Smoothing Vehicular Traffic Flow with VGrid

Behrooz Khorashadi[1], Bojin Liu[2], Haining Du[6]
Dipak Ghosal[3], Chen-Nee Chuah[4], Michael Zhang[5]
bkhorashadi@ucdavis.edu[1], frdliu@ucdavis.edu[2]
dghosal@ucdavis.edu[3], chuah@ucdavis.edu[4]
hmzhang@ucdavis.edu[5] , hndu@ucdavis.edu[6]
Department of Computer Science[2,3], Department of Electrical & Computer Engineering[1,4]
Civil and Environmental Engineering[5,6]
University California, Davis

Abstract

In this paper we explore a distributed application used to smooth traffic and reduce long term traffic incidents. To do this we use VGrid: an ad hoc networking and computing grid formed by leveraging inter-vehicle wireless communications. In addition to exchanging data between vehicles, VGrid actively uses pertinent data to perform computations for solving traffic-related problems. We examine the problem of smoothing vehicular traffic flow through the use of real-time position and velocity information exchanged over the network. This is accomplished through the application of 1) accident alert messages sent from the accident site or obstruction point and 2) dynamically calculated variable speed limit leveraging the available vehicular grid computing network. In order to evaluate these application we develop an extensive simulation tool which incorporates both a comprehensive network model and a realistic vehicle mobility model. Using this simulation tool we show that VGrid reduces vehicle speed variance, resulting in more homogeneous vehicle behavior in both freeflow and obstructed-lane scenarios.

I. Introduction

According to a recent report [1], the US loses billions of dollars due to traffic congestion. Therefore any traffic management measure that reduces traffic congestion by even a small proportion could translate into huge savings. Although the specific triggers of traffic congestion are numerous, one can broadly attribute them to ”bottlenecks,” or hot spots on the road. Two types of bottlenecks can be found: fixed bottlenecks and dynamic bottlenecks. The former can be places where two or more roads merge together, number of lanes drop to fewer numbers, grades become steeper, or curves become tighter. On the other hand dynamic bottlenecks can be locations of accidents that temporally blocks one or more lanes, or slowly moving vehicles in a traffic stream (such as trucks on a steep climb). It has been reported that accidents caused more than half of congestion hours in this country [1]. As such, measures to reduce the occurrences of traffic accidents are highly desirable from both safety and congestion management perspectives. One such measure, variable speed limit control [2], has been demonstrated to hold great potential in stabilizing traffic flow and reducing the number of accidents on the road [3] .

The FCC has recently allocated the 5.85-5.95GHz portion of the spectrum for inter-vehicle communication and DSRC (Dedicated Short Range Communication). This has prompted a recent surge of new design applications, which include driver-vehicle safety applications, infotainment, and mobile in-vehicle Internet services for passengers [4]. In addition to these applications, the newly allocated spectrum also provides an opportunity to leverage vehicular ad hoc networks (VANET) for intelligent distributed vehicular traffic flow control.

In this paper, we introduce a new framework called VGrid (Vehicular based Networking and Computing Grid), in which we harness DSRC-enabled vehicles to form a distributed, ad hoc computing grid. VGrid is used not only to monitor/sense real-time vehicular traffic and road conditions, but also to exchange information among vehicles to perform distributed computation to 'control' or 'smooth' traffic flow. This networking/computing capability can enable vehicle driver safety applications. For example, to reduce "stop-and-go" traffic and unnecessary deceleration or stopping, VGrid can help locate accidents and alleviate congestion through traffic metering and early warning messages.

In most driving conditions, a given driver can see one or at most two cars ahead and, as a result, a vehicle can only react to its immediate neighbors. If a vehicle stalls or crashes, vehicles far upstream may be forced to a stop or decelerate due to the "shockwave" effect. However, using VGrid, drivers have the ability to see farther down the road and react to incidents early. Instead of waiting to reach a crash site and then changing lanes, upstream vehicles can be alerted far ahead of time giving them much more time to react and switch out of the affected lane. Additionally, VGrid can be used to dynamically implement Variable Speed Limits, which can result in smoother traffic flow when accurately calculated, by leveraging inter-vehicle communication to exchange and propagate individual vehicle speeds to upstream vehicles.

The contributions of this paper are as follows:

- First we propose and introduce the VGrid, a novel new framework which uses vehicle to vehicle communication and distributed computation. VGrid can then be used as a platform for performing a variety of useful tasks such as traffic control, user infotainment applications and internet services.
- Second we evaluate two distributed traffic smoothing application, VSL and Accident Alert, which leverage the VGrid inter-vehicular communication and distributed computing capabilities. This evaluation is done using the VGSim simulation platform specifically designed to simulate the VANET/VGrid environment [5].

The remainder of this paper is structured as follows: First we discuss related work that are pertinent to this study, followed by and introduction to the VGrid framework and the benefits it has over fixed-infrastructure sensing and control systems such as inductive loops and road signs. Second, we introduce two methods by which VGrid can improve driver safety and smoothen traffic flow: accident alert, an early warning system which gives drivers significantly greater time to react to roadway obstructions, and variable speed limit (VSL), a dynamic speed limit calculated based on traffic characteristics. Third, we discuss the mobility and network models as well as the model parameters utilized in the simulations. Finally, we present the simulation results for both free-flow traffic (no obstructions) as well as a scenario in which one lane is obstructed. The results show that VGrid reduces speed variance in all scenarios. We also show that performance increases with the percentage of equipped vehicles.

