Distributed Neighbor Distribution Estimation with Adaptive Compressive Sensing in VANETs

Yunxiang Cai[1], Hongzi Zhu[1], Xiao Wang[1], Shan Chang[2], Jiangang Shen[1], Minyi Guo[1]
[1]Shanghai Jiao Tong University, China
[2]Donghua University, China
{caiyunxiang, hongzi}@sjtu.edu.cn, changshan@dhu.edu.cn

***Abstract*—Acquiring the geographical distribution of neighbors can support more adaptive media access control (MAC) protocols and other safety applications in Vehicular ad hoc network (VANETs). However, it is very challenging for each vehicle to estimate its own neighbor distribution in a fully distributed setting. In this paper, we propose an online distributed neighbor distribution estimation scheme, called PeerProbe, in which vehicles collaborate with each other to probe their own neighborhood via simultaneous symbol-level wireless communication. An adaptive compressive sensing algorithm is developed to recover a neighbor distribution based on a small number of random probes with non-negligible noise. Moreover, the needed number of probes adapts to the sparseness of the distribution. We conduct extensive simulations and the results demonstrate that PeerProbe is lightweight and can accurately recover highly dynamic neighbor distributions in critical channel conditions.**

***Index Terms*—neighbor distribution estimation; adaptive compressive sensing; vehicular ad hoc network; OFDM; Bloom filter**

I. Introduction

Vehicular ad hoc networks (VANETs) are emerging as a new landscape of mobile ad hoc networks, aiming to provide a wide spectrum of safety and comfort applications to drivers and passengers. In VANETs, vehicles equipped with wireless communication devices can exchange data with each other (vehicle-to-vehicle communications). It is essential for a vehicle to acquire the context information of the network, especially, the geographical distribution of its neighbors. We refer to the *neighbor distribution estimation problem* as the problem that a vehicle can actively estimate the respective number of neighboring vehicles within a large set of different communication ranges. Figure 1 illustrates the neighbor distribution of a vehicle within three communication ranges. Such neighbor distribution information can be utilized to design efficient media access control (MAC) protocols. For instance, a vehicle can choose proper communication ranges and broadcasting periodicities for various safety applications. Moreover, given the number of neighbors within a chosen communication range, adaptive channel allocation schemes can be achieved.

A feasible scheme for the neighbor distribution estimation problem in VANETs, however, has to meet three rigid requirements as follows: 1) due to the fast movement of vehicles, such a scheme should be fast and efficient in terms of both communication and computation costs; 2) it is often the case that there is no roadside unit available, which means that such a scheme should reliably work without any centralized unit in the network; 3) such a scheme should achieve very high accuracy as the derived information could be utilized in critical driving safety applications.

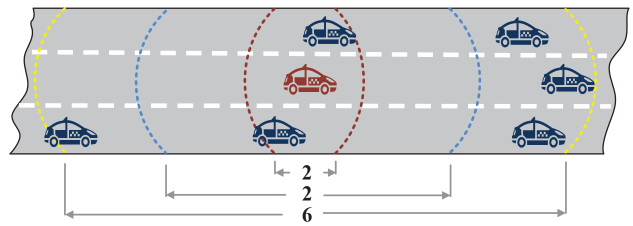


Fig. 1. Illustration of the neighbor distribution of a red vehicle, e.g., the number of neighbors is 2, 2, and 6, respectively, within three consecutive communication ranges

In the literature, existing vehicle density estimation schemes and vehicle mobility modeling methods can be explored to derive neighbor distribution information. Infrastructure-based density estimation schemes need certain infrastructure support such as video monitoring and surveillance system [1]–[3], base stations [4], roadside units (RSUs) [5], or other fixed traffic detectors [6] [7] [8] [9]. Other infrastructure-free methods either require the prior knowledge about the distribution of vehicles [10] or need massive communication among vehicles [11]. Mobility modeling schemes [1] [12] [13] [14] usually take a centralized methodology to study vehicle mobility models and distributions and cannot be directly utilized to estimate the neighbor distribution for each individual vehicle.

In this paper, we propose a fully distributed neighbor distribution estimation scheme, called *PeerProbe*, in which vehicles in vicinity collaborate with each other to probe the neighborhood via symbol-level wireless communication. Instead of letting each vehicle measure the number of neighbors within all communication ranges to get the neighbor distribution, inspired by the insight that the geographical distribution of vehicles normally could be sparse, the core idea of PeerProbe is for each vehicle to utilize compressive sensing to recover its neighbor distribution by randomly measuring the number of neighboring vehicles only within a few randomly selected communication ranges.

Three main challenges are encountered. First, it is non-

Hongzi Zhu is corresponding author.

trivial to count the number of neighbors of each vehicle within a specific communication range, especially when there is no central unit available to coordinate data transmission and to collect information sent from individual vehicles. To address this challenge, a Bloom filter is constructed for each vehicle in a distributed fashion by all neighbors within the communication range. More specifically, with common half-duplex radios, each vehicle randomly chooses to be a transmitter or a receiver. When a vehicle acts as a transmitter, it broadcasts an *individual bitmap* generated by making a hash of its own identity. When the vehicle act as a receiver, it decodes superposed signals to collect bitmaps from transmitter neighbors. This procedure repeats until each vehicle has collected enough bitmaps from most of its neighbors with high probability.

Second, it is challenging to efficiently and reliably exchange individual bitmaps among vehicles. It would be infeasible if each vehicle broadcasts individual bitmaps by packets in turn. To tackle this challenge, we use On-OFF Keying (OOK) mapping to embed the individual bitmap of a vehicle into Orthogonal Frequency Division Multiplexing (OFDM) symbols, letting each subcarrier of an OFDM symbol carry one bit. Furthermore, all transmitter neighbors simultaneously transmit their OFDM symbols while a receiver can successfully decode these superposed OFDM symbols by an effective de-mapping scheme, deriving a *combined bitmap* which is the logical OR result of all transmitted individual bitmaps.

Third, the geographical distribution of vehicles varies, which could be neither even (e.g., due to traffic lights) nor sparse (e.g., due to traffic jams) on a road, which makes it difficult to recover the neighbor distribution with normal compressive sensing methods. To deal with this challenge, we propose an adaptive compressive sensing scheme, in which the number of measures needed for a vehicle to recover its neighbor distribution adapts to the sparseness of its neighborhood. Moreover, probabilistic measurement noise is first filtered out via Total Variation (TV) regularization before being applied to compressive sensing to recover the final neighbor distribution.

PeerProbe is a fully distributed scheme and requires no special hardware. Extensive simulations are conducted and the results demonstrate that the estimation errors of the number of neighbors in one measure can reach up to 10% due to demodulation errors and the probabilistic noise. Nevertheless, the adaptive compressive sensing scheme is resilient to measurement noise and outperforms traditional compressive sensing schemes at low measurement costs.

We highlight the main contributions made in this work as follows:

- An efficient number-of-neighbors estimation scheme is proposed using symbol-level wireless communication with half-duplex radios.
- An adaptive compressive sensing scheme is developed to estimate neighbor distributions, which can handle massive measurement noise and adapt to varying densities of vehicles.
- Extensive simulations are conducted to demonstrate the efficiency of PeerProbe in achieving high accuracy neighbor distribution estimation under different channel conditions and node distributions.

