GrooveSim: A Topography-Accurate Simulator for Geographic Routing in Vehicular Networks

Rahul Mangharam† Daniel S. Weller† Daniel D. Stancil† Ragunathan Rajkumar†
Jayendra S. Parikh§

†Dept. of Electrical & Computer Engineering
Carnegie Mellon University, U.S.A
{rahulm, dweller, stancil, raj}@ece.cmu.edu

§General Motors Corporation
Warren, MI. USA
jayendra.s.parikh@gm.com

ABSTRACT

Vehicles equipped with wireless communication devices are poised to deliver vital services in the form of safety alerts, traffic congestion probing and on-road commercial applications. Tools to evaluate the performance of vehicular networks are a fundamental necessity. While several traffic simulators have been developed under the Intelligent Transport System initiative, their primary focus has been on modeling and forecasting vehicle traffic flow and congestion from a queuing perspective. In order to analyze the performance and scalability of inter-vehicular communication protocols, it is important to use realistic traffic density, speed, trip, and communication models. Studies on multi-hop mobile wireless routing protocols have shown the performance varies greatly depending on the simulation models employed. We introduce GrooveSim, a simulator for geographic routing in vehicular networks to address the need for a robust, easy-to-use realistic network and traffic simulator. GrooveSim accurately models inter-vehicular communication within a real street map-based topography. It operates in five modes capable of actual on-road inter-vehicle communication, simulation of traffic networks with thousands of vehicles, visual playback of driving logs, hybrid simulation composed of real and simulated vehicles and easy test-scenario generation. Our performance results, supported by field tests, establish geographic broadcast routing as an effective means to deliver time-bounded messages over multiple-hops.

Categories and Subject Descriptors

I.6.8 [**Simulation and Modeling**]: Types of simulation – *discrete event and visual*. C.2.2. [**Network Protocols**] *Routing protocols.*

General Terms

Design, Experimentation, Performance

Keywords

Multi-hop wireless networks, vehicular networking modeling and simulation.

1. INTRODUCTION

Over the past five years, travelers in the top 75 urban areas in the USA spent over 3.5 billion hours in traffic delays annually. The cost of congestion due to wasted fuel and hourly wages is estimated to be on the order of $63 billion per-annum. Vehicular traffic incidents account for twenty-five percent of the delay [1]. Instantaneous broadcasts alerting local travelers of vehicle incidents and favorable alternate routes will alleviate such congestion. Such delay-sensitive messages with local relevance may be propagated from vehicle-to-vehicle by equipping each with wireless interfaces capable of multi-hop networking. The performance of protocols designed for vehicular networks must be tested over a spectrum of realistic traffic scenarios and network conditions.

This paper presents GrooveSim, a topography-accurate street-map based vehicle network simulator and GrooveNet, a geographic routing protocol for vehicular networks. We have deployed GrooveSim over 400 miles of city and rural driving in five vehicles to evaluate network protocols and extracted mobile wireless propagation models. As shown in Fig. 1, each vehicle is equipped with a GrooveNet portable networking kit consisting of a Dedicated Short Range Communication (DSRC) based 5.9GHz transceiver, a differential Global Positioning System (GPS) receiver, a cellular modem, and audio/video equipment. A Linux-based laptop ran GrooveSim in "Drive" mode so all vehicles could communicate over the air interface across multiple hops. In this paper, we assume all vehicles have an IEEE 802.11a-based DSRC transceiver and know their position via GPS.

While on-road driving provides a suitable environment for evaluating the robustness of our protocol, we designed GrooveSim to analyze its performance and scalability. GrooveSim can simulate and evaluate protocols across thousands of vehicles driving along US roads while communicating with neighbors within transmission range, a feat we cannot easily accomplish with real vehicles. The focus of this paper is on GrooveSim, its design and the performance analysis of geographic broadcast routing protocols. GrooveSim includes various mobility, trip, communication and traffic density models. It operates in five modes capable of actual on-road inter-vehicle communication, simulation of traffic networks with thousands of vehicles, visual playback of driving logs, hybrid simulation composed of real and simulated vehicles and easy test-scenario generation. We aim to provide the Vehicular Ad hoc Networking (VANET) community with a stable and easy-to-use vehicular network simulator.

As opposed to Mobile Ad hoc Networking (MANET) routing protocols, the central premises for vehicular networking protocols are to (a) provide safety alerts and emergency warnings quickly to *all local vehicles*, (b) ensure that not only the destination vehicle receives the message but also *all approaching traffic* and (c) maintain the event notification *along the routed path* for the *lifetime* of the event.

VANET'05, September 2, 2005, Cologne, Germany.

Figure 1. GrooveNet vehicular networking test kit

While adapting the proposed MANET protocols is tempting, vehicular networks differ in four key ways. Vehicular networks are characterized by rapid (relative speeds up to 300kmph) but predictable topology changes, a small effective diameter, frequent fragmentation and limited redundant paths. Furthermore, vehicular networks have well-specified application categories which favor broadcast protocols over generic path-based end-to-end MANET protocols.

1.1 Vehicular Networking Application Classes

Instead of attempting to cover a broad range of applications, we focus on three key application categories, which users currently regard as high value services: time-critical safety alerts, non-critical traffic updates and commercial services [2].

1) **Safety Alerts:** These encompass time-sensitive messages such as vehicle crash warnings, sudden braking and other crisis notifications. As shown in Fig. 2, we require a geographically-aware routing protocol that may choose a sequence of waypoints as intermediate hops to inform all approaching traffic along a well-defined highway when broadcasting an emergency event. Vehicles receiving the emergency notice may inform local authorities and avoid congestion by opting for favorable alternate routes and exiting the highway up to 2-3 miles away from the event. In urban settings, we keep the message "alive" around the intersection of interest to alert approaching vehicles. Finally, in a rural setting with sparse traffic, specifying a bounding-box to define the valid zone for accepting and forwarding the message will leverage vehicles in the vicinity to maintain connectivity.

2) **Traffic Updates:** This category includes useful services such as traffic congestion probing to compute one's estimated time of arrival, collaborative driving among a fleet of vehicles and local traffic updates from infrastructure nodes.

3) **Commercial Services:** Location-based services, such as room availability in local hotels and multimedia downloads at gas stations are also of interest.

