Analysis of Distributed Algorithms for Density Estimation in VANETs (Poster)

Nabeel Akhtar*, Sinem Coleri Ergen† and Oznur Ozkasap*
*Department of Computer Engineering
†Department of Electrical and Electronics Engineering
Koc University, Turkey
Email: nakhtar@ku.edu.tr, sergen@ku.edu.tr, oozkasap@ku.edu.tr

***Abstract*—Vehicle density is an important system metric used in monitoring road traffic conditions. Most of the existing methods for vehicular density estimation require either building an infrastructure, such as pressure pads, inductive loop detector, roadside radar, cameras and wireless sensors, or using a centralized approach based on counting the number of vehicles in a particular geographical location via clustering or grouping mechanisms. These techniques however suffer from low reliability and limited coverage as well as high deployment and maintenance cost. In this paper, we propose fully distributed and infrastructure-free mechanisms for the density estimation in vehicular ad hoc networks. Unlike previous distributed approaches, that either rely on group formation, or on vehicle flow and speed information to calculate density, our study is inspired by the mechanisms proposed for system size estimation in peer-to-peer networks. We adapted and implemented three fully distributed algorithms, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation. The extensive simulations of these algorithms at different vehicle traffic densities and area sizes for both highways and urban areas reveal that Hop Sampling provides the highest accuracy in least convergence time and introduces least overhead on the network, but at the cost of higher load on the initiator node.**

I. Introduction

Vehicular ad hoc network (VANET) is a promising Intelligent Transportation System technology that aims to improve the traffic safety via emergency message transmission to the drivers and traffic efficiency by transferring road monitoring and location information to traffic managers while also providing a wide range of applications such as the Internet access and entertainment content to passengers [1] [2] [3] [4]. The wireless communication standard IEEE 802.11p [5] has been developed for VANETs as an improvement to the IEEE 802.11 family [6]. IEEE 802.11p particularly deals with the physical and medium access control layers of the network stack allowing communication among the vehicles without relying on any infrastructure.

Vehicular density is one of the main metrics used for monitoring road traffic conditions and provides a good estimate for the congestion on the road. Various methods have been used in the literature to estimate vehicular density [7]-[16]. 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 [7] [8] [9]. However, these techniques suffer from high deployment cost and high rate of failures, resulting in high maintenance cost and limited coverage.

The other group of methods adopt centralized or grouping approaches (e.g. clustering in [10]) where data is gathered at centralized locations for processing and then disseminated across the network. However, the centralized approach not only makes the system vulnerable to single point of failure, but also makes it harder to implement such system for a fully distributed and self organizing traffic information system, where vehicles can only directly communicate with their neighboring vehicles, and they do not rely on infrastructure or centralized information system to establish communication.

The goal of this paper 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. Although some distributed approaches have been previously proposed in literature, they either rely on group formation [11], or vehicle speed and flow information to calculate density of vehicles. We use a completely different network size calculation technique to estimate the density of vehicles on the road. To the best of our knowledge, this approach has not been previously applied for density estimation in VANETs. The main contributions of this paper 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.
- The algorithms have been rigorously tested for validity and performance based on real life data across eight different traffic scenarios including different traffic densities and different area sizes for both highway and urban roads.

The rest of the paper is organized as follows. Section II provides an overview of the related work. Section III describes

978-1-4673-4996-3/12/$31.00 ©2012 IEEE

TABLE I: Related Work on Vehicle Density Estimation in VANETs

Ref	Infrastructure used	Category	Method used
[7]	Yes	Centralized	Road-side camera images using Kalman filtering
[8]	Yes	Centralized	Neural networks
[9]	Yes	Centralized	Capturing road video using cameras and applying Kalman filtering
[12]	No	Distributed	Local density used to estimate global density
[10]	No	Clustering	Extension of [12] by using clusters
[11]	No	Distributed	Group formation
[13]	No	Distributed	Traffic-flow model using vehicle's speed and flow
[14]	No	Distributed	Random sampling of vehicles
[15]	No	Distributed	Vehicle's speed and acceleration information
[16]	No	Distributed	Fluid dynamics and car follow model

the distributed algorithms used for the density estimation in VANET. Section IV presents the simulation environment and scenarios, the performance metrics used for the comparison of the density estimation algorithms, and the simulation results. Concluding remarks and future directions are given in Section V.

II. 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 I.

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 [7], [9], [8]. Road side camera images are used for traffic monitoring and density estimation in [7]. 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 [9] 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 [8].

