Vehicle Mobility, Communication Channel Modeling and Traffic Density Estimation in VANETs

by

Nabeel Akhtar

A Thesis Submitted to the
Graduate School of Sciences and Engineering
in Partial Fulfillment of the Requirements for
the Degree of

Master of Science

in

Computer Science and Engineering

Koç University

July 15, 2013

Koç University

Graduate School of Sciences and Engineering

This is to certify that I have examined this copy of a master's thesis by

Nabeel Akhtar

and have found that it is complete and satisfactory in all respects,

and that any and all revisions required by the final

examining committee have been made.

Committee Members:

Assoc. Prof. Oznur Ozkasap (Advisor)

Assist. Prof. Sinem Coleri Ergen (Advisor)

Prof. Ceyda Oguz

Prof. Attila Gursoy

Assoc. Prof. Serdar Tasiran

Date:

To the people of Pakistan

ABSTRACT

Vehicular Ad-Hoc Network (VANET) is a promising Intelligent Transportation System (ITS) technology that aims to improve road traffic conditions and safety of passengers.

First part of our work deals with providing a realistic analysis of the VANET topology characteristics over time and space using various key metrics of interest. In this analysis, we integrate real-world road topology and real-time data extracted from the Freeway Performance Measurement System (PeMS) database into a microscopic mobility model to generate realistic traffic flows along the highway. Moreover, we use a more realistic, recently proposed, obstacle-based channel model and compare the performance of this sophisticated model to the most commonly used more simplistic channel models including the unit disc and log-normal shadowing models. Our investigation on the key metrics reveals that both log normal and unit disc models fail to provide realistic VANET topology characteristics. We therefore propose a matching mechanism to tune the parameters of the lognormal model according to the vehicle density and a correlation model to take into account the evolution of the link characteristics over time. The proposed method has been demonstrated to provide a good match with more sophisticated but computationally expensive and difficult to implement obstacle based model and validated over the real data of two different highways in California.

Second part of our work deals with distributed algorithms for density estimation in VANETs. Vehicle density is an important system metric used in monitoring road traffic conditions. Most of the existing methods for vehicular density estimation either use infrastructure, or use local neighbor information to estimate global vehicle density. These techniques however suffer from low reliability and limited coverage

as well as high deployment and maintenance cost. We adapted and implemented three fully distributed algorithms for density estimation, inspired by the mechanisms proposed for system size estimation in peer-to-peer networks. Results show that system size estimation technique can be used for density estimation in VANETs. Moreover, we proposed a completely distributed algorithm *CluSampling* which has been specifically tailored for VANETs. The extensive simulations of these algorithms at different vehicle traffic densities and area sizes for both highways and urban areas reveal that *CluSampling* is robust to changes in the network and it provides high accuracy in least convergence time and introduces less overhead on the network and the initiator node.

ÖZETÇE

Geçici Araç Ağları (GAA), yol trafik durumunu iyileştirme ve yolcuların güvenliğini sağlama özellikleriyle ümit veren bir Akıllı Taşıma Sistemleri teknolojisidir.

Bu tez çalışmasının ilk bölümü, çeşitli temel başarım ölçütleri kullanarak GAA'ların zaman ve konum tabanlı topoloji özelliklerinin gerçekçi analizinin sağlanmasıyla ilgilidir. Bu analizde, gerçek yol topolojileri ve PeMS veritabanından alınan gerçek zamanlı veriler, otoyollarda gerçekçi trafik akışları üretebilmek için mikroskobik hareketlilik modeliyle birleştirilmektedir. Ayrıca, daha gerçekçi, yakın zamanda önerilmiş engel-tabanlı kanal modeli kullanılmış ve bu karmaşık sistemin başarımı en çok kullanılan sabit-disk ve log-normal gibi daha basit kanal modelleriyle karşılaştırılmıştır. Temel başarım ölçütleri üzerindeki araştırmamız sabit-disk ve log-normal kanal modellerinin gerçekçi GAA topoloji özellikleri sağlamada yetersiz olduğunu açığa çıkarmıştır. Bu nedenle, link özelliklerinin zamana bağlı değişimlerini hesaba katmak amacıyla, araç yoğunluğu ve bir korelasyon modeline göre log-normal modelin parametrelerini adapte eden bir eşleştirme mekanizması önermekteyiz. önerilen yöntemin karmaşık, hesaplama açısından pahalı ve gerçekleştirme açısından zor olan engel-tabanlı kanal modeliyle iyi bir eşleme sağladığı gösterilmiş ve modelin işlevselliği Kaliforniya'da bulunan iki farklı otoyoldan alınan gerçek verilerle doğrulanmıştır.

Çalışmamızın ikinci bölümü, GAA'larda yoğunluk hesaplaması için dağıtık algoritmalarla ilgilidir. Araç yoğunluğu yol trafik durumunun gözlemlenmesinde kullanılan önemli bir sistem ölçütüdür. Araç yoğunluk hesaplaması için önerilmiş algoritmaların çoğu ya belli bir altyapıya dayanmakta ya da sisteme genel araç yoğunluğunu hesaplamak için yerel komşu bilgisini kullanmaktadır. Ancak, bu algoritmalar yüksek yerleştirme ve bakım masraflarının yanı sıra düşük güvenilirlik ve kısıtlı kapsamadan dolayı dezavantajlıdırlar. Çalışmamızda, görevdeş ağlarda sistem boyutunu

hesaplamak için önerilmiş mekanizmalardan esinlenerek üç farklı tamamen dağıtık algoritma tasarladık. Sonuçlarımız sistem büyüklüğü hesaplama tekniği GAA'larda trafik yoğunluğunu hesaplamak için de kullanılabileceğini gösterdi. Buna ek olarak, GAA'lar için özel olarak tasarlanmış tamamen dağıtık CluSampling algoritmasını önerdik. Hem otoyollarda hem de şehir-içi bölgelerde, farklı bölge genişlikleri ve trafik yoğunluklarında yapılan kapsamlı benzetimler CluSampling algoritmasının ağdaki değişikliklere dayanıklı, yüksek kesinlikte ve en az zaman gerektiren çözüm sunan bir algoritma olduğunu ve bunları sağlarken ağ ve öncü araç üzerinde daha az yük oluşturduğunu göstermiştir.

ACKNOWLEDGMENTS

This research project would not have been possible without the support of many people. The author wishes to express his gratitude to his supervisors, Dr. Sinem Ergen and Dr. Oznur Ozkasap who were abundantly helpful and offered invaluable assistance, support and guidance. Special thanks also to all his graduate friends. The author wishes to express his love and gratitude to his Turkish friends for their support, understanding & assistance, throughout the duration of his studies.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

VANET is a promising Intelligent Transportation System (ITS) technology that offers many applications such as safety message dissemination [1, 2, 3], dynamic route planning [4], content distribution, gaming and entertainment [5]. The majority of the VANET research effort on protocol design has relied on simulations due to the prohibitive cost of deploying real world test-beds. Building a realistic simulation environment for VANET is therefore essential in judging the performance of the protocols proposed at various layers.

The first part of our work deals with analysis of vehicular mobility and communication channel modeling for VANET simulation. VANET simulation environment should be realistic requiring an accurate representation of the vehicular mobility and signal propagation among the vehicles, and also efficient necessitating a reasonable amount of simulation time. Realistic representation of the vehicle mobility requires using real-world road topology, accurate microscopic mobility modeling and real-data based traffic demand modeling whereas a realistic representation of the signal propagation among the vehicles requires reproducing the actual physical radio propagation process for a given environment. On the other hand, an efficient representation of the vehicle mobility and signal propagation model requires analyzing the closeness to the realistic representations in terms of both the key metrics summarizing the dynamics of the VANET topology in time and space, and the runtime of the simulations. As summarized in Table 2.1, the literature on VANET topology characteristics focuses on realistic channel models tested on simplistic vehicle mobility models [6, 7], realistic mobility models without considering realistic signal propagation models [8, 9, 10] or simplistic models for both vehicle mobility and communication channel [11, 12], [13, 14, 15, 16, 17]. Moreover, these models are often compared using classical metrics such as node degree, number of clusters, link duration and quality [11, 8, 12, 13] or related metrics such as packet loss probability and connectivity probability [14, 15, 6] [7, 17, 10]. They include only a small subset of important network-wide metrics summarizing the state of the network such as closeness centrality measuring how long it takes for the information to spread in a network, clustering coefficient giving information about the degree to which nodes tend to cluster and size of the largest group of connected vehicles in the network [12, 9].

The goal of this part of our work is to analyze VANET topology characteristics on a large highway section by integrating realistic microscopic mobility traces generated using real-world road topology and real-data based traffic demand with realistic channel models taking into account the effect of vehicles on the received signal power. We compare the performance of this realistic scenario to the most commonly used more simplistic channel models using various metrics of interest. The original contributions of this work are listed as follows:

- We incorporate real-world road topology and real-time data from PeMS database [18] into the microscopic mobility model provided by Simulation of Urban Mobility (SUMO) [19]. PeMS database allows modeling realistic traffic flows on the highway by adjusting the parameters of the SUMO simulator to match the real data. This is the first work to analyze VANET topology characteristics over a large scale highway using an open source database for measuring vehicle traffic and speed.

- We incorporate more realistic recently proposed obstacle-based channel model into the analysis of VANET topology characteristics and compare its performance to the most commonly used more simplistic channel models including unit disc and log-normal shadow fading models. This is the first work to analyze the effect of using the obstacle-based channel model on the VANET topology characteristics.

- Since it is hard to integrate the obstacle based model into the modern simulators due to its high complexity and computational cost, we propose a matching mechanism to tune the parameters of the lognormal model according to the vehicle density and a correlation model to take into account the evolution of the link characteristics over time. We validate the performance of the proposed method over two different highways in California. This is the first work to propose a method to adjust the parameters of the lognormal model and introduce time correlation depending on the vehicle density for more realistic and efficient VANET simulation.

- This is the first work to perform an extensive analysis of the VANET topology characteristics based on the realistic vehicle mobility and channel models. This analysis includes not only node degree, link duration, number of clusters but also neighbor distribution, closeness centrality, size of largest cluster and clustering coefficient.

The second part of our work deals with the analysis of distributed algorithms for density estimation in VANETs. Road traffic density estimation provides important information in VANETs and intelligent transportation systems for road traffic control, intelligent vehicular routing and efficient data dissemination. Various methods have been used in the literature to estimate vehicular density. Traditionally most of the methods rely on building an infrastructure, such as pressure pads, inductive loop detectors deployed under the road surface, roadside radar, infra-red counters, cameras, or even manual counts, to measure the *speed* and *flow* of the vehicles in the estimation of the vehicular density. However, these techniques suffer from high deployment cost, high rate of failures and vulnerable to single point of failure, resulting in high maintenance cost and limited coverage. In addition, most of the current traffic information system rely on a centralized communication model, where all the data is processed at one central location. Therefore this approach for data processing is not suitable for an emerging self-organizing traffic information system.

The other group of methods adopt distributed or infrastructure-free approaches for density estimation. These approaches varies from using clustering or group based approach [20, 21] to using local neighbor information to calculate local density which is then used to estimate global density [22, 20, 23, 24]. These methods are suitable for self-organizing traffic information system since they do not use centralized communication model.

The goal of this part of the work is to adapt fully distributed algorithms developed for system size estimation in peer-to-peer (P2P) networks, to the infrastructure-free vehicle density estimation in highly mobile VANET, and analyze their performance over a wide range of scenarios including both highways and urban areas at different traffic densities and area sizes. The main challenge of VANET is its highly dynamic and mobile behavior compared to P2P networks where vehicles enter and leave very quickly, and new connections are made and existing connections are broken very often. We use a completely different network size calculation technique to estimate the density of vehicles on the road. After adapting these algorithms from P2P network size estimation, we proposed a fully distributed infrastructure free density estimation algorithm *CluSampling* specially tailored for for VANETs. The main contributions of this work are summarized as follows:

- Three fully distributed algorithms for system size estimation, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation, have been adapted and implemented for density estimation in VANETs for the first time.

- We proposed a fully distributed and self organizing vehicle density estimation algorithm *CluSampling* which use network wide information using clustering and sampling technique to estimate the density of vehicles on the road.

- *CluSampling* make use of simple clustering mechanism that has limited load on the network and is robust to changes in the network.

- These algorithms are tested on eight different traffic scenarios for both highway and urban areas using a realistic data-set used for microscopic vehicle mobility and traffic generation. We have used different traffic densities and different road sizes for the validation of the algorithms.

- To test the validity of *CluSampling*, we compared it with four fully distributed algorithms previously proposed in the literature for density estimation. These algorithms include Sample & Collide, Hop Sampling, Gossip-based Aggregation and Local Density-based Algorithm.

- These algorithms are rigorously tested across different performance metrics like convergence time, overhead on network, percentage error and load on initiator.

The rest of the thesis is organized as follows. Chapter 2 explains the vehicular mobility and communication channel modeling. Chapter 3 explains the matching mechanism used to tune the parameters of Log normal model. Chapter 4 explains three fully distributed algorithms inspired from system size estimation in p2p networks. In Chapter 5 we propose a completely distributed and infrastructure free traffic density estimation algorithm for Vehicular Networks. Finally, concluding remarks and future work are given in Section 6.

Chapter 2

VEHICULAR MOBILITY AND COMMUNICATION CHANNEL MODELING

This chapter explains the realistic vehicular mobility and communication channel modeling. Realistic representation of the vehicle mobility requires using accurate microscopic mobility modeling, real-world road topology and real-data based traffic demand modeling. For modeling communication channel modeling, we first describe simplistic channel models including unit disc and log-normal shadowing models that are commonly used in the analysis of VANET topology characteristics. We then describe a recently proposed more realistic channel model called obstacle-based channel model that incorporates the effect of the moving obstacles (i.e. vehicles) on the received signal power due to their dominating influence.

2.1 Related Work

2.1.1 Vehicle Mobility Models

Vehicular mobility simulators have been growing their complexity and features over time encompassing realistic road topologies and microscopic vehicular models, where each vehicle is represented as a separate entity and the behavior of vehicles depends on the neighboring vehicles [25]. SUMO [19], VISSIM [26], DIVERT [27], MMTS [28] are examples of such simulators.

The simulations are usually performed on small portions of a road with user generated traffic flows in these simulators and only recently have been extended to larger areas while incorporating a realistic model for the macroscopic mobility of the vehicles where the traffic flows are determined based on the real data. The real data used for this purpose in the recent studies on the analysis of VANET topology characteristics includes mobility traces gathered through various measurement campaigns [8] and statistics performed by the urban planning and traffic engineering communities [9, 10]. None of these studies however analyze VANET topology characteristics on a large-scale highway considering real data based traffic demand of vehicles.

2.1.2 Communication Channel Models

Realistic representation of the signal propagation among vehicles requires reproducing the actual physical radio propagation process for a given environment based on the ray-tracing method [29, 6]. Ray-tracing approach generates the complex impulse response of the channel by determining possible paths or rays from the transmitter to the receiver according to the rules of geometrical optics. Such a model however is impractical since it requires a detailed description of the site-specific propagation environment.

Stochastic models on the other hand determine the physical parameters of the vehicular channel in a completely stochastic manner without presuming any underlying geometry [30]. The distance-dependent path loss, large scale and small scale fading distribution are the parameters to be estimated in these stochastic models as a result of extensive measurement campaigns. The path loss represents the local average received signal power relative to the transmit power as a function of the distance between the transmitter and receiver. The path loss exponent of $n = 1.8 - 2.7$ was observed on highways in [31, 32, 33, 34]. The large-scale fading models the effect of the surrounding obstacles on the mean signal attenuation at a given distance. The surrounding obstacles may be mobile (e.g. other vehicles), or static (e.g. buildings in urban environments). Most of the channel modeling activities aim at averaging the additional attenuation due to these obstacles resulting in a log-normal distribution around the mean received signal power [34, 35]. Although some of these models estimate different values for the variance of this large-scale fading distribution at high and low traffic densities [35], only recently a mechanism for incorporating the effect of vehicles and static obstacles on the received signal power has been proposed in [36] and [37] respectively. Finally, the small-scale fading models the effect of the reception of multiple replica of the transmitted signal at the receiver. Various distributions have been proposed for small-scale fading including Rice [38], Nakagami [34] and Weibull [39, 35] distributions.

Although the signal propagation has great impact on the performance of the communication protocols, most of the recent work on the analysis of VANET topology characteristics are using unit disc as the signal propagation model, where the vehicles can communicate with each other if they are within a threshold distance and cannot communicate otherwise [11, 12, 8, 13, 14, 15, 16, 9, 10]. Although more recently such analysis employs more sophisticated stochastic signal propagation models including both large-scale fading [13, 15, 17] and small-scale fading [14, 17], none of these models incorporate the effect of the vehicles on the signal propagation.

Table 2.1: Related Work on Topology characteristics in VANETs

Ref	Topology	Channel Model	Vehicle Mobility	Database for Vehicle Mobility	Performance Criteria
[11]	Urban	Unit disc with LOS and NLOS	Cellular Automata based traffic mobility model on Manhattan-grid	No database used	Link duration, Link re-healing time
[12]	Mostly Urban	Unit disc	Constant Speed Motion, Manhattan, Fluid Traffic Motion, Intelligent Driver Model	No database used	Link duration, Node degree, No. of cluster, Cluster coefficient, Vehicular density distribution
[8]	Urban	Unit disc	Microscopic road traffic	TAPASCologne Dataset by German Aerospace Institute of Transportation Systems [40]	No. of clusters, Node degree
[13]	Urban	Unit disc, Quasi-Unit disc, Log-normal shadowing	Microscopic road traffic	Map of city of Porto, no database for real traffic	Node degree, Link duration
[14]	Urban	Unit disc, Rayleigh Fading, Ricean Fading	Microscopic multi-model traffic	Map of Berlin and Frankfurt, no database for real traffic	Packet reception probability
[15]	Highway	Unit disc, Log-normal shadowing	Randomly generated	No database used	Access probability, Connectivity probability
[16]	Urban	Unit disc	Manhattan Model	No database used	No. of clusters
[9]	Urban	Unit disc	Multi-agent microscopic traffic	Zurich traffic data used [41]	Node degree, Link duration, Diameter, Centrality, No. of clusters, Clustering coefficient
[6]	Urban	Ray-tracing-derived models	Intelligent Driver Model on Manhattan-grid	No database used	Packet loss ratio, Average end-to-end delay
[7]	Urban	Two ray ground	Chain topology with nodes arranged at a regular interval	No database used	Packet loss distribution
[17]	Urban	Log-normal shadowing, Multi-path fading	Microscopic road traffic	No database used	Packet delivery ratio
[10]	Urban, Highway, Suburban	Unit Disk, Two ray Model	Microscopic road traffic	DRIVE-IN Project [42]	Packet delivery rate, delay

2.2 Vehicle Mobility Model

Realistic representation of the vehicle mobility requires using accurate microscopic mobility modeling, real-world road topology and real-data based traffic demand modeling. The input and parameters of the microscopic mobility simulator are determined based on the real traffic flow and speed values measured by the road sensors deployed along the highway as detailed next.