II. Related Works

Many kinds of road-side infrastructures (e.g., fixed sensors) for monitoring highway conditions and road-to-vehicle communications have been in place. They are used to support various traffic management tasks. For example, controlling traffic flow through variable speed limits (VSL) has seen increased usage in Europe in recent years. In VSL control, electronic message boards are installed on freeways to advise or enforce certain dynamic speed limits. Such VSL systems aim at reducing speed differences within the traffic stream and are reported to lead significant reductions in accidents (up to 30% in a German experiment) and severity of shock waves [2]. Research has also been done to develop more effective variable speed limit control algorithms [6], [7], [8], [9], [10], [11], [12], [3], [2]. These control systems are often in the form of a centralized system with roadside sensors providing the real-time measurements to evaluate system performance. While such control systems have demonstrated its potential in managing traffic congestion, they require considerable initial investment in fixed infrastructure and constant maintenance after installation, which has in some ways limited the wide application of such systems. Adding wireless networking capability to vehicles further expands the telematic services in the automobile industry, and offers new ways to collect data and manage traffic. Existing research efforts have studied two approaches to provide this communication capability: with and without infrastructure. For instance, COMCAR [13], DRiVE [14], OverDRiVE [15], and MCP [16] investigate vehicular communications using cellular networks. A second approach aims to provide seamless integration of safety and entertainment units in the in-vehicle network [17]. Inter-vehicle communication uses wireless technology (in particular Bluetooth) for high mobility ad hoc networking.

FleetNet is a research project which involves the cooperation from both industry (DaimlerChrysler, NEC, Bosch, Siemens, etc) and academia (University of Mannhein, University of Hannover, etc). It attempts to develop a wireless multi-hop ad hoc network for inter-vehicle communication to improve the driver's and passenger's safety and comfort. Location awareness and position data play a crucial role for FleetNet applications and the communication protocols deployed. NEC Network Labs Europe and University of Mannheim design and evaluate position-based routing and forwarding strategies [18], [19] for vehicular ad hoc networks within the framework of FleetNet. A follow-up project, NOW (Network on Wheels), was initiated in Germany in 2004 to further address key questions on communication protocol and data security, as well as standardization for vehicle-to-vehicle communication .

There are several other projects that focus on developing intelligent vehicles based on this wireless ad hoc communication capability, including Electronic Toll Collection service (ETC), Advanced Cruise-Assist Highway System

(ACAHS), and Vehicle Information and Communication System (VICS) [20], and Automated Highway Systems (AHS)[21]. ACAHS aims at reducing traffic accidents, enhancing safety, improving transportation efficiency, and reducing the operational involvement of drivers. On the other hand, VICS allows drivers to obtain road and traffic information in real time.

The AutoNet project [22] focuses on developing a distributed and self-organizing transportation management and control system. The autonomous information network is composed of vehicles containing GPS equipped computers that communicate with each other information about traffic conditions. The AHS project examines how vehicle automation technology can be used to relieve traffic congestion. The premise is that vehicles operating in tight coordination (platooning) in an automated highway system can provide a significant increase in highway throughput (vehicles per lane per hour moving along the highway). The high-performance vehicle control system also increases the safety of highway travel, reduces driving stress and tedium, and provides a very smooth ride. Another related project, entitled Zero-Infrastructure [23], introduces and studies a fully decentralized traffic information system that is based only on data exchanged by equipped vehicles and does not require any infrastructure support. Vehicles exchange traffic information as they move through the network, which allows drivers to adjust their routes and avoid or be prepared for congestion, incidents or other hazards.

However, none of the projects mentioned above have outlined a unified framework to formulate and study the data fusion, traffic-state estimation, and dynamic traffic control problems. To evolve away from a fully-centralized approach, our work explores a system where distributed sensors and computers can complement the existing infrastructure to perform data collection and traffic flow management.

While there are several projects that use grid computing [24], [25] (or more generally, distributed computing) techniques in order to solve large scale problem (e.g., SETI@home [26] and Google Compute [27]), the most relevant to our work is the flash mob computing [28]. This work coordinates a large number of computers with wireless connections within a concentrated area to form an ad-hoc, distributed computer for solving complex problems. This work has two important characteristics that are relevant to VGrid: 1) the network is formed in an ad-hoc fashion, and 2) the problems are solved using distributed algorithms.

III. VGrid Framework, Architecture and Simulation

We envision an architecture in which the results of computations of various nodes are shared with neighbors, thereby influencing those neighbors' computations. Applications that utilize VGrid can operate in a number of different scopes: ranging from a single car to a platoon of cars that form a peer space, as discussed in [29]. However, for some applications, we may need a larger scope that combines a collection of peer spaces to achieve wider-scale results. Finally, we may think in terms of wide-area networking, in cases where data is sent back to a central server, or Internet services are utilized.

VANET, unlike mobile ad hoc networks (MANET), does not suffer the same constraints that are seen in typical MANET devices such as limited battery and computing power. On the other hand, the highly mobile individual nodes that make up VANET create a system with much more dynamic topology and channel conditions. One important aspect of VANET is the lack of need for fixed roadside infrastructure.

In VGrid, we describe a fully-distributed traffic monitoring and control system in which individual vehicles share their mobility and sensor information with their neighbors in order to collectively alter their traffic flow patterns in real-time. Unlike traditional grid computers, in VGrid, both the topology and the node membership change with time. When the vehicles forming the ad hoc grid computer are themselves part of the vehicular traffic flow that is being controlled, the system has a characteristic of being a self-referential system. Such a system has the following unique feature: the capacity of the grid computer increases as congestion increases, which is precisely when more capacity is needed (up to a certain point) to accurately determine the vehicular flow characteristics that can smooth and homogenize the traffic flow and thereby optimize the associated delay and throughput.