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 [12]. This work has been extended with a clustering approach [10] 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 [11] where group leader computes vehicle density and disseminates this information among other members of the group. In [13], a relationship between speed, flow and density is used to estimate local density using traffic-flow model. A similar approach is used in [15] where vehicle tracks its own speed and acceleration patterns to estimate the local density. In [14], 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 [16].

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 [11], 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.

III. 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.

A. 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 [17].

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 [17]. 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 $N = {C_L}^2/2L$ [17].

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 [17]. If a high T value is selected by the initiator node, the system becomes more asymptotically unbiased while increasing the communication overhead.

B. Hop Sampling

Hop Sampling algorithm is based on the principle of probabilistic polling [18]. 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.

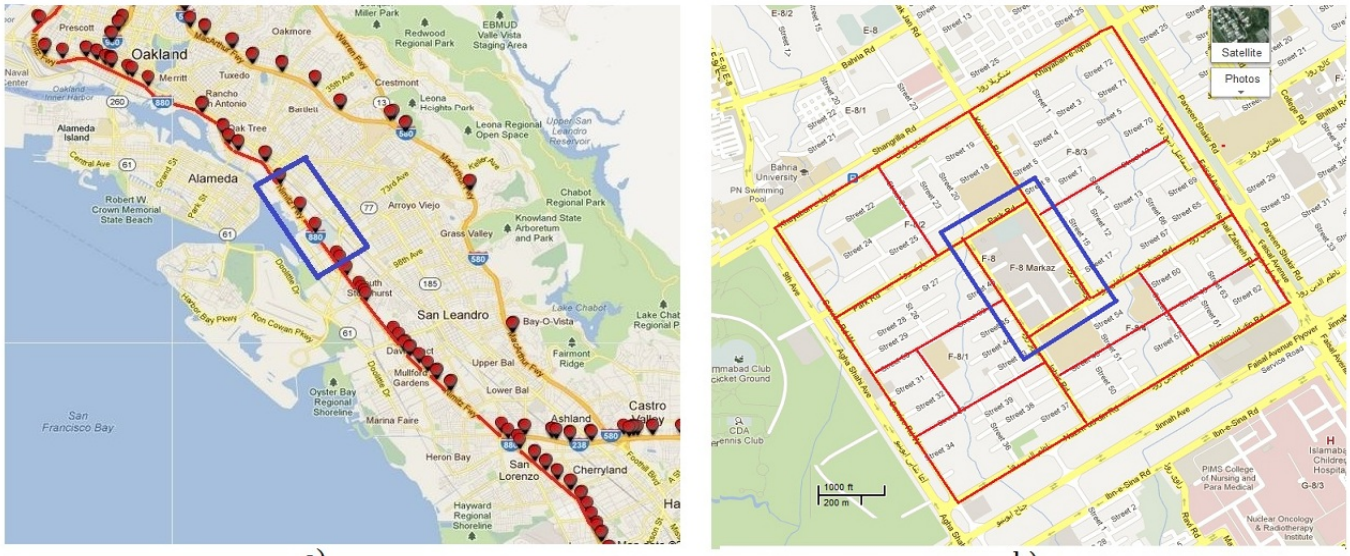
In our simulations, the values of *minHopsReporting* and *gossipTo* are set to 2. Density of vehicles D_a within area of size a is then obtained by $D_a = N/l_a$, where N is the number of vehicles on the road, and l_a is the total length of the road within area of size a.

C. 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 [19]. 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

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



a) b)

Fig. 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)

- 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.

IV. Simulation & Results

A. 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) [20] 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.

TABLE II: 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

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 [21]. 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