2.2.1 Microscopic Mobility Modeling

SUMO [19] is used to simulate the microscopic mobility of vehicles. SUMO, generated by the German Aerospace Center, is an open-source, space-continuous, discrete-time traffic simulator capable of modeling the behavior of individual drivers. The path of each driver is determined based on the origin/destination matrix provided as an input to the simulator. The movement of each driver is implemented using the surrounding vehicles via Krauss' car-following model that regulates its acceleration and Krajzewicz's lane-changing model that regulates its overtaking decisions [43]. The parameters of the simulator that determine the driver's acceleration and overtaking decisions include the distance to the leading vehicle, the traveling speed, the acceleration and deceleration profiles, and dimension of the vehicles.

2.2.2 Traffic Demand Modeling

PeMS collects historical and real-time data from highways in the State of California with the goal of providing a comprehensive assessment of highway performance [18]. PeMS was developed by the Department of Electrical Engineering and Computer Sciences at University of California Berkeley, in co-operation with the California Department of Transportation, California Partners for Advanced Transit and Highways, and Berkeley Transportation Systems. The flow and speed data are collected in real time from over 25,000 individual road sensors located over all major metropolitan areas in the state of California. The sampling period of the flow and speed data ranges from 30 seconds to 5 minutes. We used road I-880S in Alameda County, Bay Area, California for our simulation. Fig. 2.1 shows the road sensors located on I-880S.

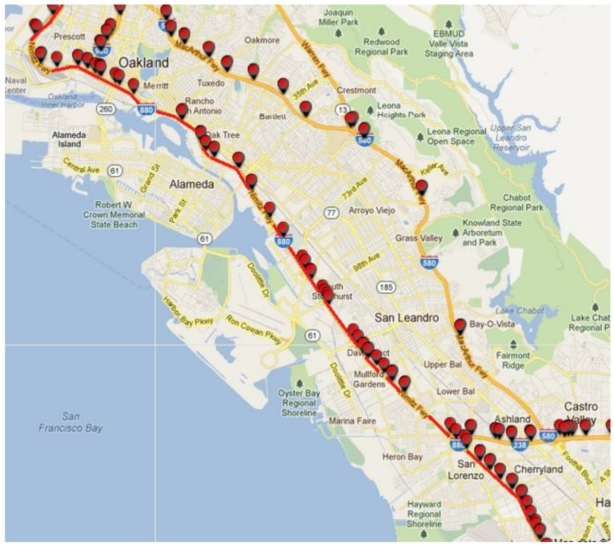


Figure 2.1: Road sensors located on I-880S in Alameda County, Bay Area, California

2.2.3 *Realistic Mobility Generation*

The first step in generating the realistic mobility model is to determine the input of SUMO based on the real-data based vehicular traffic flows over the road. The input of SUMO including the number of vehicles injected at each entry of the highway (the starting point of the vehicle) and the probability that each vehicle leaves the highway from the exits (destination of vehicles) is determined such that the expected number of vehicles passing through each road sensor location in the simulation closely matches the flow measured at that sensor on the actual road. However, matching the traffic flow in the simulation to that of the PeMS database does not guarantee that the average speed of the vehicles in the simulation also matches the speed measured through PeMS. Therefore, the second step in generating realistic mobility model is to determine the parameters of SUMO such that the average speeds of vehicles determined by the simulation and PeMS agree with each other. The parameters of SUMO adjusted for this purpose include the distance to the leading vehicle, the initial speed, the acceleration and deceleration profiles.

Figs. 2.2 and 2.3 show the flow and speed of vehicles recorded from the simulation and PeMS database. The data from 419 road sensors on highway I880-S, as shown in Fig. 2.1, are extracted for both high traffic density, i.e. at $18:00$, and low traffic density, i.e. at $01:00$. As shown in the figures, once the system stabilizes at around 10-th minute, both the flow and speed from simulation and PeMS database agree with each other.

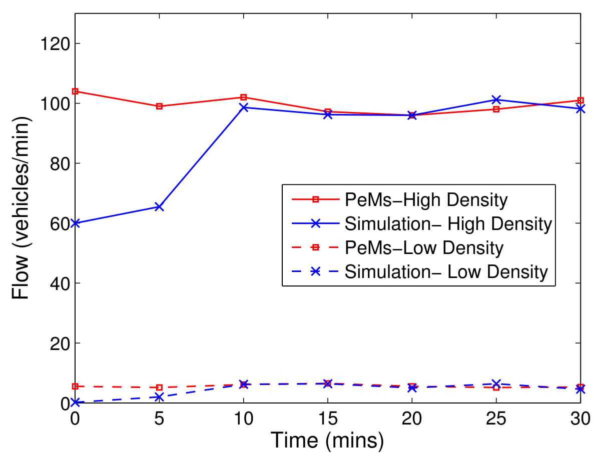


Figure 2.2: Flow of vehicles extracted from the PeMS database and obtained from the simulation at low and high vehicle traffic density.

2.3 *Vehicular Channel Models*

In this section, we will first describe simplistic channel models including unit disc and log-normal shadowing models that are commonly used in the analysis of VANET topology characteristics. We will then describe a recently proposed more realistic channel model called obstacle-based channel model that incorporates the effect of the moving obstacles (i.e. vehicles) on the received signal power due to their dominating influence as illustrated in [36]. The obstacle-based model has not been used before in the VANET topology analysis.

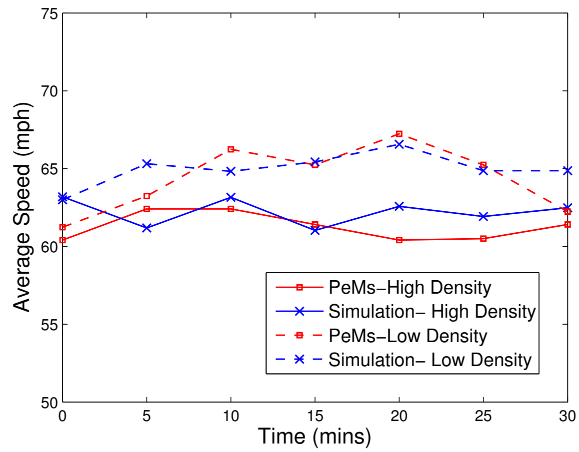


Figure 2.3: Speed of vehicles extracted from the PeMS database and obtained from the simulation at low and high vehicle traffic density.

2.3.1 Unit Disc Model

In the unit disc model, the vehicles can communicate with each other if they are within a threshold distance and cannot communicate otherwise. This model is widely used in the analysis of the VANET topology characteristics due to its simplicity [11, 12, 8, 13, 14, 15, 16, 9]. However, the sharp cut-off at the threshold distance not only fails to capture the random noise that can make even nearby nodes unreachable but also does not take into account the effect of obstacles on the received signal strength.

2.3.2 Classical Log-Normal Shadowing Model

In the classical log-normal shadowing model, rather than calculating the additional attenuation due to each obstacle between the transmitter and receiver, the probabilistic distribution of the additional attenuation is modeled with a log-normal probability density function resulting in the following formulation for the received signal power [34, 35]:

$$P_{rx}(d) = P_0 - 10n \log_{10} \frac{d}{d_0} + N \tag{2.1}$$

where d is the distance between the transmitter and the receiver, d_0 is the reference distance, $P_{rx}(d)$ is the received signal power at distance d (in dBm), P_0 is the received signal power at the reference distance d_0 (in dBm) , n is the path loss exponent and N is zero mean Gaussian random variable with variance σ^2. A vehicle can communicate with another vehicle if P_{rx} is greater than a certain threshold value [15]. Note that the log-normal shadowing model reduces to the unit disc model if $\sigma = 0$. The parameters of the log-normal model is chosen such that the mean transmission range is equal to the threshold distance in the unit disc model to have a fair comparison. The parameters n and σ of the model are chosen based on the channel measurement results reported in [31, 32, 33, 34, 35]: $n = 2.5$, $\sigma = 5.5$dB.

2.3.3 Obstacle-Based Channel Model

In the obstacle-based channel models, algorithms to incorporate the effect of the surrounding obstacles such as other vehicles, walls and buildings on the received signal strength have been proposed [35, 36] rather than modeling the average additional attenuation due to these obstacles by a stochastic large-scale fading model. Usually there are a few buildings around the highway, mostly far from the vehicles. That is why, in this study, we only consider the impact of the surrounding vehicles as obstacles. Since the additional obstacles can only further reduce the probability of the line-of-sight (LOS) between the transmitter and receiver vehicles, this approach gives a best case analysis for the probability of LOS as stated in [36].

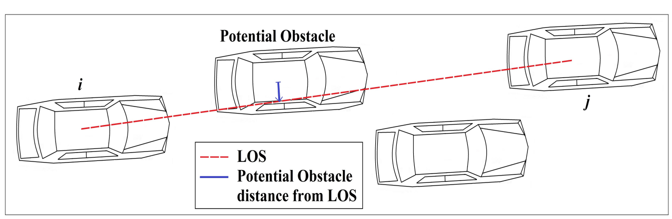


Figure 2.4: Determining the vehicles potentially obstructing the LOS between vehicles i and j

The algorithm proposed and validated in [36] is implemented for calculating the additional attenuation due to the vehicles. This algorithm consists of three main parts:

First, the vehicles potentially obstructing the LOS between the transmitter vehicle i and receiver vehicle j are determined ($getPotentialObs(i, j)$): If the distance from the center of the vehicle to the LOS line between vehicles i and j is less than half the width of the vehicle, the vehicle is considered as a potential obstacle as illustrated in Fig. 2.4 (Line 1 of Algorithm 1).

Algorithm 1 Obstacle Based Model: Calculation of the additional attenuation between vehicles i and j due to surrounding vehicles as obstacles

1: $[PotentialObs] = getPotentialObs(i, j)$
2: **if** $size([PotentialObs]) \neq 0$ **then**
3: $\quad [ObsVeh] = getLOSobs([PotentialObs])$
4: $\quad$ **if** $size([ObsVeh]) \neq 0$ **then**
5: $\quad\quad addAttenuation = calAttenuation([ObsVeh])$
6: $\quad$ **else**
7: $\quad\quad addAttenuation = 0$
8: $\quad$ **end if**
9: **else**
10: $\quad addAttenuation = 0$
11: **end if**

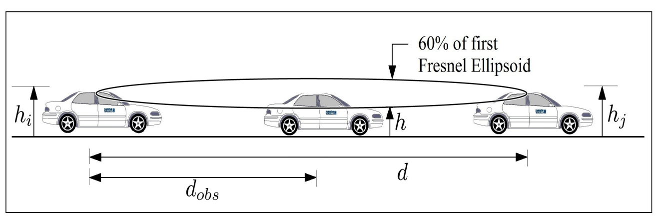


Figure 2.5: Determining the vehicles that obstruct the LOS between vehicles i and j (For simplicity, vehicle antenna heights (h_a) are not shown).

Second, the vehicles that obstruct the LOS between vehicles i and j are chosen from the set of the potential obstructing vehicles identified in the previous step ($getLOSobs([PotentialObs])$): From the electromagnetic wave propagation perspective, the LOS is not guaranteed with the existence of the visual sight line between the transmitter and receiver. Any vehicle that obstructs the Fresnel ellipsoid might affect the transmitted signal. The effective height of the LOS line that connects vehicles i and j at the potential obstacle vehicle location when we use the first Fresnel ellipsoid is given by

$$h = (h_j - h_i)\frac{d_{obs}}{d} + h_i - 0.6r_f + h_a \tag{2.2}$$

where h_i and h_j are the heights of the transmitter vehicle i and receiver vehicle j respectively, d_{obs} is the distance between the transmitter and the obstacle, d is distance between the transmitter and receiver, h_a is the height of the vehicle antennas, and r_f is the radius for the first Fresnel zone ellipsoid which is given by

$$r_f = \sqrt{\frac{\lambda d_{obs}(d - d_{obs})}{d}} \tag{2.3}$$

with λ denoting the wavelength. Fig. 2.5 illustrates these parameters. If the height of each potentially obstructing vehicle is known beforehand, the vehicle will obstruct the LOS between the transmitter and receiver if h is greater than its height. Based on the assumption that the vehicle heights follow a normal distribution as also assumed in [36], the probability of the LOS for the link between vehicles i and j is calculated as

$$\Pr(LOS|h_i, h_j) = 1 - Q(\frac{h - \mu}{\sigma}) \tag{2.4}$$

where μ and σ are the mean and standard deviation of the height of the obstacle vehicle (Line 3 of Algorithm 1).

Third, the additional attenuation in the received signal power is calculated for the LOS obstructing vehicles determined in the previous step ($calAttenuation([ObsVehicles])$). The existing models to calculate the attenuation are empirical and vary from optimistic [44] to pessimistic approximations [45, 46]. To calculate the additional attenuation, we used the ITU-R method based on the multiple knife edge model [47] as suggested in [36]. In this model, a complete profile is created for all the LOS obstructing vehicles, and the signal attenuation is calculated based on the vehicle height, distance from the transmitting vehicle, wavelength of electromagnetic waves and position of the vehicles (Line 5 of Algorithm 1).

2.4 Performance Metrics

The performance metrics are used in the comparison of different signal propagation models. For the formal definition of these metrics, we represent the vehicular network topology at time t by a graph $G(t) = (V, E(t))$ where V is the set of vehicles and $E(t) \subset V \times V$ are the (undirected) edges representing the wireless communication links between the vehicles.

2.4.1 Node Degree

Node degree of a vehicle is defined as the number of neighboring vehicles it can communicate with. Let us denote the set of neighbors of vehicle i at time t by $N_i(t)$ such that $N_i(t) = \{j|(i,j) \in E(t)\}$. The degree of node i is then formulated as $d_i(t) = |N_i(t)|$. Node degree measures the density of the network from the physical connectivity point of view.

2.4.2 Neighbor Distance Distribution

Neighbor distance distribution is defined as the distribution of the distance of the neighbors of the vehicles in the network. Let us denote the set of neighbors at distance d away from vehicle i at time t by $N_i(t,d)$ such that $N_i(t,d) = \{j|(i,j) \in E(t), d_{ij} = d\}$ where d_{ij} is the distance between vehicles i and j. The neighbor distance distribution as a function of the distance d denoted by $f(d)$ is then formulated as $f(d) = \frac{1}{NT}\sum_{t=1}^{T}\sum_{i=1}^{N}|N_i(t,d)|$ where $N = |V|$ is the total number of vehicles in the network and T is the total simulation time. Neighbor distance distribution measures the distribution of the communicating nodes over space.

2.4.3 Link Duration

Link duration is defined as the time span between the instants at which the communication link between two vehicles is established and lost. Let us denote the times when the link between vehicles i and j is established and broken by t_0 and t_f respectively such that $(i,j) \notin E(t_0 - \epsilon)$, $(i,j) \notin E(t_f + \epsilon)$ for arbitrarily small $\epsilon > 0$ and $(i,j) \in E(\tau)$, $\forall \tau \in [t_0, t_f]$. Then the duration of the link between vehicles i and j denoted by l_{ij} is formulated as $l_{ij} = t_f - t_0$. Link duration measures how stable a connection is over time.

2.4.4 Closeness Centrality

Closeness centrality is defined as the inverse of the sum of the distances to all other nodes in the network. Formally, the closeness centrality denoted by $CC_i(t)$ is formulated as

$$CC_i(t) = \frac{1}{\sum_{j \in [1,N], j \neq i} d_{ij}} \tag{2.5}$$

More central nodes have a lower value for the total distance to all other nodes thus higher value for closeness centrality. Closeness centrality measures how long it will take information to spread from a given vehicle to other vehicles in the network.

2.4.5 Number of Clusters

Number of clusters is defined as the number of co-existent, non-connected groups of nodes at a given instant. We define cluster as a connected group of vehicles within which there exists a path between any pair of nodes. Formally, let us denote the existence of a path between vehicles i and j at time t by the binary variable $p_{ij}(t)$ such that $p_{ij}(t)$ takes value 1 if $(i,j) \in E(t)$ or there exists k for which $(i,k) \in E(t)$ and $p_{kj}(t) = 1$, and value 0 otherwise. Let us also define the cluster in which vehicle i is located at time t as $C_i(t) = i \cup \{j | p_{ij}(t) = 1\}$. The set of unique clusters in the network at time t is formulated as

$$C(t) = \{C_i(t) | C_i(t) \cap C_k(t) = \emptyset, \forall k \in [1, i-1]\} \tag{2.6}$$

The number of clusters denoted by $c(t)$ is then equal to $|C(t)|$. Number of clusters measures the degree of fragmentation in the network in terms of the number of mutually isolated groups of vehicles.

2.4.6 Size of the Largest Cluster

Size of the largest cluster is defined as the number of nodes in the largest cluster of the network. Formally, the size of the largest cluster denoted by $c_{\max}(t)$ is formulated as $c_{\max}(t) = \max_{i \in [1,N]} |C_i(t)|$.

2.4.7 Clustering coefficient

Clustering coefficient is defined as the ratio of the number of links within a cluster to the maximum number of links that could exist within a cluster. Let us denote the set of links within cluster $C_i(t)$ by $E_{C_i}(t) = \{(i,j) | (i,j) \in E(t); i,j \in C_i(t)\}$. The clustering coefficient of the same cluster denoted by $k_{C_i}(t)$ is then formulated as

$$k_{C_i}(t) = \frac{|E_{C_i}(t)|}{|C_i(t)|(|C_i(t)| - 1)} \tag{2.7}$$

Clustering coefficient measures the degree of connectivity of the vehicles within a cluster. Note that the clustering coefficient has a maximum value 1 if the cluster is a clique.

Chapter 3

MATCHED LOG-NORMAL SHADOWING MODEL & PERFORMANCE ANALYSIS

In this chapter, we propose a matching mechanism to tune the parameters of Log-normal shadowing model such that the performance metrics summarizing the link characteristics over space agree with those of the obstacle-based model.

3.1 Matched Log-normal shadowing model

The classical log-normal shadowing model is based on specifying the probabilistic distribution of the additional attenuation due to the vehicles instead of calculating the attenuation due to each obstacle separately. The runtime of the simulations using this model therefore is reasonable. However, the parameters of this model, including the path loss exponent and the standard deviation of the Gaussian distribution, are fixed independent of the density of the surrounding vehicles leading to unrealistic simulations.

The obstacle-based channel model, on the other hand, incorporates the effect of each vehicle on the received signal strength separately. However, the accurate representation of the channel comes at the cost of high complexity and computational burden preventing the integration into the network simulators. Fig. 3.1 shows the average runtime of the analysis of the performance metrics based on the 1800 sec simulation of the scenario where the vehicles are generated using the real data at different traffic densities and communicating with mean transmission range of 500 m under unit disc, classical log-normal and obstacle-based channel models denoted by $Unit$, Log_C and Obs respectively. We observe that the average runtime of the obstacle-based model is about 100 times more than that of the unit disc and classical log-normal models.

