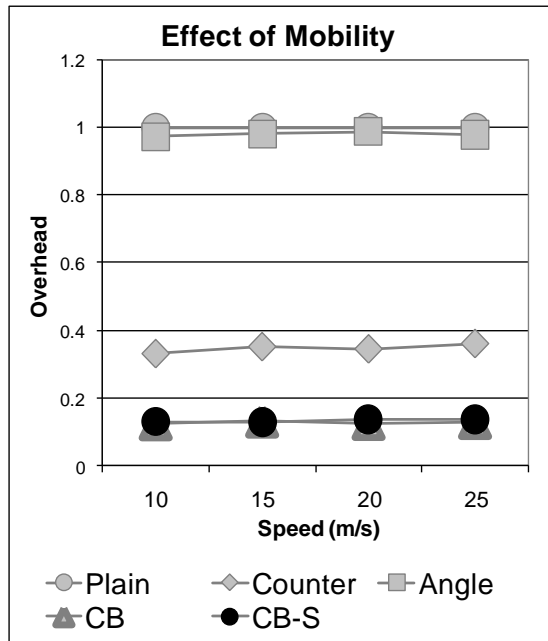
Figure 35. Effect of Mobility in Overhead.
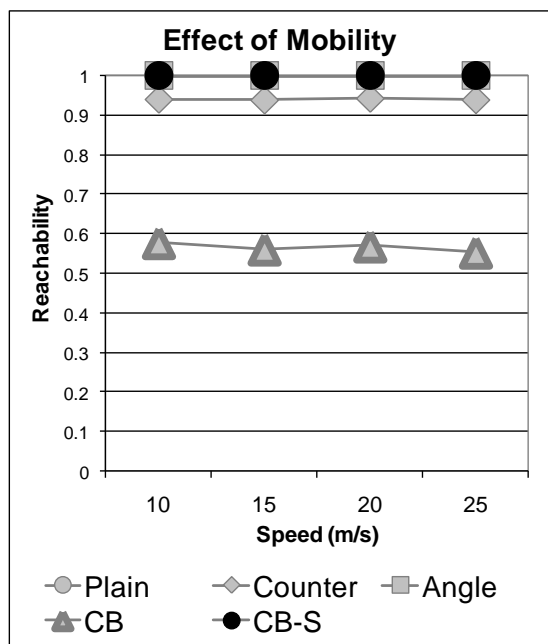
Figure 36. Effect of Mobility in Reachability.

89
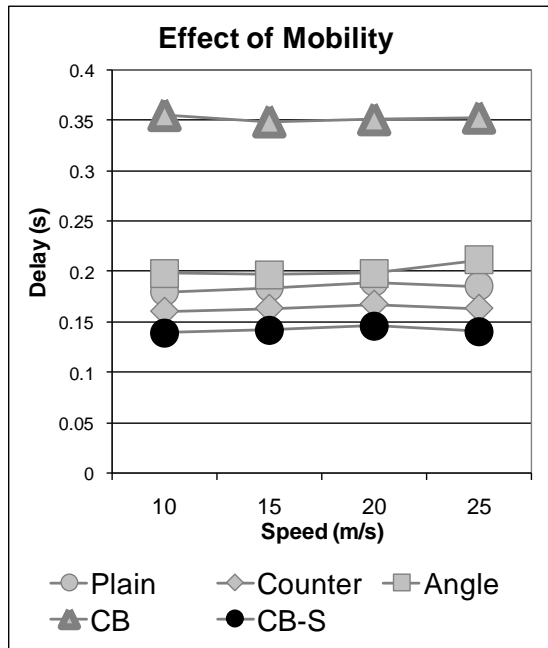
Figure 37. Effect of Mobility in Delay.

Figure 38. Effect of Building Obstruction in Overhead.
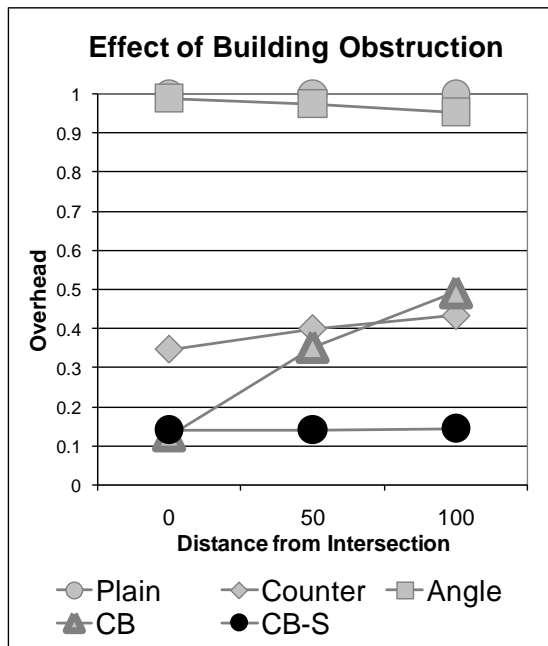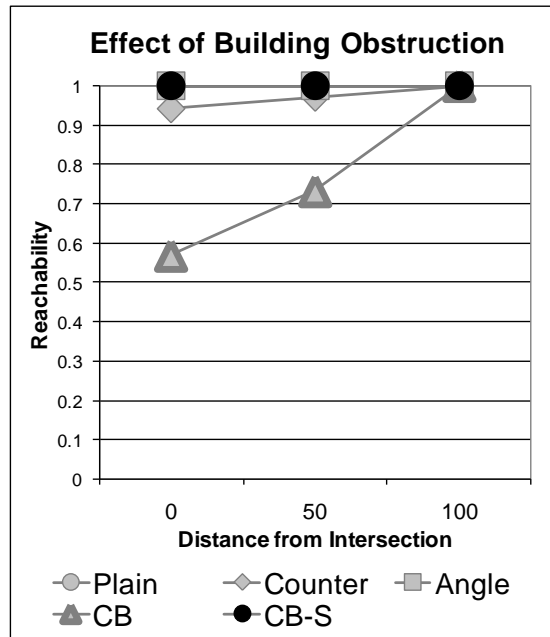
Figure 39. Effect of Building Obstruction in Reachability.
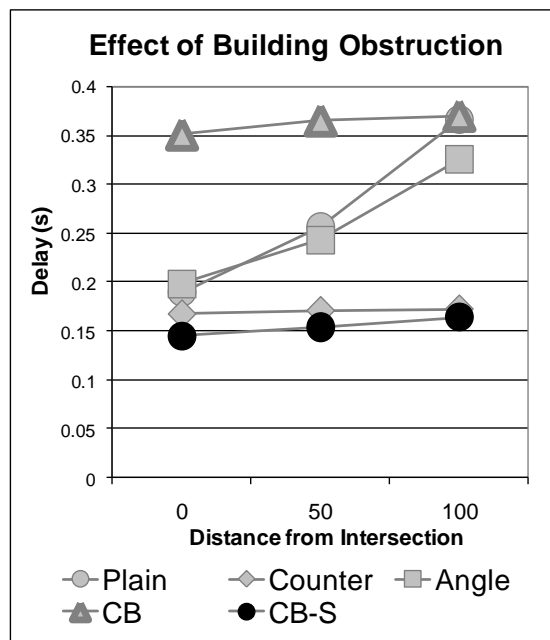
Figure 40. Effect of Building Obstruction in Delay.
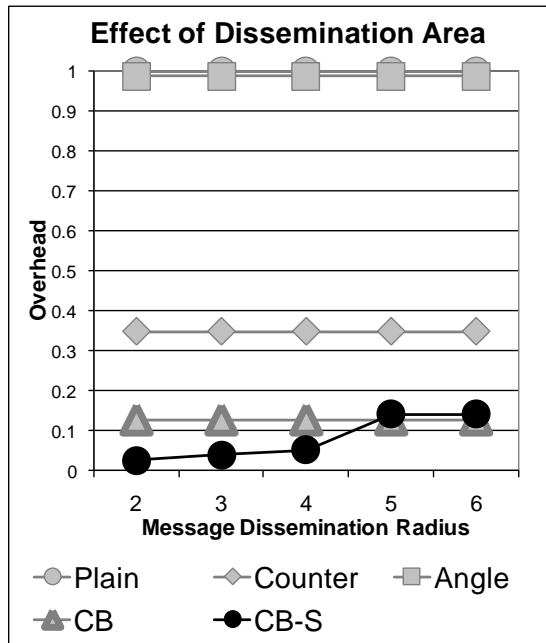
Figure 41. Effect of Dissemination Radius in Overhead.
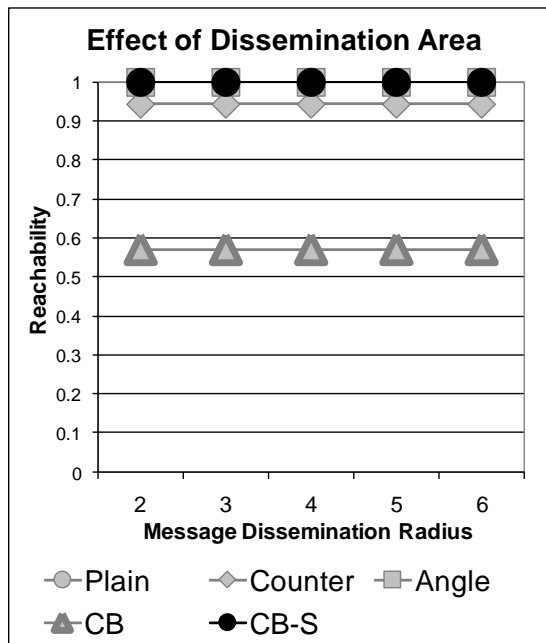
Figure 42. Effect of Dissemination Radius in Reachability.