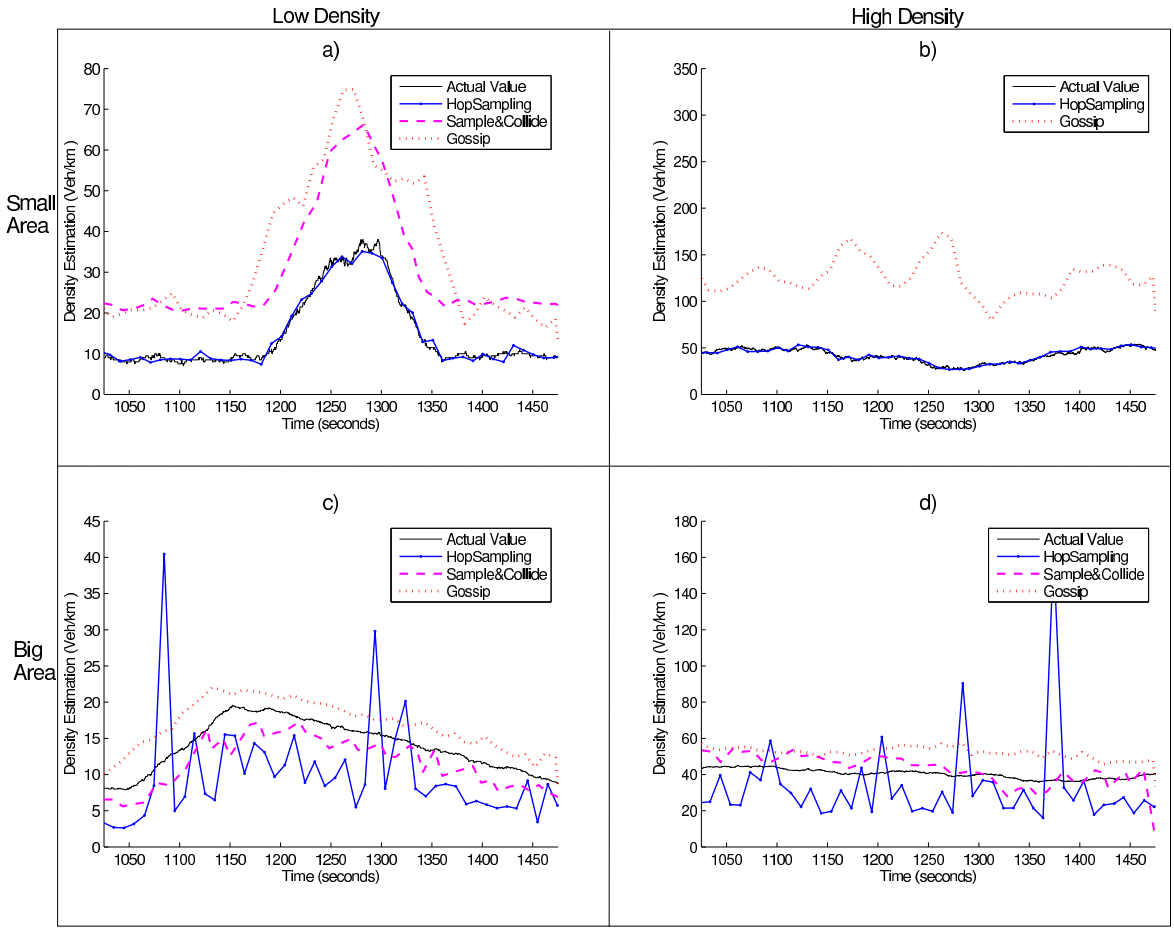


Fig. 2: Density Estimation-Highway Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

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. 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 II.

2) Urban Simulation: We used one of the urban areas in Islamabad, Pakistan shown in Fig. 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 [22]. 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. 1-b) and big areas (12.9 km of road, red lines, Fig. 1-b) for both low and high density traffic. Simulation parameters are given in Table II.