As shown in Fig. 2, VGrid leverages in-vehicle sensors and wireless communications to exchange position and velocity information. Real time vehicular flow statistics from multiple sources need to be ”fused” together (sometimes with historical profile) to infer traffic conditions for traffic management decisions. We will explore a new approach to this data fusion problem: A distributed solution where in-vehicle control units exchange information collected through in-vehicle sensors with other vehicles to control the current traffic state, this flow is diagramed in Fig. 1.

The collected data from distributed sensors will be combined to detect congestion or problems and compute desired driving speed. In this paper we will explore the application of VGrid to dynamic speed control with increasing complexity: a) accident warning and b) variable speed limits (VSLs). The first type of speed control provides early warnings of drastic changes in travel speed ahead, so that drivers can respond to these changes in time to avoid collisions. VSLs assign dynamic speed limits to road sections to smooth traffic flow. This is made possible through the sensing, communication and computation capabilities of VGrid.

Applications that utilize VGrid can operate in a number of different scopes: ranging from a single car to a platoon of cars that form a peer space, as discussed in [29]. The platoon scope is important since many potential VGrid applications will require knowledge of data contained in a local area of interest. However, for some applications, we may need a larger scope that combines a collection of peer spaces to achieve wider-scale results. Compared with fixed sensor network or computing grids, VGrid offers several advantages. Because of its mobility, VGrid can be deployed wherever and whenever it is needed in a road network. Because of its ad hoc nature, information can propagate along many directions. Because of its finite information propagation speed, information is disseminated in a staggered fashion, avoiding over-flooding or the so called herding effect in broadcast based information systems.

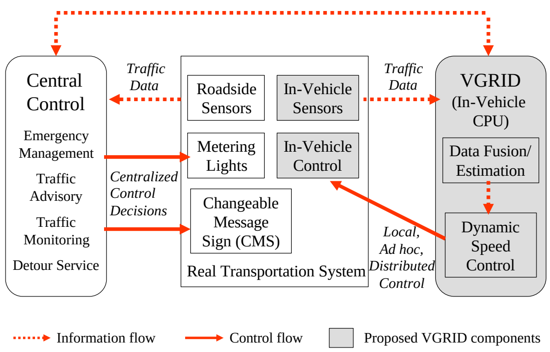

Fig. 1. Introducing local, distributed traffic control loops with the aid of VGrid.

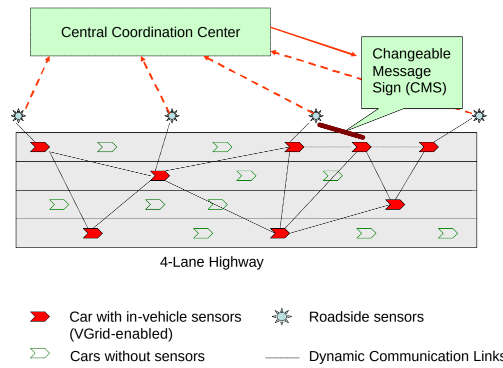

Fig. 2. Existing ITS uses fixed road side sensors that communicate with the central coordination center. VGrid leverages DSRC-enabled vehicles as not only mobile sensors and data relay nodes, but ad hoc computation units to make distributed traffic flow control decisions.

VGrid can analyze problems on the fly using its computing resources.

In order to study these applications we use VGSim [5] a simulation platform specifically designed to simulate vehicular adhoc networks. This tool uses a finer resolution Nagel-Schreckenburg (NS) mobility model [30] for vehicle mobility simulation and Jist/Swans [31] for network simulation.

IV. Application of VGrid to Vehicular Flow Homogenization

The goal of VGrid is to smoothen traffic flow through the use of a distributed computation grid network and the dissemination of traffic data through VANET. In order to quantify this, we define a set of metrics we will use to describe traffic flow. The primary metric is the speed variance normalized to the average speed, which indicates the amount of acceleration and deceleration vehicles experience. Without VGrid, drivers lack precise speed and position information about the vehicles around them, which may lead to overreaction to changes in speed, causing accidents or unneeded slowdowns. If variance is low, traffic flows at a more constant rate and this scenario occurs less often. This homogenization also affects the throughput, the number of vehicles exiting a section of roadway in a fixed time interval, and the latency, the amount of time it takes vehicles to exit a section of roadway, are indicators of the overall performance of the system. Therefore, the goal is to minimize variance and latency and to maximize throughput.

A. Accident Alert

One basic application of on-board wireless communications is to alert upstream vehicles of the presence of an obstruction in the road ahead, allowing them to change out of impacted lanes earlier and preventing them from changing into those lanes. This may include accidents, construction zones, and debris. Emergency or construction vehicles as well as nearby automobiles capable of identifying such incidents will broadcast their GPS location to other cars in the area, which will relay this information upstream.

In addition to the message being sent at the location of the accident, we also take into account the queuing of vehicles behind an accident. We define a vehicle as enqueued if it is below a certain speed threshold and a within a certain distance from either an accident or another enqueued vehicle. That is:

1) The speed of the vehicle is less than v_{min}.
2) There is a vehicle in front that is either stalled or in a queue.
3) This stalled or enqueued vehicle is less than d_{min} cells away

When such a queue is formed, the vehicle at the end of the queue may be considered to be the location of the accident, allowing drivers to avoid it. The actual stall vehicle still generates the initial message but enqueued vehicles modify the accident message before forwarding it on so that vehicles upstream know the accident location of the queue tail.