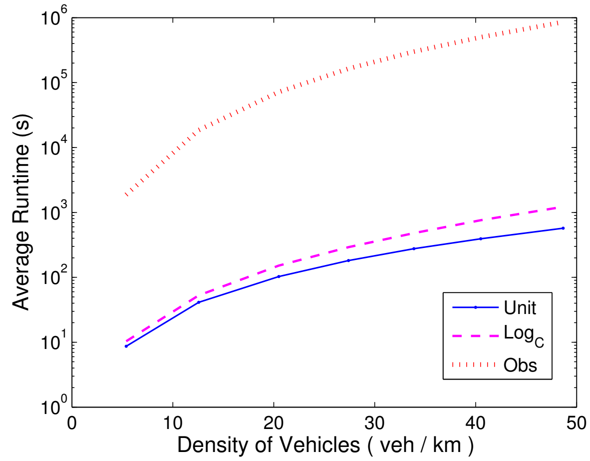


Figure 3.1: Average runtime of the analysis of the performance metrics based on the 1800 sec simulation of the scenario where the vehicles are generated using the real data at different traffic densities and communicating with mean transmission range of 500 m under different channel models.

In this section, we propose a matching mechanism to tune the parameters of the lognormal model such that the performance metrics summarizing the link characteristics over space agree with those of the obstacle-based model (Section 3.1.1). We also introduce a correlation model to take into account the evolution of the link characteristics over time and propose a mechanism to tune the parameters of this model to match the performance metrics summarizing the time characteristics of the links to those of the obstacle-based model (Section 3.1.2). The resulting matched log-normal model provides performance close to the obstacle-based model at much lower computational cost and implementation complexity allowing its integration into the network simulators.

3.1.1 Matching Parameters of Log-normal Model

The parameters of the log-normal model that need to be matched for the spatial evolution of the link characteristics include the path loss exponent and standard deviation of the Gaussian random variable.

Algorithm 2 Matching Parameters of Log Normal Shadowing Model

Input: $nValues, \sigma Values$

Output: n_m, σ_m

```
 1: NDObs = cdfND(ObsModel);
 2: Error_min = ∞;
 3: for all nValues do
 4:     for all σValues do
 5:         NDLog = cdfND(LogModel, n, σ) ;
 6:         Error = Diff(NDObs, NDLog) ;
 7:         if Error < Error_min then
 8:             Error_min = Error;
 9:             n_m = n;
10:             σ_m = σ;
11:         end if
12:     end for
13: end for
```

Our matching algorithm to tune these path loss exponent and standard deviation parameters is given in Algorithm 2. The inputs of the algorithm are the set of possible values for the path loss exponent and standard deviation denoted by $nValues$ and $\sigma Values$ respectively. The outputs of the algorithm are the values for the path loss exponent and standard deviation that provide the best match to the obstacle based model denoted by n_m and σ_m respectively.

The algorithm starts by extracting the cumulative distribution function (cdf) of the node degree metric for the obstacle-based model by using function $cdfND$ with the parameter $ObsModel$ representing obstacle-based model and storing the resulting cdf in the variable $NDObs$, and initializing minimum error to infinity by using variable $Error_{min}$ (Lines $1-2$). The algorithm then computes the cdf of the node degree metric of the log-normal model, represented by $LogModel$, and storing the resulting cdf in the variable $NDLog$ for every possible value of the path loss exponent and standard deviation, stored in the variables n and σ respectively in each iteration (Lines $3-5$). The difference between the cdf of the obstacle-based and log-normal models is then calculated by using Kolmogorov-Smirnov statistic defined as $sup_x|NDObs(x) - NDLog(x)|$ in the function called $Diff$ (Line 6). The values of the path loss exponent and standard deviation that provide the minimum difference are then selected to provide the best match to the obstacle based model (Lines $7-11$).

The reason for choosing the cdf of the node degree in the matching algorithm is that node degree is a measure of the spatial distribution of the nodes in the network. The above process is repeated for the neighbor distance distribution metric to validate the values n_m and σ_m. The matching of the remaining performance metrics are justified in Section 3.2.

3.1.2 Matching Time Correlation

The instances of the Gaussian variable used in the classical log-normal model are calculated independently at each time step of the simulation resulting in zero correlation of the link characteristics over time. The obstacle-based model on the other hand provides the time correlation of the link characteristics implicitly due to the slow changes in the relative locations of the obstacles between the transmitter and receiver vehicles. We therefore extend the classical log-normal model to include a correlation model taking into account the temporal evolution of the link characteristics.

We used Gudmunson model with exponential correlation function for this work [48]. This model has been previously used for spatially correlated processes [49, 50]. However, as described in [51], this model can also be used for time correlation. The model describes the correlation of the shadowing process at time difference Δt by

$$R(\Delta t) = \sigma^2 . \exp(-\alpha \Delta t)$$

where σ is the standard deviation of the Gaussian variable at each time instant, and α is the correlation factor.

The matching algorithm to tune the value of the correlation factor is given in Algorithm 3. The inputs of the algorithm are the values of the path loss exponent and standard deviation providing the best match with the spatial link characteristics of the obstacle-based model, i.e. n_m and σ_m respectively, and the set of possible values

Algorithm 3 Matching Correlation Factor for Log Normal Shadowing Model

Input: $n_m, \sigma_m, \alpha Values$

Output: α_m

1: $LDObs = cdfLD(ObsModel);$
2: $Error_{min} = \infty;$
3: **for all** $\alpha Values$ **do**
4: $\quad LDLog = cdfLD(LogModel, n_m, \sigma_m, \alpha)$
5: $\quad Error = Diff(LDObs, LDLog)$
6: $\quad$ **if** $Error < Error_{min}$ **then**
7: $\qquad Error_{min} = Error;$
8: $\qquad \alpha_m = \alpha;$
9: $\quad$ **end if**
10: **end for**

for the correlation factor denoted by $\alpha Values$. The output of the algorithm is the value of the correlation factor that provides the best match with the temporal link characteristics of the obstacle-based model denoted by α_m.

The algorithm starts by determining the cdf of the link duration metric for the obstacle based model by using function $cdfLD$ and storing the resulting cdf in the variable $LDObs$, and initializing minimum error to infinity (Lines $1-2$). The reason for choosing the cdf of the link duration is that link duration is a measure of the stability of the links over time. The algorithm continues by computing the cdf of the link duration metric of the log-normal model with the matched parameters n_m and σ_m, and every possible value of the correlation factor stored in the variable α, and storing the resulting cdf in the variable $LDLog$ in each iteration (Lines $3-4$). The difference between the cdf of the link duration of the log-normal and obstacle-based models is then calculated by using Kolmogorov-Smirnov statistics in the function $Diff$ (Line 5). The value of the correlation factor giving the minimum difference is selected as the best match value σ_m (Lines $6-9$).

3.2 Simulation Results

The goal of the simulations is to compare the effect of different channel models including the unit disc, classical log-normal fading, obstacle-based and matched log-normal channel models on the topology characteristics of VANET located on a large-scale highway by comparing the node degree, neighbor distance distribution, link duration, closeness centrality, number of clusters, size of the largest cluster and clustering coefficient metrics of the resulting communication graphs as explained in Section 2.4. In all the figures in this section, the unit disc, classical log-normal, matched log-normal and obstacle-based channel models are denoted by $Unit$, Log_C, Log_M and Obs respectively, and R refers to the mean transmission range of vehicles.

The topology of the VANET is obtained by using the accurate microscopic mobility modeling of SUMO while determining its input and parameters based on the PeMS database as explained and validated in detail in Section 2.2. The flow and speed data of 419 road sensors on highway I880-S as shown in Fig. 2.1 at both high traffic density, i.e. 121 vehicles/km at 18 : 00, and low traffic density, i.e. 11 vehicles/km at 01 : 00, are used for this purpose. The performance metrics are extracted after the system stabilizes around the 10-th minute as illustrated in Figs. 2.2 and 2.3. The vehicle mobility output of SUMO is then input to MATLAB where the channel models are implemented and the performance metrics are derived and plotted.

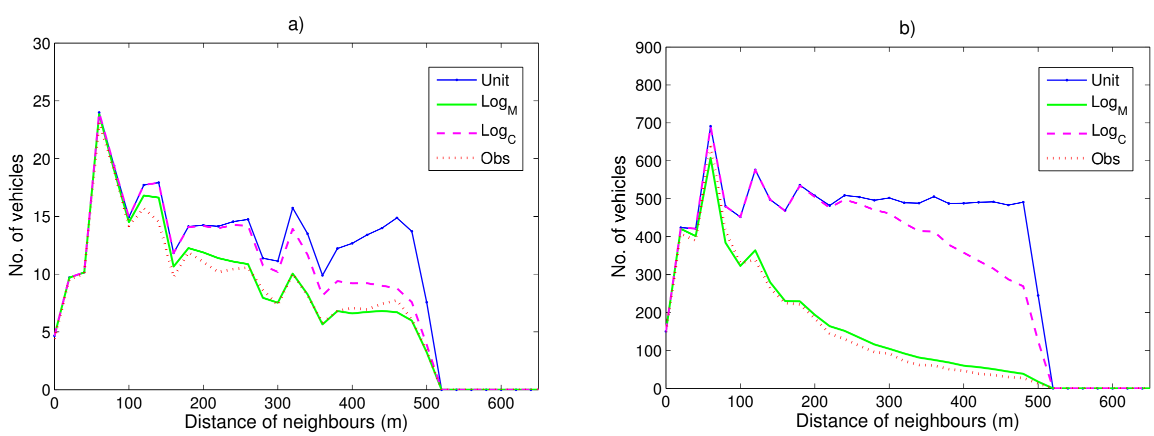


Figure 3.2: Neighbor distance distribution for a) low density and b) high density networks at 500 m transmission range.

Figs. 3.2 a) and b) show the neighbor distance distribution for low and high density networks respectively. The matched log-normal model follows the obstacle-based model closely. The difference between the obstacle-based model and commonly used unit disc and classical log-normal fading models on the other hand increases as the vehicle density increases and the transmission range becomes greater than 100 m. To elaborate on the effect of this different behavior in the performance metrics, we therefore plot the rest of the graphs at transmission ranges of 100 m and 500 m for both low and high vehicle traffic densities.

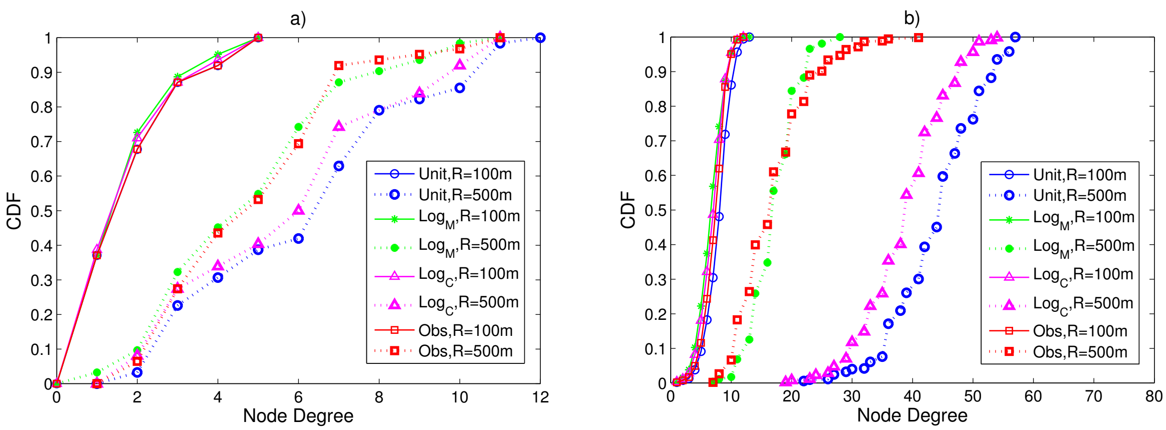


Figure 3.3: Cdf of the node degree metric for different channel models and transmission ranges in a) low density and b) high density networks

Figs. 3.3 a) and b) show the cdf of the node degree metric for different channel models and transmission ranges in low and high density network respectively. All channel models generate the node degree distribution very close to each other at low transmission range for both low and high density networks. As the transmission range and the density of the network increase, the discrepancy between the obstacle based model and unit disc and classical log-normal fading models increases as expected from the difference observed in the neighbor distance distribution. The matched log-normal model on the other hand still agrees with the obstacle-based model for all scenarios.

Figs. 3.4 a) and b) show the cdf of the link duration metric for different channel models and transmission ranges in low and high density network respectively. The link duration for obstacle based model is smaller than that of the unit disc model and larger than that of the lognormal model. The main reason is that the nodes

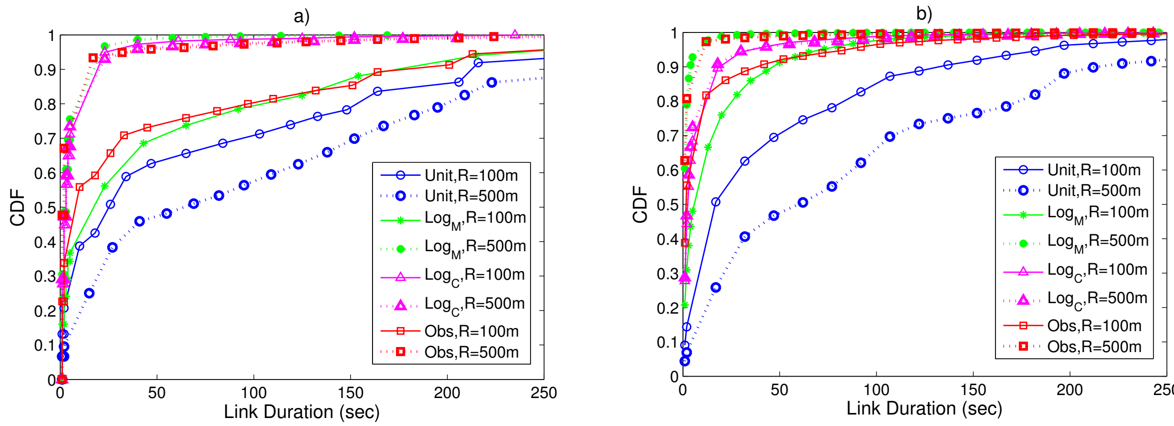


Figure 3.4: Cdf of the link duration metric for different channel models and transmission ranges in a) low density and b) high density networks

can always communicate with each other within a threshold distance for the unit disc model creating high correlation of the connectivity behavior over time so much higher link duration. On the other hand, the connections between the vehicles are determined probabilistically for the lognormal model where the probability is chosen independently in each step creating low correlation of the connectivity behavior so much lower link duration. The link duration for the obstacle based model is closer to the unit disc model at low transmission range and closer to the lognormal model at high transmission range meaning the correlation of the connectivity behavior decreases as the transmission range increases in the obstacle-based model. The link duration of the matched log-normal model again is very close to that of the obstacle-based model under all conditions.

Fig. 3.5 shows the cdf of the number of clusters metric for different channel models and transmission ranges in low density network. Since the number of clusters is very low for high density networks when the transmission range is between 100 m and 500 m, we did not include a separate graph for the high density network. The distributions of the number of clusters based on the unit disc and obstacle based models are very close to each other. The main reason for this similarity even at different transmission ranges is that the vehicle which acts as an obstacle between two

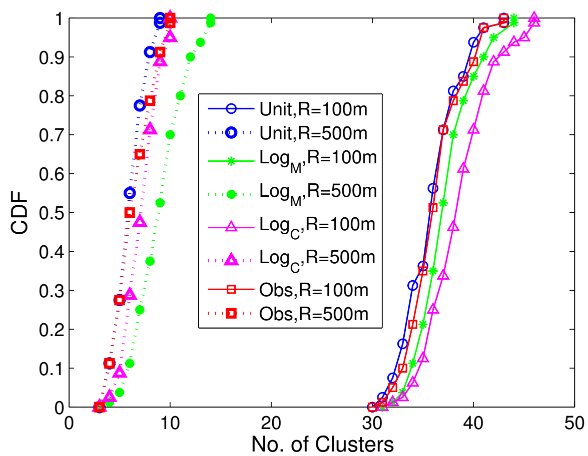


Figure 3.5: Cdf of the number of clusters metric for different channel models and transmission ranges in low density network

vehicles also acts at the same time as a bridge between them resulting in an indirect connection through the obstructing vehicle. The vehicles are directly connected when unit-disc model is used whereas they are connected through the obstacles in the obstacle-based model, resulting in the same number of vehicles within clusters. The number of clusters of the matched log-normal model is also close to that of the obstacle-based model with a slight difference resulting from not including the spatial correlation of the Gaussian variables in the log-normal model. However, the classical log normal model does not provide a good matching to the obstacle-based model.

Figs. 3.6 a) and b) show the cdf of the size of the largest cluster for different channel models and transmission ranges in low and high density network respectively. Similar to the behavior of the number of clusters metric, the size of the largest cluster for unit disc and matched log-normal model is very close to that of the obstacle-based model whereas the largest cluster size is very different for classical log-normal and obstacle-based models.