In the above applications, we employ packet diffusion for traffic updates and commercial services. This scheme is based on simple message exchange among neighbors where receivers aggregate, filter and rebroadcast information relevant to the supported application services. For safety alerts which define a target audience region, we use directed broadcast where messages are accepted and rebroadcast periodically while the vehicle is within the target region. Vehicles outside the geographic region drop the message, thus controlling message flooding. However, for directed broadcast, unlike MANET protocols [3, 4], a destination vehicle is not defined and unlike position-based protocols [5, 15] the broadcast region is defined only by intermediate waypoints. Furthermore, the protocol periodically re-broadcasts messages over the event lifetime.

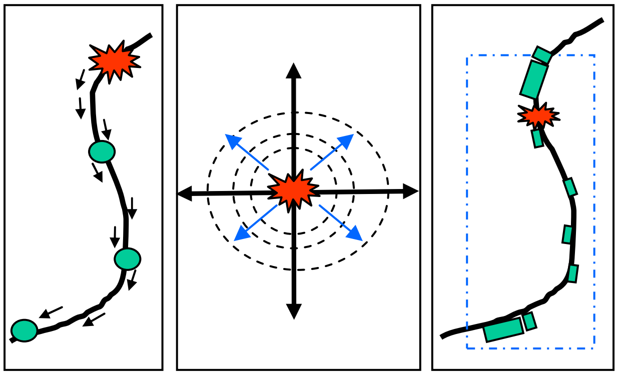


Figure 2(a) Waypoint based broadcast routing for highway driving, (b) intersection alerts for city driving and (c) bounding-box routing for rural driving

1.2 Related Work

A vehicular network is a special class of mobile ad hoc network with well-defined applications and more severe operating constraints in terms of network fragmentation. Mobile wireless and vehicular research has been primarily been pursued by the MANET and Intelligent Transportation System (ITS) communities.

Over the past two decades, there have been a large number of proposals for path-based end-to-end routing protocols under the auspices of MANET [3, 4]. These protocols aim to establish a connection between a source node and one or more destination nodes. However, in a vehicular networking context, the principal descriptors of a vehicle are its position, heading and speed. In [6], path-based MANET protocols are demonstrated to be unsuitable for vehicular networking because the protocols do not provide stable paths across multiple vehicles for even moderate durations. It is important to minimize any handshaking and shared state information among nodes.

Furthermore, as most MANET protocols are proposed to be generic solutions, they are evaluated with arbitrary or unrealistic mobility models such as random waypoint, random direction, random walk, and probabilistic versions of random walks with correlated speed and direction. Several mobility studies [7, 8, 9] show that the results obtained from different models vary widely and do not realistically represent vehicular traffic speeds, directions and trips.

On the other hand, the ITS initiative has developed over 40 vehicular traffic simulators [10] based on car following and lane changing algorithms to determine vehicle movements. The primary focus of these has traditionally been on traffic flow analysis and forecasting.

GrooveSim has been developed to simulate mobile wireless protocols in realistic traffic settings by employing a range of mobility and communication models within the spatial framework of street maps. Since the same implementation is used for actual on-road driving tests, GrooveSim is well-suited to both analyze and stress-test vehicular networks.

1.3. Organization of the Paper

We first provide an overview of the design of the GrooveSim simulation tool. In section III, the GrooveNet routing protocol is described. Section IV presents the experiments conducted and their performance results followed by the conclusion.

2. GROOVSIM SIMULATOR DESIGN

We now describe the design of the GrooveSim tool, the different component-based models included and its modes of operation. GrooveSim includes basic speed, trip, communication and traffic models and has been designed to be extensible so researchers may include their own models with ease.

2.1 GrooveSim Architecture

GrooveSim generates street level maps for any place in the USA by importing TIGER/Line (Topologically Integrated Geographic Encoding and Referencing) [12] files available free from the US Census Bureau. The TIGER/Line files constitute a digital database of geographic features, such as roads, railroads, rivers, lakes, and legal boundaries, covering the entire United States. The database contains information about road segments as records which include their location in latitude and longitude, name, type, address ranges, speed limits, and other related information. The database is composed of text records and does not include any graphical images. GrooveSim is based on open-source roadnav [16] with significant additions including a graph-based abstraction of streets, networking, simulation models, and a cross-platform graphical user interface in Qt [17].

In order to represent the network of roads as traversal paths, GrooveSim abstracts the TIGER records into a planar graph with an array of edges and an array of vertices. Each record corresponds to an edge, which includes the street segment's attributes and points to at most two vertices (e.g. street intersections). Each vertex points to an array of records corresponding to the incident edges.

Using this graph abstraction, we are able to implement efficient vehicle (not packet) minimum weight routing such as the shortest path or fastest path between two vertices. In addition, each county's map is divided into a 10x10 grid of map-regions. GrooveSim has been implemented for Linux-based platforms using C++ and the street maps are rendered by the Qt graphics library. The example in Fig 3 illustrates 1000 vehicles initialized in a Chicago, IL suburb. Each vehicle has a specified origin, start time, trip model, explicit destination, speed model and communication model.

2.2. GrooveSim Operation Modes & Features

GrooveSim has five distinct modes of operation which are enumerated below:

1) Drive Mode: GrooveSim can process National Marine Electronics Association (NMEA) 0183 format data from a GPS unit to provide a real-time map of the vehicle's current location. It can communicate using the 5.9GHz modified 802.11a/DSRC transceiver with multiple vehicles and display all connected vehicles on the map. It is able to form UDP connections with other vehicles, spanning multiple hops with the routing protocol. A vehicle is able to trigger and send safety message broadcasts and stream files between reachable vehicles. In addition, a vehicle can establish TCP connections over a 1xRTT cellular connection so a remote node may monitor the driving activity on the road via the Internet. GPS coordinates and vehicle positions are updated periodically with a maximum rate of 5Hz. This mode evaluates the robustness and protocol testing in real traffic and channel conditions.