We also introduce a lane changing behavior into the model. At every time step, each vehicle has a chance to change into another lane with probability p_{change}, where, p_{base} is a fixed lane changing probability, $max\_distance$ is the maximum distance considered for accident alert, and $stall\_distance$ is the distance to the accident. This lane change probability is only used when a vehicle detects that it is in a lane with an accident present. If no accident is present p_{base} is the lane change probability.

$$P_{change} = min(p_{base} + (1 - p_{base})\frac{max\_distance - stall\_distance}{max\_distance}, 1)$$

In order to propagate vehicle location information vehicles use a beaconing message which will be described later in the paper, but because accidents are time sensitive events it was prudent to designate specific accident alert messages(ALMs) instead of piggy backing accident information onto current beacon messages. Much like the beaconing messages ALMs use UDP broadcast in order to propagate information, a description of the information in these messages is given in Table I.

Unlike beaconing messages ALMs are forwarded by vehicles ”upstream” of the accident. The accident vehicle or accident point broadcasts an ALM every 4 ($0.25Hz$) seconds. In order to propagate information upstream in an efficient manner we use a time delay suppression mechanism. The goal of this mechanism is to have the vehicle that is farthest upstream forward the message. When a vehicle receives an ALM a delay is inserted before the message is re-broadcasted. The delay is calculated as follows:

TABLE I
NETWORK PARAMETERS

Accident Parameter	Description
POSITION	Position of accident
LANE	Accident lane
TIMESTAMP	Time at which the message was first created, used to uniquely identify the message
SOURCE_TX_power	Transmission range of the last vehicle to forward this ALM
SOURCE_Postion	Position of the last vehicle to forward this ALM

$$time\_delay = \frac{T_{delay} * (tx_{range}^2 - D_{Src\_Dst}^2)}{tx_{range}^2}$$

Where $time\_delay$ is the time vehicles wait before retransmitting the ALM. T_{delay} is a constant set to 500 milliseconds, tx_{range} is the transmission range of last vehicle to transmit this ALM and D_{Src_Dst} is the distance between the source of the ALM and the receiving vehicle. Vehicles closer to boundary or edge of the source vehicles transmission range will quickly retransmit the ALM. Vehicles downstream from the retransmission will then overhear the retransmission and as a result will cancel or suppress forwarding the ALM. This means that when a source vehicle transmits an ALM the furthest vehicle from that source will be the first to retransmit and all the vehicle in between the source and the retransmitting vehicle will suppress their retransmissions. This was done in order to avoid a unnecessary flooding in the propagation of the ALM.

Although the focus of this study is not the propagation method of the alert messages we choose to evaluate the method in order to ensure the message is successfully propagated to all upstream vehicles. During simulation we found that a simple probabilistic broadcast yielded scenarios in which vehicles reached a critical area without first receiving the alert message. This was due to the fact that as the number of VGrid vehicles increased so did the number of rebroadcasted messages. At very high penetration rates and high traffic densities this resulted in over congestion of the wireless medium and large numbers of packet loss, at which point a significant number of VGrid beacon messages were lost and thus resulted in degraded performance of the distributed VSL application. Conversely the number of packets generated using the suppression technique did not increase significantly with penetration rate and density. Through simulation we found that this suppression method succeeded in reaching all VGrid vehicles 200 meters prior to those vehicles reaching the incident location. Additionally it had a minimal impact on network traffic in terms of packet generation and collision. We found that there was only a 3%-7% increase in packet loss with the suppression technique where as with simple probabilistic flooding packet loss increased by up to 60%.

We show that with suppression forwarding we achieve a level of performance that successfully propagates the message to all vehicles and also regulates the amount of network traffic generated. We argue that although this method may not be optimal, it is functional, and can be replaced with better methods which would only lead to improved performance of the applications.

B. Computing Optimal Variable Speed Limits

The fundamental purpose of VSL is to reduce the variance in the speed of vehicles in order to limit the sudden acceleration and deceleration that leads to accidents. We leverage the VGrid framework and construct a fully ad-hoc and distributed VSL application to control vehicle speeds and smooth traffic flow. In order to do this each VGrid vehicle periodically decimates, through the use of a udp broadcast message, vehicle specific traffic information which includes the current speed, position, lane and forward spacing information of that vehicle. These messages are single-hop broadcast messages, meaning they are not forwarded by subsequent receiving vehicles. This information is then collected and aggregated by all VGrid vehicle within range of the broadcast. The information is then used to, in a distributed fashion, coordinate and set a local VSL for each vehicle. We show in section V that this has the effect of homogenizing traffic flow.

This distributed VSL application is implemented as follows. We break the application into two separate phases. In phase one each vehicle disseminates the VGrid beacon messages. As previously mentioned all vehicles within range collect this information and aggregate them. The forward spacing information in the beacon message is used then to estimate the local traffic density. In order to achieve a VSL speed we first build a function which maps the average traffic speed given a traffic density. A constant delta value is then added to this function to create the VSL function. Equation 1 illustrates the VSL function where $density$ refers to the traffic density and $Map(density)$ returns the average speed of a given traffic flow. A δ value of 4 is used for these simulations and aximum speed is still capped at 30 meters/sec.

$$VSL(density) = Min(30, Map(density) + \delta) \tag{1}$$

In the second phase of this distributed VSL application the vehicles broadcast the estimated VSL to surrounding vehicles. This process again uses a single hop UDP message which is periodically re-broadcasted everyone second. Vehicles continually collect VGrid beacon messages and update this value based on the collected surrounding traffic information. The idea behind the second phase broadcast is to allow the vehicles within a region to converge on