Figs. 3.7 a) and b) show the cdf of the clustering coefficient metric for different channel models and transmission ranges in low and high density network respectively. Although the unit disc and obstacle-based models agree with each other in the number

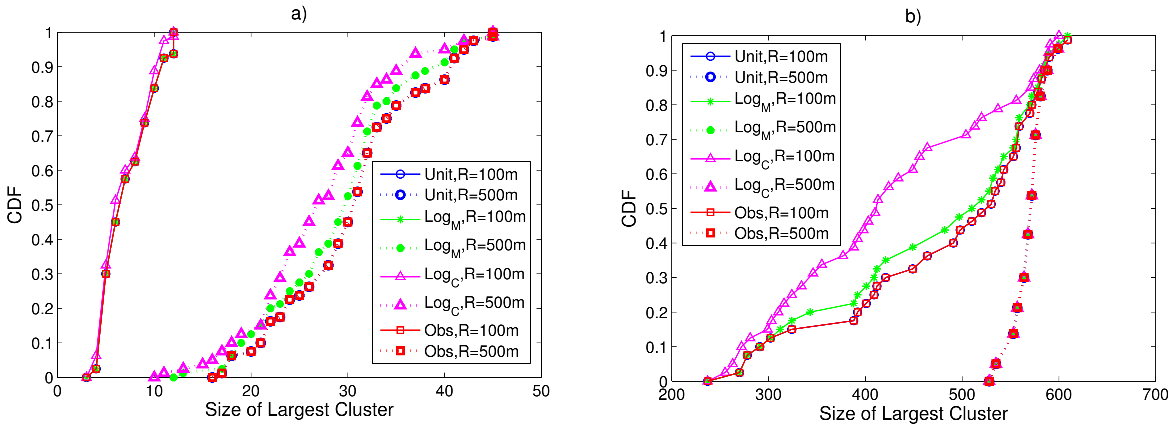


Figure 3.6: Cdf of the size of largest cluster metric for different channel models and transmission ranges in a) low density and b) high density networks

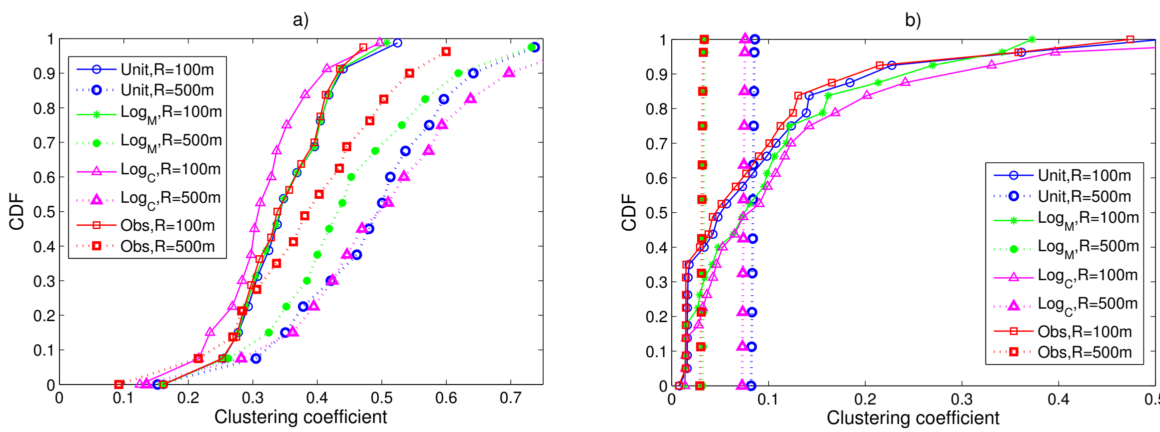


Figure 3.7: Cdf of the clustering coefficient metric for different channel models and transmission ranges in a) low density and b) high density networks

of clusters and size of largest cluster metrics, we observe that their performance is very different when the clustering coefficient is considered. The reason is that the clustering coefficient provides the degree of connectivity of the vehicles within a cluster, which differentiates between the direct connectivity and the connection through the obstacles unlike the number of clusters and size of largest cluster metrics. The matched log-normal model on the other hand again agrees with the obstacle-based model in the clustering coefficient metric unlike the classical log-normal model.

Figs. 3.8 a) and b) show the cdf of the closeness centrality metric for different

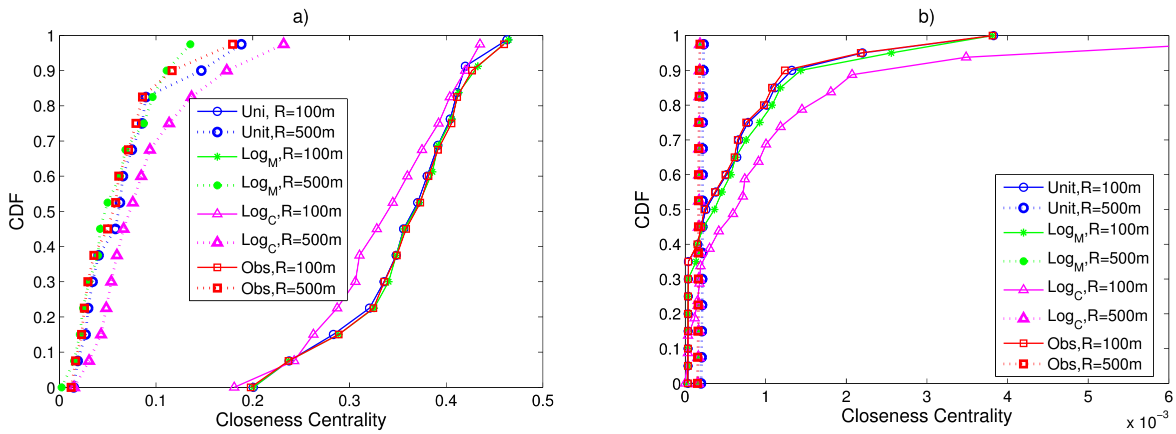


Figure 3.8: Cdf of the closeness centrality metric for different channel models and transmission ranges in a) low density and b) high density networks

channel models and transmission ranges in low and high density network respectively. Again, the matched log normal model provide very close performance to the obstacle based model unlike the classical log normal model.

3.3 Validation of Results

To validate the proposed matched log-normal model, we have extended the simulations for various vehicle traffic densities and an additional highway road I5-S near Los Angeles, California, and checked the agreement of the resulting matched parameters including path loss exponent, standard deviation and correlation factor for matched lognormal model on I5-S and I-880S highway roads. The I5-S road is very different from the I-880S road with much higher vehicle traffic density due to the proximity to Los Angeles, higher number of lanes and intersections. The realistic mobility over the I5-S road is generated by determining the input and parameters of the microscopic mobility simulator SUMO based on the flow and speed information provided by the PeMS database as explained in detail in Section 2.2. The vehicle mobility output of SUMO is then input to MATLAB where the values of the path loss exponent, standard deviation and correlation factor that provide the best match to the obstacle-based model are determined.

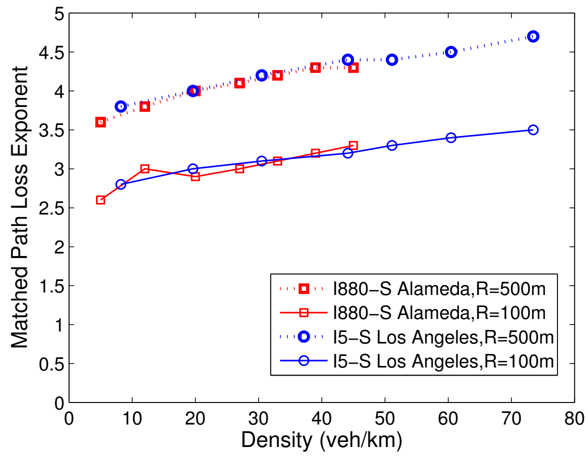


Figure 3.9: Matched path loss exponent for I880-S and I5-S highway roads at different vehicle traffic densities and transmission ranges.

Figs. 3.9, 3.10 and 3.11 show the matched path loss exponent, standard deviation and correlation factor values respectively for I880-S and I5-S highway roads at different vehicle traffic densities and transmission ranges. We observe that the matched values of these parameters are consistent across different highways and different vehicle traffic densities.

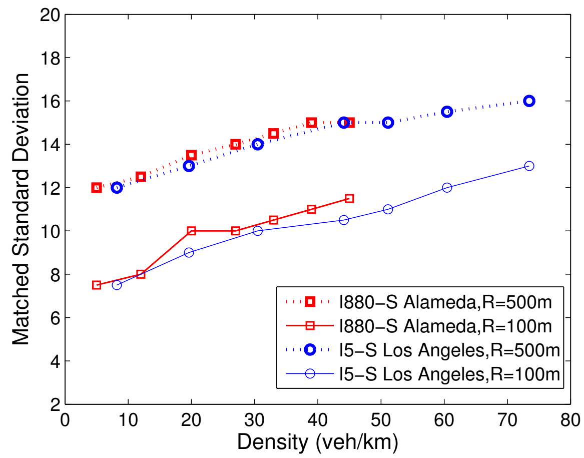


Figure 3.10: Matched standard deviation for I880-S and I5-S highway roads at different vehicle traffic densities and transmission ranges.

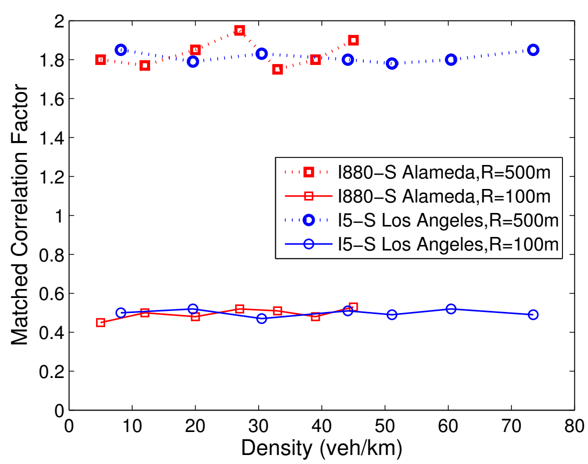


Figure 3.11: Matched correlation factor for I880-S and I5-S highway roads at different vehicle traffic densities and transmission ranges.

Chapter 4

USING SYSTEM SIZE ESTIMATION IN P2P NETWORKS FOR DENSITY ESTIMATION IN VANETS

In this chapter, we adapted and implemented three fully distributed infrastructure free road traffic density estimation algorithms inspired from system size estimation techniques used in P2P networks.

4.1 Related Work

The existing methods for density estimation in VANETs can be broadly divided into two main categories: (1) Infrastructure-based and (2) Infrastructure-free. A summary of these methods are given in Table 4.1.

In the infrastructure-based methods, dedicated infrastructure such as loop detectors, roadside sensors or cameras are used to determine the presence of the vehicles on the road [52, 53, 54]. Road side camera images are used for traffic monitoring and density estimation in [52]. Using Kalman filter-based background estimation, the difference between the incoming image and the calculated background is used to mark vehicles and then to estimate the density of vehicles on the road. A similar approach using data fusion has been proposed in [53] in which the flow measured from video cameras on the road and travel time measured from GPS are used to estimate the density of vehicles. A neural network technique is applied on the data collected using video monitoring system to estimate the density of vehicles in [54].

In the infrastructure-free methods, vehicles co-operate with each other to estimate the size of the network. A probe vehicle uses information of number of its neighbors to calculate the local density, which is then used to estimate the global density, assuming that the inter-vehicular spacing is exponentially distributed in [22]. This work

Table 4.1: Related Work on Vehicle Density Estimation in VANETs

Ref	Infrastructure used	Category	Method used
[52]	Yes	Centralized	Road-side camera images using Kalman filtering
[54]	Yes	Centralized	Neural networks
[53]	Yes	Centralized	Capturing road video using cameras and applying Kalman filtering
[22]	No	Distributed	Local density used to estimate global density
[20]	No	Clustering	Extension of [22] by using clusters
[21]	No	Distributed	Group formation
[55]	No	Distributed	Traffic-flow model using vehicle's speed and flow
[56]	No	Distributed	Random sampling of vehicles
[23]	No	Distributed	Vehicle's speed and acceleration information
[57]	No	Distributed	Fluid dynamics and car follow model

has been extended with a clustering approach [20] where the cluster heads gather information about the cluster members which is then used to estimate the global density. A fully distributed grouping approach is used for density estimation in [21] where group leader computes vehicle density and disseminates this information among other members of the group. In [55], a relationship between speed, flow and density is used to estimate local density using traffic-flow model. A similar approach is used in [23] where vehicle tracks its own speed and acceleration patterns to estimate the local density. In [56], vehicles are uniformly sampled from a road section, and their neighbor information is then used to estimate the density. Fluid dynamics and car follow models are utilized to estimate the vehicle density in [57].

In this work, we propose fully distributed and infrastructure-free mechanisms for the density estimation in VANETs. Unlike previous distributed approaches which either use group formation [21], or rely on vehicle speed and flow information to calculate density of vehicles, we use network size information (i.e. number of vehicles in a particular geographical location) to estimate the density of vehicles on road. To the best of our knowledge, network size estimation approach has not been previously applied to VANETs for density estimation.

4.2 Density Estimation Algorithms

Inspired by the mechanisms for system size estimation in P2P networks, we adapted and implemented three fully distributed algorithms, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation, for vehicular density estimation. The algorithms are used in calculating the number of vehicles within a particular geographical region specified by the Global Positioning System (GPS) coordinates. Once we calculate the network size (number of vehicles) within a particular geographical region, we divide the network size by the length of the roads in that area to estimate the density of vehicles. Details of the algorithms are given next.

4.2.1 Sample & Collide

Sample & Collide algorithm is based on uniformly sampling the nodes from a population, and then estimating the system size depending on how many samples of the nodes are collected, before an already sampled node is re-selected [58].

The approach is built upon the *inverted birthday paradox*. According to the inverted birthday paradox, in a room of 57 or more people, the probability of two people having the same birthday is at least 99%. We can calculate the probability $p(N, K)$ of at least two people having birthday on the same date in a group of K people for N = 365 days. Sample & Collide is built on inverting such evaluations. We determine the number of people $X(N)$ that needs to be sampled, one at a time, until two people share the same birthday. It turns out that for large N, value of $X(N)$ converges to $\sqrt{2N}$. In the vehicle density estimation, the number of days corresponds to the number of nodes in the network, and sample of people having the same birthday corresponds to the number of the nodes selected until two samples coincide. The number of samples that are obtained before this happens gives the estimate for the number of nodes N, where $N = X^2/2$.

The accuracy of the algorithm relies heavily on the sampling technique used. Sampling technique of Sample & Collide is asymptotically unbiased in contrast to the previously proposed sampling techniques in graphs with heterogeneous node degrees

[58]. The unbiased sampling of Sample & Collide proceeds as follows.

- An initiator node sets timer T to some predefined value ($T > 0$) in the sampling message, and sends the message to one of its neighboring nodes.

- Upon receiving a sampling message, a node i does the following operations. It picks a random number U uniformly distributed between $[0, 1]$. It then decrements T by $\log(1/U)/d_i$ (i.e. $T \leftarrow T - \log(1/U)/d_i$), where d_i is the degree of the current node i. If the updated value $T \leq 0$, then the current node i is selected as the sampled node. Otherwise, it forwards the updated timer value T to one of its neighbors selected uniformly at random, and the sampling process continues.

- Samples are collected by the initiator node until a node, which has already been sampled, is re-selected. Initiator node counts the number of samples C obtained before the same node is re-selected. Estimated value for the number of nodes is given by $N = C^2/2$.

- To improve the accuracy of the algorithm, the fixed control parameter L is used. Initiator node picks an integer $L > 0$ and starts the sampling process. The process is continued until L collisions occur, i.e. same nodes are re-selected L times. The initiator node counts the number of samples C_L obtained until L collisions occur. Using inverted birthday paradox, size of the network (i.e. number of vehicles) can then be estimated as $N = {C_L}^2/2L$ [58].

Once we calculate size of network, we estimate the density of vehicles D_a within area of size a by $D_a = N/l_a$, where N is the number of vehicles on road, and l_a is the total length of road within area of size a. As explained in the algorithm, we introduced fixed control parameter L in our implementation to improve the accuracy and performance of the algorithm for dynamic networks like VANETs.

The value of T should also be carefully selected so that there is negligible bias in selecting the samples from the pool of nodes [58]. If a high T value is selected by the initiator node, the system becomes more asymptotically unbiased while increasing the communication overhead.

4.2.2 Hop Sampling

Hop Sampling algorithm is based on the principle of probabilistic polling [59]. The initiator node spreads a message to all the nodes in the network using gossiping. The nodes reply back to the initiator probabilistically depending on their distance from it. Based on the replies that the initiator node gets from other nodes in the network, it estimates the size of the network. The algorithm works as follows.

- The *hopCount* value is initialized to zero by the initiator, and the message is sent to the neighboring nodes of the initiator.

- Upon receiving a gossip message, a node checks if it has previously received that gossip message. If the node has not received the gossip message, it saves the value for *hopCount*. Otherwise, the node compares the newly received *hopCount* value with stored value of *hopCount*. If the new value is less than the stored value, the node replaces old *hopCount* value with the new value, and forwards the message to its neighboring nodes with *hopCount* value equal to $hopCount+1$. Otherwise, the node ignores the message. Minimum value of *hopCount* received by node represents the distance of the node from the initiator node.

- Depending on the distance of the node from the initiator, each node probabilistically replies back to the initiator. This is to save the initiator node from massive flood of incoming messages. Message is sent back with probability 1 if $hopCount < minHopsReporting$, and with probability $1/gossipTo^{hopCount-minHopsReporting}$ otherwise, where *minHopsReporting* and *gossipTo* are system parameters and their values are set by the initiator node.

- Upon receiving the messages from the nodes, the initiator node calculates the size of the network depending on the responses it gets back from the nodes at different distances. For instance, if the value of *minHopsReporting* and *gossipTo* is set to 2, only $1/2^{4-2}$ fraction of the total nodes (i.e. 25%) at distance 4 hops, will reply to the initiator node.

In our simulations, the values of *minHopsReporting* and *gossipTo* are set to 2. Density of vehicles D_a within area of size a is then obtained by $D_a = N/l_a$, where N is the number of vehicles on the road, and l_a is the total length of the road within area of size a.

4.2.3 Gossip-based Aggregation

Gossip-based aggregation algorithm has been proposed for large-scale overlay networks, where each peer periodically exchanges information with one of its neighbours picked at random to estimate the size of the network [60]. In this study, gossip-based aggregation algorithm has been adapted for dynamic VANETs. In the algorithm, if one node in the system holds weight value equal to 1, and rest of the nodes hold weight value equal to 0, then the average of the weight values in the system would be $1/N$, where N is the size of the network. The algorithm works as follows.

- Initiator node samples K vehicles at random.

- These K vehicles then initialize their weight values to 1 and all other nodes in the system initialize their weight to 0. K nodes then start gossiping with one of their neighbors selected randomly.

- At each predefined cycle, the nodes which have previously received a gossip message, randomly select one of their neighboring nodes, to exchange the values of their weights. These nodes then update their weight by the average of their current weight and the weight of their neighbor as

$$weight \leftarrow \frac{weight_{currentNode} + weight_{neighborNode}}{2}$$

- The gossiping is repeated for a certain number of *gossipRounds* until the value of the weight of the nodes converges. The size of the network is then estimated at each node by using equation $N = \frac{K}{weight}$.

One of the drawbacks of using gossip-based aggregation algorithm in dynamic networks is that if the nodes leave the network during the initial phase of the algorithm after receiving the gossip message, the accuracy of the algorithm decreases significantly. To make the algorithm perform better in dynamic situations, we introduced the scheme of initiating the algorithm by selecting K distinct vehicles at random, instead of widely used approach of running the algorithm with one initiator. Sampling technique we used for selecting K vehicles at random by the initiator is similar to the technique used for *Sample & Collide*. The initiator sets timer T to some predefined value ($T > 0$) and sends message to one of its neighbors. A node i, after receiving the message, decrements T by $\log(1/U)/d_i$ (i.e. $T \leftarrow T-$ $\log(1/U)/d_i$), where d_i is the degree of node i and U is uniformly distributed random number between $[0, 1]$. If $T \leq 0$, current node is selected as one of the K nodes to start the gossip algorithm. This process is repeated K times to select K initiator nodes for gossip based algorithm.