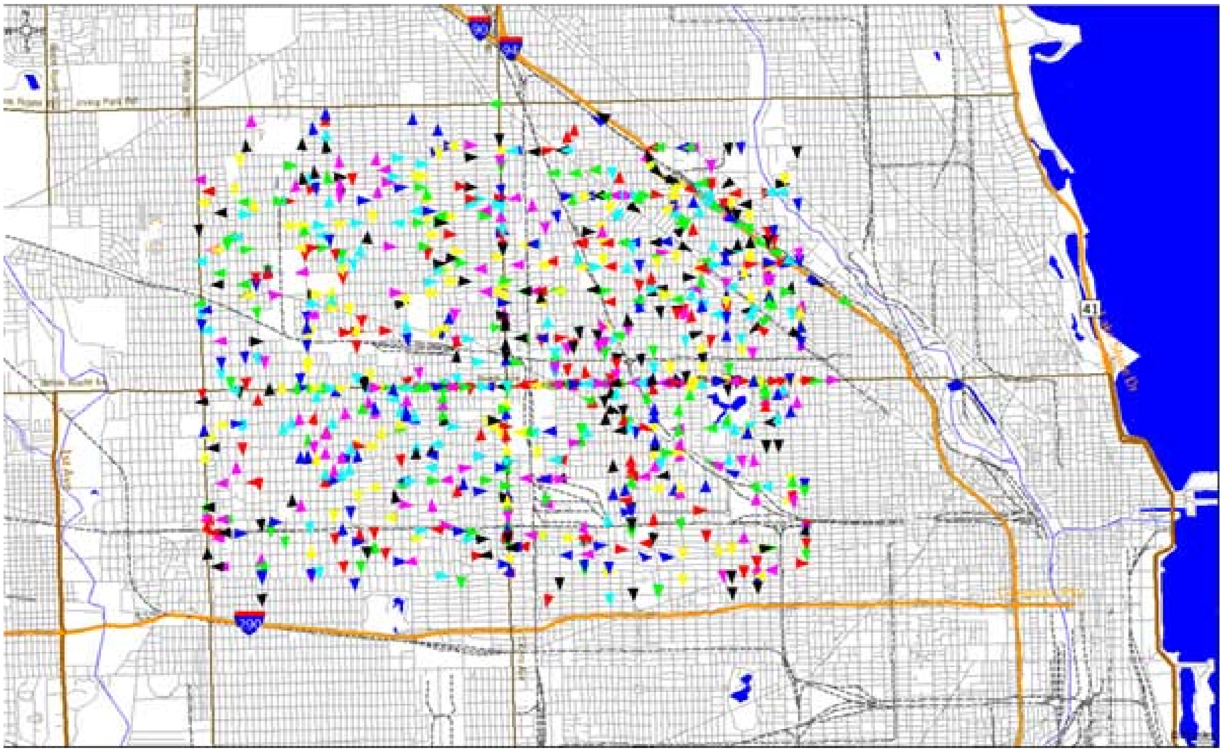


Figure 3. A simulation with 1,000 vehicles with a 200m communication range in Chicago, IL.

2) Simulation Mode: GrooveSim can support thousands of concurrently moving and communicating vehicles. Each vehicle may have its own mobility, trip, and communication model. Vehicles may start after a specified delay and travel in "simulation time" so a day of travel for 75 vehicles can be completed in 20 actual minutes. Users may also choose to view all vehicles graphically. This helps evaluate the scalability and performance under various traffic loads. GrooveSim automatically downloads and extracts the required TIGER/Line files via the Internet if they are not available locally.

3) Playback Mode: Both *Drive* and *Simulation* modes log the movement and communication of vehicles using the same format. Trips may be played-back using VCR-like controls and users can fast-forward and rewind through a log file using the graphical interface. This permits visual analysis with reproducible results. In addition, performance data such as message penetration distance and delay, vehicle group sizes, packet transmission/reception and other metrics are analyzable. Propagation-related evaluators such as received signal-to-noise ratio, distance between vehicles, relative speed, packet drops, and retransmissions are logged. A Matlab-based analysis tool was built to estimate the path-loss exponent from each drive. In addition, GPS-related information such as number of visible satellites, heading, position and timing errors are also logged.

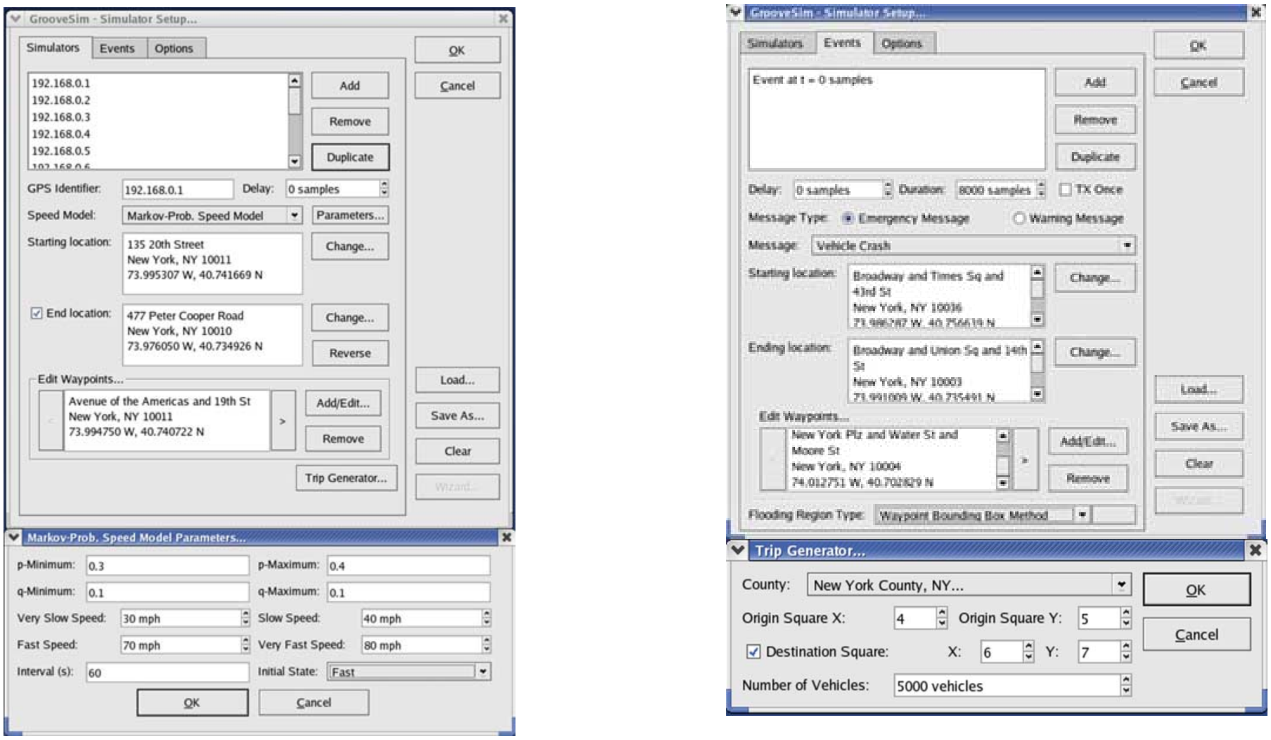
4) Hybrid Simulation Mode: In this mode, both real vehicles on the road and virtual vehicles interact with each other. A virtual vehicle may be simulated at the time of the test or played back from a log file. Each real vehicle may host multiple virtual vehicles or virtual vehicles may be hosted by the remote node connected via the 1xRTT cellular connection. This enables the evaluation of a large number of vehicles with realistic channel and traffic conditions along select links.