II. System Model and Design Goals

A. System Model

In the neighbor distribution estimation problem, we consider that there is no centralized unit in the network and vehicles are peers with equal capabilities as follows:

- **Communication:** vehicles can communicate with each other via Dedicated Short Range Communications (DSRC), which adopts OFDM modulation and can actively control transmission power to change the communication range. Given the analysis based on real-world DSRC trace [15], we consider symmetric channels, i.e., a vehicle x is in the communication range of vehicle y if and only if y is in the communication range of x when they use the same transmission power to communicate. In addition, we consider the disk model where the communication range controlled by certain transmission power is a disk.
- **Computation:** vehicles can perform basic operations such as random number generation, hash operation, and common matrix algebra operations.
- **Synchronization:** vehicles are equipped with global positioning system (GPS) receivers, which are used as the time reference to synchronize vehicles. GPS can provide 1 pulse per second (PPS) signal with an accuracy of less than 100ns even for low-end GPS receivers [16]. Note that we do not directly use GPS for positioning as the reported location is quite inaccurate in urban settings [17].

B. Design Goals

We consider the following goals in designing our distributed neighbor distribution estimation scheme:

- **High Accuracy.** The estimated neighbor distribution information could be used for driving safety applications or for MAC protocol design. These applications have the urge for high accuracy of the estimated distribution with a fine spacial granularity (e.g., at meter level).
- **Cost Efficiency.** The communication cost for neighbor distribution estimation should be extremely low to reserve most channel resource for data communication. Moreover, the computation complexity of the algorithms should be low to achieve fast response time.
- **Good Reliability.** The scheme should reliably work under highly dynamic vehicular environments, for example, uneven vehicle distributions due to various traffic conditions.
- **Easy Deployment.** The scheme should have a minimal hardware requirement, consisting of only widely-available cheap sensors and commercial off-the-shelf (COTS) vehicle-to-vehicle communication radio devices, which makes it easy to deploy.

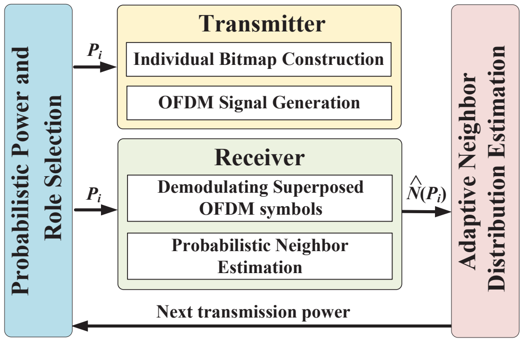


Fig. 2. The system architecture of PeerProbe

III. DESIGN OF PEERPROBE

A. Overview

To facilitate individual vehicles to efficiently estimate their neighborhood, PeerProbe integrates three techniques as follows. First, a vehicle only needs to randomly measure the number of neighbors in a few communication ranges, leveraging the insight that vehicle distribution could be sparse and the power of the compressive sensing theory. Moreover, the number of measures required for a vehicle adapts to the sparseness of its neighborhood. Second, a vehicle counts the number of neighbors within a communication range with a Bloom filter, which is neatly constructed in a distributed fashion by its neighbors. Third, OFDM symbols are designed so that superposed OFDM symbols simultaneously transmitted by multiple vehicles can be successfully demodulated, which significantly increases the utility of channel resources.

The system architecture as depicted in Figure 2 consists of four major components:

Probabilistic Power and Role Selection. A vehicle randomly chooses a transmission power level P_i according to the network time. As vehicles are synchronized, it means that all vehicles would randomly select the same transmission power at a time. Given the transmission power level P_i, the vehicle chooses to be a transmitter (or a receiver) with the probability of p (or $1-p$). The probabilistic role selection repeats until a receiver has collected sufficient information from the transmitter neighbors within the communication range controlled by the transmission power level P_i.

Role of Transmitter. If the vehicle selects to be a transmitter, it first constructs an individual bitmap, which consists of a hashed identity (e.g., using the MAC address of the vehicle) and an indication bit field and then embeds the bitmap into OFDM symbols using OOK mapping with each subcarrier conveying one bit. Finally, the vehicle transmits these OFDM symbols with the transmission power level of P_i.

Role of Receiver. If the vehicle selects to be a receiver, it demodulates received OFDM symbols, which are the superposition of those OFDM symbols sent from transmitter neighbors, and derives a combined bitmap. As one combined bitmap only contains the ID information of partial neighbors, the vehicle keeps collecting more combined bitmaps when it selects to be a receiver again. Finally, by aggregating all combined bitmaps, the vehicle can obtain a full bitmap of a Bloom filter, with which the number of neighbors within the communication range controlled by P_i, can be estimated, denoted by $\widehat{N}(P_i)$.

Adaptive Neighbor Distribution Estimation. Each time when a new $N(P_i)$ is estimated, the vehicle tries to recover the whole neighbor distribution using compressive sensing and compares the new result with previous result. If the difference is smaller than a given threshold, the neighbor estimation process converges and ends; otherwise, the vehicle chooses the next random transmission power level P_j and starts to estimate $N(P_j)$. Moreover, the vehicle also informs its neighbor in the communication range controlled by P_j of the new measure demand by setting the indication bit field in its individual bitmap.

B. Probabilistic Power and Role Selection

Let $\{R_0, R_1, \cdots, R_{n-1}\}$ denote the set of communication ranges of interest in the neighbor geographical distribution estimation problem and let $\{P_0, P_1, \cdots, P_{n-1}\}$ denote the corresponding transmission power levels[1]. To randomly select a transmission power level P_i from $\{P_0, P_2, \cdots, P_{n-1}\}$, a vehicle uses the synchronized network time as the seed of a random number generator and $i = rand()\%n$.

As there is no centralized unit in the network, in order for each vehicle to collect information from neighbors within communication range R_i, we propose a probabilistic scheme, in which a vehicle v selects to be a transmitter with a probability of p and selects to a receiver with a probability of $1-p$. When v is a transmitter, v broadcasts its own information with power P_i, and when v is a receiver, v can receive information from all transmitter neighbors within R_i. We have the following theorems:

***Theorem** 1: If each receiver vehicle expects to collect information from a proportion of λ of all its neighbors within R_i, then the probabilistic role selection procedure should be repeated at each vehicle for at least $\frac{1}{(1-p)} \cdot log_{(1-p)}(1-\lambda)$ times.*

Proof: We assume that the probabilistic role selection procedure should be repeated at least for x times and the number of neighbors is $N(P_i)$. Then, the expected number for a vehicle v to be a receiver is $x(1-p)$. When v is a receiver for the first time, the expected number of neighbors that v can receive information from is $N(P_i)p$. When v is a receiver for the second time, the expected number of new neighbors that v can receive information from is $N(P_i)(1-p)p$. Similarly, when v is a receiver for the i-th time, the expected number of new neighbors that v can receive information from is $N(P_i)(1-p)^{i-1}p$.Therefore, the total expected number of neighbors that v can receive information from is $N(P_i)(1-$

[1]The specific propagation model about a transmission power level and the corresponding communication range is not the concern of this work. We hereafter use the transmission power level to refer to the corresponding communication range without the loss of generality.