The density of the vehicles D_a within area of size a is then obtained by $D_a = N/l_a$, where N is the number of vehicles on the road, and l_a is the total length of road within area of size a.

Table 4.2: Parameters for Highway and Urban Scenarios

	Length of roads (km)	Density	No. of Vehicles	Average Speed (km/h)	Maximum Speed (km/h)	Acceleration (m/s^2)
Highway	Small (2km) &	Low	903	104	120	2
	Big (11.5km)	High	4436	102	120	2
Urban	Small (1.8km) &	Low	1566	52	60	2
	Big (12.9km)	High	4666	47	60	2

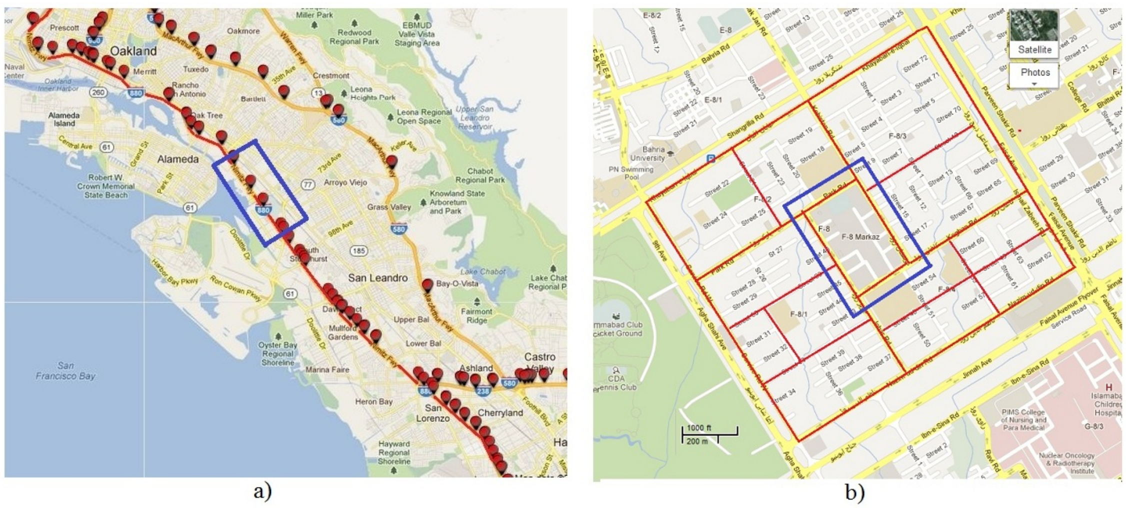


Figure 4.1: Road Maps: a) Highway: Big Area (Red Line- 11.5 km of road), Small area (Blue Box- 2 km of road) b) Urban: Big Area (Red Lines- 12.9 km of road), Small area (Blue Box- 1.8 km of road)

4.3 Simulation & Results

4.3.1 Simulation Environment

For realistic analysis of the proposed algorithms, we used a rational representation of vehicle mobility based on the accurate microscopic mobility modeling, real-world road topology and real-data based traffic demand modeling for both highway and urban environments. SUMO (Simulation of Urban Mobility) [18] is used to simulate the microscopic mobility of vehicles. SUMO is an open-source, space-continuous, discrete-time traffic simulator developed by the German Aerospace Center, capable of modeling the behavior of individual drivers. The path of each driver is determined based on the origin/destination matrix provided as an input to the simulator. The input of SUMO is determined for different scenarios at low and high density in small and big areas for both highway and urban environments as detailed next.

Highway Simulation

We used Performance Measurement System (PeMS) data to create realistic vehicle simulation for the highway. PeMS is developed by the department of Electrical Engineering and Computer Science at the University of California Berkeley in co-operation with the California Department of Transportation, California Partners for Advanced Transit and Highways, and Berkeley Transportation Systems [18]. The data is collected in real time from over 25,000 individual detectors. The system is deployed over all major metropolitan areas of the state of California. PeMS data provides information about the flow, speed and occupancy of the road. These data are then input to SUMO for a realistic flow of vehicles. For the purpose of our simulations, we downloaded the data of 419 road sensors at highway I880-S in Alameda County for both high traffic density, i.e. at $18:00$, and low traffic density, i.e. $01:00$, as shown in Fig. 4.1-a. The traffic density algorithms are then tested for small area (2 km road) and large area (11.5 km road) for both low and high density traffic. Other simulation parameters are given in Table 4.2.

Urban Simulation

We used one of the urban areas in Islamabad, Pakistan shown in Fig. 4.1-b. There are two types of traffic generated for Urban area.

- $TransitVehicles$: The destination of the vehicles is not inside the area that is vehicles pass through this area.

- $ArrivalVehicles$: The destination of the vehicles is inside the area. Vehicles enter the area and then after reaching their destination they stop and leave the network.

Vehicles entering the network follow Poisson distribution which is considered as a realistic model [61]. Vehicles randomly select a starting point and a destination. Destination can lie either within the area of interest ($ArrivalVehicles$) or outside the area ($TransitVehicles$). The vehicular density algorithms are then tested for small area (1.8 km of road in blue box, Fig. 4.1-b) and big areas (12.9 km of road, red lines, Fig. 4.1-b) for both low and high density traffic. Simulation parameters are given in Table 4.2.

4.3.2 Performance Metrics

The following performance metrics are used in the comparison of the density estimation algorithms:

Density Estimation is defined as the vehicular density estimated by the algorithm.

Convergence Time is defined as the time duration between the starting time and the convergence time of the algorithm.

Overhead is defined as the total number of messages transmitted over the network during the execution of the algorithm until it converges.

Error Ratio is defined as the ratio of the difference between an estimated value $Value_{estimated}$ and the actual value $Value_{actual}$

$$ErrorRatio = \frac{|Value_{estimated} - Value_{actual}|}{Value_{actual}}$$

Load on initiator is defined as the ratio of the total number of messages sent or received by the initiator ($Messages_{initiator}$) to the total number of messages sent over the network ($Messages_{network}$)

$$Load_{initiator} = \frac{Messages_{initiator}}{Messages_{network}}$$

In our simulations, *Sample & Colide* algorithm parameters T and L are set to 5 and 50 respectively, the *minHopsReporting* parameter of *Hop Sampling* is set to 2, and K and T parameters of *Gossip-based Aggregation* are set to 10 and 5 respectively.

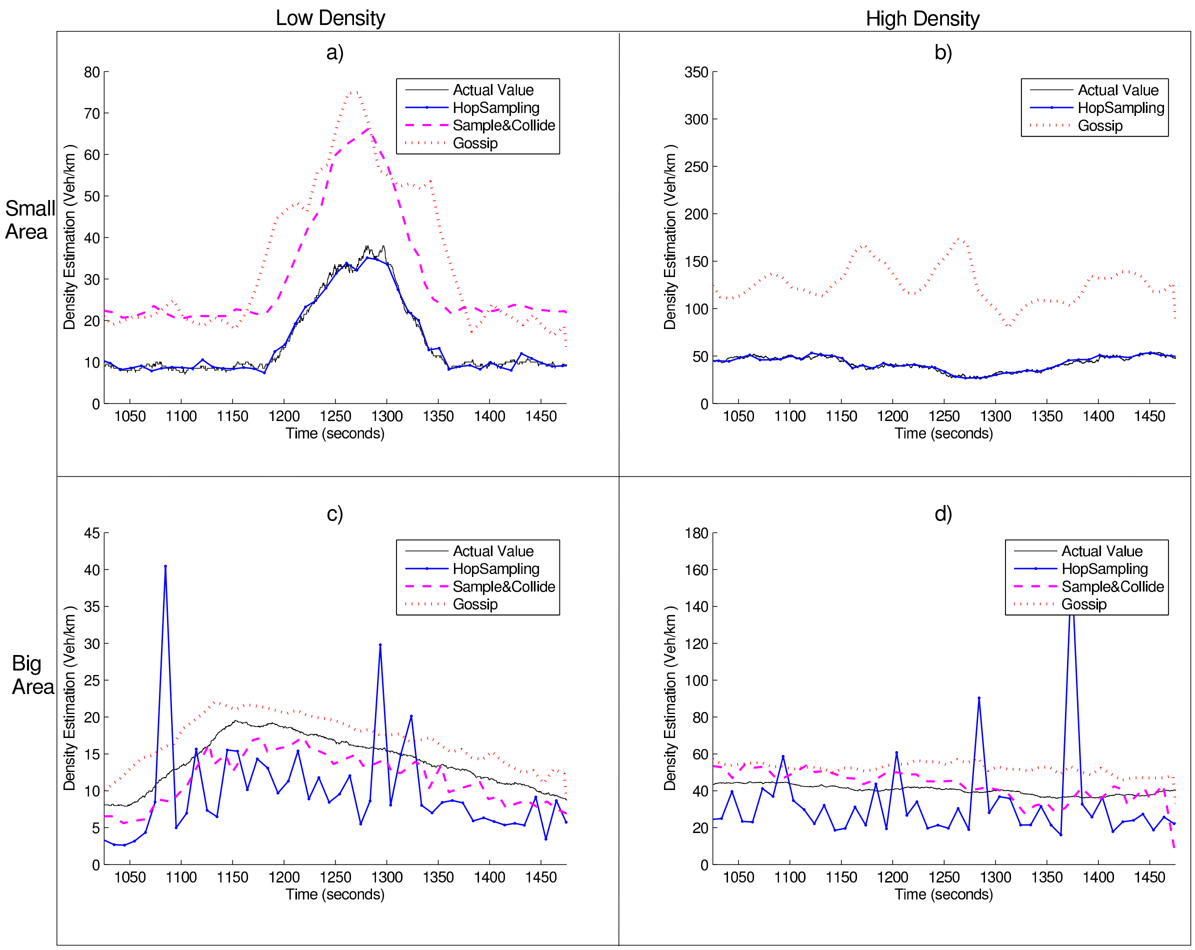


Figure 4.2: Density Estimation-Highway Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

4.3.3 Simulation Results

Figs. 4.2 and 4.3 show the estimated density values over time for the algorithms and the actual density at both low and high density traffic for different area sizes of highway and urban environment respectively. The density estimation of the Hop Sampling is very close to the actual value for both small and large areas of the urban road, and small areas of the highway. The main reason why the density estimation

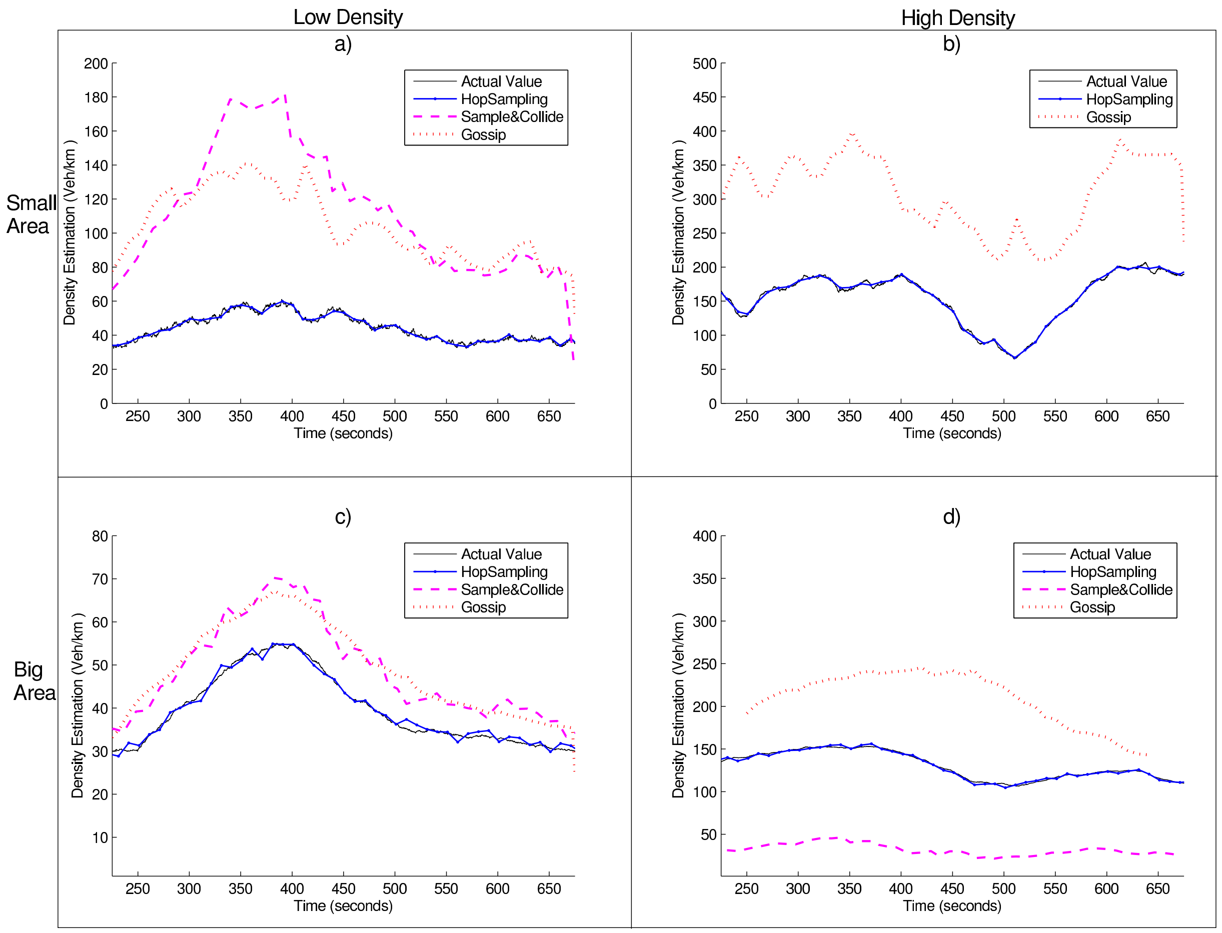


Figure 4.3: Density Estimation-Urban Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

is not very close to the actual value for Hop Sampling algorithm in large areas of the highway is that the accuracy of this algorithm decreases as the distance (the number of hops between the initiator vehicle and other vehicles) increases. Since we have a long stretch of straight highway, the vehicle at one end of the highway is farther away from the vehicles at the other end of the highway when compared to the urban scenario where there is a network of roads with multiple paths between the initiator vehicle and other vehicles which decreases the hop count values. Sample & Collide and Gossip-based aggregation perform worse than Hop Sampling because in a highly dynamic network like VANETs, connections are continuously made and broken, and vehicles are frequently entering and leaving the network. In Sample & Collide, when

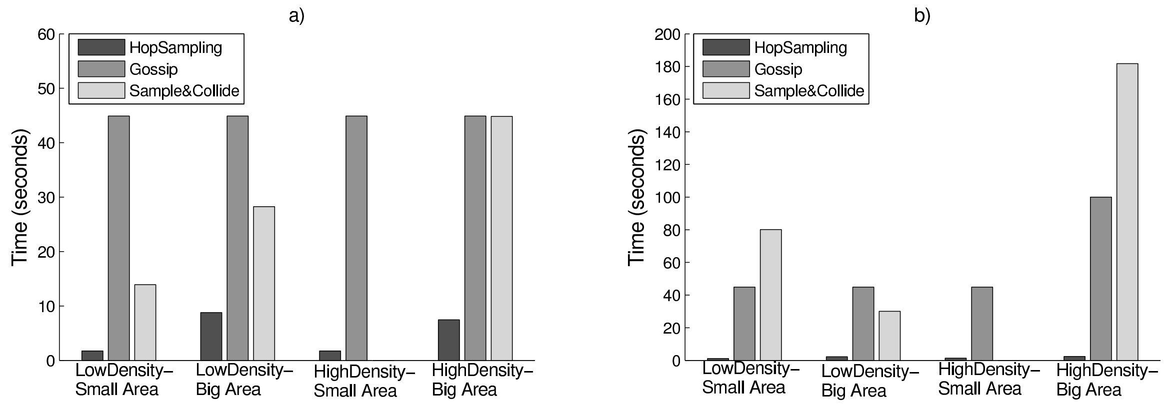


Figure 4.4: Convergence Time: Total time needed for the algorithms to converge for (a) Highway and (b) Urban scenarios

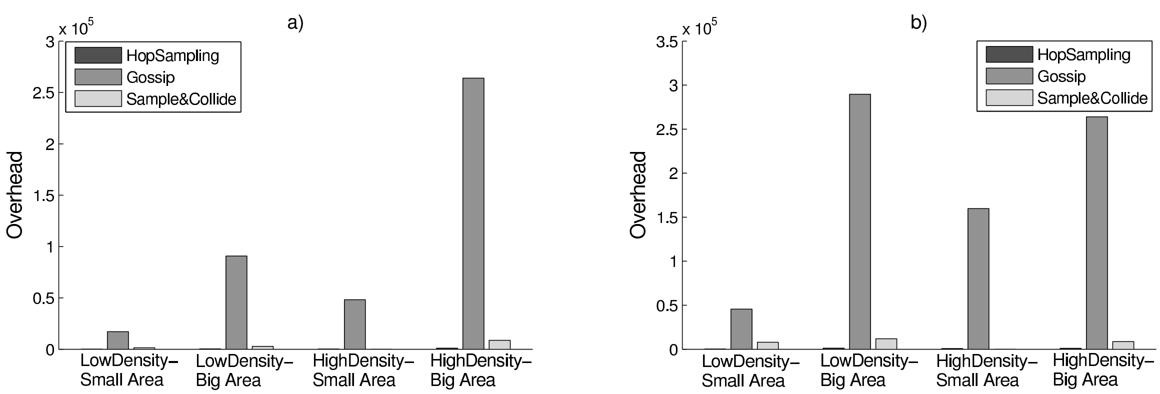


Figure 4.5: Overhead: Total number of messages sent for (a) Highway and (b) Urban scenarios

a vehicle which has already been sampled before leaves the network, the probability of selecting a sampled vehicle again decreases. Results for Sample & Collide in Fig. 4.2-b) and 4.3-b) are not included because in small area with high vehicle density, the sampled vehicle leave the network more quickly thus the algorithm converges in a very long time with inaccurate results. High mobility has a similar effect on Gossip-based aggregation. When a vehicle which is part of gossiping leaves the network, important information is lost with the vehicle. The average weight of the system, which should

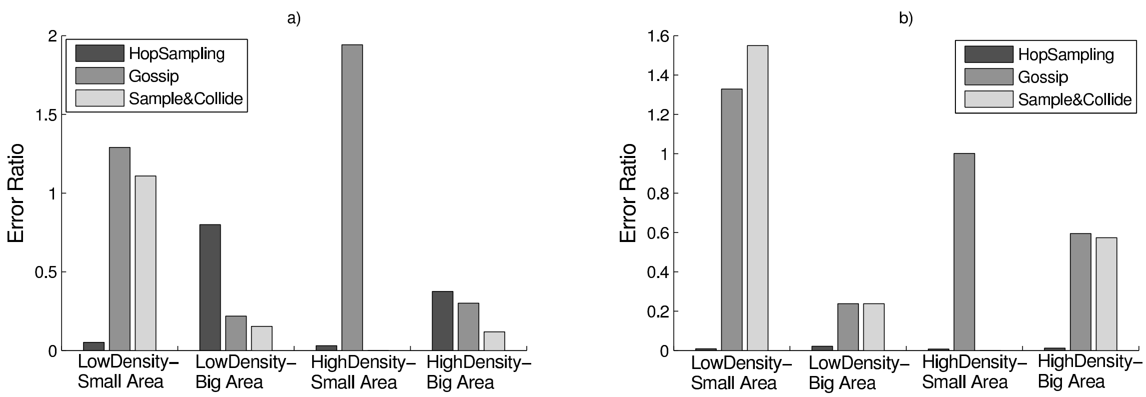


Figure 4.6: Error Ratio: (a) Highway (b) Urban scenarios

be equal to K, becomes less than K thus the estimated value is always more than the actual value. However, Hop Sampling is the most suitable algorithm in terms of accurately estimating the vehicle density.