Effect of Dissemination Area
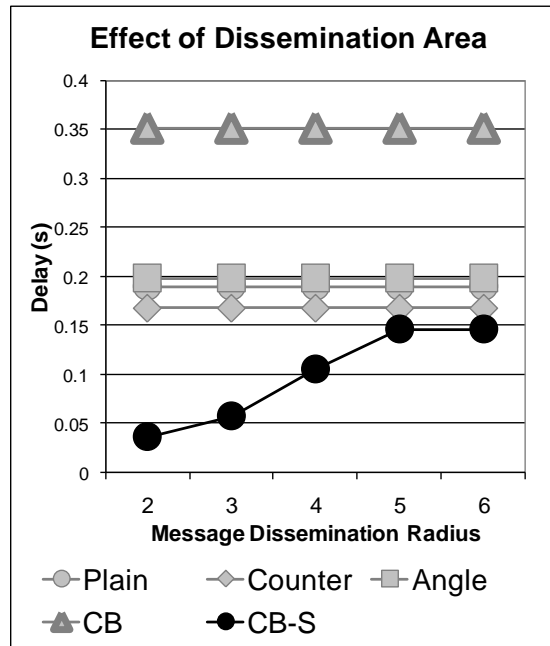
Figure 43. Effect of Dissemination Radius in Delay.
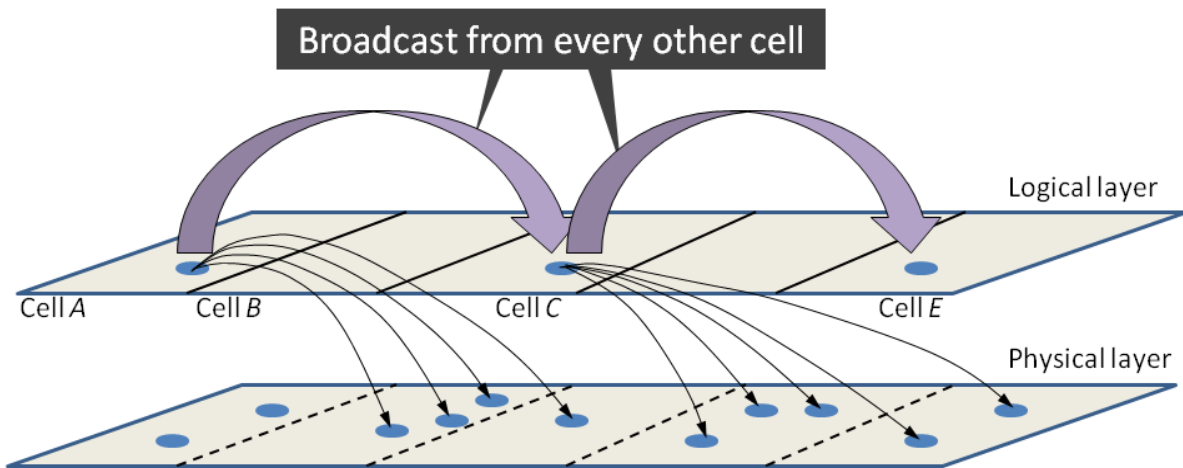
Figure 44. Two layers of the CB-S approach.

5.5 Analysis of CB-S

In this section we analysis the overhead, delay (in terms of number of hops), and reachability of the proposed protocol. CB-S can be viewed as a dual-layer design as illustrated in Figure 44, which consists of a logical layer and a physical layer. The physical layer is the actual mobile ad hoc network with the physical nodes. The logical layer consists of the cells, each treated as a virtual node. If every two adjacent virtual nodes are considered as connected in the logical layer, we have a special kind of overlay network formed over the mobile ad hoc network. From this perspective, a rebroadcast from a physical node in one virtual node to another physical node in the next virtual node can be seen as a broadcast from the first virtual node to the next virtual node. That is, a virtual node derives its communication functionality from the capability of the physical nodes currently within its area of operation.

To disseminate a message to all the nodes in the physical layer, the algorithms in Section 5.2 are used to implicitly construct a *broadcast graph* in the overlay network, on demand. This is possible because the virtual nodes in the logical overlay layer, unlike the physical nodes, are stationary. This broadcast graph connects every other virtual node (Figure 44) to include half of the virtual nodes in the overlay of one street. During data dissemination, the rebroadcasts are carried out at each virtual node in the broadcast graph starting from the source node (i.e., the virtual node in the logical cell where the broadcast is initiated). Each rebroadcast, however, is an actual node broadcast occurring in the physical layer. The rebroadcast data can reach all physical nodes in the next two virtual nodes. The second of these two virtual nodes is part of the broadcast graph, and it in turn relays the message to its own two next virtual nodes. This process

94

transmits the data packet to all the physical nodes inside these two virtual nodes. With the data relay proceeding in this manner, the data packet will eventually propagate downstream to eventually cover all virtual nodes in the broadcast graph, and will therefore reach every physical node in the physical layer.