5) Test Generation Mode: This mode provides an easy test scenario generation with 1000's of vehicles, each possibly with different models and parameters. As shown in Fig. 4(a), a vehicle ID, speed model, its origin, destination and intermediate waypoints along its route are specified. In Fig. 4(b), events of different types may be generated at a specified start time. The geographic region for message traversal is specified by the event origin, intermediate waypoints and destination. The flooding region type dictates the shape of the flooding region delineated by the waypoints. Finally, the trip model and start times may be specified via the GUI. In addition, speed and start-time distributions may be generated via an external application (e.g. Matlab, Excel) and input as a text file. This ensures easy repeatability of experiments and permits the user to modify settings before each simulation run.

2.3 GrooveSim Vehicular Network Model

In order to fully describe the network model of a vehicle equipped with a communication device, we need to specify the start position and start time of the vehicle, the route that it selects, the speed at which it travels, and its communication range and reliability. Furthermore, several factors such as speed and communication reliability are functions of the density of nodes in close proximity and cannot be determined *a priori*. We describe basic models in each category by parameters settable at runtime. The models are easily extensible to represent more complex behavior.

1) Mobility Model: A vehicle's speed and trajectory are specified by its mobility model. GrooveSim supports four basic mobility models: (a) Uniform speed model with a minimum and maximum speed, (b) 4-state Markov-based probabilistic model, (c) Load-based model and (d) Street-map based maximum speed



192.168.0.1
192.168.0.2
192.168.0.3
192.168.0.4
192.168.0.5

GPS Identifier: 192.168.0.1
Delay: 0 samples
Speed Model: Markov-Prob. Speed Model
Starting location: 135 20th Street
New York, NY 10011
73.995307 W, 40.741669 N
☑ End location: 477 Peter Cooper Road
New York, NY 10010
73.976050 W, 40.734926 N
Edit Waypoints...: Avenue of the Americas and 19th St
New York, NY 10011
73.994750 W, 40.740722 N

p-Minimum: 0.3
p-Maximum: 0.4
q-Minimum: 0.1
q-Maximum: 0.1
Very Slow Speed: 30 mph
Slow Speed: 40 mph
Fast Speed: 70 mph
Very Fast Speed: 80 mph
Interval (s): 60
Initial State: Fast

Event at t = 0 samples

Delay: 0 samples
Duration: 8000 samples
☐ TX Once
Message Type: ● Emergency Message ○ Warning Message
Message: Vehicle Crash
Starting location: Broadway and Times Sq and 43rd St
New York, NY 10036
Ending location: Broadway and Union Sq and 14th St
New York, NY 10003
Edit Waypoints...: New York Plz and Water St and Moore St
New York, NY 10004
74.012751 W, 40.702829 N
Flooding Region Type: Waypoint Bounding Box Method

County: New York County, NY...
Origin Square X: 4
Origin Square Y: 5
☑ Destination Square: X: 6 Y: 7
Number of Vehicles: 5000 vehicles

Figure 4(a) Vehicle simulation settings with speed model and (b) traffic event settings with vehicle trip generator

Figure 5. Vehicles routed with least cost paths gradually migrate to higher speed roads in Chicago, IL suburb

model specified by actual road speed limits. For vehicles with initial positions given in Fig. 3, we observe that over time, as vehicles are routed with least-cost paths, they prefer main roads to tributary streets since main roads have higher speed limits, as in Fig 5.

The Markov model employs four states: two states for slower "city" driving and two states for faster "highway" driving. Each state specifies the current speed of the vehicle and state transition probabilities between fast and slow states are settable parameters. The parameters can be set to match the mean and n^{th}-percentile speed from measured data [13, 14]. The load-based model is useful for uninterrupted travel speed and is described by a simple function:

$$v_u = 3600/t_u = 3600/(t_r + d_{tu})$$

where v_u is the uninterrupted travel speed, t_u is the uninterrupted travel time per unit distance, d_{tu} is the traffic delay per unit distance and is a function of the capacity saturation on the given road, and t_r is the free flow travel time per unit distance.

2) Trip Model: Vehicles can embark on three types of trips: (a) a random walk where vehicles are biased against taking a u-turn, (b) an explicit origin, a destination and intermediate waypoints, and (c) a random origin and destination constrained within map regions. Both vehicles in (b) and (c) are routed with least-cost routing where the cost is the travel distance or time. This permits the experimenter, for example, to generate swarms of vehicles randomly walking in congested downtown areas and also large groups of vehicles leaving downtown for the suburbs. Special scenarios, such as at the end of an NFL football game, can thus be conveniently modeled.

3) Communication Model: The communication model employs three schemes to model the channel state and multiple access schemes: (a) a simple two-state Gilbert-Elliot Markov model to generate packet errors with a known mean error rate, (b) a random access collision model and (c) a channel model with packet error rates and path loss determined by actual on-road measurements. The transmission range has a default of 100m and may be modified at runtime. Fig. 6 illustrates on-road measurements of signal-to-noise ratio and the packet error rate for various distances between moving vehicles. The measurements were taken in an urban environment at a mean speed of 40mph. We observe that beyone 100m, the SNR falls below the sensitivity threshold specified by the 802.11a standard [18].

4) Traffic Generation: Vehicular traffic may be generated and distributed according to annual average daily volumes or with probabilistic start times. A list of start times from real or synthetic traces may be input via a text file.