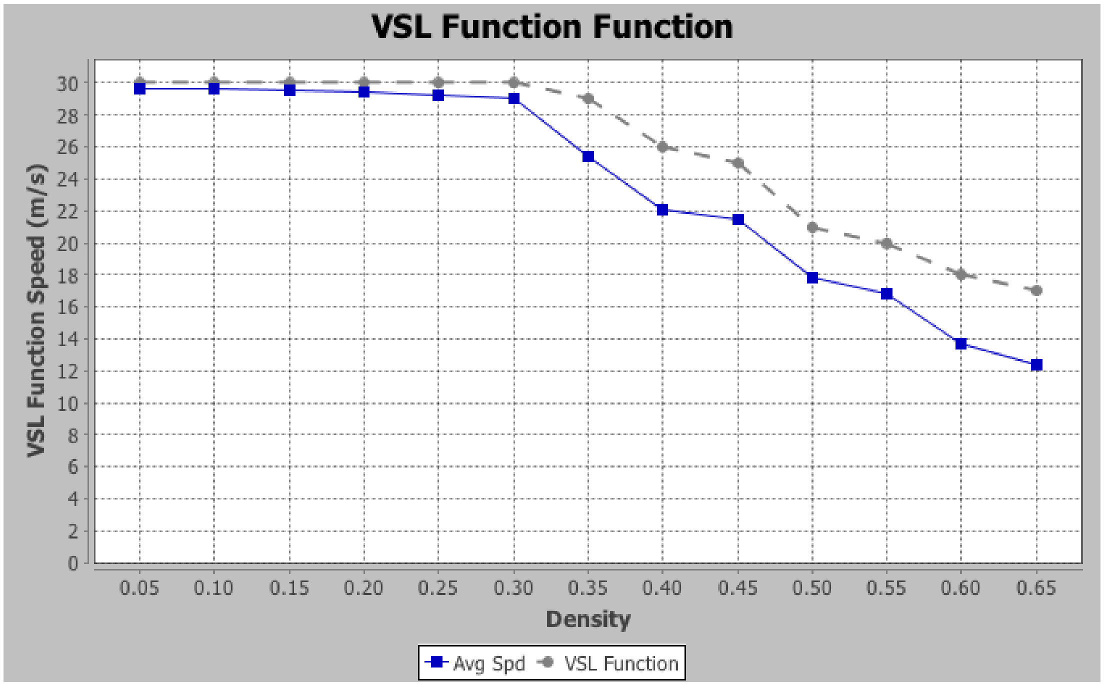

Fig. 3. The function mapping traffic density to average speed and the corresponding VSL function with $\delta = 4$.

like VSL values. Vehicle that collect this broadcasted VSL speed limit then aggregate them and obtain an average which is then used as the final VSL speed for that particular vehicle. This allows for vehicles to apply a VSL that is an aggregate of surrounding vehicles. An additional benefit is that as congestion downstream increases the calculated VSL speed will dynamically adjust to the increasing traffic density. This is due to that fact that a vehicle just entering a congested area will calculate heavy traffic density ahead and light traffic behind. This will result in a middle ground traffic density calculation and as a result a moderate VSL. In addition, in the second phase of the algorithm we should see much slower VSL speeds from forward vehicles and slow VSL speed from back vehicles. This will result in the vehicle applying a VSL that combines the forward VSLs and back VSLs allowing for a gradual reduction of VSL from one area to the next. This, in fact, is precisely what is need in the implementation of VSL.

In the 4-lane highway scenario, we look at two accident configurations: no accidents; 1 fixed accident in lane 1 in cell 12000. The accident is fixed for the duration of the simulation. We use a 1.5 km circular road in order to maintain the specified traffic density. In the scenario with no accidents, we look at the effect of VSL alone. In the one accident scenario, we look at the effect of both VSL and Accident Alert. For each case we first evaluate the traffic variance but also evaluate average traffic speed for the entire road. Although speed is not our primary metric we want to also take it into consideration as the goal of traffic smoothing is not only to reduce accident but also increase traffic throughput. We calculate variance as follows: At each 1 second interval we find the average speed of all vehicles on the road. We then calculate the variance using each vehicle speed from that average. These values are accumulated over the life of the simulation and then averaged to obtained a final calculated variance. The simulation duration is 30 minutes so a total of 1800 variance calculations are accumulated and averaged for the final calculation. Finally we examine the effect of penetration rate of enabled vehicles. The non-participating vehicles will not be VGrid enabled and thus will not have access to VSL or accident alert applications. We refer to the participation rate as the penetration rate of VGrid vehicles in the system. A 0% penetration rate means that no VGrid vehicles are present.

V. Results

We first evaluate the VSL only scenario in which there are no accidents. Figures 4 and 5 present the results of the distributed VSL algorithm. We see that with 100% penetration we have an average variance reduction of around %22 percent. We concider this figure by calculating variance decrease when traffic density begins to cause congestion found around 30% density. In the best case we see a 37% decrease in variance. Figure 5 shows that there is minimal decrease in speed. In some cases VSL improves on the average speed of the system. Additionally it is important to note that our simulation do not show the long term effect of reduction in speed variance in traffic flow. Long term reduction in variance should translate to accident reduction and lower long term average travel time.