From the above description of the broadcast operation, we observe that a broadcast utilizes k physical nodes, where k is approximately half the number of cells in one particular street and one third of cells in an entire street terrain. Given the length of each cell is about one third of the nominal radio range, k is nearly minimal. We will discuss this property further when we analysis the overhead in Section 5.5.4.

5.5.1 Desirable Properties of Broadcast Protocol in a Street Environment

Protocols [11][20][54][58][83] rely on neighborhood information are not considered desirable in high speed environments because of their large overhead to maintain the neighborhood information. Protocols such as Plain Flooding and Angle Based that require every node to rebroadcast the packet are not considered desirable because of large overhead. A protocol such as CB optimized for open terrain is not desirable for street environments because it does not reach all the nodes easily. Protocols like Counter and Probabilistic Based are not desirable because they do not reach all the nodes in minimal hops by not utilizing nodes in intersections whenever possible.
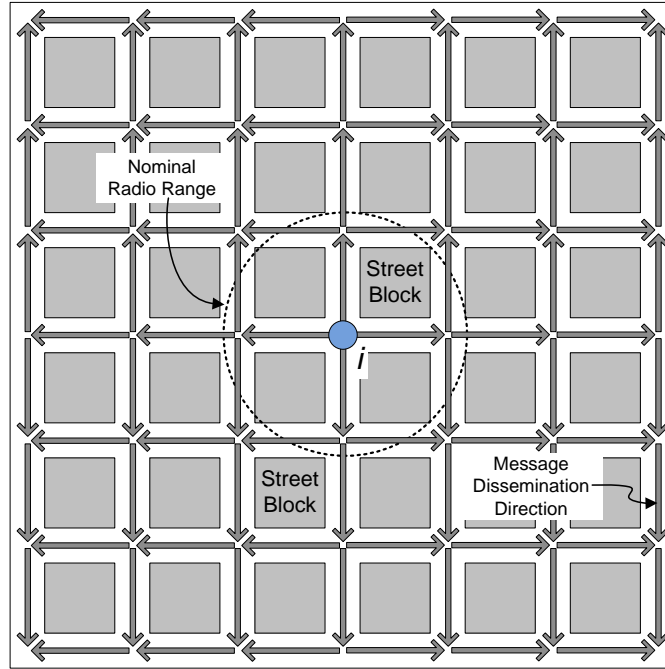
Figure 45. Desirable message relaying pattern

In this section, we will consider the desirable properties to broadcast a packet in a street environment and provide analysis that demonstrates that CB-S possess these properties. Consider a nominal radio range that is about the size of three consecutive cells (or two intersections and the segment in between) as defined in Section 5.2.1, a desirable relaying pattern is illustrated in Figure 45. For simplicity of the analysis we consider the nominal radio range to be not much larger than the three consecutive cells as defined in Section 5.2.1. In practice, one can increase the radio range to reach more nodes in one rebroadcast with tradeoff of increased power consumption. This desirable pattern uses a small number of rebroadcasts to transmit a packet to a large number of nodes and do so with little delay. More formally, for physically absolute optimality, the time t_n at which a node n (at distance d_n from the source node) receives a

message originating at the source node at time $t_0=0$ should be the same time required by electromagnetic radiation to traverse said distance d_n. This distance cannot be the minimal Euclidean distance in the street model for any broadcast technique because signals are assumed not to propagate through occlusions such as buildings. Therefore, in the street model, the distance between the origin and any node n is calculated using the Manhattan distance metric (also known as the taxicab metric and rectilinear distance). Also, for the simplicity of the analysis, we assume that nodes are distributed evenly across the terrain.
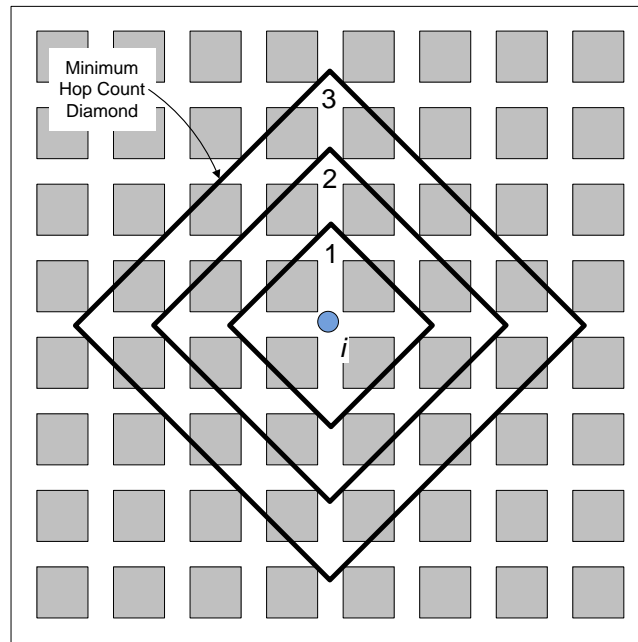
Figure 46. Minimum Hop Count Diamond

5.5.2 *Proof of Minimal Delay and Full Reachability of the Desirable Relaying Pattern*

In this section, we prove that the desirable relaying pattern can reach each of the nodes in minimal number of hops. From the desirable pattern in Figure 45, a diamond shape can be drawn to indicate the area where nodes are within h hops from node i in the desirable pattern. We call this shape a Minimum Hop Count Diamond as this diamond indicates the minimal number of hops to reach an enclosed node near the edge of the diamond. For examples, diamond 1 in Figure 46 indicates an area where nodes are 1 hop away from node i and the area between diamonds 1 and 2 indicates an area where nodes are 2 hops away from i.

We can prove by induction that the nodes in between diamond h and diamond h-*1* required at least h hops from Node i to reach. We use the following equation to denote this:

$$\mathrm{hop}(h) = h$$

Basis: $\mathrm{hop}(1) = 1$.

From Figure 45 and the definition of the nominal radio range defined in Section 5.2.1, nodes can hear i's broadcast are located on the segment cells adjoining i's cell and their adjoining intersection cells. Hence, they require at least one hop from the initiating node i to be reached and $\mathrm{hop}(1) = 1$.

Induction Hypothesis: $\mathrm{hop}(k) = k$ for $1 \le k \le n$ and $n \ge 1$.

Induction Step: We want to show that $\mathrm{hop}(n+1) = n+1$.

The nodes between diamond $n+1$ and diamond n are within radio range of the nodes on the intersection cells at the edge of diamond n. Thus we can rewrite the $\text{hop}(n+1)$ expression as follows:

$$\text{hop}(n+1) = \text{hop}(n) + 1$$
$$= n+1 \qquad \text{by Induction Hypothesis}$$

In Section 5.5.3, we will also empirically show that CB-S exhibits this message relaying pattern by demonstrating that CB-S requires a similar number of hops as this desirable message relaying pattern to reach every node.

Since every node can be reached by the minimum number of hops indicated by its minimum hop diamond, we also show that the desirable relaying pattern can reach every node.

5.5.3 Reachability and Hop Count Analysis of CB-S

From the simulation result in Section 5.4, it is clear that CB-S can consistently broadcast a packet to every node in the network. In the rest of this section, we verify that nodes are reached in a relaying pattern similar to the one in Figure 45 by analyze the number of hops a packet travels before reach a node in the terrain. Figure 47 shows the minimal hop count to reach each node using CB-S to broadcast a packet initiated by a node in the middle of the center diamond. The result is obtained from a simulation run with even node distribution and the nominal radio range allows a node in one intersection to communicate with a node in another intersection. The hop count is represented with different node colors, with red being 1-hop away from the initiate node in the middle of the red diamond and navy blue being 10-hops away. The figure shows the

following: the majority (76%) of the nodes are reachable by the minimal hop count depicted by their encompassing diamond, 21% are reached by 1 more hop than the minimum, 2% are reached by 2 or 3 hops, and no node requires more than 3 hops than the minimum to be reached.