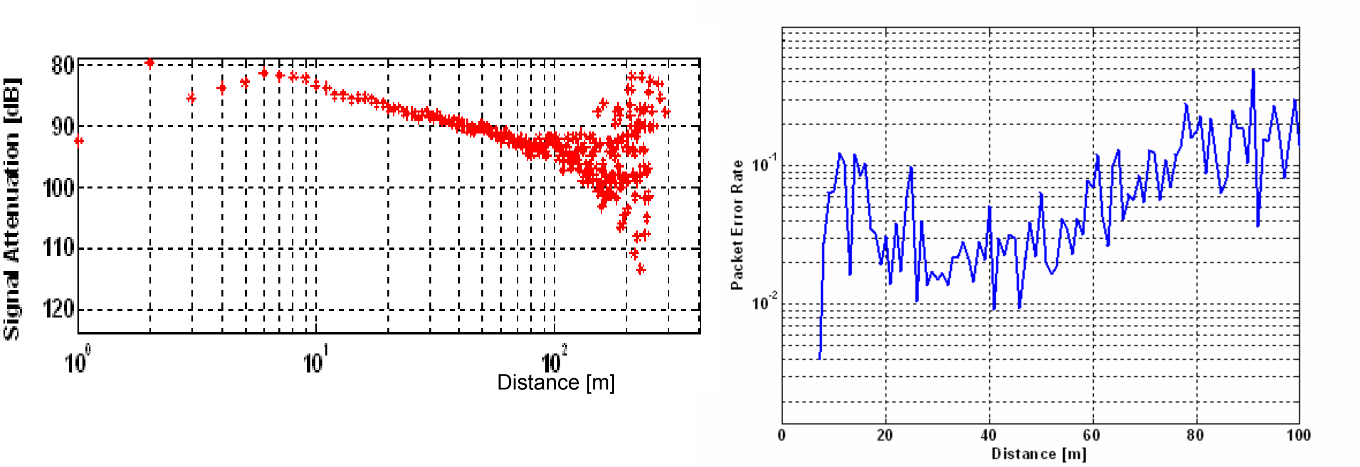


Figure 6. (a) Variation of Signal to Noise Ratio with relative distance between vehicles. (b) Variation of packet error rate with distance

5) Limitations: While the TIGER map database provides sub-50m accurate spatial data, it unfortunately does not state which roads are one-ways, the number of lanes on each road, the altitude of overlapping roads (to determine intersections versus overpasses or underpasses) and does not mark any traffic signals. While the load-based speed model results in vehicular congestion regions, we do not employ a car-following model as in microscopic simulators mentioned in [10]. The focus of this paper is on providing basic models and illustrating their impact on key performance metrics. GrooveSim's extensible architecture facilitates addition of user-defined modules with ease.

2. GROOVENET PROTOCOL

We briefly describe GrooveNet, the geographic broadcast protocol used to evaluate GrooveSim. Traffic incidents are primarily relevant to vehicles in the vicinity and more so to vehicles approaching the event. This local relevance and need for rapid message dissemination forms the basis of the GrooveNet protocol. As vehicular networks are by definition in a state of constant and rapid movement, the network is frequently and widely fragmented thereby eliminating the use of cluster-based and end-to-end path-based protocols. We verified this using the Linux implementation of AODV routing protocol [19] and a standardized implementation of the H.323 streaming protocol [20] for low-rate audio and video communication between five vehicles driving through urban areas. While the streaming worked well over a single hop, the outage rate across multiple hops was significant (>50%) due to frequent route errors.

In order to locally disseminate safety messages, GrooveNet is designed to be an opportunistic broadcast protocol with minimal handshaking between sending and receiving parties and with little or no shared state information among neighboring vehicles. To maintain the local relevance and minimize traffic in regions not affected by the traffic incident, the originator of the message, termed as the *event originator*, specifies a valid routing region surrounding itself. This region may be described by a set of physical waypoints along a road (Fig. 2a), a circle centered at the origin with a radius *r* (Fig. 2b) or vertices of a polygonal bounding box (Fig. 2c). Nodes within the valid routing region accept and periodically re-broadcast the message over the lifetime of the event. Vehicles outside the valid routing region drop the packet and do not participate in any forwarding. The message's relevance is based on the relative position of the event originator and the receiving node and not on the receiving node's identifier. Unlike location-aided routing protocols [5] which aim to find a node destination with logical address, GrooveNet's goal is to keep the message "alive" along a path specified by physical positions. In other words, continuous re-broadcasting the event's message within the routing region is more important than reaching a particular destination. As all vehicles are synchronized by GPS, the event lifetime for which vehicles may re-broadcast the message results in alerting more vehicles.

2.1 GrooveNet Implementation

In our implementation, GrooveNet supports two modes of messaging: diffusion and directed broadcast. In message diffusion mode, vehicles periodically exchange non-critical data such as speed information for congestion probing, collaborative driving updates and road conditions. Vehicles accept all packets in promiscuous mode, filter the data and re-broadcast a message with data relevant to its travel path.

General Frame Header [4B] -
[2b] Protocol Version
[4b] Frame Type
[4b] Frame Sub-Type
[12b]Frame Sequence Number
[4b] Fragment Sequence Number
[1b] More Fragments Flag
[1b] Message Direction: From/To Gateway
[4b] Reserved

Net work Sub-header [50B] -
[8B] Vehicle Globally Unique ID
[4b] Routing Region Type
[40B] Routing Region Description
[4b] Rebroadcast Frequency
[4b] Event Duration
[4b] Reserved

GPS Sub-header [26B] -
[4B] Source GPS Time stamp
[4B] Source Latitude
[4B] Source Longitude
[9b] Source Direction
[9b] Source Speed
[4B] Destination Latitude
[4B] Destination Longitude
[9b] Destination Direction
[5b] Reserved

Figure 7. GrooveNet frame format where b and B signifies length in bits and bytes respectively

In directed broadcast mode for time-critical alert messages, the event has an origin, a destination position or routing region defined by intermediate waypoints - all described by physical locations and not by mobile node identifiers. The waypoints are GPS coordinates and function as routing hints in a source routing fashion. Due to the intermediate waypoints, messages are forwarded in a greedy fashion and do not suffer from the local maxima problem present in traditional greedy geographic routing [11]. A single 80-byte frame header is described in Fig 7.

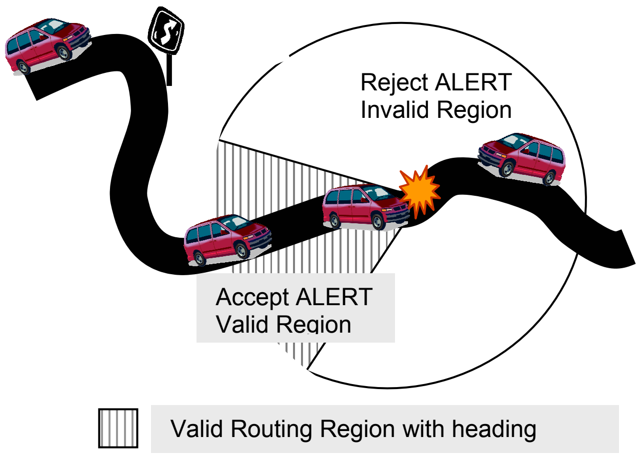


Figure 8. GrooveNet sector-based routing region. All vehicles within the sector may accept and re-broadcast the message.