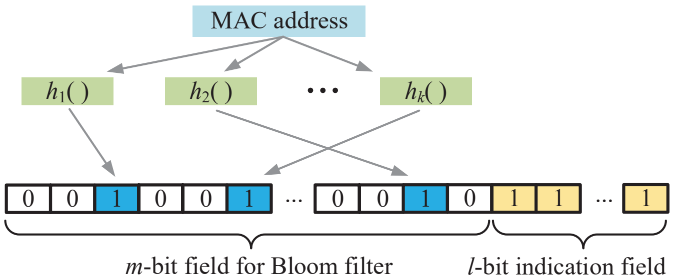


Fig. 3. An individual bitmap of a vehicle, consisting of an m-bit Bloom filter field and an l-bit indication field

$p)^0p + N(P_i)(1-p)^1p + \cdots + N(P_i)(1-p)^{x(1-p)-1}p$, which is no less than $\lambda N(P_i)$. We have

$$x \geq \frac{1}{(1-p)} \cdot log_{(1-p)}(1-\lambda), \tag{1}$$

which concludes the proof. ∎

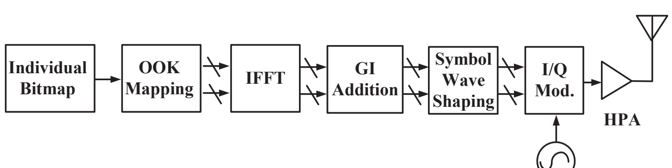
Given the above theorem, x reaches the minimum when $p = 1 - \frac{1}{e}$.

***Theorem** 2: If each transmitter vehicle expects to inform a proportion of γ of all its neighbors within R_i, then the probabilistic role selection procedure should be repeated at each vehicle for at least $\frac{1}{p} \cdot log_p(1-\gamma)$ times.*

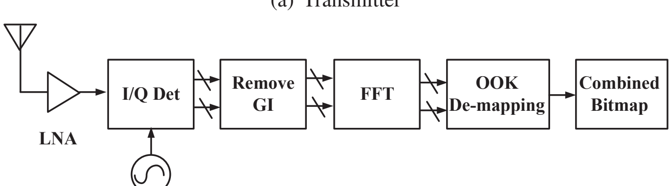
The proof is similar as that of Theorem 1 and is omitted due to the page limitation. Given the above theorem, x reaches the minimum when $p = \frac{1}{e}$. Considering both Theorem 1 and Theorem 2, we set $p = 0.5$. As a result, if we expect each receiver vehicle to collect information from $\lambda = 95\%$ of its neighbors and each transmitter vehicle to inform $\gamma = 95\%$ of its neighbors within the communication range R_i, each vehicle needs to repeat the probabilistic role selection procedure for nine times.

C. Role of Transmitter

When a vehicle selects to be a transmitter, it helps other receiver neighbors to construct the bitmap of a Bloom filter by broadcasting its own individual bitmap through OFDM symbols with the transmission power level P_i.



(a) Transmitter



(b) Receiver

Fig. 4. The signal processing of transmitter and receiver with standard OFDM-based scheme

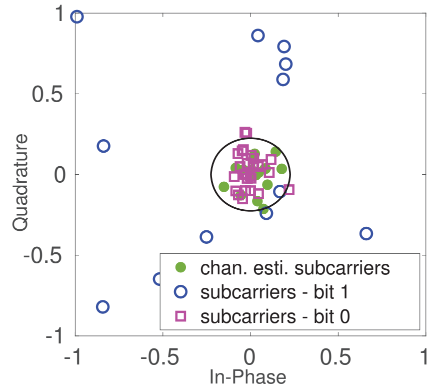


Fig. 5. De-mapping algorithm, where the Euclidean norm of a complex number is used to determine bit 1 or bit 0

*1) **Individual Bitmap Construction**:* As depicted in Figure 3, an individual bitmap consists of two bit fields, i.e., an m-bit Bloom filter field and an l-bit indication field. A transmitter vehicle sets up the m-bit Bloom filter field according to the results of a group of k hash functions, calculated with its MAC address. The vehicle sets the indication bits to inform other vehicles that it also expects to collect information from other neighbors in this communication range. Note that l bits are set at the same time to mitigate the impact of demodulation error (see more details in Subsubsection III-D1 and Subsubsection III-E4).

*2) **OFDM Symbol Generation**:* We adopt OOK mapping to embed each bit in an individual bitmap into one corresponding subcarrier of an OFDM symbol. Specifically, bit 1 and bit 0 are mapped to the symbol (1, 0) and (0, 0) in the constellation diagram, respectively. As depicted in Figure 4(a), the signal processing of a transmitter is compatible with IEEE 802.11p radios. Specifically, 48 subcarriers with index [-26:-22, -20:-8, -6:-1, 1:6, 8:20, 22:26] are used for bitmap transmission and the rest 16 subcarriers, including four pilot subcarriers, eleven null subcarriers and the DC subcarrier, always transmit bit 0, i.e., symbol (0, 0), for channel estimation. Consequently, when the length of an individual bitmap is larger than 48, the bitmap is divided into multiple OFDM symbols. With the generated OFDM symbols, a transmitter vehicle broadcasts each OFDM symbol with the chosen transmission power level P_i according to the synchronized network time.

D. Role of Receiver

When a vehicle v selects to be a receiver, it collects combined individual bitmaps from transmitter neighbors and estimates the number of neighbors when enough combined bitmaps are collected.

*1) **Demodulating Superposed OFDM Symbols**:* Given the number of neighbors $N(P_i)$, there are on average $pN(P_i)$ transmitter neighbors, simultaneously broadcasting their individual bitmaps via OFDM symbols. These OFDM symbol signals are superposed before they hit the antenna of v. Figure 4(b) plots the signal processing of a receiver. To demodulate superposed OFDM symbols, after FFT, a complex number $z_i = I_i + jQ_i$ can be obtained from subcarrier i. We introduce a

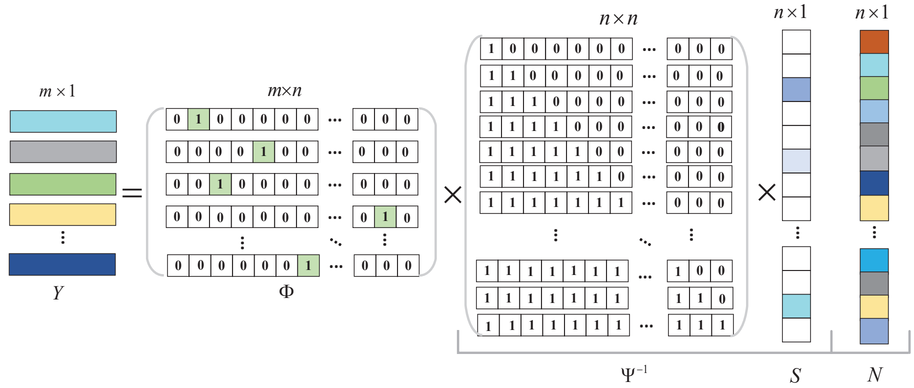


Fig. 6. Illustration of compressive sensing, where Y is the measurement vector; Φ is an $m \times n$ measurement matrix; Ψ is an $n \times n$ differential basis; S is the transformed vector in a sparse domain; N is the expected neighbor distribution vector

simple yet effective OOK de-mapping algorithm. Specifically, the Euclidean norm of the complex on each channel estimation subcarrier is calculated, and the maximal norm is used as the threshold to determine bit 1 or bit 0 on other data subcarriers.