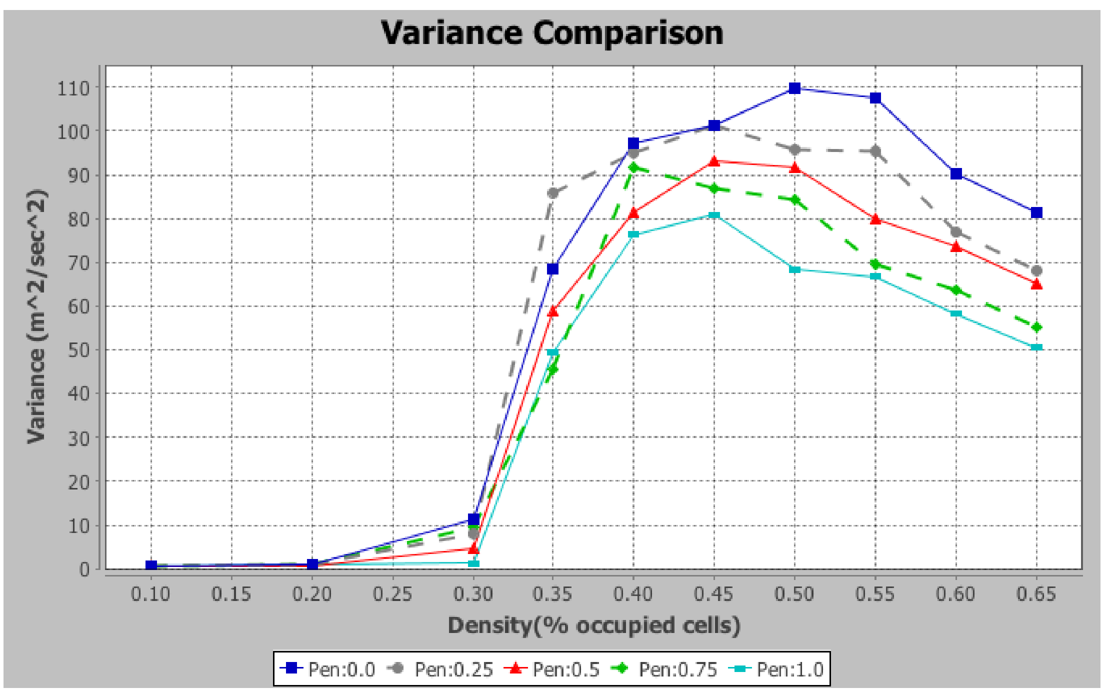

Fig. 4. Variance performance of a fully distributed VSL application. This figure shows the effect of both traffic density and penetration rate on the performance of VSL.

We also found that this distributed VSL application is effective even with less than 100% penetration rates. As expected the improvement is linearly correlated with the penetration rate itself. However, even with a 25%

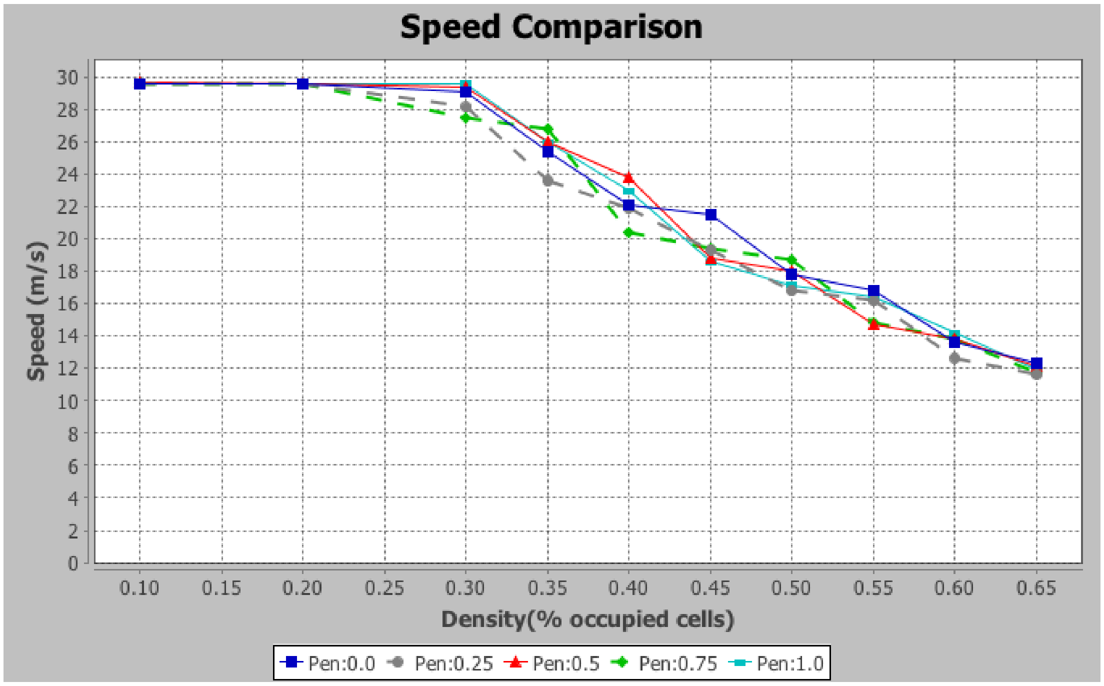

Fig. 5. Average speed of traffic with the use of VSL in the no accident scenario.

penetration rate there is still improvement in the traffic variance.

Next we evaluate the accident scenario in which a road blockage exist in the final third of the road. In this case the obstruction itself broadcast a warning message which is propagated using multi-hop udp messages to all vehicle upstream from the incident point. These are then used to allow vehicles to merge prior to the accident or incident point. Figures 6 7 reflect the results of the VSL + Accident Alert application working together to homogenize the traffic flow in this section of road. We can see that the combination of these two application greatly reduces the variance in traffic speed. As shown in figure 6 we found that a variance decrease of up to 70% can be achieved when these distributed applications are applied. On average there is a 47% decrease in variance in traffic densities of over 30%.