Figure 9. 200 vehicles traveling in the same direction along I-80 in Ohio. Four events are marked at different locations.

Frame types for critical alerts, traffic updates and commercial messages have been defined. Each time-critical alert specifies a routing region based on the severity of the event. The routing region is expressed by a region type and a maximum of five waypoints which describe the region. The destination position marks the only required point of the routing region and is usually the furthermost point of the region. For example, as in Fig. 8, we implemented a sector-based routing region. A vehicle that issues the alert triggered by an airbag deployment specifies a circular region with its position and a radius in meters. In addition, a heading range is defined. Only vehicles within the circle and the heading range may accept and rebroadcast the message. In order to prevent flooding of the network, the non-critical message re-broadcast frequency is set to one Hertz by default. Also, messages are cached and not re-broadcast during an interval a neighbor broadcasts the same message.

In our on-road experiment (*drive* mode), we set the heading range to be +/- 30 degrees of the event vehicle's heading so only vehicles approaching the event may be notified. We measured the distance around the circumference at which the vehicles stopped re-broadcasting as they left the circle and resumed re-broadcasting as they re-entered the circle. For vehicles traveling at 40mph, we observed an accuracy of 12 meters. While the GPS receiver has an accuracy of 2m for a stationary node, the larger error is due to the 1 second message rebroadcast interval and the fact that the GPS receiver uses a stabilization algorithm which results in a small lag until the true position is realized.

4. PERFORMANCE RESULTS

The TIGER map database restricts vehicle trajectories to roadways, which results in a different connectivity distribution when compared to a random walk in open space. As mobility and topology have a major influence on the connectivity and therefore performance of multi-hop wireless networks, the street structure delivers more realistic performance results. In order to evaluate the usefulness of vehicular networking, we first list the degrees of freedom and the evaluation metrics. In a vehicular network, speed, vehicle density, travel direction, size of routing region, message rebroadcast frequency, transmission power and start time distribution are the key degrees of freedom that influence the connectivity between vehicles.

Three simulation tests were conducted to evaluate the impact of the above degrees of freedom on the message penetration distance, message lifetime and end-to-end message delay. The message penetration distance is the maximum distance from the event the message traverses in the valid routing region. The message lifetime is defined as the duration a message is actively re-broadcasted and received by nodes in the valid routing region. Finally, the end-to-end message delay is the time it takes to send a message from the start position of an event within the valid routing region to its destination.

4.1 Impact of Vehicular Density, Clustering & Direction on Message Penetration Distance

The number of vehicles in a region and the distribution of the vehicles have a large impact on the message penetration distance. In this test our goal is to quantify this statement so that by observing vehicles at one point along a highway, we may predict the penetration distance at events located further down the highway. The messages are routed from vehicle to vehicle along multiple hops.

In Fig. 9, we routed 200 vehicles along a 20km stretch of Interstate-80 in Ohio from west to east. We placed seven events along the route and recorded the message penetration distance in the direction of the oncoming traffic. The events are valid only for vehicles west of the event and re-broadcasting ceases as vehicles pass to the east of the event.

The maximum penetration distance is a function of the size of the longest connected group of vehicles. As vehicles are started sequentially, each vehicle is initially spaced from its previous and subsequent vehicle. This is accomplished by randomly starting a vehicle within an average of S seconds from the start of the previous vehicle. By varying S we can control the traffic density and vehicle clustering. Vehicles use a uniform speed model where the speed of a vehicle is selected between 30-40mph at its start time. Each vehicle has a transmission range of 200m.

At each event's location, we observe the arrival rate (or how spaced apart the vehicles are) and note the variation of message penetration distance. In Fig. 10(a), we observe the message penetration is low at the closest event location – limited to five times the transmission range. It then gradually increases and peaks around the 10Km marker to 15 times the transmission range.

This behavior can be explained from the variation of the mode of the group size (i.e. the size of the group of connected vehicles that occurred most often at that event). In Fig. 10(b), initially the most common group size is large indicating the vehicles are very close to each other and then gradually spread out due to the 30-40mph speed differential. When the vehicles are closely packed, the message penetration distance is small as the vehicles are in large

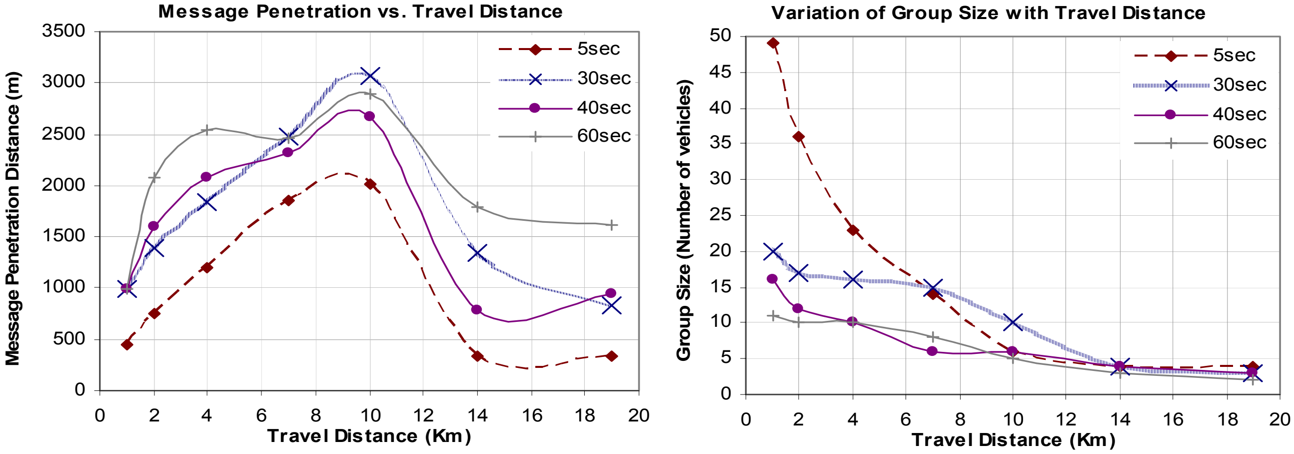


Figure 10. (a) Message penetration distance and (b) vehicle communication group size over travel distance

but tightly-packed discrete clusters. As they spread out (due to small variations in their initial speed) they form longer chains around the 10Km marker and then begin to rarefy and disconnect. This trend is most evident in vehicles spaced out by an average of *S*=5 seconds from their previous and subsequent neighbors. These vehicles are initially within 70m of the closest neighbor as the minimum speed is 30mph. Due to the spread of speed between 30-40mph, the vehicles gradually rarefy and the group size decreases.