For example, Figure 5 plots the constellation diagram for one superposed OFDM symbol through channel. The dark circle in the figure indicates the calculated threshold. It can be seen that most bits can be correctly demodulated except that there are three bits being wrong. As a result, the demodulated bitmap is actually the result of logical OR operation of those individual bitmaps. Moreover, bit errors would happen in the combined bitmap due to channel impairments. Figure 7 illustrates an example of an extracted combined bitmap from superposed individual bitmaps.

*2) **Probabilistic Neighbor Estimation**:* Each time vehicle v selects to be a receiver, it can get a combined bitmap. According to Theorem 1, after v repeats probabilistic role selection procedure for $\frac{1}{(1-p)} \cdot log_{(1-p)}(1-\lambda)$ times, it can collect information from a proportion of λ of all its neighbors within R_i. After that, v can get a *full bitmap* of a Bloom filter [18] by performing the logical OR operation on all combined bitmaps. Such a full bitmap can be used to estimate $N(P_i)$ as

$$\widehat{N}(P_i) = -\frac{mln(1-\frac{c(Z)}{m})}{k} \tag{2}$$

where m is the length of the full bitmap; k is the number of hash functions and the function $c(\cdot)$ counts the number of ones within the full bitmap.

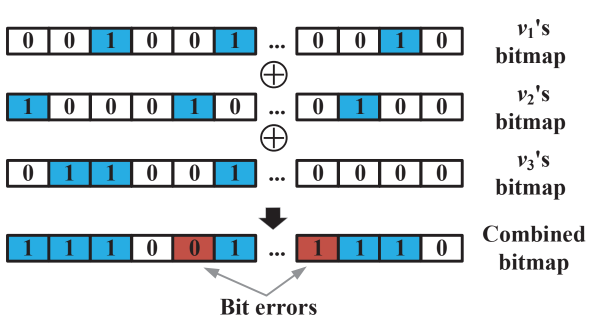


Fig. 7. An example of derived combined bitmap, which is actually the logical OR result of three individual bitmaps

E. Adaptive Neighbor Distribution Estimation

As for the geographical distribution of vehicles, we have the following observation.

***Observation** 1: The geographical distribution of vehicles on a surface road could be sparse and the sparseness of the distribution varies over space and time.*

Recent advances in the field of compressive sensing have developed reliable recovery algorithms for inferring sparse representations if one can randomly measure arbitrary linear combinations of the signal, then the signal could be reliably reconstructed through solving an optimization problem. Given the above observation, we leverage compressive sensing theory to recover the neighbor distribution of individual vehicles at the cost of a small number of random measures. Moreover, the number of measures needed for a vehicle adapts to the sparseness changes of its neighbor distribution.

*1) **Determining the Representation Basis**:* We define the geographical neighbor distribution of a vehicle v as a sequence of two-tuples $\{(P_0, N(P_0)), (P_1, N(P_1)), (P_2, N(P_2)), \cdots, (P_{n-1}, N(P_{n-1}))\}$, where $P_0 < P_1 < \cdots < P_{n-1}$. It is clear that $N(P_0) \le N(P_1) \le \cdots \le N(P_{n-1})$ and the vector $N = (N(P_0), N(P_1), \cdots, N(P_{n-1}))$ is not sparse. However, as depicted in Figure 6, N could be transformed to a sparse vector S, i.e., $S = \Psi N$ and $N = \Psi^{-1} S$, where Ψ is an $n \times n$ differential basis and Ψ^{-1} is a lower triangular matrix,

$$\Psi = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}, \tag{3}$$

$$\Psi^{-1} = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \dots & 1 \end{pmatrix}. \tag{4}$$

For instance, $N = (2, 2, 6)$ and $S = (2, 0, 4)$ in the example of Figure 1.

*2) **Constructing the Measurement Matrix**:* The measurement matrix Φ is an $m \times n$ matrix, where m equals to the number of measures and each row corresponds to the transmit power levels from P_0 to P_{n-1}. As described in Subsection III-B, each vehicle randomly chooses a transmission power level $P_i \in \{P_0, P_1, \cdots, P_{n-1}\}$ according to the network time for m times and estimates the corresponding number of neighbors $\widehat{N}(P_i)$. As shown in Figure 6, if P_j is chosen in the i-th measure, then Φ_{ij} is set to 1. Let Y denote the vector of the m measures. Since each measure $\widehat{N}(P_i)$ has errors, Y can be represented as follows:

$$Y = \Phi N + \eta = \Phi \Psi^{-1} S + \eta, \tag{5}$$

where η denotes the vector of estimation errors.

*3) **Recovering the Neighbor Distribution Signal**:* According to (5), to recover N in the sparse domain is to solve the following l_0 optimization problem:

$$\widehat{S} = \arg\min_{S} \|S\|_0 \quad \text{s.t. } Y = AS \tag{6}$$

where Y and $A = \Phi\Psi^{-1}$ are known. The above minimization problem can be resolved with orthogonal matching pursuit (OMP) algorithms [19] [20] [21]. We adopt the CoSaMP algorithm [22] as it can identify many components during each iteration, which allows the algorithm to run faster for many types of signals.

According to the compressive sampling theory, a K-sparse signal can be reconstructed from m measurements, if m satisfies $m \geq b \cdot \mu^2(\Phi, \Psi) \cdot K \cdot logn$ where b is a positive constant, and $\mu(\Phi, \Psi)$ is the coherence between measurement matrix Φ and representation basis Ψ [23]. The coherence metric measures the largest correlation between any two element of Φ and Ψ, defined as: $\mu(\Phi, \Psi) = \sqrt{n} \cdot \max_{1 \leq i,j \leq n} |\langle \phi_i, \psi_j \rangle|$. We can see that the smaller the coherence between Φ and Ψ is, the less measurements are needed to reconstruct the signal. Because Φ is randomly generated, Φ is largely incoherent with any fixed representation basis Ψ. $m = 3K$ to $4K$ is usually sufficient to perfectly recover the signal.