Fig. 4.4 shows the convergence time of the algorithms under all the traffic scenarios. Hop Sampling takes the least amount of time to converge when compared to other algorithms, with convergence time usually less than 10 seconds.

Fig. 4.5 shows the overhead of different algorithms under all the traffic scenarios. Hop Sampling has the least overhead on the network followed by Sample & Collide and Gossip-based aggregation algorithms.

Fig. 4.6 shows the error ratio of the algorithms under all the traffic scenarios. Hop Sampling has the least error ratio except for Highway big area scenarios where the distance or the number of hops between the initiator and other vehicles increases, thus decreasing the efficient of the algorithm.

Fig. 4.7 shows the load on the initiator for running the density algorithms. Hop Sampling has the highest load on the initiator because once the initiator starts the algorithm, all the nodes reply back to the initiator with some probability. Thus, the initiator has to constantly receive messages from other nodes to accurately estimate the size of the network.

From the results, it can be concluded that the Hop Sampling performs better than the other algorithms for density estimation under different traffic scenarios. Hop Sampling provides the highest accuracy with the least overhead and convergence time. However, this comes at the cost of higher load on the initiator.

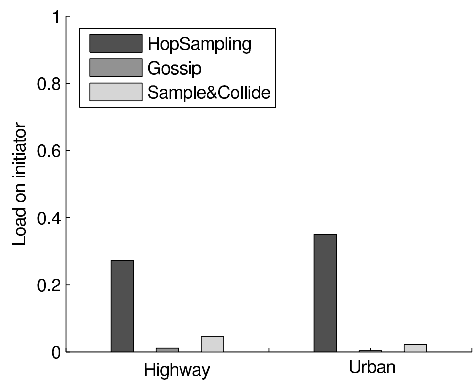


Figure 4.7: Load on the initiator: Highway and Urban scenarios

Chapter 5

$C_{LU}S_{AMPLING}$: DISTRIBUTED ALGORITHM FOR DENSITY ESTIMATION

In this chapter, we propose a completely distributed and infrastructure free density estimation algorithm *CluSampling* specifically tailored for highly dynamic vehicular networks.

5.1 Related Work

The existing methods for density estimation in VANETs can be broadly divided into two main categories: (1) Infrastructure-based and (2) Infrastructure-free. A summary of these methods are given in Table 5.1.

In the infrastructure-based methods, dedicated infrastructure such as loop detectors, roadside sensors or cameras are used to determine the presence of the vehicles on the road [52, 54, 53]. Road side camera images are used for traffic monitoring and density estimation in [52]. Using Kalman filter-based background estimation, the difference between the incoming image and the calculated background is used to mark vehicles and then to estimate the density of vehicles on the road. A similar approach using data fusion has been proposed in [53] in which the flow measured from video cameras on the road and travel time measured from GPS are used to estimate the density of vehicles. A neural network technique is applied on the data collected using video monitoring system to estimate the density of vehicles in [54]. A modified extended Kalman filter-based approach for density estimation is presented in [62]. In [63], commulative road acoustics were used in estimating road traffic density and the impact of noise on the estimation. In [64], Lagrangian state estimator-based approach for density estimation has been proposed. The main drawback of such infrastructure-based approach is that it needs large investment from government and other agencies to build, maintain and manage a huge infrastructure. Infrastructure based approach is rigid, difficult to maintain and upgrade, and the information is also limited to only those roads where infrastructure is deployed.

In the infrastructure-free methods, vehicles co-operate with each other to estimate the size of the network. Vehicles communicate with each other and share information to compute the density vehicles on the road. Different approaches have been proposed in the literature. In [22, 20, 23, 24], a probe vehicle uses local information from its neighbors to calculate the local density which in then used to give an estimate for global density. [22, 20, 24] assumes that inter-vehicular distances are exponentially distributed and based on this property, global vehicle density can be estimated using the local density information. However, their validation in based on simplistic vehicle mobility scenarios where traffic is randomly generated [20, 24] or vehicles move in free-flow condition independent of other vehicles around it [22, 20].

In [21], location based grouping scheme is used for density estimation. Vehicles are divided into groups with a group leader. Group leader is responsible for collecting information for its group and calculate the density of vehicles in a group. This approach suffers from the overhead of group formation. It also requires that the group size is fixed making it unsuitable for scenarios where vehicles density changes frequently over time.

Previous work also has limited focus on a realistic traffic scenario for both vehicle mobility and road traffic conditions which can have a great impact on the performance of the proposed algorithms. In [21], only city traffic conditions are taken into account and simulation is performed on a small straight road section of 2500m. In [20, 55, 57], only highway scenarios are used for simulation. In [57] a straight one way and single lane highway road is used for simulation. In [56], both highway and urban scenarios are used but validation of algorithm is limited to highway of size 2000m and small urban road with only one intersection. In [22, 20, 57, 24], realistic vehicular mobility is not taken into account while computing the road traffic.

It is also important that the validity of an algorithms is done using different performance metrics. In [20], mean of 10,000 trials and in [22], mean of 100,000 trials is used to calculate the mean absolute error without giving any information about the deviation in the computed results. In [57], only density information is used for comparison without giving any information about the error, convergence time and overhead of the proposed algorithm.

In [65] three fully distributed algorithms inspired from system size estimation techniques from peer-to-peer networks has been used. Algorithms are validated using realistic mobility at different traffic densities and area sizes for both highway and urban scenario.

In this work, we propose fully distributed and infrastructure-free algorithm *CluSampling* for the density estimation in VANETs. Unlike previous distributed approaches which either use local information [22, 20, 23, 24], or rely on vehicle speed, acceleration and flow information to calculate density of vehicles [55, 23, 57], *CluSampling* use network wide information using clustering and sampling technique to estimate the density of vehicles on the road. We have also rigorously validated algorithm across Highway and Urban scenarios using a realistic data-set used for microscopic vehicle mobility and traffic generation. We have used different traffic densities and different road sizes for the validation of the algorithm. We compared *CluSampling* with previously proposed four fully distributed algorithms and tested these algorithms across different performance metrics like convergence time, overhead on network, percentage error and load on initiator.

Table 5.1: Related Work on Vehicle Density Estimation in VANETs

Ref.	Infrastructure used	Category	Method used	Realistic Mobility	Validation Scenarios	Others
[52]	Yes	Centralized	Road-side cameras	Yes	Urban only	High deployment cost
[54]	Yes	Centralized	Neural networks	Yes	Urban only	High deployment cost
[53]	Yes	Centralized	Road-side cameras	Yes	Urban only	High Deployment cost
[62]	Yes	Centralized	Extended Kalman Filter	Yes	Highway only	High Deployment cost
[63]	Yes	Centralized	Roadside Acoustic signals used	Yes	Urban only	High Deployment cost, single point of failure
[64]	Yes	Centralized	Lagrangian state estimation based approach	Yes	Highway only	High Deployment cost, single point of failure
[22]	No	Distributed	Local density used for global density estimation	No	Small straight road	Assumption inter- vehicular spacing exponentially distributed
[20]	No	Clustering	Extension of [22] by using clusters	No	Highway only	Assumption inter- vehicular spacing exponentially distributed
[21]	No	Distributed	Location based group formation	Yes	Small straight 2500m road	Group formation overhead. Fixed group size
[55]	No	Distributed	Traffic-flow model using vehicle's speed and flow	Yes	Highway only	Local density estimation
[56]	No	Distributed	Random sampling of vehicles	Yes	Small Highway & Urban road	Directional information needed
[23]	No	Distributed	Vehicle's speed and acceleration information	Yes	Small section of road used	Local density estimation
[57]	No	Distributed	Fluid dynamics and car follow model	No	Highway only	One-way single lane road used
[24]	No	Distributed	Local density used to estimate global density	No	Single straight road	Inter-vehicular spacing exponentially distributed. Randomly generated traffic
[65]	No	Distributed	System size estimation technique	Yes	Highway & Urban road	Validated for low and high density traffic

5.2 CluSampling

CluSampling is a fully distributed infrastructure-free algorithm to estimate the density of vehicles using *Clu*stering and *Sampling* technique. Algorithm can be divided into two main parts: *Clustering* and *Sampling*. During *Clustering*, vehicles in the network are divided into geographical clusters based on the density of vehicles on the road. Once the clustering is completed and a cluster head is selected, each vehicle become member of a cluster depending on its geographical location. Within each cluster, *Sampling* technique is used to estimate the density of vehicles on the road. Small fraction of vehicles reply to cluster head with information about the local density of vehicles on the road which is then used to estimate the global density.

For ease of description, we assume that each vehicle in the network is equipped with communication device (IEEE 802.11p interface) and a global positioning system device. Target area is defined as the geographical region where we are interested in estimating the vehicle traffic density. We assume that each vehicle has complete map of the road within the target area and has a nominal transmission range R.

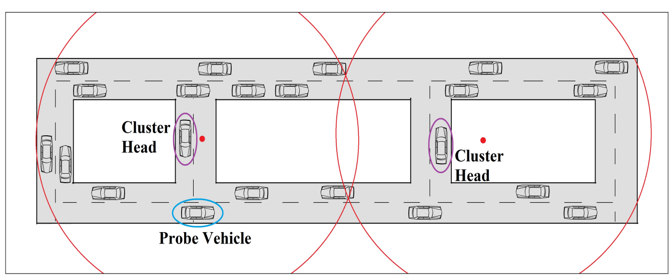


Figure 5.1: Vehicles grouped into clusters

5.2.1 Clustering

Fig.5.1 shows an example of the vehicles grouped into different clusters. The density algorithm is started by a probe vehicle which is interested in estimating the density of vehicle in the target area. The algorithm is explained below:

Algorithm 4 Calculating Total Number of Clusters

Input: T, V_{probe}, R
Output: C

1: $Neg_{probe} = getNeighbors(V_{probe}, R)$;
2: $L_{probe} = getRoadLength(V_{probe}, R)$;
3: $\sigma_{local} = \frac{Neg_{probe}}{RoadLength_{probe}}$;
4: $N_{estimate} = \sigma_{local} \times L_{TargetArea}$;
5: **if** $N_{estimate} \leq T$ **then**
6: $\quad C = 1$;
7: **else**
8: $\quad C = \frac{N_{estimate}}{T}$;
9: **end if**

- Probe vehicle is responsible for dividing the target area into clusters. Cluster size is determined depending on the estimate for the number of vehicles within the cluster. Algo.4 shows how the total number of clusters for a target area is determined. The input to the algorithms are T, V_{probe} and R, where T is the threshold for the total number of vehicles that can be part of cluster, V_{probe} is the probe vehicle and R is the transmission range of the vehicles in the cluster. Since probe vehicle has limited knowledge about the number of vehicles in the target area, it uses its local density (line 1-4 in Algo.4) to postulate the number of vehicles within the target area. Probe vehicle V_{probe} determines total number of cluster C as shown in line 5-9.

- Once C is calculated by probe vehicle, it uses this information to divide area into different regions such that the total estimated number of vehicles within a cluster is approximately equal to threshold T. Load on the cluster head can be managed by changing the value of threshold T.

- Once the target area is divided into different clusters, probe vehicle broadcasts the coordinates of the center of the each cluster to all the nodes in the network.

- Each vehicle calculates its distance from the cluster centers and becomes member of the cluster whose center is closest to the vehicle.

- Vehicle which is closest to the cluster center is selected as cluster head as shown in Fig.5.1. We assume that each vehicle has information about the locations of all its neighboring vehicles. This information can be exploited by vehicles to select a cluster head as shown in Algo.5. Input to the algorithm are N_c and C_{center} which is the total number of vehicles in a cluster and the cluster center coordinates respectively. For all the vehicles in the network, if distance of vehicle V_i to the cluster center D_{center} is less than all its neighbors, it broadcasts message containing information about its coordinates to all the nodes in the network, asking to become the cluster head. Other vehicles on receiving the message compares their distance to cluster center with distance of the vehicle V_i. If a vehicle is closer to cluster center compared with V_i, it sends $ConflictMessage$ to the V_i. If V_i do not receive any $ConflictMessage$ within a certain time period t, it is selected as cluster head.

Algorithm 5 Selecting Cluster Head in a Cluster

Input: N_c, C_{center}
Output: $V_{CluHead}$

for $V_i = 1 : N_c$ **do**
    $D_{center} = CalDis(C_{center});$
    $D.Neg_{center} = CalDisNeg(C_{center});$
    if $D_{center} \leq min(D.Neg_{center})$ **then**
        $BroadCastMsg(D_{center});$
    end if
    $Wait(t);$
    if $ReceiveConflictMessage = False$ **then**
        $V_{CluHead} = V_i;$
    end if
end for

5.2.2 Sampling

Sampling process is started by cluster head. Cluster head broadcasts starter message to all the nodes within a cluster. Each node replies back to the cluster head with a small probability p, independent of other vehicles, sending information about its local density σ_{local}. A vehicle calculate its local density as $\sigma_{local} = \frac{N}{L}$, where N and L are total number of vehicles and total length of road within its transmission range respectively.

5.2.3 Density Estimation

Cluster head estimates the density of vehicles within a cluster as $\sigma_{cluster} = \frac{\sum_{i=1}^{n} \sigma_i}{n}$, where σ_i and n are the local density of i_{th} vehicle and the total number of vehicles that reply to cluster head respectively.

Cluster head sends information about density of vehicles within a cluster $\sigma_{cluster}$ to the probe vehicle. Probe vehicle calculate global density as $\sigma_{global} = \frac{\sum_{c=1}^{C} \sigma_c}{C}$, where σ_c and C are the density of vehicles in c_{th} cluster and total number of clusters respectively.

5.3 Comparison Algorithms

To compare CluSampling with other density estimation algorithms, we implemented algorithms from two different categories of traffic density estimation i.e. using system size to estimate density of vehicles and using local neighbor information to estimate global density.

5.3.1 System Size for Global Vehicle Density Estimation

Inspired by the mechanisms for system size estimation in P2P networks, we implemented three fully distributed algorithms, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation, for vehicular density estimation [65]. The algorithms are used in calculating the number of vehicles within a particular geographical region specified by the Global Positioning System (GPS) coordinates. Once we calculate the network size (number of vehicles) within a particular geographical region, we divide the network size by the length of the roads in that area to estimate the density of vehicles. Details of the algorithms are given next.

Sample & Collide

Sample & Collide algorithm is based on uniformly sampling the nodes from a population, and then estimating the system size depending on how many samples of the nodes are collected, before an already sampled node is re-selected [58].

The approach is built upon the *inverted birthday paradox*. The detail of this algorithm has been previously explained in section 4.2.1.

Hop Sampling

Hop Sampling algorithm is based on the principle of probabilistic polling [59]. The initiator node spreads a message to all the nodes in the network using gossiping. The nodes reply back to the initiator probabilistically depending on their distance from it. Based on the replies that the initiator node gets from other nodes in the network, it estimates the size of the network. The detail of this algorithm has been previously explained in section 4.2.2.

Gossip-based Aggregation

Gossip-based aggregation algorithm has been proposed for large-scale overlay networks, where each peer periodically exchanges information with one of its neighbours picked at random to estimate the size of the network [60]. In this study, gossip-based aggregation algorithm has been adapted for dynamic VANETs. The detail of this algorithm has been previously explained in section 4.2.3.

5.3.2 Local Information for Global Vehicle Density Estimation

In local information based density estimation algorithms, probe vehicle use local information to give an estimate for the global density. This approach has been extensively used for density estimation is VANETs [22, 20, 23, 24]. We implemented *Two-Hop-Neighbor Scheme* for density estimation proposed in [20]. In *Two-Hop-Neighbor Scheme*, probe vehicle not only uses its neighbors information, but also takes into account the number of neighbors of its neighboring vehicles when estimating the density of vehicles on the road.

Lets consider probe vehicle P and the vehicle U, where U is the farthest vehicle within the transmission range of P as shown in Fig.5.2. R is the transmission range of the vehicles and X_u is the distance of U from P. It is important to note that we only consider neighbors which are behind the probe vehicle and their relative positions can be obtained using positioning techniques. Lets N_p be the number of neighbors behind vehicle P and N_u be the number of neighbors behind vehicle U. Thus from Fig.5.2, it can be seen that total number of vehicles are $N_p + N_u$ within the distance of length $X_u + R$. Thus density is calculated as $\sigma = \dfrac{N_u + N_p}{X_u + R}$.