From a practical perspective, if the average vehicle arrival rates and speeds are observed at the first event location, we can predict the penetration distance at event locations further down the highway. With this viewpoint, as vehicles travel with a constant speed (e.g. use of cruise control in different lanes), the density and cluster sizes are a function of the distance traveled.

We repeated the same test for vehicles in both directions (west to east and east to west) and observed that the message penetration was solely a function of the speed of the vehicle in the opposite direction rather than multi-hopping across vehicles. This is due to the fact that while the mobility model tends to cluster vehicles due to the close start times, the clusters are spaced far apart from each other due to the difference in fixed speeds. Thus to communicate between clusters the protocol must leverage vehicles traveling in the opposite direction and the range speeds must be large.

4.2 Effect of Routing Region on Message Lifetime

In this test we are interested in the duration a message is successfully re-broadcasted within the valid routing region. This is useful when a traffic incident has occurred at an intersection and there is a need to continuously warn vehicles approaching form all directions.

As shown in Fig. 11, we simulated 200 vehicles routed with random origins and destinations within two map-regions in central Manhattan. The vehicles moved with speeds uniformly distributed between 30-40mph. The emergency event occurs at the intersection of Broadway Avenue and Avenue of Americas in Manhattan, New York City.

In order to view the impact of the routing region on message lifetime, we run the test across seven circular routing regions with radius ranging from 50m to 500m. Only vehicles traveling within the valid region may accept packets broadcasted by other nodes within transmission range. While the results illustrated in Fig. 12 show the expected increase in lifetime as the routing region is enlarged, it also highlights that with a radius of just 300m the message lifetime is at one hour. Beyond the 300m radius, the message lifetime flattens as the message propagation was limited due to moderate network connectivity.

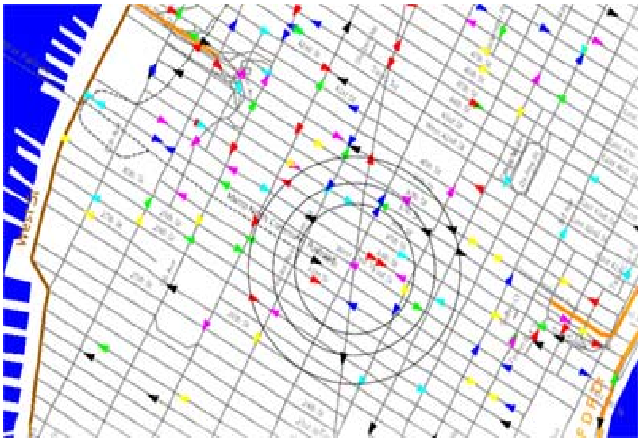


Figure 11. Message transmission about an intersection in Manhattan, New York City