VI. Conclusion and Future Work

There are countless applications for inter-vehicle wireless communications, and as vehicular computational and networking capabilities increase, what once required significant infrastructure, can be achieved in an ad hoc, distributed manner. In this paper, we outlined such a distributed framework for a vehicular computing grid that allows for the monitoring and control of traffic flow. Using these applications, we look at the homogenization or smoothing of traffic in order to reduce accidents by lowering the variance of vehicle speeds on a highway. We found that VGrid does significantly reduce traffic jitter while having a minimal impact on simulated travel velocity. We argue that even with the slight increase in travel time, in the long run, VSL and accident alert VGrid applications will reduce traffic accidents and thus reducing travel time. Finally we also showed that even a partial deployment

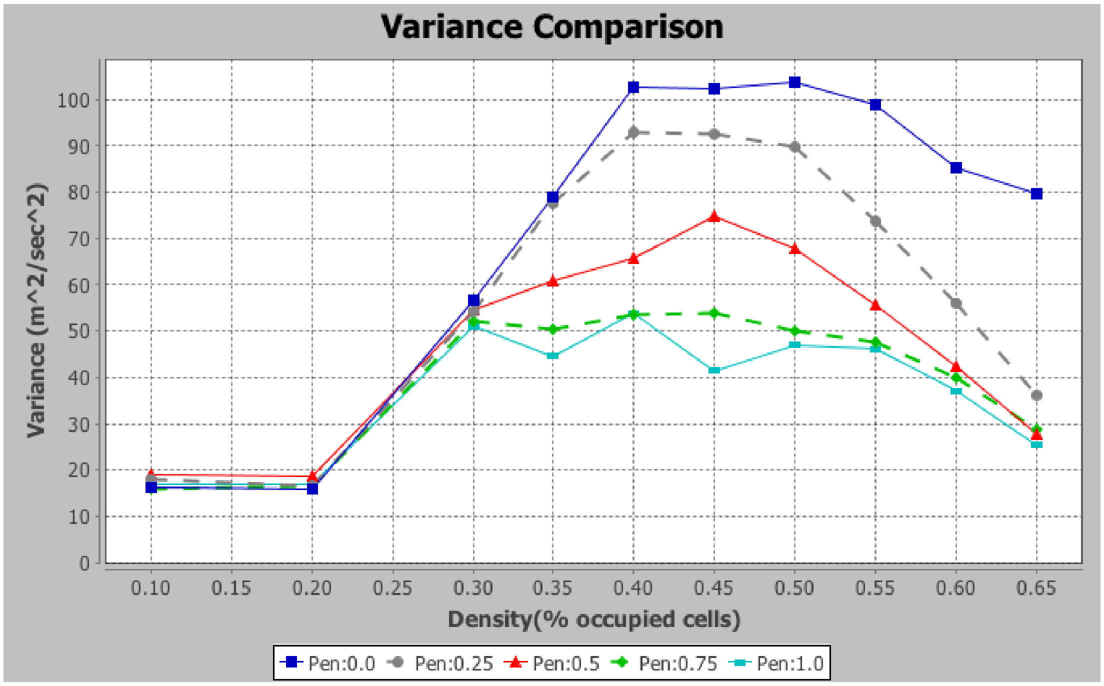

Fig. 6. Variance performance of a fully distributed VSL and Accident Alert application. This figure shows the effect of both traffic density and penetration rate on the performance of VSL.

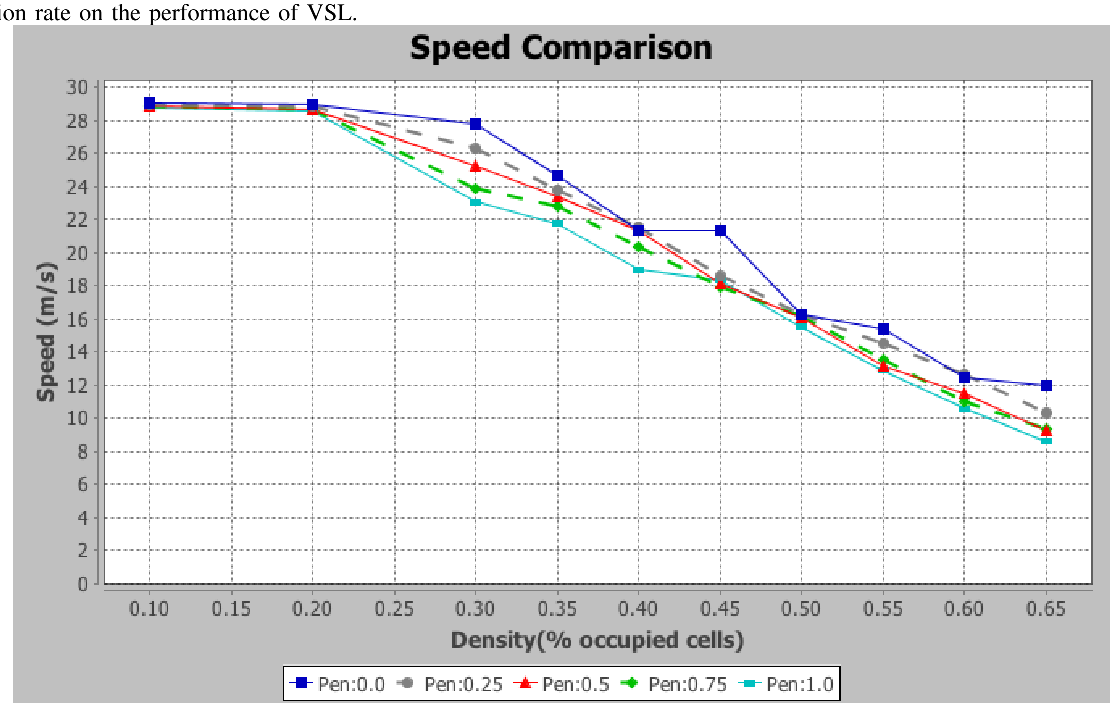

Fig. 7. Average speed of traffic with the use of VSL and Accident Alert applications in the one accident scenario.

of the VGrid infrastructure can improve traffic smoothing.