In the most ideal scenario of broadcast that does not include stochastic delay, 100% of the nodes should be reachable by the minimum hop count. However, we do not require such high levels of minimal hop count recipients, nor can they be realistically achieved under many circumstances. What impedes having all nodes receive packets at the minimal hop count is the need to do probabilistic collision avoidance using random delays. This constraint is a core assumption of this method and of many other broadcast protocols [29][52][60][71][77][86] where the goal is to reduce message reception delay and attempt to minimize retransmissions without requiring precise control using a distributed deterministic collision avoidance mechanism. Such a mechanism is more feasible for fixed wired or wireless networks using circuit switching, where communication parameters are well known. However, the feasibility drops significantly when mobility is introduced, when packet switching (and associated message time origination non-determinism) is used, and when reducing the cost of communication devices such as radios is desired.

Figure 48 represents the minimal hop count to reach each cell in Figure 47. The hop count of a cell is the lowest hop count required to reach a node on the cell. The hop count is represented with different cell colors with red being 1-hop away from the source in the middle of red diamond and navy blue 10-hops away. The figure shows that the majority (82%) of the cells are reached by the minimal hop count depicted by the diamond they are in, 16% are reached by 1

more hop than the minimum, 2% are reached by 2 more hops, less than 1% by 3 hops, and no cell requires more than 3 hops than the minimum to be reached. Figure 47 and Figure 48 show that CB-S is very close to the desirable low delay by reaching nodes and cells with a nearly minimal number of hops. We summarize the result of the figures in Table I with the result from Figure 47 listed under "Percentage of nodes reached b**y"** column and Figure 48 under "Percentage of cells reached by", respectively.

Table 1. Percentages of nodes and cells reached by minimal hops

Additional hops to the minimal	Percentage of nodes reached by	Percentage of cells reached by
+ 0	76%	82%
+ 1	21%	16%
+ 2	1%	2%
+ 3	1%	0%
> 3	0%	0%

The percentage ($p_{minimal}$) of nodes that is reached by the minimal number of hops will vary depending on many factors. These include topological information on nodes, radio signal propagation environment due to buildings and other occlusions, travel speed of individual vehicles, the minimum desired number of retransmissions due to collisions among simultaneous radio transmissions, and various others. A controllable parameter that can be adjusted that will affect this number is the node transmission delay range or t as discussed in Section 4.2.4.

Adjusting this parameter needs to be done carefully to match the given scenario. Setting the value of t too high causes an increased message reception time across the network because nodes are waiting longer to retransmit. Setting the value too low also causes an increased message reception time because the collision probability due to nodes broadcast at same time increases with a smaller value of t. Also as mentioned in Section 4.2.4, this value can be adjusted based on historical data on node density. Tuning this parameter in the simulation runs has allowed for the relatively high 76% value of $p_{minimal}$ in the result in this section.

5.5.4 Proof of Minimal Overhead of the Desirable Relaying Pattern

In this section, we study the overhead required by the desirable relaying pattern. From the relaying pattern in Figure 45, one can see the minimal overhead is one rebroadcast from every intersection. This is approximately half of the cells in one particular street. For a terrain with HS horizontal streets and VS vertical streets similar to Figure 45, the number of intersections I and segments S can be computed as following:

$$I = HS \times VS \quad \text{and} \quad S = (HS-1) \times VS + (VS-1) \times HS = 2 \times HS \times VS - HS - VS$$

We can compute the percentage of cells needed for broadcast as percentage of intersection cells in all the cells. Thus, we have the following equation :

$$\frac{I}{(I+S)} = \frac{HS \times VS}{HS \times VS + 2 \times HS \times VS - HS - VS} = \frac{HS \times VS}{3 \times HS \times VS - HS - VS}$$

The above function has a limit as the following:

$$\lim_{HS \to \infty, VS \to \infty} \frac{HS \times VS}{3 \times HS \times VS - HS - VS} = \frac{1}{3}$$

The above equation shows that about one-third of cells in Figure 45 are intersections. The following equation computes the minimal overhead needed for a terrain with C cells and N nodes:

$$\frac{I}{N} = \frac{HS \times VS}{N} = \frac{C}{3N} \tag{1}$$
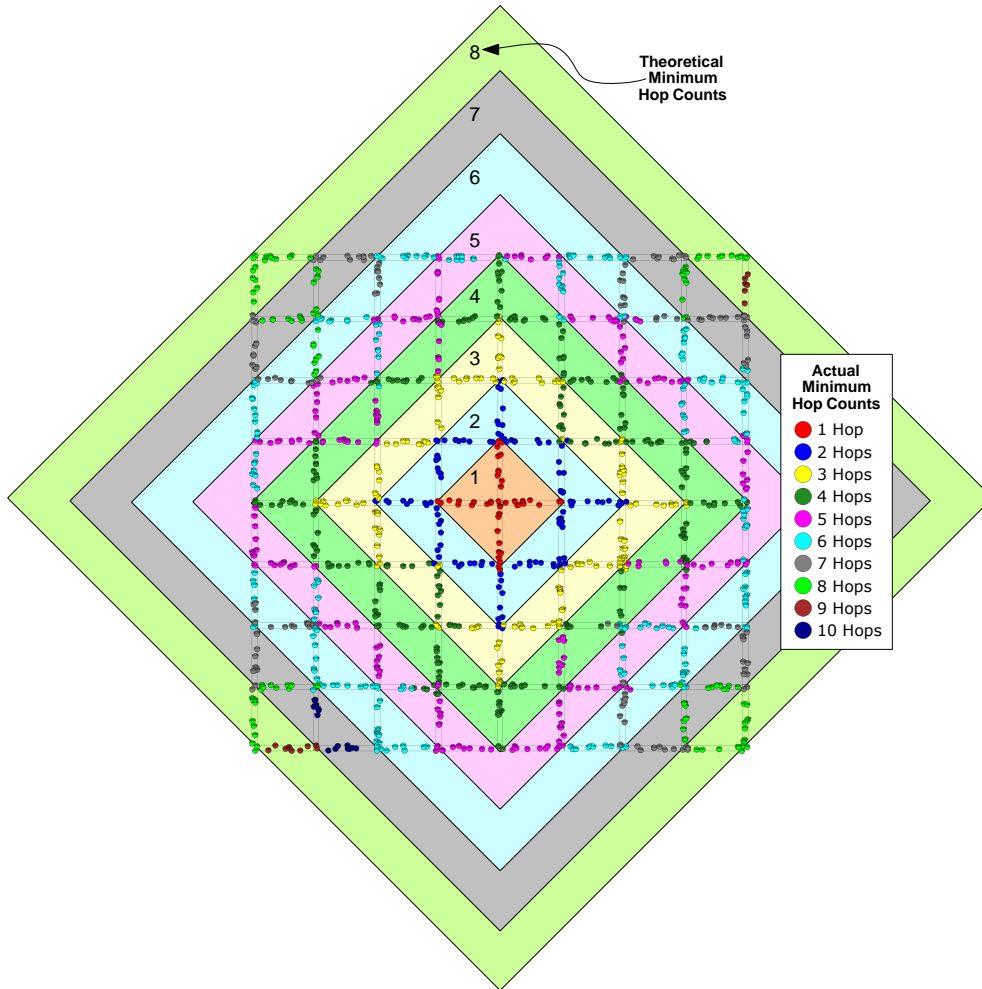
Figure 47. Minimal number of hops to reach each node.
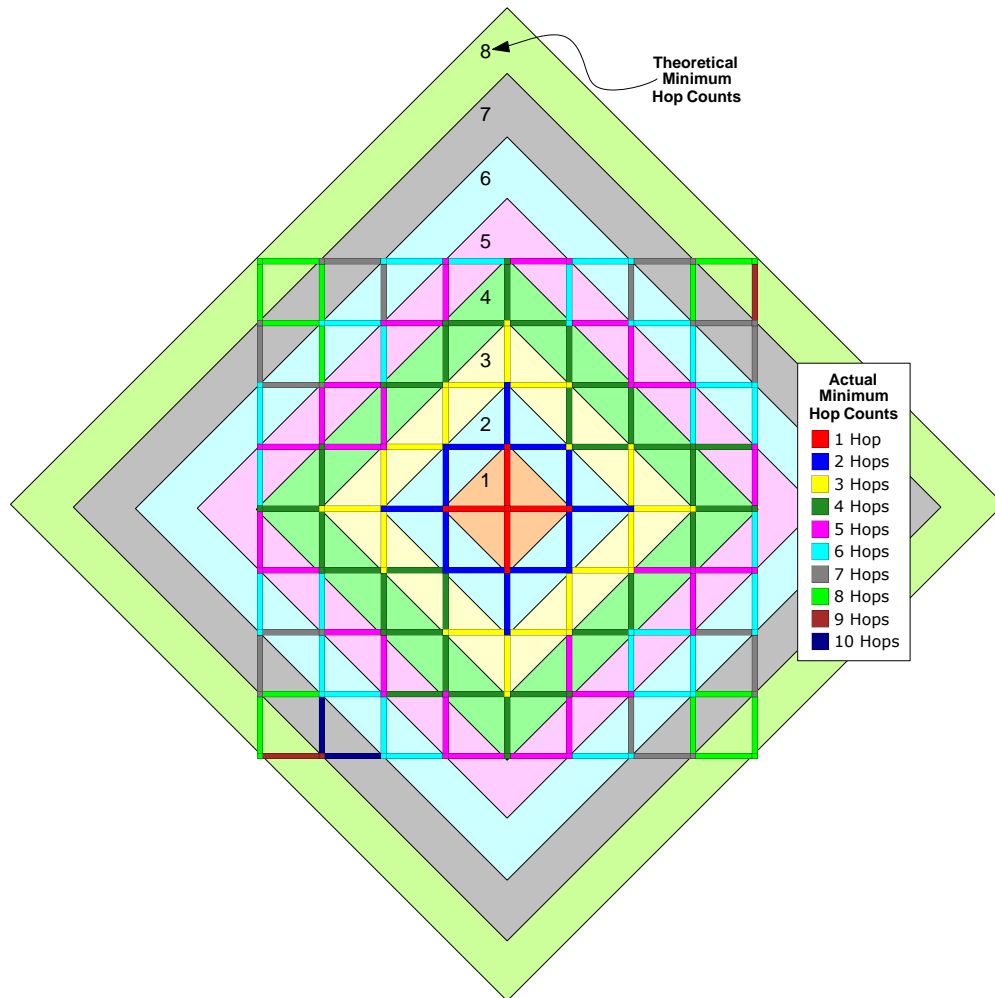
Figure 48. Minimal number of hops to reach each node.
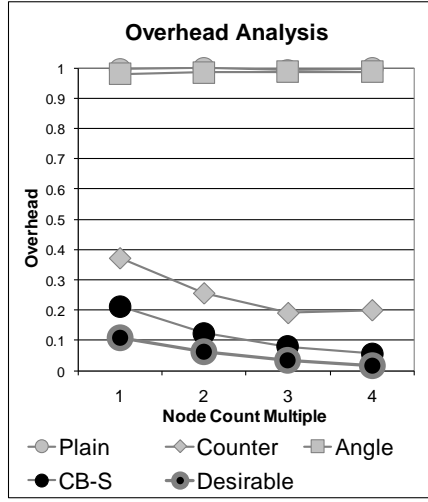
Figure 49. Overhead analysis

5.5.5 *Overhead Analysis of CB-S*

In this section, we verify CB-S has an overhead close to the low overhead of the desirable

relaying pattern. Using the terrain and the initiating node in the middle of the terrain as in Figure

47, we study the overhead of the protocols discussed in Section 5.4 and the desirable relaying

pattern. We vary the number of nodes or *N* in Equation (1) and show the result obtained from the

simulation in Figure 49. For clarity, the number of nodes is shown as multiples of the node count

of the fewest node setting. The overhead of the protocols are obtained from simulation result and

the overhead of the desirable relaying pattern is obtained using Equation (1) with different

number of nodes in the terrain. Only protocols with full reachability are shown in the figure.

The figure shows that among all the protocols with full reachability, CB-S has the lowest

overhead and thus is closest to the overhead of the desirable relaying pattern. Like the desirable

relaying pattern, CB-S also exhibits a decreasing trend in overhead as the number of nodes increases. Both Plain and Angle Based use all the nodes to rebroadcast and do not respond to the node increase. Countered Based shows an unstable trend when the number of nodes becomes large. Counter Based also has a higher overhead than CB-S in all node density settings.

5.6 <u>Conclusion</u>

In this paper, we proposed the CB-S broadcast technique for street environments. We defined a desirable message dissemination pattern and proved how such pattern can reach each node with the minimal overhead and the minimal number of hops away from the initiate node of the message. Our analysis of simulation result shows that CB-S is able to reach every node in a minimal or almost-minimal number of hops. We also performed simulation studies to compare it with four other techniques. The results indicate that only CB-S performs well under all three performance metrics, namely reachability, overhead, and delay. Its reachability results are consistently at 100% for all simulation settings, always successfully disseminating a message to every node in the street network. This perfect coverage is achieved with the least overhead. In fact, the analysis indicates that its overhead is closed to the overhead of the desirable relaying pattern (i.e., the number of nodes needed to disseminate a message is small, and the number of hops is also small). This efficient property gives CB-S the advantage of very small delay. In summary, CB-S can always disseminates messages to all the nodes in the least amount of time using the least number of rebroadcast nodes.

6. CONCLUSIONS

6.1 <u>Concluding Remarks</u>

In this dissertation, we propose a virtual router approach to address mobility issue in ad hoc networks. Two virtual router approaches are introduced: *Static Virtual Router* (SVR) and *Dynamic Virtual Router* (DVR). In Chapter 2, we apply static virtual router to vehicular networks where nodes move in high speed. The static virtual routers are modified to adapt to the unique terrain of a street environments [30]. Rather than a square shape as in open environment, virtual routers in a street environment are defined by intersections and street segments. Data packets are relayed by nodes in intersections whenever possible. The simulation results show that this adaption makes virtual router approach performs better in street environments than other routing protocols that rely on physical links that are easily broken especially the links in between two nodes moving along different streets.