*4) **Adapting to Varying Sparseness**:* Different vehicles have distinct neighbor distributions, i.e., sparseness of N varies among vehicles, which means each vehicle needs to conduct different number of measures. To this end, an adaptive neighbor distribution estimation algorithm is proposed. Specifically, after z ($z \geq 1$) measures have been conducted, a vehicle v ranks the z measures according to the corresponding transmission power levels and gets measurement vector $M = \{\widehat{N}(P_{i0}), \widehat{N}(P_{i1}), \cdots, \widehat{N}(P_{iz})\}, P_{i0} < P_{i1} < \cdots < P_{iz}$. As the probabilistic neighbor estimation would introduce non-negligible measurement noise, M is first de-noised using the TV regularization [24]. Then, the de-noised z measures are used to recover the neighbor distribution, denoted as $\widehat{N}^{v,z}$. Similarly, when vehicle v continues to perform the $(z+1)$-th measure and it can get an updated recovered neighbor distribution $\widehat{N}^{v,z+1}$. Vehicle v calculates an *indicator* as

$$indicator = \frac{\|\hat{N}^{v,z+1} - \hat{N}^{v,z}\|_2}{\|\hat{N}^{v,z}\|_2} \tag{7}$$

If the indicator is smaller than a threshold ϵ, the neighbor distribution estimation process ends. In this case, vehicle v clears the l-bit indication field in its individual bitmap in all future broadcastings, but still needs to repeat the probabilistic role selection procedure as described in Subsection III-B until the indication fields in all received combined bitmaps are cleared[2] (i.e., the neighbor estimation process ends for all its neighbors).

IV. Performance Evaluation

A. Methodology

We conduct extensive simulations using the VENUS simulator (http://lion.sjtu.edu.cn/project/projectDetail?id=17) to evaluate the performance of PeerProbe with respect to various vehicle distributions and critical channel conditions. Specifically, we randomly generate a neighbor distribution, following three types of vehicle distributions, i.e., uniform distribution, Gaussian distribution, and Poisson distribution. For each distribution type, we vary the total number of vehicles, ranging from 20 to 100. We consider a set of distinct BERs to reflect the impact of particular channel models. We define two accuracy metrics as follows:

- **Neighbor Estimation Error Ratio (NEER)**: refers to the ratio of the absolute error of the estimated number of neighbors with certain transmission power level P_i to the ground truth, calculated as $\frac{|\widehat{N}(P_i) - N(P_i)|}{N(P_i)}$.
- **Neighbor Distribution Estimation Error (NDEE)**: we measure the difference between the estimated neighbor distribution of a vehicle v and the ground truth using RMS error, calculated as $\sqrt{\frac{\sum_{i=1}^{n}(\widehat{N}_i^v - N_i^v)^2}{n}}$.

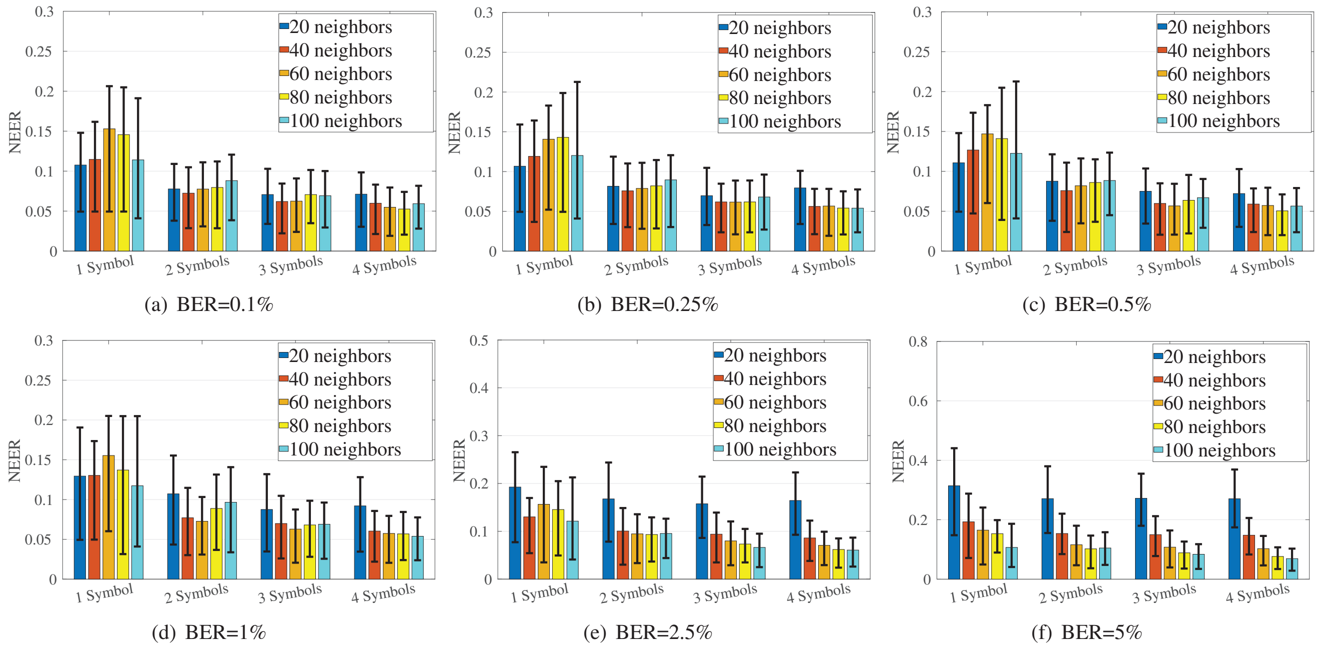
We compare our scheme with two typical compressive sensing methods, i.e., CS-OMP [21] and CS-CoSaMP [22] and the following two interpolation methods:

- **Intp-NN**: The nearest neighbor interpolation algorithm selects the value of the nearest point, yielding a piecewise-constant interpolant.
- **Intp-Linear**: The linear interpolation algorithm uses linear polynomials to construct new data points within the range of a discrete set of known data points.

B. Accuracy of Estimating the Number of Neighbors

*1) **Effect of Individual Bitmap Length**:* We randomly generate uniform neighbor distributions with the number of neighbors varying from 20 to 100 with an interval of 20. We set the repeating times of the probabilistic role selection to nine and use two hash functions. We vary the length of individual bitmaps, ranging from using one OFDM symbol (i.e., 48 bits) up to using four OFDM symbols (i.e., 192 bits) and study the