B. 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 ($Messages_{network}$)

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

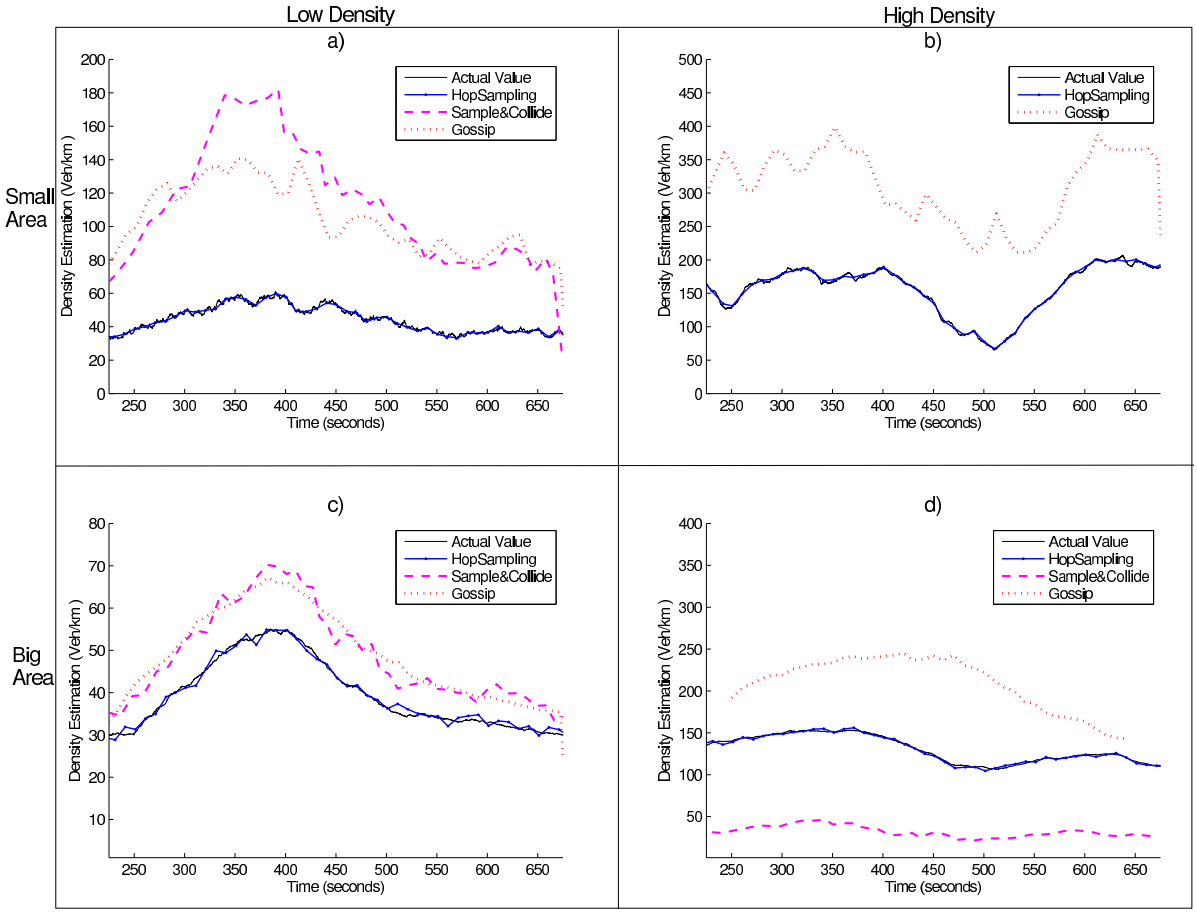


Fig. 3: Density Estimation-Urban Scenarios a) Small Area-Low Density b) Small Area-High Density c) Big Area-Low Density d) Big Area-High Density

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.

C. Simulation Results

Figs. 2 and 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 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 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. 2-b) and 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 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 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

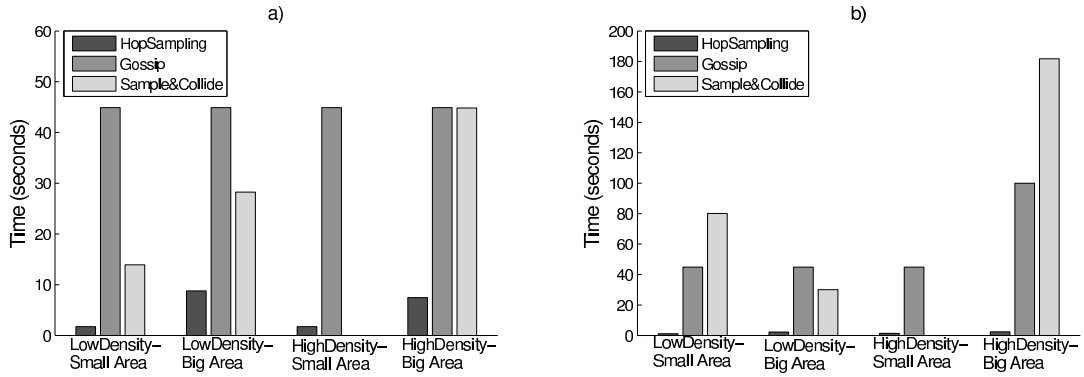


Fig. 4: Convergence Time: Total time needed for the algorithms to converge for (a) Highway and (b) Urban scenarios