In Chapter 3, we propose a dynamic virtual router (DVR) approach [32]. Unlike static virtual routers that are shared by all communication paths, dynamic virtual routers are created as needed for communication sessions and deprecated after use. Dynamic virtual routers also do not need to be defined with geographic coordinates and thus alleviate nodes from install positioning devices such as GPS. Our simulation results show that DVR has less control overhead than static virtual router approach (SVR) due to fewer nodes involved in update and has lower delay due to better utilization of underlying network topology by dynamic virtual routers.

107

In Chapter 4, we apply virtual router concept (particularly static virtual routers) to reduce overhead to broadcast a message (such as location request) to all the nodes in the terrain [29]. Using virtual routers, the proposed broadcast protocol requires minimal nodes to relay a packet to all the nodes in the terrain. With modifications to virtual routers and addition of dissemination radius as discussed in Chapter 5, virtual routers can be applied to disseminate a message to vehicles in street environments [31]. The extensive simulation study and performance analysis [33] indicate the virtual routers outperform other broadcast protocols that also do not use neighborhood information.

6.2 **Future Works**

In this dissertation, virtual routers are applied to environments where node topology changes fast due to high node speed with or without presence of obstruction. In the future, I am interested in study the effectiveness of virtual router in a low speed environment that could also experience frequent topology change due to obstructions. An example of such environment would be people walking inside a building with hallways. Nodes or mobile devices on two different hallways could experience frequent link disconnect and connect as the people carrying the devices walking around the corner of walls that block the line of sights between nodes on two different hallways. I am interested in comparing dynamic virtual routers that do not rely on position information of nodes with static virtual routers that rely on node locations.

In this dissertation, virtual routers are applied to handle the topology change of an individual communication session between a pair of source and destination. Future study can

investigate techniques to optimize scenarios where many communication paths overlap in space. Existing works such as [38][37] merge communication streams in a static wired or mesh network. It would be interesting to understand how virtual routers can be applied to merge communication streams in a more dynamic ad hoc network.

REFERENCES

[1] D. Aguayo, J. Bicket, S. Biswas, G. Judd, and R. Morris. "Link-level measurements from an 802.11b mesh network," *SIGCOMM Comput. Commun*. Rev. 34, pp. 121-132.

[2] J. B. Alegiani, J. L. Buxton, and S. K. Honey, "An in-vehicle navigation and information system utilizing defined software services," *Conf. Rec. 1989 IEEE Conf. Vehicle Navigation and Information Systems*, Sep. 1989, pp. A3-A8.

[3] M.J. Ammann "A wideband monopole for reconfigurable multiband radio terminals," *Proc. of 2001 IEEE Antennas and Propagation Society International Symposium*, vol.1, pp.170-173.

[4] M. J. Ammann, Z. N. Chen, "Wideband monopole antennas for multi-band wireless systems," *Antennas and Propagation Magazine*, IEEE, vol.45, no.2, pp. 146- 150, April 2003.

[5] A. Aved, T. Do, G. Hamza-Lup, A. H. Ho, L. Hoang, L. Hsia, K. A. Hua, F. Liu, and R. Peng, "A Real-Time Route Diversion Management System," In *Proc. of the International Conference on Intelligent Transportation Systems* (ITSC), pp. 1131-1136, September, 2007.

[6] A. Balasubramanian, Y. Zhou, W. B. Croft, B. N. Levine and A. Venkataramani, "Web Search From a Bus", *Proc. of the Second Workshop on Challenged Networks* (CHANTS 2007), pp. 59-66.

[7]   S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B.A. Woodward, "A distance routing effect algorithm for mobility (DREAM)," *Proc. of the Fourth annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp.76–84, October 25-30, 1998.

[8]   S. Biswas, and R. Morris. "ExOR: opportunistic multi-hop routing for wireless networks". *Proc. of the 2005 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* (SIGCOMM '05). ACM Press, New York, NY, pp. 133-144.

[9]   S. Biswas and R. Morris. "Opportunistic routing in multi-hop wireless networks". *SIGCOMM Comput. Commun. Rev.*, 34(1):69–74, 2004.

[10]   L. Blazevic, S. Giordano, and J. L. Boudec. "Anchored Path Discovery in Terminode Routing". *Proc. of the Second international Ifip-Tc6 Networking Conference on Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications. Lecture Notes in Computer Science*, vol. 2345. Springer-Verlag, London, pp. 141-153, 2002.

[11]   Ying Cai, Kien A. Hua, and Aaron Phillip. "Leveraging 1-hop Neighborhood Knowledge for Efficient Flooding in Wireless Ad Hoc Networks," *Proc. of IEEE Int'l Performance Computing and Communications Conference*, pp. 347-354, April 7-9, 2005.

[12]   C. C. Chiang and M. Gerla, "Routing and multicast in multihop, mobile wireless networks," *Proc. of IEEE 6th International Conference on Universal Personal Communications Record*, (ICUPC '97). Vol.2, pp. 546-551.

[13]   P. Christy, "Virtual Routing: What's The Goal? And What's Beyond?",

http://www.netsedgeonline.com/pdf_files/virtual_routing.pdf, August 2001. Downloaded

on August 29, 2008.

[14]   J. E. Córcoles and P. González, "A specification of a spatial query language over GML,"

Proc. of the 9th ACM International Symposium on Advances in Geographic Information

Systems, 2001, pp. 112-117.

[15]   S. Corson and J. Macker, "Mobile Ad Hoc Networking (MANET): Routing Protocol

Performance Issues and Evaluation Consideration," Request for Comments 2501, Internet

Engineering Task Force, 1999.

[16]   D. J. Deng, H. C. Chen, H. C. Chao and Y. M. Huang, "A Collision Alleviation Scheme

for IEEE 802.11p VANETS", *Wireless Personal Communications*, vol. 56, no. 3, pp.

371-383, February 2011.

[17]   F. De Rango, F. Veltri, S. Marano, "Hierarchical trajectory-based routing protocol for

Vehicular Ad Hoc Networks," *Proc. of International Symposium on Performance*

Evaluation of Computer and Telecommunication Systems (SPECTS '08), pp.189-196.

[18]   S. Dolev et al, "Virtual Mobile Nodes for Mobile Ad Hoc Networks," Computer Science

and Artificial Intelligence Laboratory Technical Report, MIT-CSAIL-TR-2004-009,

Massachusett Institute of Technology, February 26, 2004.

[19]   G. Dommety and R. Jain, "Potential networking applications of global positioning system

(GPS)," Tech. Rep. TR-24, CS Dept., The Ohio State University, April 1996.

[20]   M. Durresi, A. Durresi, L. Barolli, "Emergency broadcast protocol for inter-vehicle communications," *Proc. of 11th International Conference on Parallel and Distributed Systems*, 2005, vol.2, pp. 402-406.

[21]   H.C.C. Fernandes, M.S.M. Damascena, N.A. Pinto, "New analysis and 3D results of microstrip antenna arrays," *Proc. of 1999 International Microwave and Optoelectronics Conference*, 1999. SBMO/IEEE MTT-S, APS and LEOS - IMOC '99. International , vol.1, pp.250-253 vol. 1, 1999.

[22]   H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad hoc networks," *Ad Hoc networks*, vol. 1, issue 4, pp.351-369, November 2003.