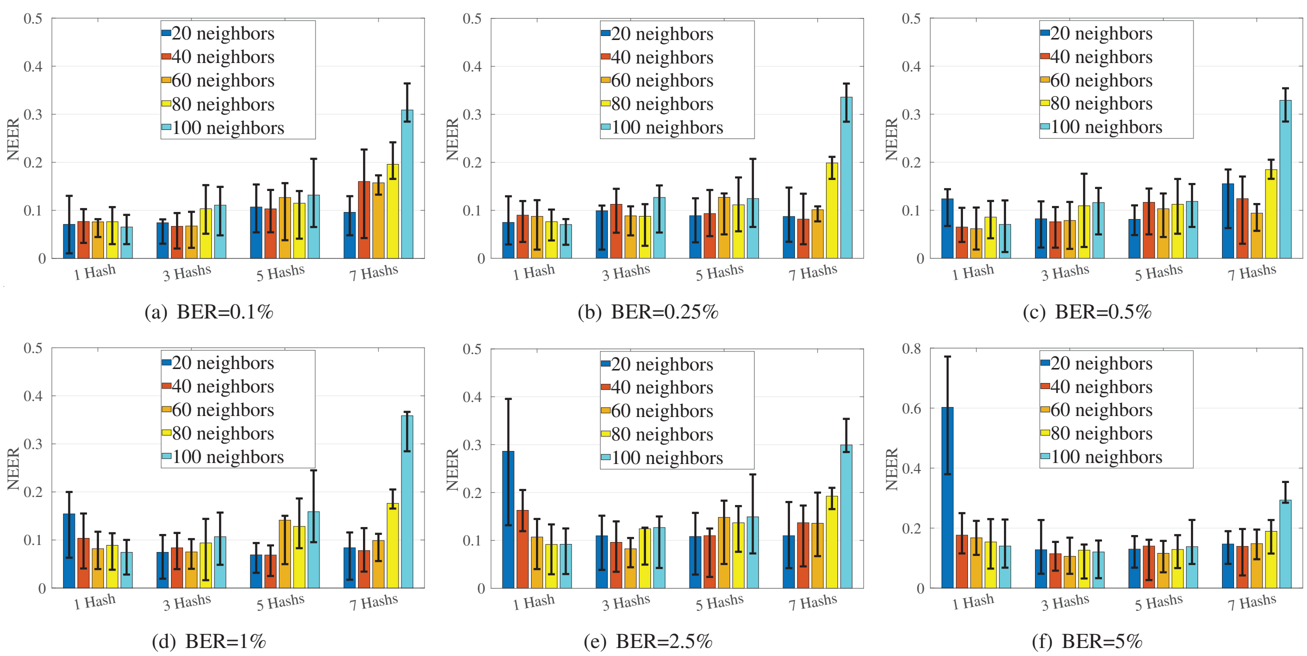
[2] To mitigate the impact of demodulation errors, the indication field of a combined bitmap is considered clear if there are more than $\frac{l}{2}$ bits are zero.



(a) BER=0.1% (b) BER=0.25% (c) BER=0.5%

(d) BER=1% (e) BER=2.5% (f) BER=5%

Fig. 8. Neighbor estimation error ratio using different bitmap lengths under different BERs



(a) BER=0.1% (b) BER=0.25% (c) BER=0.5%

(d) BER=1% (e) BER=2.5% (f) BER=5%

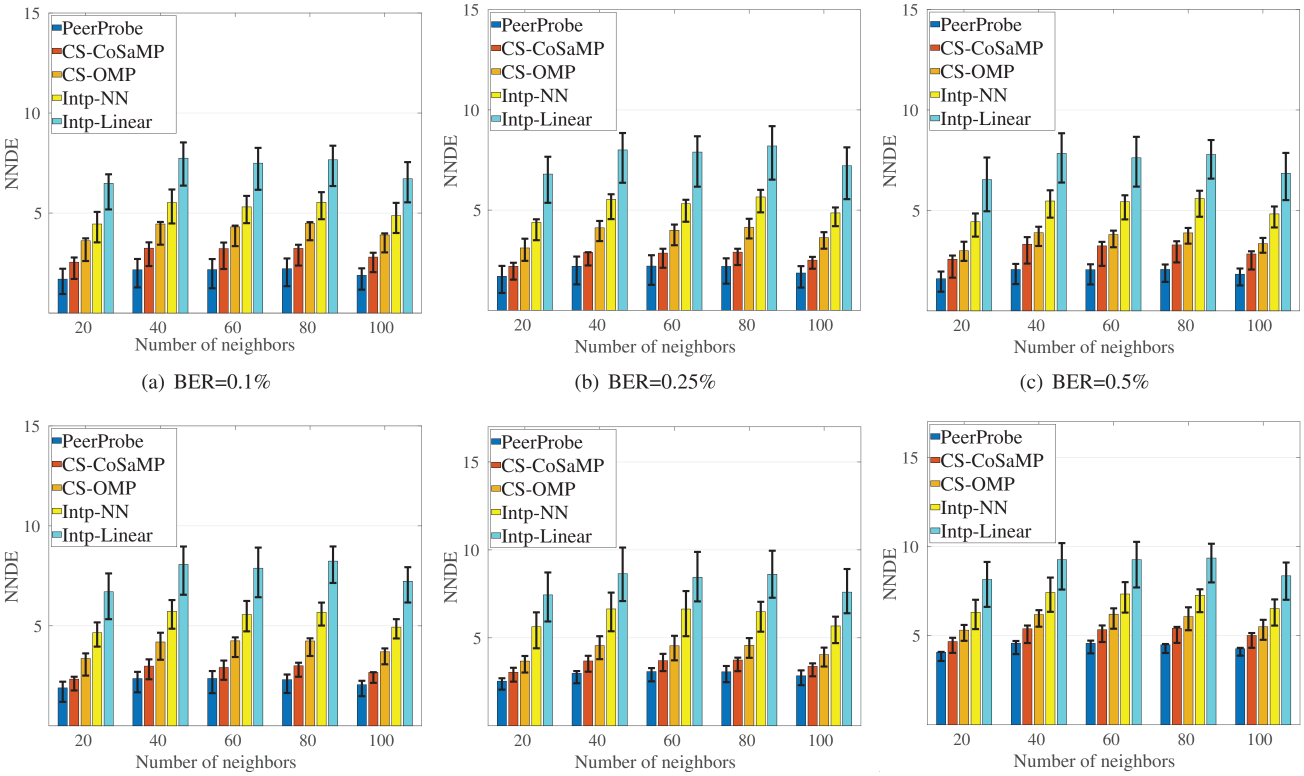
Fig. 9. Neighbor estimation error ratio using different number of hash functions under different BERs

NEER under six BERs, i.e., 0.1%, 0.25%, 0.5%, 1%, 2.5% and 5%. For each setting, we run the experiment for 500 times.

Figure 8 plots the NEER as the function of individual bitmap length. It is clear to see that the NEER decreases as the bitmap length increases. Moreover, it can also be seen that a long bitmap can deal with large BERs and a dense neighbor distribution. In practice, low response time is essential for vehicular applications, which limits the maximal length of individual bitmaps. Therefore, the strategy is to choose the longest bitmap as long as the Quality of Service (QoS) of application scenarios can be guaranteed.

*2) **Effect of the Number of Hash Functions**:* The settings of this experiment are similar to the above experiment except that we fix the length of individual bitmaps using two OFDM symbols. We vary the number of hash functions, ranging from one to seven with an interval of two and study the NEER under the same six BERs. For each setting, we run the experiment for 500 times.

Figure 9 plots the NEER as the function of the number of hash functions. It can be seen that, on one hand, according to (2) the estimated $\widehat{N}(P_i)$ is inversely proportional to the number of hash functions, which can tolerate more bit errors



(a) BER=0.1% (b) BER=0.25% (c) BER=0.5%

(d) BER=1% (e) BER=2.5% (f) BER=5%

Fig. 10. Impact of the neighbor density under different BERs

when more hash functions are involved (e.g., when BER is large, the NEER first drops as the number of hash functions increases); on the other hand, more hash functions will saturate the bitmap, which decreases the estimation accuracy of a Bloom filter (e.g., when BER is large, the NEER increases as the number of hash functions is larger than three as shown in Figure 9). Given the length of individual bitmaps, the optimal number of hash functions can be determined.