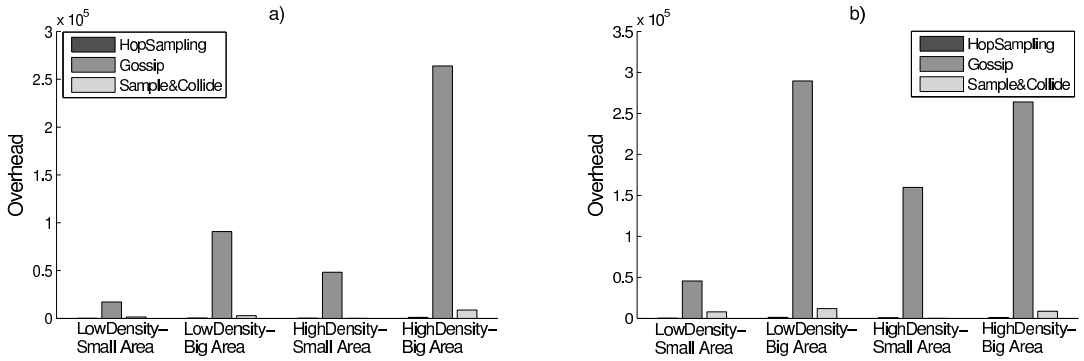


Fig. 5: Overhead: Total number of messages sent for (a) Highway and (b) Urban scenarios

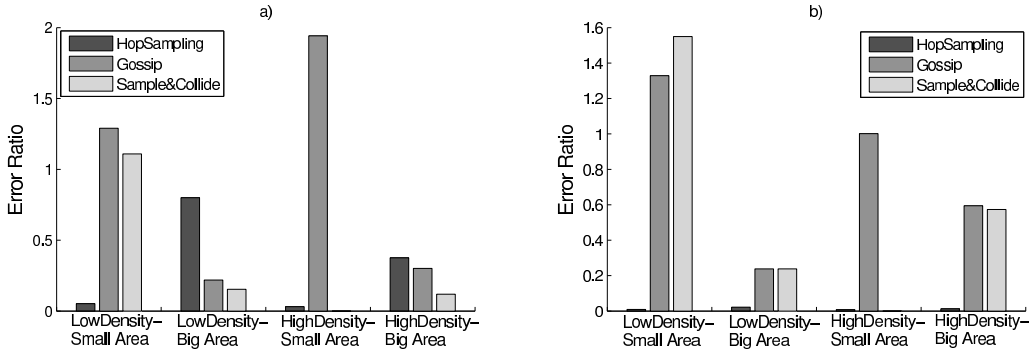


Fig. 6: Error Ratio: (a) Highway (b) Urban scenarios

convergence time usually less than 10 seconds.

Fig. 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. 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. 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.

V. Conclusion

In this paper, we 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

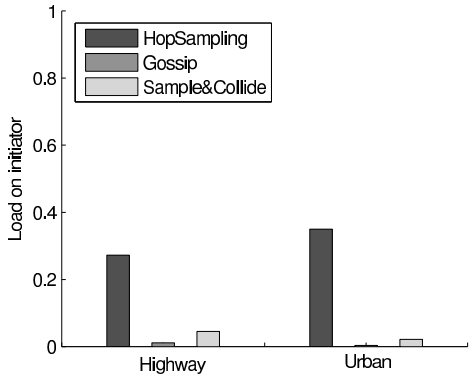


Fig. 7: Load on the initiator: Highway and Urban scenarios

implemented three fully distributed algorithms, namely Sample & Collide, Hop Sampling and Gossip-based Aggregation for VANETs. The 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 Hop Sampling provides the highest accuracy in the least convergence time by introducing the least overhead to the network but at the cost of higher load on the initiator node. The high performance of Hop Sampling algorithm 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.

Our ongoing work involves incorporating the effect of background traffic on the efficiency of the algorithms. The tradeoff between the accuracy of estimation and the network load will be investigated. Then, we aim to propose a new distributed protocol especially tailored for VANETs, taking advantages from the strong aspects of the three algorithms used in this work.

ACKNOWLEDGEMENT

Our work was supported by Turk Telekom under Grant Number 11315-07. We would like to thank Irem Nizamoğlu and Orkhan Badirkhanli for their invaluable contribution during the initial phase of our work.

REFERENCES

[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, 2010.

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

[3] J. Rao and T. Giuli, "Evaluating vehicular ad hoc networks for group applications," in *Pervasive Computing and Communications Workshops (PERCOM Workshops), IEEE International Conference on*, march 2011, pp. 594 –599.

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

[5] "IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments," *IEEE Std 802.11p*, pp. 1 –51, 07 2010.

[6] "IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications," *IEEE Std 802.11 (Revision of IEEE Std 802.11-1999)*, pp. C1 –118, 06 2007.

[7] 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.

[8] 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.

[9] 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.

[10] S. Panichpapiboon and W. Pattara-atikom, "Exploiting wireless communication in vehicle density estimation," *Vehicular Technology, IEEE Transactions on*, vol. 60, no. 6, pp. 2742 –2751, july 2011.

[11] 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.

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

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] 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

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

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

[22] 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.