[23]   R. S. Gray, et al. "{CRAWDAD} trace dartmouth/outdoor/routing/20031017 (v. 2006-11-06)", Downloaded at http://crawdad.cs.dartmouth.edu/dartmouth/outdoor/routing/20031017, Nov. 2006.

[24]   C. Gui and P. Mohapatra, "A framework for self-healing and optimizing routing techniques for mobile ad hoc networks," *Wirel. Netw.* vol. 14, iss. 1 (Jan. 2008), pp. 29-46.

[25]   Z.J. Haas and M.R. Pearlman, "The Zone Routing Protocol (ZRP) for Ad-Hoc Networks," IETF MANET working group, Internet Draft, draft-ietf-manet-zone-zrp-02.txt, June 1999.

[26]   G. L. Hamza-Lup, K. A. Hua, R. Peng, A. H. Ho. "A Maximum Flow Approach to

        Dynamic Handling of Multiple Incidents in Traffic Evacuation Management". *Proc. of

        the 8th Int'l IEEE Conf. on Intelligent Transportation Systems* (ITSC'05), pp. 1147-1152.

[27]   M. Heissenbuttel, T. Braun, T. Bernoulli and M. Wälchli, "BLR: Beacon-Less Routing

        Algorithm for Mobile Ad-Hoc Networks," *Computer Communications Journal*, vol. 27,

        iss. 11, 1 July 2004, pp. 1076-1086.

[28]   N. Herscovici, Z. Sipus, P.-S. Kildal, "The cylindrical omnidirectional patch antenna,"

        IEEE Trans. Antennas and Propagation, vol. 49, pp. 1746-1753, Dec. 2001.

[29]   A. H. Ho, A. J. Aved, and K. A. Hua. "A Novel Broadcast Technique for High-Density

        Ad Hoc Networks," *Proc. of International Wireless Communications and Mobile

        Computing Conference* (IWCMC 2006), pp. 425-430.

[30]   A. H. Ho, Y. H. Ho, and K. A. Hua. "A Connectionless Approach to Mobile Ad Hoc

        Networks in Street Environments," *Proc. of IEEE Intelligent Vehicles Symposium* (IV

        2005), pp. 575- 582.

[31]   A. H. Ho, Y. H. Ho, and K. A. Hua, "An Efficient Broadcast Technique for Vehicular

        Networks," *Journal of Information Processing Systems* (JIPS), vol.7, no.2, pp. 221-240,

        June 2011.

[32]   A. H. Ho, Y. H. Ho, and K. A. Hua, "Handling High Mobility in Next-Generation

        Wireless Ad Hoc Networks," *Special Issue on Next Generation Networks (NGNs) of the

        International Journal of Communication Systems*, pp. 1057-1309, 2010.

[33]   A. H. Ho, Y. H. Ho, and K. A. Hua, and R. Villafane. "A near-optimal broadcast technique for vehicular networks," *Proc of 2009 Conference on Wireless Telecommunications Symposium* (WTS 2009). pp. 413-420.

[34]   Y. H. Ho, A. H. Ho, and K. A. Hua, "Connectionless Protocol – A Localized Scheme to Ad Hoc Network." *Proc. of International Journal of Ad Hoc and Ubiquitous Computing* (IJAHUC), vol. 2, iss. 1/2, pp. 21 - 35, 2007.

[35]   Y. H. Ho, A. H. Ho, K. A. Hua, and G. L. Hamza-Lup. "A Connectionless Approach to Mobile Ad hoc Networks," *Proc. of the Ninth IEEE Symposium on Computers and Communications* (ISCC 2004), vol. 1, pp. 188 - 195.

[36]   C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for Reliable Multicast in Multi-hop Ad Hoc Networks," *Proc. of the Int'l Workshop on Discrete Algorithms and Methods for Mobile Computing and Communication*, 1999, pp. 64-71.

[37]   Kien A. Hua and Fei Xie, "A Dynamic Stream Merging Technique for Video-on-Demand Services over Wireless Mesh Access Networks," in *Proceedings of IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks* (SECON 2010), pp.208-216.

[38]   K. A. Hua, Y. Cai, and S. Sheu, "Patching: A Multicast Technique for The Video-on-Demand Services," in *Proc. of ACM Multimedia '98*, Bristol, UK, pp. 191-200.

[39]   C. J. Huang, Y. T. Chuang, and D. X. Yang, "An Adaptive QoS-Aware Inter-Vehicle Routing in Vehicular Networks", *Journal of Internet Technology*, Vol. 9, No. 3, pp. 289-295, July 2008.

[40]   O. Hussein, T. Saadawi, "Ant routing algorithm for mobile ad-hoc networks (ARAMA)," *Proc. of 2003 International Performance, Computing, and Communications Conference*, pp. 281-290.

[41]   A. Iwata, C.- C. Chiang, G. Pei, M. Gerla, and T.- W. Chen, "Scalable routing strategies for ad hoc wireless networks*," IEEE Journal on Selected Areas in Communications*, vol.17, no.8, pp.1369-1379, 1999.

[42]   Jinyang Li, John Jannotti, Dogulas S. J. De Couto, David R. Karger, and Robert Morris. "A Scalable Location Service for Geographic Ad Hoc Routing," *Proc. of the 6th Annual international Conference on Mobile Computing and Networking* (MobiCom '00), pp. 120-130.

[43]   D. B. Johnson and D. A. Maltz. "The dynamic source routing (DSR) protocol for mobile ad hoc network," Internet Draft, Mobile Ad Hoc Network (MANET) Working Group, IETF, October 1999.

[44]   E. D. Kaplan, *Understanding GPS Principles and Applications*. Artech House Publishers, Norwood, MA, USA, 1996. pp.413-425.

[45]   B. Karp and H.T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *Proc. of the 6th Annual international Conference on Mobile Computing and Networking* (MobiCom '00), pp. 243–254.

[46]   M. Kasemann, H. FiiBler, H. Hanenstein, and M. Mauve. "A Reactive Location Service for Mobile Ad Hoc Networks," Technical Report TR-02-014, Department of Computer Science, University of Mannheim. Nov. 2002.

[47]   Y. B. Ko and N. H. Vaidy, "Location-aided routing (LAR) in mobile ad hoc network,"
       Wirel. Netw. vol. 6, iss. 4, pp. 307-321, July 2000.

[48]   J. Kong, X. Hong, Y. Yi, J. Park, J. Liu, and M. Gerla, "A secure ad-hoc routing approach
       using localized self-healing communities," *Proc. of the 6th ACM international
       Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc '05), pp. 254-265.

[49]   G. Korkmaz, E. Ekici, and F. Özgüner, "Black-burst Based Multi-Hop Broadcast
       Protocols for Vehicular Networks," *IEEE Transactions on Vehicular Technology*, vol. 56,
       no. 5, pp. 3159 - 3167, September 2007.

[50]   G. Korkmaz, E. Ekici, F. Özgüner, and Ü Özgüner, "Urban multi-hop broadcast protocol
       for inter-vehicle communication systems," *Proc. of the 1st ACM international Workshop
       on Vehicular Ad Hoc Networks* (VANET '04). pp. 76-85.

[51]   J. Li, J. Jannotti, D. S. De Couto, D. R. Karger, and R. Morris, "A scalable location
       service for geographic ad hoc routing," *Proc. of the 6th Annual international Conference
       on Mobile Computing and Networking* (MobiCom '00). pp. 120-130.