C. Accuracy of Estimating Neighbor Distribution

*1) **Impact of Neighbor Density**:* Based on previous experiments, the bitmap length is fixed using two OFDM symbols (i.e., 96-bit long) and three hash functions are used. The number of repeating times of probabilistic role selection is set to nine. We refer to the number of controllable transmission power levels as the dimension of neighbor distributions and set the dimension of neighbor distributions to 100. ϵ takes an empirical threshold of 0.03. We randomly generate uniform neighbor distributions with the number of neighbors varying form 20 to 100 with an interval of 20 and study the NDEE under six BERs, i.e., 0.1%, 0.25%, 0.5%, 1%, 2.5% and 5%. For each setting, we run the experiment for 500 times.

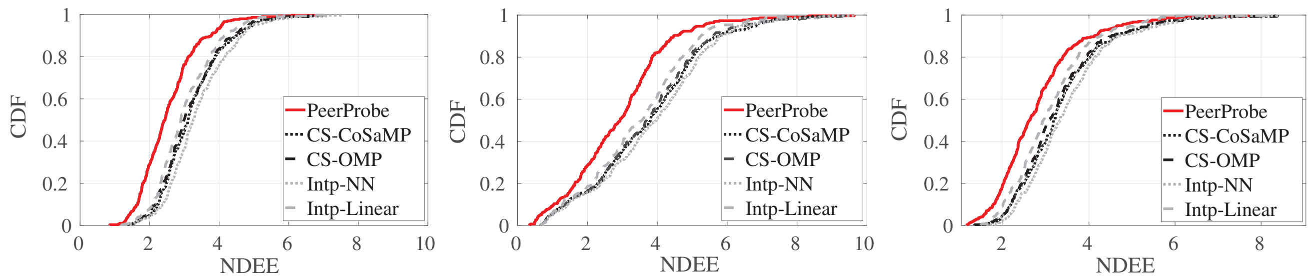
Figure 10 plots the NDEE of five candidate estimation methods as the function of neighbor density. It can be seen that, in general, all methods can achieve stable performance when the neighbor density changes. BER has significant influence on the ultimate neighbor distribution estimation accuracy. NDEE increases as the neighbor density and BER increase. PeerProbe outwits other methods.

*2) **Impact of Vehicle Distribution Types**:* The setting of this experiment is similar with the above experiment, except that we generate three typical types of geographical distributions of vehicles, i.e, uniform distributions, Gaussian distributions, and Poisson distributions. More specifically, 2000 vehicles are generated on a 4-lane road of 10km at a granularity of one meter according to different distribution types. For uniform distributions, vehicles are uniformly distributed. For Gaussian distributions, the distribution parameter μ and σ are set to $5,000 meters$ and $2,500 meters$, respectively. For Poisson distributions, the distribution parameter λ is set to 0.2. Given a generated vehicle distribution, we estimate the neighbor distribution of each vehicle, using PeerProbe. The same measures used for PeerProbe is also applied to other four candidate methods for comparison.

Figure 11(a)-(c) plot the CDFs of NDEE for all vehicles of all distribution types. Figure 12 plots the cumulative density function (CDF) of the number of measures needed by each vehicle. It can be seen that more than 90 percent of vehicles requires less than 40 measures to recover a 100-dimension neighbor distribution. It can be seen that PeerProbe can always achieve higher estimation accuracy.

D. Communication and Computation Costs

The communication delay equals to the needed number of OFDM symbols per bitmap $\times$ repeating times of role selection



(a) Uniform distribution

(b) Gaussian distribution

(c) Poisson distribution

Fig. 11. Impact of distinct vehicle distribution types, i.e., uniform distributions, Gaussian distributions, and Poisson distributions

$\times$ the needed number of measures $\times$ the duration of an OFDM symbol. For instance, for 2-symbol bitmaps, nine repeating times, 40 measures and 8μs OFDM symbol duration, the communication delay for estimating one neighbor distribution is about 5.7ms. The main computational cost stems from the CoSaMP algorithm which has a computational complexity of $O(K \cdot m^2 \cdot n)$ where K, m and n are the sparseness of the signal, the number of measures and the dimension of the signal. For example, the average running time of CoSaMP algorithm on a laptop with a 4-core 1.4GHz Intel Core i5 CPU and a 16GB memory is about 9ms in the experiment described Subsubsection IV-C2.

V. Related Work

Two categories of existing work are relevant to our work.

Vehicle density estimation. Infrastructure-base vehicle density estimation methods require extra devices to be installed as infrastructure. Computer vision techniques are used to estimate vehicle density with video monitoring and surveillance system [1]–[3]. Such schemes are severely affected by the weather and light conditions. Some other studies utilize base stations [4], RSUs [5], or other fixed traffic detectors such as dual loop detectors [6], wireless vehicle sensor [7], roadside-installed microphone [8], and highway toll stations [9]. These methods suffer from low reliability and limited coverage as well as high deployment and maintenance costs. Infrastructure-free vehicle density estimation methods either require the prior knowledge about the distribution of vehicles such as lognormal or exponential inter-vehicle spacing distribution [10], or need massive communication among vehicles such as clustering or grouping methods [11].

Vehicle mobility modeling. Infrastructure such as surveillance cameras [1], cellular towers [12] are used for vehicle distribution modeling. The relationship between vehicle movement and wireless signals is also studied [13], [25], [26]. Smartphones [27] [28] are used to track vehicles in real time. Yang et al [14] proposed a method to get the distribution and trace in real-time by using Electric Toll Collection data. These schemes take a centralized methodology to study vehicle mobility models and distributions and cannot be directly utilized to estimate the neighbor distribution for each individual vehicle.

The most relevant work to our scheme is CoReCast [29] where a bloom filter constructed at packet level is used to count the number of neighbors of a vehicle within the communication range under duplex system. CoReCast needs reliable packet communication and does not study the neighbor distribution. In contrast, PeerProbe is a distributed neighbor distribution estimation scheme without packet communication or any other prior assumption.

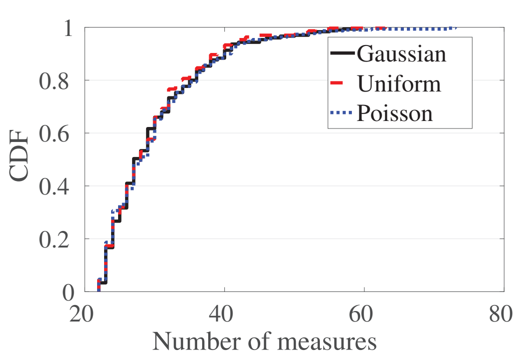


Fig. 12. CDF of the number of needed measures

VI. Conclusion

In this work, a distributed neighbor distribution estimation scheme PeerProbe has been developed in VANETs. In PeerProbe, vehicles can efficiently exchange hashed ID information through superposed OFDM symbols. With a few rough measures on the number of neighbors in randomly selected communication range, a vehicle can accurately recover the neighbor geographical distribution, leveraging the power of the combination of TV de-noising technique and the compressive sensing theory. PeerProbe needs no centralized unit or any prior knowledge about vehicle distributions. Moreover, PeerProbe is lightweight and easy to deploy with a minimal requirement on hardware. We have conducted extensive simulations. The results demonstrate the efficacy of PeerProbe.