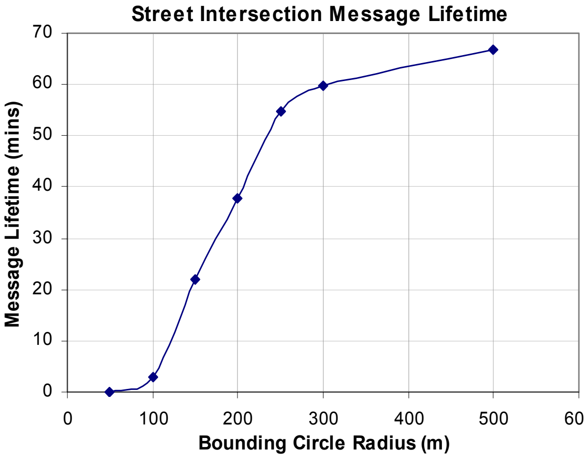


Figure 12. Variation of message lifetime with radius of valid routing region

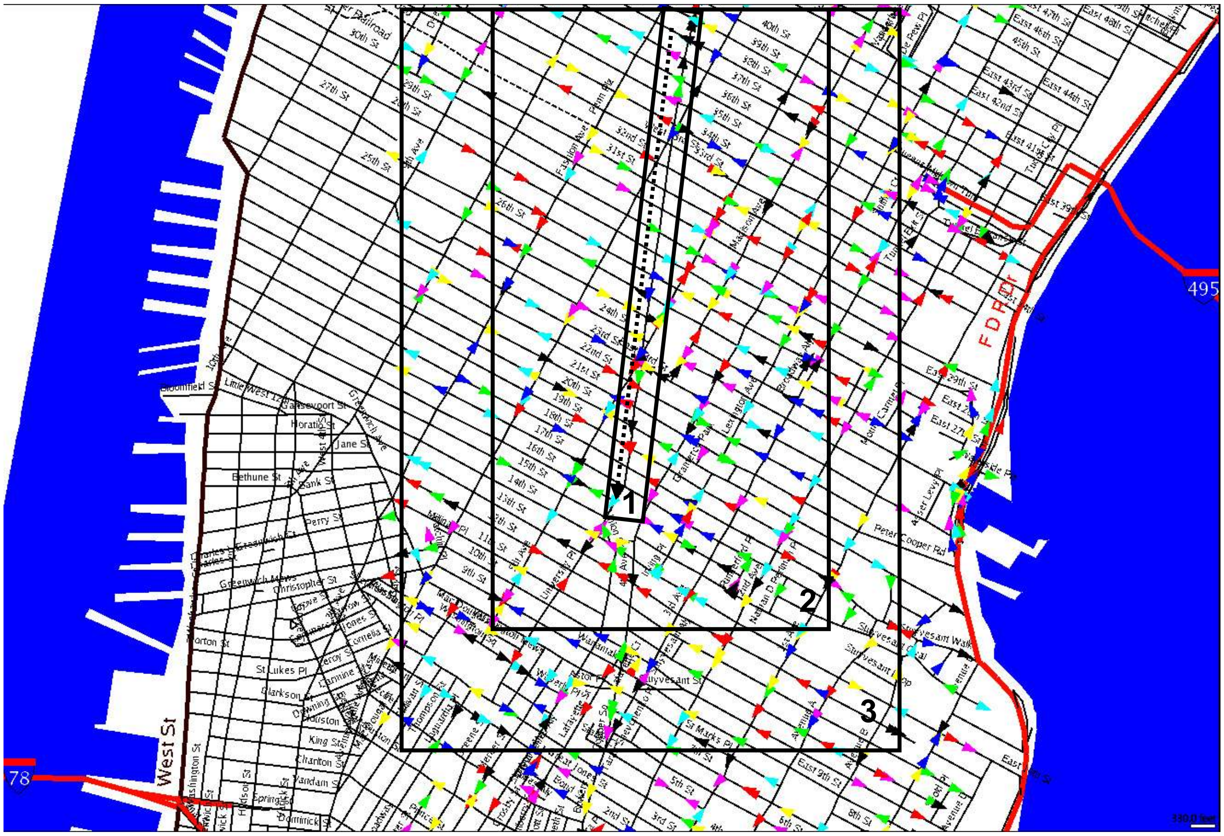


Figure 13. Directed broadcast of a message from 42nd Street and Broadway (Time Square) to 14th Street and Broadway in Manhattan, New York City. 200 vehicles overlaid with three routing bounding-boxes.

4.3 Influence of Routing Region on Message Delay

We now look at the benefit of multi-hop routing with geographically constrained flooding. In this test we focus on the time it takes a message to get from the origin of the event to the destination. By increasing the size of the routing region, we permit more vehicles to participate in the routing. In Fig 13, the event origin is at 42nd street and Broadway Avenue (Time Square in Manhattan, New York City). The routing region is described by a single waypoint located 2.4Km south at 14th street and Broadway Avenue. All messages are re-broadcast within the bounding-box containing the line connecting the source and destination locations. The metric of interest is the time it takes the message to reach the destination via directed diffusion.

If a vehicle were to drive at 30mph, it would take about 3.2 minutes or 192 seconds. As shown in Table 1, by using a bounding box that fits Broadway Avenue to just one avenue across, the message delay is reduced by almost a factor of five to 40.4 seconds. Further increasing the bounding box to two and then six avenues across, the message traversal delay is reduced to just 11 seconds. With each increase in size of the bounding box, more vehicles are available to complete the connectivity graph. Increasing the bounding box beyond a certain size does not reduce the message delay further as enough vehicles along the shortest path are within the valid region.

Table 1. Effect of Bounding Box Size on Message Delay

Bounding Box Size	# Active Vehicles	Message Delay (sec)
0	1	192
1	138	40.4
2	150	19
3	162	11

The results in the above three experiments provide estimates of values that may be observed in reality. On-road experiments with five vehicles verify the effect of the routing regions.

5. CONCLUSION

The concept of a multi-hop wireless vehicular network is a key enabling technology that will make driving safer, more efficient and entertaining. Our focus is on delivery of time-sensitive safety alerts, useful traffic updates and commercial services. To analyze the scalability and performance of message delivery between vehicles, we developed GrooveSim, a street map-based vehicle network simulator. To evaluate GrooveSim, we employ GrooveNet, a diffusion-based geographic broadcast routing protocol.

GrooveSim is a hybrid vehicular network simulation tool based on accurate street map geography. It includes a variety of mobility, trip, communication and traffic density models. It is capable of actual on-road inter-vehicle communication, simulation of traffic networks with thousands of vehicles, visual playback of driving logs, hybrid simulation composed of real and simulated vehicles and easy test-scenario generation.

Using GrooveSim, we determined the message penetration distance, message delay and lifetime in city, rural and highway contexts. Our simulation results, supported by field tests, establish geographic broadcast routing as an effective means to deliver time-bounded messages over multiple-hops.

6. REFERENCES

1. Schrank, D., Lomax, T. *The 2005 Annual Urban Mobility Report.* Texas Transportation Institute. http://mobility.tamu.edu
2. Murph, T. "Do or Die", *Ward's AutoWorld.* Volume 38, Number 10, page 56. Oct 2002.
3. E. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Communications Magazine*, vol. 6, no. 2, pp. 46-55, April 1999.
4. J. Broch, D. Maltz, D. Johnson, Y.-C. Hu, and J. Jetcheva, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," *Proceedings of ACM MobiCom,* Dallas, TX, USA, pp. 85-97, October 1998.
5. Mauve, M., Widmer, J., Hartenstein, H. "A Survey on Position-Based Routing" *Mobile Ad-Hoc Networks. IEEE Network*. Vol. 15 (2001), Nr. 6, S. 30-39
6. J. Blum and A. Eskandarian and L Hoffman. "Challenges of Intervehicle Ad Hoc Networks." *IEEE Transaction on Intelligent Transportation Systems*, 5(4):347--351. 2004.
7. T. Camp, J. Boleng, and V. Davies. "A survey of mobility models for ad hoc network research" *Wireless Communications & Mobile Computing*, 2(5):483--502, 2002.
8. J. Yoon, M. Liu, and B. Noble. "Random waypoint considered harmful" *Proceedings of Infocom '03*, pp. 1312-1321, San Francisco, USA, April 2003.
9. "Sound Mobility Models," J. Yoon, M. Liu, B. Noble, *ACM Mobicom*, San Diego, California, 2003
10. Boxill S. A. and Yu L., "An Evaluation of Traffic Simulation Models for Supporting ITS Development", *Technical Report 167602-1. Texas Southern University,* October 2000.
11. B. Karp and H. T. Kung. "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks" *Proceedings of MobiCom.* ACM Press, 2000.
12. TIGER/Line 2004. US Geological Survey (USGS) topographic maps http://www.census.gov/geo/www/tiger
13. Speed Trend Characteristics, http://www.ibiblio.org/rdu/gov-stat/speed-tr.html
14. "Traffic Speeds on English Trunk Roads: 2003", *Transport Statistics*, Department of Transport, U.K.
15. C. Maihöfer, "A Survey of Geocast Routing Protocols," *IEEE Communications Surveys & Tutorials*, vol. 6, no. 2, pp. 32-42, 2004.
16. Lynch, R., "Roadnav", http://roadnav.sourceforge.net
17. Qt Cross-platform GUI Development, http://trolltech.com
18. IEEE 802.11a, Part 11: High-speed Physical Layer in the 5 GHz Band, Supplement to IEEE 802.11 Standard, September 1999.
19. H. Lundgren, D. Lundberg, J. Nielsen, E. Nordström, C. Tschudin, "A Large-scale Testbed for Reproducible Ad hoc Protocol Evaluations"∞, WCNC 2002.
20. OpenH323 project, http://www.openh323.org/