[52]   J. Li and P. Mohapatra, "A novel mechanism for flooding based route discovery in ad
       hoc networks", *Proc. of 2003 IEEE Global Telecommunications Conference*
       (GLOBECOM '03). Vol.2, Vol.2. pp. 692- 696.

[53]   W. H. Liao, J. P. Sheu, and Y. C. Tseng, "GRID: A Fully Location-Aware Routing
       Protocol for Mobile Ad Hoc Networks," *Telecommunication Systems* 18(1-3): pp. 37-60
       (2001).

[54] H. Lim and C. Kim. Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks. *Proc. of the ACM Int'l Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems* (MSWIM '00), pages 61-68.

[55] F. Liu and K. A. Hua, and F. Xie, "On Reducing Communication Cost for Distributed Moving Query Monitoring Systems," *Proc. of International Conference on Mobile Data Management* (MDM 2008), pp. 156-164, April 27-30, 2008.

[56] T. Liu, K. Liu, "Improvements on DSDV in Mobile Ad Hoc Networks," *Proc. International Conference on of Wireless Communications, Networking and Mobile Computing* (WiCom 2007), pp.1637-1640.

[57] C. Lochert, H. Hartenstein, J. Tian, D. Herrmann, H. Füßler, and M. Mauve, "Routing strategy for vehicular ad hoc networks in city environments," *Proc. of IEEE Intelligent Vehicles Symposium* (IV2003), pp. 156-161.

[58] W. Lou and J. Wu, "On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks," *IEEE Transactions on Mobile Computing*, vol. 1, no. 2, pp. 111-123, 2002.

[59] S. Murthy and J. J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," *ACM Mobile Networks and App. J. Special Issue on Routing in Mobile Communication Networks*, pp. 183-97, October 1996.

[60] S. Ni, Y. Tseng, Y. Chen, and J. Sheu. "The broadcast storm problem in a mobile ad hoc network," *Proc. of the 5th annual ACM/IEEE international conference on Mobile computing and networking* (MobiCom '99), pp. 151-162.

[61] D. Niculescu and B. Nath, "Trajectory based forwarding and its applications," *Procs. of the 9th annual international conference on Mobile computing and networking* (MOBICOM '03), pp. 260–272.

[62] T. Nolte and N. Lynch, "A Virtual Node-Based Tracking Algorithm for Mobile Networks," *Proc. of the 27th international Conference on Distributed Computing Systems* (ICDCS 2007).

[63] Open GIS Consortium, Inc. "OpenGIS simple features specification for SQL revision 1.l OpenGIS project document 99-049 release," 1999. Available: http://www.opengeospatial.org/docs/99-049.pdf

[64] B. Parkinson and S. Gilber, "NAVSTAR: global positioning system – 10 years later," *Proc. of IEEE*, pp. 1177-1186, 1983.

[65] W. Peng and X. Lu, "AHBP: An Efficient Broadcast Protocol for Mobile Ad Hoc Networks," *Journal of Science and Technology*, Beijing, China, 2002.

[66] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for Mobile Computers," *Comp. Commun. Rev.*, pp. 234-44, Oct. 1994.

[67] C. E. Perkins, E. M. Belding-Royer, and S. R. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF MANET working group, Internet Draft, draft-ietf-manet-aodv-13.txt, February 2003.

[68] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *Proc. of 2nd IEEE Wksp. Mobile Comp. Sys. and Apps.*, pp. 90-100, February 1999.

[69]   C. E. Perkins and E. M. Royer, "Ad hoc on demand distance vector (AODV) routing," IETF Inter Draft, draft-ietf-manet-aodv-09.txt, November 2001 (Work in Progress).

[70]   A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica. "Geographic routing without location information," *Proceedings of the 9th annual international conference on Mobile computing and networking* (MobiCom '03), pp. A96-108.

[71]   Y. Sasson, D. Cavin, and A. Schiper, "Probabilistic broadcast for flooding in wireless mobile ad hoc networks," in Swiss Federal Institute of Technology, Technical Report IC/2002/54, 2002.

[72]   A. Savvides, C.-C. Han, and M. Srivastava, "Dynamic fine-grained localization in ad hoc networks of sensors," *Proc. of the 7th annual international conference on Mobile computing and networking* (MobiCom '01), pp. 166-179.

[73]   J. Scott, et al. "{CRAWDAD} data set cambridge/haggle (v. 2006-01-31)," Downloaded at http://crawdad.cs.dartmouth.edu/cambridge/haggle, January 2006.

[74]   Y. Sesson, D. Cavin, and A. Schiper, "Probabilistic broadcast for flooding in wireless mobile ad hoc network," Tech. Rep. IC/2002/54, École Polytechnique Fédérale de Lausanne, Switzerland, July 2002.

[75]   S. Sivavakeesar and G. Pavlou, "Scalable location services for hierarchically organized mobile ad hoc networks," *Proc. of the 6th ACM international Symposium on Mobile Ad Hoc Networking and Computing* (MobiHoc '05). pp. 217-228.

[76]   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, January 2002.

[77]   M. T. Sun, W. C. Feng, and T. H. Lai, "Location Aided broadcast in wireless ad hoc networks," *Proc. of 2001 IEEE Global Telecommunications Conference* (GLOBECOM '01), vol.5 pp. 2842-2846.

[78]   A. T. Toyserkani, E. G. Strom, and A. Svensson, "An efficient broadcast MAC scheme for traffic safety applications in automotive networks," *Proc. of IEEE Wireless Communications and Networking Conference* (WCNC 2006). vol.4, pp.2100-2105.

[79]   M. Tsogas, A. Polychronopoulos, and A. Amditis, "Using digital maps to enhance lane keeping support systems," *Proc. of IEEE 2007 Intelligent Vehicles Symposium*, pp.148-153.

[80]   Turner-Fairbank Highway Research Center, "Nationwide Differential Global Positioning System Program Fact Sheet," Nov. 2004; Available: http://www.tfhrc.gov/its/ndgps/02072.htm

[81]   B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks", *Proc. of the 3rd ACM international Symposium on Mobile Ad Hoc Networking &Amp; Computing* (MobiHoc '02), pp. 194-205.

[82]   L. Wischhof, A. Ebner, H. Rohling, "Information dissemination in self-organizing intervehicle networks," *IEEE Transactions on Intelligent Transportation Systems*, vol.6, no.1, pp. 90-101 , March 2005.

[83]   J. Wu and F. Dai, "A Generic Distributed Broadcast Scheme in Ad Hoc Wireless
        Networks," *Proc. of the 23rd IEEE International Conference on Distributed Computing
        Systems* (ICDCS 2003), pp. 460-467.

[84]   Y. Xu, J. Heidemann, and D. Estrin. "Geography-informed Energy Conservation for Ad
        Hoc Routing," *Proc. of the 7th annual Int'l Conf. on Mobile Comp. and Networking*,
        pp.70-84, 2001.

[85]   X. Zeng, R. Bagrodia, M. Gerla "GloMoSim: a library for parallel simulation of large-
        scale wireless network," *Proc. of the twelfth workshop on Parallel and distributed
        simulation*, pp. 154-161, May 1998.

[86]   C. Zhu, M.J. Lee, and T. Saadawi, "A border-aware broadcast scheme for wireless ad hoc
        network", *Proc. of First IEEE Consumer Communications and Networking Conference*,
        (CCNC 2004), pp. 134-139.