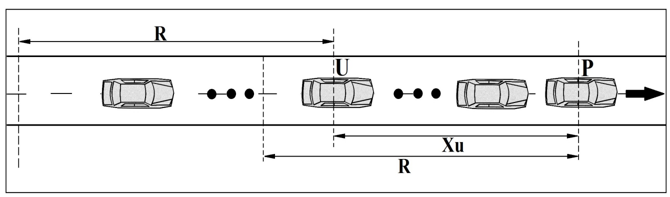


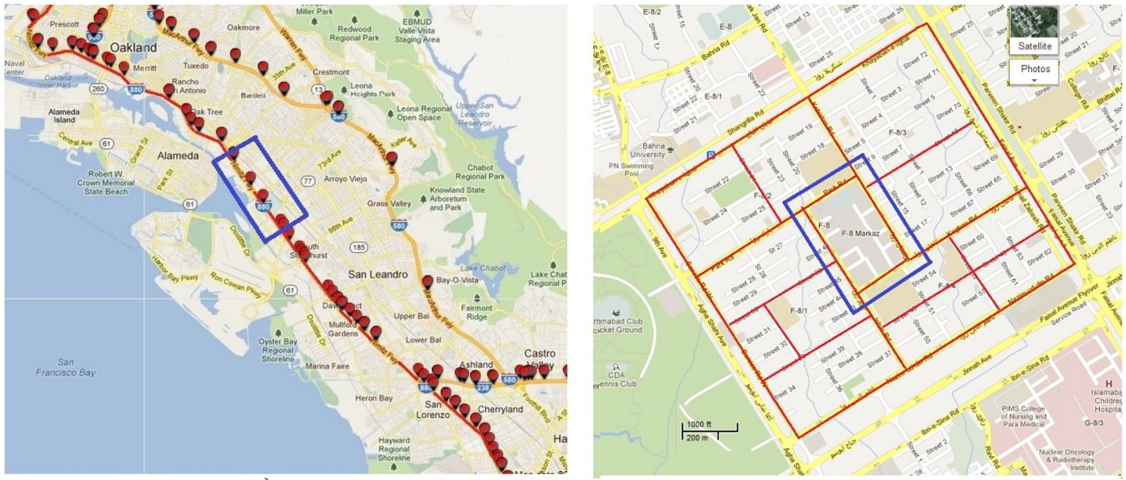
Figure 5.2: Two-Hop-Neighbor Scheme: Using local density to estimate global density

Table 5.2: Parameters for Highway and Urban Scenarios

	Length of roads (km)	Density	No. of Vehicles	Average Speed (km/h)	Maximum Speed (km/h)	Acce-leration (m/s^2)
High-way	Small (2km) &	Low	903	104	120	2
	Big (11.5km)	High	4436	102	120	2
Urban	Small (1.8km) &	Low	1566	52	60	2
	Big (12.9km)	High	4666	47	60	2

5.4 Simulation Environment

For realistic analysis of the proposed algorithm, we used a rational representation of vehicle mobility based on the accurate microscopic mobility modeling, real-world



a) b)

Figure 5.3: Road Maps: a) Highway: Big Area (Red Line- 11.5 km of road), Small area (Blue Box- 2 km of road) b) Urban: Big Area (Red Lines- 12.9 km of road), Small area (Blue Box- 1.8 km of road)

road topology and real-data based traffic demand modeling for both highway and urban environments. SUMO (Simulation of Urban Mobility) [19] is used to simulate the microscopic mobility of vehicles. SUMO is an open-source, space-continuous, discrete-time traffic simulator developed by the German Aerospace Center, capable of modeling the behavior of individual drivers. The path of each driver is determined based on the origin/destination matrix provided as an input to the simulator. The input of SUMO is determined for different scenarios at low and high density in small and big areas for both highway and urban environments as detailed next.

5.4.1 Highway Simulation

We used Performance Measurement System (PeMS) data to create realistic vehicle simulation for the highway. PeMS is developed by the department of Electrical Engineering and Computer Science at the University of California Berkeley in co-operation with the California Department of Transportation, California Partners for Advanced Transit and Highways, and Berkeley Transportation Systems [18]. The data is collected in real time from over 25,000 individual detectors. The system is deployed over all major metropolitan areas of the state of California. PeMS data provides information about the flow, speed and occupancy of the road. These data are then input to SUMO for a realistic flow of vehicles. For the purpose of our simulations, we downloaded the data of 419 road sensors at highway I880-S in Alameda County for both high traffic density, i.e. at 18 : 00, and low traffic density, i.e. 01 : 00, as shown in Fig. 5.3-a. The traffic density algorithms are then tested for small area (2 km road) and large area (11.5 km road) for both low and high density traffic. Other simulation parameters are given in Table 5.2.

5.4.2 Urban Simulation

We used one of the urban areas in Islamabad, Pakistan shown in Fig. 5.3-b. There are two types of traffic generated for Urban area.

- $TransitVehicles$: The destination of the vehicles is not inside the area that is vehicles pass through this area.
- $ArrivalVehicles$: The destination of the vehicles is inside the area. Vehicles enter the area and then after reaching their destination they stop and leave the network.

Vehicles entering the network follow Poisson distribution which is considered as a realistic model [61]. Vehicles randomly select a starting point and a destination. Destination can lie either within the area of interest ($ArrivalVehicles$) or outside the area ($TransitVehicles$). The vehicular density algorithms are then tested for small area (1.8 km of road in blue box, Fig. 5.3-b) and big areas (12.9 km of road, red lines, Fig. 5.3-b) for both low and high density traffic. Simulation parameters are given in Table 5.2.

5.4.3 Performance Metrics

The following performance metrics are used in the comparison of the density estimation algorithms:

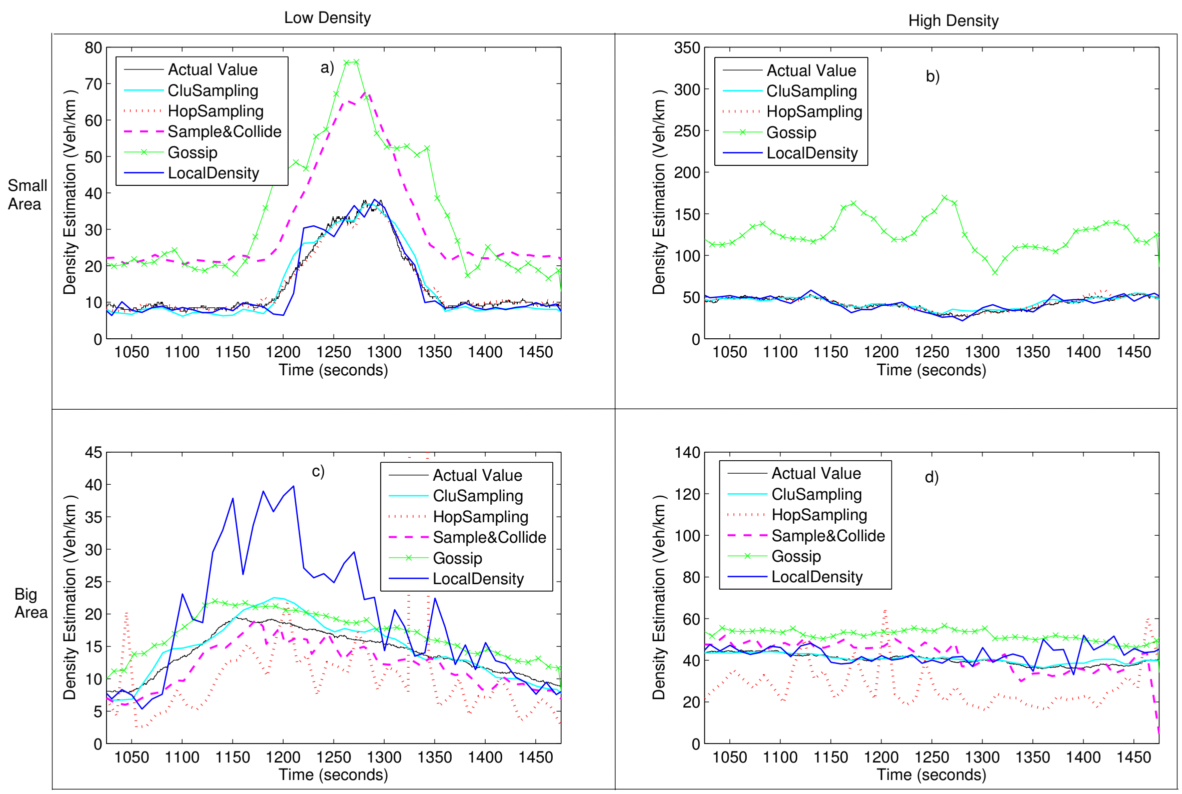


Figure 5.4: Density Estimation-Highway Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

Density Estimation

is the vehicular density estimated by the algorithm.

Convergence Time

is the time duration between the starting time and the convergence time of the algorithm.

Overhead

is the total number of messages transmitted over the network during the execution of the algorithm until it converges.

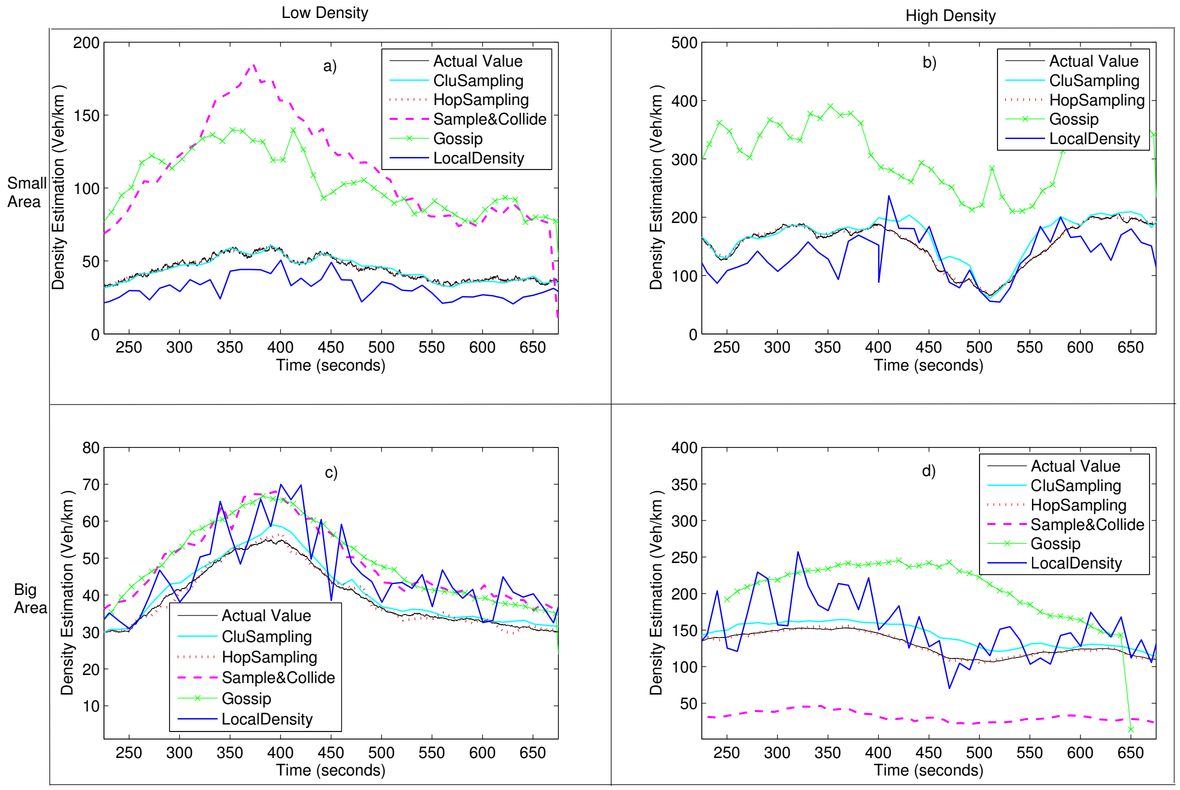


Figure 5.5: Density Estimation-Urban Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

Percentage Error

is the percentage error of the difference between an estimated value $Value_{estimated}$ and the actual value $Value_{actual}$

$$PercentageError = \frac{|Value_{estimated} - Value_{actual}|}{Value_{actual}} \times 100\%$$

Load on Initiator

is the ratio of the total number of messages sent or received by the initiator ($Messages_{initiator}$) to the total number of messages sent over the network ($Messages_{network}$)

$$Load_{initiator} = \frac{Messages_{initiator}}{Messages_{network}}$$

5.5 Simulation Results

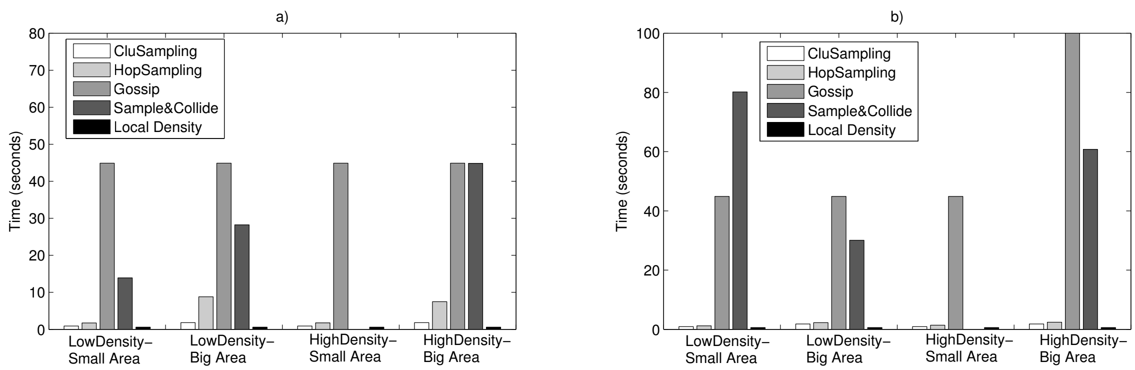


Figure 5.6: Convergence Time: Total time needed for the algorithms to converge for (a) Highway and (b) Urban scenarios

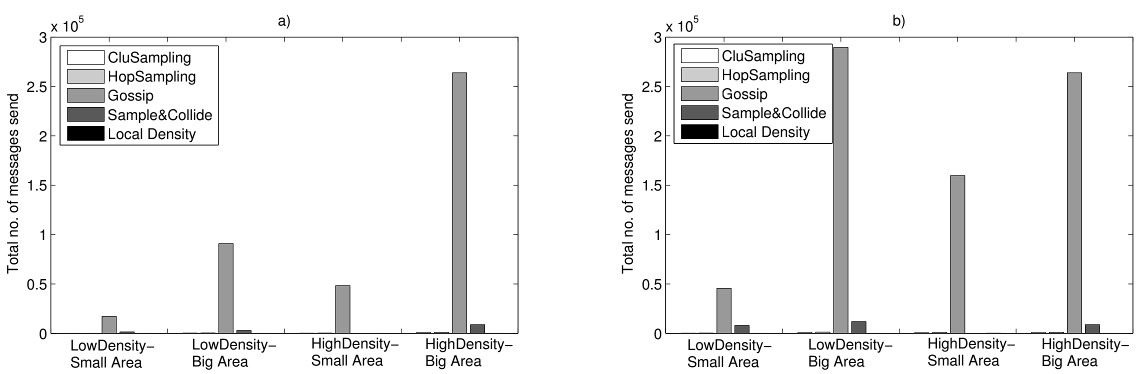


Figure 5.7: Overhead: Total number of messages sent for (a) Highway and (b) Urban scenarios

Figs. 5.4 and 5.5 show the estimated density values over time for the algorithms and the actual density at both low and high density traffic for different area sizes of highway and urban environment respectively.

Sample & Collide and *Gossip-based aggregation* perform worst when compared with other algorithms because in a highly dynamic network like VANETs, connections are continuously made and broken, and vehicles are frequently entering and leaving the network. In Sample & Collide, when a vehicle which has already been sampled before

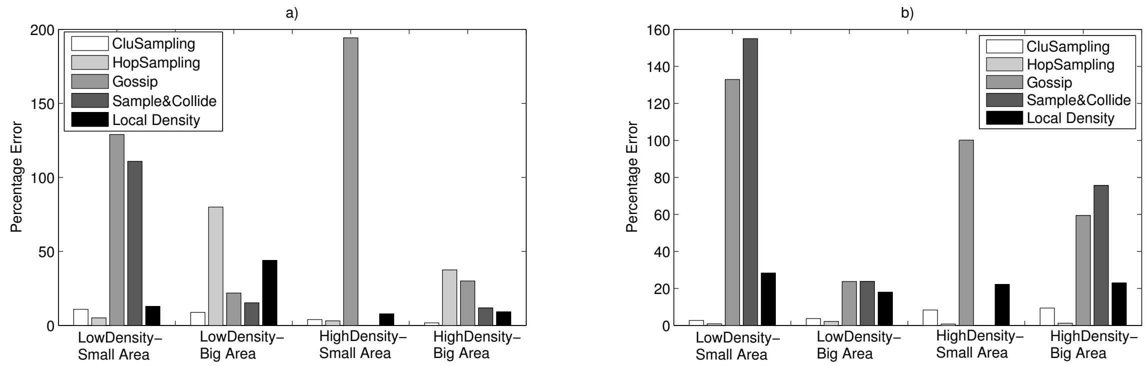


Figure 5.8: Percentage Error: (a) Highway and (b) Urban scenarios

leaves the network, the probability of selecting a sampled vehicle again decreases. Results for Sample & Collide in Fig. 5.4-b) and 5.5-b) are not included because in small area with high vehicle density, the sampled vehicle leave the network more quickly thus the algorithm convergence time is high with inaccurate results. High mobility has a similar effect on Gossip-based aggregation. When a vehicle which is part of gossiping leaves the network, important information is lost with the vehicle. The average weight of the system, which should be equal to K, becomes less than K thus the estimated value is always more than the actual value.

The density estimation by *Hop Sampling* is very close to the actual value for both small and large areas of the urban road, and small areas of the highway. However, it does not perform good in large areas of highway. The main reason why the density estimation is not very close to the actual value for Hop Sampling algorithm in large areas of the highway is that the accuracy of this algorithm decreases as the distance (i.e. number of hops between the initiator vehicle and other vehicles) increases. Since we have a long stretch of straight highway, the vehicle at one end of the highway is farther away from the vehicles at the other end of the highway when compared to the urban scenario where there is a network of roads with multiple paths between the initiator vehicle and other vehicles which decreases the hop count values.

Local Density based algorithm performs better only in the high density highway scenario and low density small highway. It performs better since in high density highway scenario, vehicles are more uniformly distributed since roads are straight and there are no obstacles that disrupts the uniformity of traffic on the road. Thus local density gives a good estimate for global density. However when the density of vehicles on the road is low and its big area, vehicles are not uniformly distributed thus giving inaccurate results. In urban scenario, *Local Density* based algorithm completely fails to get accurate estimation for vehicle density. This is because in Urban scenario, vehicles are not uniformly distributed because of obstacles and different traffic densities at different roads at the intersections of the roads. This disrupts the uniform flow of traffic and local density fails to predict the global density of vehicles on the road.

CluSampling performs the best in different traffic scenarios for both highway and urban roads. This is because *CluSampling* is more robust and mobility of vehicles has limited effect on the performance of the algorithm. Clustering divides the target area into smaller manageable cluster and thus density can be more accurately estimated for each cluster. Since cluster size is determined by the number of vehicles on the road, different traffic densities have limited effect on performance of the algorithm. Large number of vehicles mean more clusters thus dividing the tasks into smaller manageable clusters.

Fig. 5.6 shows the convergence time of the algorithms under all the traffic scenarios. *CluSampling*, *Hop Sampling* and *Local Density* takes the least amount of time to converge when compared to other algorithms, with convergence time usually less than 10 seconds.

Fig. 5.7 shows the overhead of different algorithms under all the traffic scenarios. *CluSampling*, *Hop Sampling* and *Local Density* has the least overhead on the network followed by Sample & Collide and Gossip-based aggregation algorithms.