Acknowledgements

This research was supported in part by National Natural Science Foundation of China (Grants No. 61772340, 61672151, 61972081), National Key R&D Program of China (Grants No. 2018YFC1900700), Shanghai Rising-Star Program (Grant No.17QA1400100), and DHU Distinguished Young Professor Program.

References

[1] S. Zhang, G. Wu, J. P. Costeira, and J. M. Moura, "FCN-rLSTM: Deep Spatio-Temporal Neural Networks for Vehicle Counting in City Cameras," in *Proceedings of IEEE ICCV*, 2017, pp. 3667–3676.

[2] C. Ozkurt and F. Camci, "Automatic Traffic Density Estimation and Vehicle Classification for Traffic Surveillance Systems Using Neural Networks," *Mathematical and Computational Applications*, vol. 14, no. 3, pp. 187–196, 2009.

[3] O. Asmaa, K. Mokhtar, and O. Abdelaziz, "Road Traffic Density Estimation Using Microscopic and Macroscopic Parameters," *Image and Vision Computing*, vol. 31, no. 11, pp. 887–894, 2013.

[4] N. Caceres, L. M. Romero, F. G. Benitez, and J. M. del Castillo, "Traffic flow estimation models using cellular phone data," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1430–1441, 2012.

[5] Y. Ma, M. Chowdhury, A. Sadek, and M. Jeihani, "Real-time Highway Traffic Condition Assessment Framework Using Vehicle–Infrastructure Integration (VII) with Artificial Intelligence (AI)," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 4, pp. 615–627, 2009.

[6] G. Jinli and P. Xianwu, "Monitoring the evolution of traffic on main city roads," *Journal of Transportation Systems Engineering and Information Technology*, vol. 13, no. 5, pp. 30–36, 2013.

[7] K. Singh and B. Li, "Estimation of traffic densities for multilane roadways using a markov model approach," *IEEE Transactions on Industrial Electronics*, vol. 59, no. 11, pp. 4369–4376, 2011.

[8] V. Tyagi, S. Kalyanaraman, and R. Krishnapuram, "Vehicular Traffic Density State Estimation Based on Cumulative Road Acoustics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 3, pp. 1156–1166, 2012.

[9] R. Mao and G. Mao, "Road Traffic Density Estimation in Vehicular Networks," in *Proceedings of IEEE WCNC*, 2013, pp. 4653–4658.

[10] S. Panichpapiboon and W. Pattara-Atikom, "Exploiting wireless communication in vehicle density estimation," *IEEE transactions on vehicular technology*, vol. 60, no. 6, pp. 2742–2751, 2011.

[11] A. N. Mian, I. Fatima, and R. Beraldi, "Traffic Density Estimation Protocol Using Vehicular Networks," in *Proceedings of International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, 2012, pp. 1–12.

[12] A. Thiagarajan, L. Ravindranath, H. Balakrishnan, S. Madden, and L. Girod, "Accurate, Low-energy Trajectory Mapping for Mobile Devices," in *Proceedings of USENIX NSDI*, 2011, pp. 1–14.

[13] A. Balasubramanian, R. Mahajan, A. Venkataramani, B. N. Levine, and J. Zahorjan, "Interactive WiFi Connectivity for Moving Vehicles," *ACM SIGCOMM Computer Communication Review*, vol. 38, no. 4, pp. 427–438, 2008.

[14] Y. Yang, X. Xie, Z. Fang, F. Zhang, Y. Wang, and D. Zhang, "VeMo: Enabling Transparent Vehicular Mobility Modeling at Individual Levels with Full Penetration," in *Proceedings of ACM MobiCom*, 2019, pp. 1–16.

[21] J. A. Tropp and A. C. Gilbert, "Signal Recovery from Random Measurements via Orthogonal Matching Pursuit," *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.

[15] F. Bai, D. D. Stancil, and H. Krishnan, "Toward Understanding Characteristics of Dedicated Short Range Communications (DSRC) from a Perspective of Vehicular Network Engineers," in *Proceedings of ACM MobiCom*, 2010, pp. 329–340.

[16] H. A. Omar, W. Zhuang, and L. Li, "VeMAC: A TDMA-Based MAC Protocol for Reliable Broadcast in VANETs," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1724–1736, 2013.

[17] H. Zhu, S. Chang, L. Lu, and W. Zhang, "RUPS: Fixing Relative Distances among Urban Vehicles with Context-Aware Trajectories," in *Proceedings of IEEE IPDPS*, 2016, pp. 19–35.

[18] B. H. Bloom, "Space/Time Trade-Offs in Hash Coding with Allowable Errors," *Communications of the ACM*, vol. 13, no. 7, pp. 422–426, Jul. 1970.

[19] S. G. Mallat and Z. Zhang, "Matching Pursuits with Time-frequency Dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, 1993.

[20] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal Matching Pursuit: Recursive Function Approximation with Applications to Wavelet Decomposition," in *Proceedings of IEEE Asilomar Conference on Signals, Systems and Computers*, 1993, pp. 40–44.

[22] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, 2009.

[23] E. J. Candès, J. Romberg, and T. Tao, "Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[24] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear Total Variation based Noise Removal Algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.

[25] Y. Li, C. Peng, Z. Yuan, J. Li, H. Deng, and T. Wang, "MobileInsight: Extracting and Analyzing Cellular Network Information on Smartphones," in *Proceedings of ACM MobiCom*, 2016, pp. 202–215.

[26] F. Bai, K. R. Moghadam, and B. Krishnamachari, "A Tale of Two Cities–Characterizing Social Community Structures of Fleet Vehicles for Modeling V2V Information Dissemination," in *Proceedings of IEEE SECON*, 2015, pp. 506–514.

[27] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson, "VTrack: Accurate, Energy-aware Road Traffic Delay Estimation Using Mobile Phones," in *Proceedings of ACM SenSys*, 2009, pp. 85–98.

[28] Y. Zhao, S. Li, S. Hu, L. Su, S. Yao, H. Shao, H. Wang, and T. Abdelzaher, "Greendrive: A Smartphone-based Intelligent Speed Adaptation System with Real-time Traffic Signal Prediction," in *Proceedings of ACM/IEEE ICCPS*, 2017, pp. 229–238.

[29] T. Das, L. Chen, R. Kundu, A. Bakshi, P. Sinha, K. Srinivasan, G. Bansal, and T. Shimizu, "CoReCast: Collision Resilient Broadcasting in Vehicular Networks," in *Proceedings of ACM MobiSys*, 2018, pp. 217–229.