In the future we plan to leverage the VGrid framework to an even greater degree. Such work includes event driven VSL algorithms such that forward vehicle explicitly inform back vehicles to increase or reduce speeds. The N-S mobility model also includes random slowdown and speed up probabilities. With the use of VGrid drivers can explicitly be told to maintain current speeds in addition to decreasing or increasing speeds. Along the same lines of more explicit traffic control accident alert applications can tell drivers to change lanes at exact points or tell drivers to slow down or speed up in order to facilitate lane changing into non-accident lanes. These decisions, among others, can be made in a distributed fashion through the use of VGrid computing.

REFERENCES

[1] Lomax and Shrank, "Mobile internet access in fleetnet. http://www.fleetnet.de/."

[2] T. R. Board, "Examples of variable speed limits applications," tech. rep., Speed Management Workshop Presentation, Washington D. C, Jan 2000.

[3] A.-A. Mohamed, D. Jeremy, and D. Albinder, "Evaluation of variable speed limits for real-time freeway safety improvement," *Accident analysis and prevention*, vol. 38, pp. 335–345, 2006.

[4] "Drive-thru internet: Ieee 802.11b for automobile users. http://www.drive-thru-internet.org/index.html."

[5] B. Liu, B. Khorashadi, D. G. H. Du, C.-N. Chuah, and M. Zhang, "Vgsim: An integrated networking and microscopic vehicular mobility simulation platform," *IEEE Communication Magazine Automotive Networking Series*, vol. 47, pp. 134–141, May 2009.

[6] M. Zarean, P. Pisano, K. Dirnberger, and M. Robinson, "Variable speed limit systems: The-state- of-the-practice," *Proceedings of the 1999 Rural Advanced Technology and Transportation Systems Conference*, 1999.

[7] D. C. Nelson, "Applications of variable speed limits and speed management in the united states and internationally." University of Maryland.

[8] A. T. R. Center, "The fuzzy variable speed limit device project," *ATRC Research Notes*, August 1998.

[9] P. Mackey and G. Lemay, *Manual for setting speed limits on municipal road networks*, 2002.

[10] J. K.Wilkie, "Using variable speed limit signs to mitigate speed differentials upstream of reduced flow locations," *Southwest Region University Transportation Center - Compendium: Graduate Student Papers on Advanced Surface Transportation Systems*, 1997.

[11] L. R. Dumke, "The signs they are a changing - a modular approach to real-time information," *Traffic Technology International*, June/July 1999.

[12] S. Smulders, "Control of freeway traffic flow by variable speed signs," *Transportation Research Part B*, vol. 24, pp. 111–132, 1990.

[13] "Comcar (communication and mobility by cellular advanced radio)." http://www.comcar.de/.

[14] "Drive (dynamic radio for ip-services in vehicular environments)." http://www.istdrive.org/index2.html.

[15] "Overdrive (spectrum efficient uni- and multicast services over dynamic-radio networks in vehicular environments)." http://www.comnets.rwth-aachen.de/ o drive/index.html.

[16] "Multimedia car platform (mcp)." http://www.aramis-research.ch/d/7229.htm.

[17] "Bluetooth ad-hoc networking for inter-vehicle communication." http://www.comms.scitech.susx.ac.uk/research/bluetooth.php.

[18] H. Fussler, JorgWidmer, M. Kasemann, M. Mauve, and H. Hartenstein, "Contention-based forwarding for mobile ad-hoc networks," *Elsevier's Ad Hoc Networks*, pp. 351–369, Nov 2003.

[19] C. Lochert, H. Hartenstein, J. Tian, and etc., "A routing strategy for vehicular ad-hoc networks in city environments," *58th IEEE Semiannual Vehicular Technology Conference (VTC-Fall)*, pp. 156–161, October 2003.

[20] R. B. ITS, "Japanese ministry of land, infrastructure and transport." http://www.its.go.jp/ITS/index.html.

[21] U. Berkeley, "Partners for advanced transit and highways (path)." http://www-path.eecs.berkeley.edu/.

[22] "Autonet: Ad hoc peer to peer information technology for traffic networks." http://www.its.uci.edu/ mcnally/mgm-autonet.html.

[23] "A zero infrastructure traffic information system." http://trans.civil.northwestern.edu/ thanasis/ research/projects/zi.html.

[24] C. K. I. Foster, "Computational grids," *The Grid: Blueprint for a new Computing Infrastructure*, 1999.

[25] S. T. I. Foster, C. Kesselman, "The anatomy of the grid," *MILCOM*, 2001.

[26] D. P. Anderson, E. K. J. Cobb, M. Lebofsky, and D. Werthimer, "Seti@home: An experiment in public-resource computing," *Communications of the ACM*, vol. 45, pp. 56–61, November 2002.

[27] "Google compute." http://toolbar.google.com/dc/faq dc.html, Accessed Feb. 10 2006.

[28] "Flash mob computing." http://www.flashmobcomputing.org/, Feb 15 2006.

[29] I. Chisalita and N. Shahmehri, “A peer-to-peer approach to vehicular communication for the support of traffic safety applications,” *ITSC. IEEE*, 2002.

[30] H. Du, M. Zhang, C.-N. Chuah, and D. Ghosal, “A finer resolution cellular automata model for inter vehicle communication applications,” *Transportation Research Board 2008 Annual Meeting*, vol. 13, Jan. 2008.

[31] “Jist / swans. http://jist.ece.cornell.edu/.”