Fig. 5.8 shows the percentage error of the algorithms under all the traffic scenarios. *CluSampling* has small percentage error in all traffic scenarios and its is usually below 10%. *HopSampling* also has low percentage error but for Highway big area scenarios where the distance or the number of hops between the initiator and other vehicles increases, efficiency of the algorithm decreases drastically. *Local Based* algorithm also gives poor results for Urban roads and low density big highway road.

Fig. 5.9 shows the load on the initiator for running the density algorithms. Hop Sampling has the highest load on the initiator because once the initiator starts the algorithm, all the nodes reply back to the initiator with some probability. Thus, the initiator has to constantly receive messages from other nodes to accurately estimate the size of the network. However, other algorithms has limited load on the initiator vehicle.

From the results, it can be concluded that the *CluSampling* performs better than the other algorithms for density estimation under different traffic scenarios. *CluSampling* provides good accuracy and has less overhead, convergence time and load on the initiator or probe vehicle.

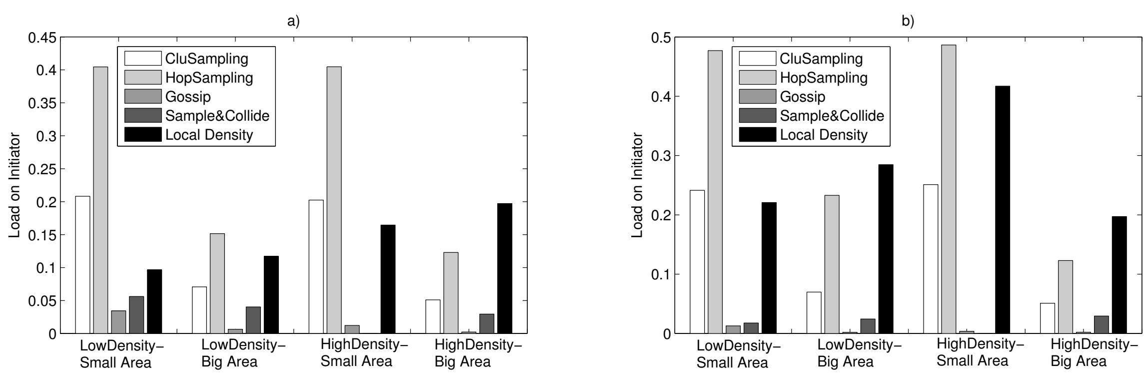


Figure 5.9: Load on the initiator: a) Highway and b) Urban scenarios

Chapter 6

CONCLUSION

In the first part of our work, we analyze the spatial and temporal evolution of the VANET topology characteristics by using both realistic large-scale mobility traces and realistic channel models. The realistic large-scale mobility traces are obtained by using accurate microscopic mobility modeling of SUMO, determining its input and parameters based on the vehicle flow and speed data extracted using the PeMS database. The realistic channel model is obtained by implementing a recently proposed obstacle-based channel model that takes all the vehicles around the transmitter and receiver into account in determining the received signal strength. The performance of the obstacle-based model is compared to the most commonly used more simplistic channel models including unit disc and log-normal shadowing models. The extensive investigation of the system metrics regarding the link characteristics over both time and space including node degree, neighbor distance distribution, number of clusters and link duration reveals that tuning the parameters appropriately and introducing time correlation for the Gaussian random variable in the log-normal model provides a good match with the more sophisticated and computationally expensive obstacle based model. We validate the consistency of the values of these parameters for various vehicle traffic densities over two different highways in California.

In the second part of our work, we first propose and analyze fully distributed and infrastructure-free mechanisms for vehicle density estimation in vehicular ad hoc networks. Inspired by the mechanisms for the system size estimation in P2P networks, we adapted and implemented three fully distributed algorithms, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation for VANETs. We also proposed a fully distributed infrastructure free density estimation algorithm *CluSampling* especially tailored for VANETs. These algorithms are then analyzed rigorously for validity and performance over eight traffic scenarios, including low and high density traffic, for different sizes of highway and urban environments, based on a realistic representation of the vehicle mobility, using accurate microscopic mobility modeling, real-world road topology and real-data based traffic demand modeling. The analysis demonstrates that *CluSampling* is more robust to changes in the network and provides high accuracy in least convergence time and introduces less overhead on the network and the initiator node. The good performance of these algorithms supports the usage of distributed approach in the density estimation in VANETs, instead of using infrastructure based solutions that suffers from limited coverage, high deployment and maintenance cost.

Future work for mobility and channel modeling can extend current work to realistic simulation model and matching mechanism for urban traffic scenarios. For density estimation, future work can involve incorporating the effect of background traffic on the efficiency of the algorithm. The tradeoff between the accuracy of estimation and the network load can be investigated.

BIBLIOGRAPHY

[1] R. Chen, W.-L. Jin, and A. Regan, "Broadcasting safety information in vehicular networks: issues and approaches," *IEEE Network*, vol. 24, no. 1, pp. 20–25, Jan-Feb 2010.

[2] W. Chen and S. Cai, "Ad hoc peer-to-peer network architecture for vehicle safety communications," *IEEE Communications Magazine*, vol. 43, no. 4, pp. 100 – 107, April 2005.

[3] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-vehicle communication: Fair transmit power control for safety- critical information," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3684 – 3703, September 2009.

[4] J. Nzouonta, N. Rajgure, G. Wang, and C. Borcea, "Vanet routing on city roads using real-time vehicular traffic information," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 7, pp. 3609 – 3626, September 2009.

[5] Y. Toor, P. Muhlethaler, and A. Laouiti, "Vehicle ad hoc networks: applications and related technical issues," *IEEE Communications Surveys Tutorials*, vol. 10, no. 3, pp. 74 –88, Third Quarter 2008.

[6] S. A. H. Tabatabaei, M. Fleury, N. N. Qadri, and M. Ghanbari, "Improving propagation modeling in urban environments for vehicular ad hoc networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 705 – 716, September 2011.

[7] D. Dhoutaut, A. Régis, and F. Spies, "Impact of radio propagation models in

vehicular ad hoc networks simulations," in *Proceedings of the 3rd international workshop on Vehicular ad hoc networks (VANET)*, September 2006, pp. 40–49.

[8] S. Uppoor and M. Fiore, "Large-scale urban vehicular mobility for networking research," in *IEEE Vehicular Networking Conference (VNC)*, November 2011, pp. 62–69.

[9] G. Pallis, D. Katsaros, M. D. Dikaiados, N. Loulloudes, and L. Tassiulas, "On the structure and evolution of vehicular networks," in *Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS)*, September 2009, pp. 1–10.

[10] M. Boban, W. Viriyasitavat, and O. Tonguz, "Modeling vehicle-to-vehicle line of sight channels and its impact on application-layer performance," in *Proceedings of the tenth ACM international workshop on Vehicular inter-networking, systems, and applications*, ser. VANET '13. New York, NY, USA: ACM, 2013.

[11] W. Viriyasitavat, F. Bai, and O. K. Tonguz, "Dynamics of network connectivity in urban vehicular networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 515 – 533, March 2011.

[12] M. Fiore and J. Harri, "The networking shape of vehicular mobility," in *ACM International Symposium on Mobile Ad hoc Networking and Computing (MobiHoc)*, May 2008, pp. 261–272.

[13] R. Meireles, M. Ferreira, and J. Barros, "Vehicular connectivity models: From single-hop links to large-scale behavior," in *IEEE Vehicular Technology Conference (VTC) Fall*, September 2009, pp. 1–5.

[14] R. Protzmann, B. Schunemann, and I. Radusch, "The influences of communication models on the simulated effectiveness of v2x applications," *IEEE Communications Magazine*, vol. 49, no. 11, pp. 149 – 155, November 2011.

[15] S. C. Ng, W. Zhang, Y. Zhang, Y. Yang, and G. Mao, "Analysis of access and connectivity probabilities in vehicular relay networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 140 –150, January 2011.

[16] X. Jin, W. Su, and Y. Wei, "Quantitative analysis of the vanet connectivity: Theory and application," in *IEEE Vehicular Technology Conference (VTC) Spring*, May 2011, pp. 1–5.

[17] J. Gozalvez, M. Sepulcre, and R. Bauza, "Impact of the radio channel modelling on the performance of vanet communication protocols," *Telecommunication Systems*, pp. 1–19, December 2010.

[18] Performance measurement system (PeMS). [Online]. Available: http://pems.dot.ca.gov/

[19] SUMO - Simulation of Urban MObility. [Online]. Available: http://sumo.sourceforge.net

[20] Panichpapiboon, "Exploiting wireless communication in vehicle density estimation," *Vehicular Technology, IEEE Transactions on*, vol. 60, no. 6, pp. 2742 –2751, july 2011.

[21] M. Jerbi, S.-M. Senouci, T. Rasheed, and Y. Ghamri-Doudane, "An infrastructure-free traffic information system for vehicular networks," in *Vehicular Technology Conference (VTC) IEEE 66th*, oct 2007, pp. 2086 –2090.

[22] Panichpapiboon, "Evaluation of a neighbor-based vehicle density estimation scheme," in *ITS Telecommunications, ITST. 8th International Conference on*, oct. 2008, pp. 294 –298.

[23] R. Shirani, F. Hendessi, and T. Gulliver, "Store-carry-forward message dissemination in vehicular ad-hoc networks with local density estimation," in *Vehicular Technology Conference Fall (VTC 2009-Fall), IEEE 70th*, sept. 2009, pp. 1 –6.

[24] R. Mao and G. Mao, “Road traffic density estimation in vehicular networks,” in *Wireless Communications and Networking Conference (WCNC), 2013 IEEE*, 2013, pp. 4700–4705.

[25] J. Harri, F. Filali, and C. Bonnet, “Mobility models for vehicular ad hoc networks: A survey and taxonomy,” *IEEE Communications Surveys and Tutorials*, vol. 11, no. 4, pp. 19 – 41, Fourth Quarter 2009.

[26] Vissim. [Online]. Available: http://www.english.ptv.de/cgi-bin/traffic/traf vissim.pl

[27] H. Conceicao, L. Damas, M. Ferreira, and J. Barros, “Large scale simulation of v2v environments,” in *ACM Symposium on Applied Computing*, March 2008, pp. 28–33.

[28] B. Raney, A. Voellmy, N. Cetin, M. Vrtic, and K. Nagel, “Towards a microscopic traffic simulation of all of switzerland,” in *International Conference on Computational Science*, April 2002, pp. 371–380.

[29] J. Maurer, T. Fugen, M. Porebska, T. Zwick, and W. Wiesbeck, “A ray-optical channel model for mobile to mobile communications,” in *4th MCM COST 2100*, February 2008.

[30] C. F. Mecklenbrauker, A. F. Molisch, J. Karedal, F. Tufvesson, A. Paier, L. Bernado, T. Zemen, O. Klemp, and N. Czink, “Vehicular channel characterization and its implications for wireless system design and performance,” *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1189 – 1212, July 2011.

[31] J. Karedal, N. Czink, A. Paier, F. Tufvesson, and A. F. Molisch, “Path loss modeling for vehicle-to-vehicle communications,” *IEEE Transactions on Vehicular Technology*, vol. 60, no. 1, pp. 323 – 328, January 2011.

[32] L. Cheng, B. E. Henty, F. Bai, and D. D. Stancil, "Highway and rural propagation channel modeling for vehicle-to-vehicle communications at 5.9 ghz," in *IEEE Antennas and Propagation Society International Symposium*, July 2008, pp. 1–4.

[33] G. Setiawan, S. Iskandar, S. Kanhere, and K. chan Lan, "The effect of radio models on vehicular network simulations," October 2007.

[34] L. Cheng, B. E. Henty, D. D. Stancil, F. Bai, and P. Mudalige, "Mobile vehicle-to-vehicle narrow-band channel measurement and characterization of the 5.9 ghz dedicated short range communication (dsrc) frequency band," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1501 – 1516, October 2007.

[35] O. Renaudin, V.-M. Kolmonen, P. Vainikainen, and C. Oestges, "Non-stationary narrowband mimo inter-vehicle channel characterization in the 5-ghz band," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4, pp. 2007 – 2015, May 2010.

[36] M. Boban, T. T. V. Vinhoza, M. Ferreira, J. Barros, and O. K. Tonguz, "Impact of vehicles as obstacles in vehicular ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 1, pp. 15 – 28, January 2011.

[37] J. Karedal, F. Tufvesson, N. Czink, A. Paier, C. Dumard, T. Zemen, C. F. Mecklenbrauker, and A. F. Molisch, "A geometry-based stochastic mimo model for vehicle-to-vehicle communications," *IEEE Transactions on Wireless Communications*, vol. 8, no. 7, pp. 3646 – 3657, July 2009.

[38] J. Maurer, T. Fugen, and W. Wiesbeck, "Narrow-band measurement and analysis of the inter-vehicle transmission channel at 5.2 ghz," in *IEEE Vehicular Technology Conference (VTC) Spring*, May 2002, pp. 1274–1278.

[39] I. Sen and D. Matolak, "Vehicle-to-vehicle channel models for the 5-ghz band,"

IEEE Transactions on Intelligent Transportation Systems, vol. 9, no. 2, pp. 235 – 245, June 2008.

[40] Tapascologne project. [Online]. Available: http://sourceforge.net/apps/mediawiki/sumo/index.php?title=TAPASCologne

[41] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing (Mobihoc)*, May 2006, pp. 108–119.

[42] Drive-in (distributed routing and infotainment through vehicular inter-networking). [Online]. Available: http://drive-in.cmuportugal.org

[43] D. Krajzewicz, M. Bonert, and P. Wagner, "The open source traffic simulation package sumo," in *RoboCup Infrastructure Simulation Competition*, June 2006, pp. 371–380.

[44] J. Epstein and D. Peterson, "An experimental study of wave propagation at 850 mc," *Proceedings of the IRE*, vol. 41, no. 5, pp. 595 –611, May 1953.

[45] J. Deygout, "Multiple knife-edge diffraction of microwaves," *IEEE Transactions on Antennas and Propagation*, vol. 14, no. 4, pp. 480 – 489, July 1966.

[46] C. Giovaneli, "An analysis of simplified solutions for multiple knife-edge diffraction," *IEEE Transactions on Antennas and Propagation*, vol. 32, no. 3, pp. 297 – 301, March 1984.

[47] "Propagation by diffraction," *International Telecommunication Union Radiocommunications Sector, Geneva*, p. P.526, February 2007.

[48] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electronics Letters*, vol. 27, no. 23, pp. 2145–2146, 1991.

[49] J. Gozalvez, M. Sepulcre, and R. Bauza, "Impact of the radio channel modelling on the performance of vanet communication protocols," *Telecommunication Systems*, vol. 50, no. 3, pp. 149–167, 2012.

[50] M. Sepulcre and J. Gozalvez, "Adaptive wireless vehicular communication techniques under correlated radio channels," in *IEEE Vehicular Technology Conference (VTC) Spring*, April 2009, pp. 1–5.

[51] S. Szyszkowicz, H. Yanikomeroglu, and J. Thompson, "On the feasibility of wireless shadowing correlation models," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 9, pp. 4222–4236, November 2010.

[52] M. Balcilar and A. Sonmez, "Extracting vehicle density from background estimation using kalman filter," in *Computer and Information Sciences, 2008. ISCIS '08. 23rd International Symposium on*, oct. 2008, pp. 1 –5.

[53] R. Anand, L. Vanajakshi, and S. Subramanian, "Traffic density estimation under heterogeneous traffic conditions using data fusion," in *Intelligent Vehicles Symposium (IV), IEEE*, june 2011, pp. 31 –36.

[54] C. Ozkurt and F. Camci, "Automatic traffic density estimation and vehicle classification for traffic surveillance systems using neural networks," in *Mathematical and Computational Applications*, 2009.

[55] M. Artimy, "Local Density Estimation and Dynamic Transmission-Range Assignment in Vehicular Ad Hoc Networks," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 400–412, 2007.

[56] L. Garelli, C. Casetti, C.-F. Chiasserini, and M. Fiore, "Mobsampling: V2v communications for traffic density estimation," in *Vehicular Technology Conference (VTC Spring), 2011 IEEE 73rd*, may 2011, pp. 1 –5.

[57] T. Umer, Z. Ding, B. Honary, and H. Ahmad, "Implementation of microscopic parameters for density estimation of heterogeneous traffic flow for vanet," in *Communication Systems Networks and Digital Signal Processing (CSNDSP), 7th International Symposium on*, july 2010, pp. 66 –70.

[58] L. Massoulié, E. Le Merrer, A.-M. Kermarrec, and A. Ganesh, "Peer counting and sampling in overlay networks: random walk methods," in *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, ser. PODC '06. ACM, 2006, pp. 123–132.

[59] D. Kostoulas, D. Psaltoulis, I. Gupta, K. Birman, and A. Demers, "Decentralized schemes for size estimation in large and dynamic groups," in *Network Computing and Applications, Fourth IEEE International Symposium on*, july 2005, pp. 41 –48.

[60] M. Jelasity and A. Montresor, "Epidemic-style proactive aggregation in large overlay networks," in *Proceedings of the 24th International Conference on Distributed Computing Systems (ICDCS'04)*, ser. ICDCS '04. Washington, DC, USA: IEEE Computer Society, 2004, pp. 102–109. [Online]. Available: http://dl.acm.org/citation.cfm?id=977400.978026

[61] S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Analytical model for connectivity in vehicular ad hoc networks," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 6, pp. 3341 –3356, nov. 2008.

[62] C. van Hinsbergen, T. Schreiter, F. Zuurbier, J. W. C. Van Lint, and H. Van Zuylen, "Localized extended kalman filter for scalable real-time traffic state estimation," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 1, pp. 385–394, 2012.

[63] V. Tyagi, S. Kalyanaraman, and R. Krishnapuram, "Vehicular traffic density

state estimation based on cumulative road acoustics," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 3, pp. 1156–1166, 2012.

[64] Y. Yuan, J. W. C. Van Lint, R. Wilson, F. van Wageningen-Kessels, and S. Hoogendoorn, "Real-time lagrangian traffic state estimator for freeways," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 13, no. 1, pp. 59–70, 2012.

[65] N. Akhtar, S. Ergen, and O. Ozkasap, "Analysis of distributed algorithms for density estimation in vanets (poster)," in *Vehicular Networking Conference (VNC), 2012 IEEE*, 2012, pp. 157–164.

VITA

NABEEL AKHTAR was born in Bahawalpur, Pakistan on 25th June. He received his early education from Sadiq Public School, Bahawalpur. He did his B.Sc degree in Computer Science from Lahore University of Management Sciences (LUMS), Lahore, Pakistan in 2011. He joined M.Sc program in Computer Science and Engineering at Koc University in 2011 as a research and teaching assistant. During his study he worked on Channel Modeling, Mobility modeling and Distributed density estimation algorithms for Vehicular ad hoc Networks. He has co-authored two conference papers in IEEE Vehicular Networking Conference (VNC), 2012 and IEEE Wireless Communication and Networking Conference (WCNC), 2013. He also has two journal papers under submission.