# Semi-Persistent Resource Allocation Based on Traffic Prediction for Vehicular Communications

Ping Chu, J. Andrew Zhang, Xiaoxiang Wang, Gengfa Fang, Dongyu Wang

***Abstract*—In cellular vehicular communications, high density and mobility of vehicles require frequent resource allocation, which can cause network congestion and large signalling and processing delay. To overcome this problem, we propose a novel semi-persistent resource allocation scheme based on a two-tier heterogeneous network architecture. The architecture includes a central macro base station (MBS) and multiple roadside units (RSU). In the proposed semi-persistent scheme, the MBS pre-allocates persistent resource to RSUs based on predicted traffic, and then allocates dynamic resource upon real-time requests from RSUs while vehicles simultaneously communicate using the pre-allocated resource. A simple Space-Time k-Nearest Neighbour (ST-kNN) method is developed for short-term traffic prediction, and a geometric water-filling algorithm is developed for minimizing the relative latency. Simulation results validate the effectiveness of the proposed semi-persistent scheme in comparison with two benchmark schemes.**

***Index Terms*—Vehicular communications, resource allocation, Short-term vehicular traffic prediction, water-filling algorithm**

## I. Introduction

Vehicular communication network is an enabling technology for realizing intelligent, faster, safer transportation [1], [2]. It is characterized by a dynamic environment and high mobility. Cellular networks are capable of providing wide coverage and flexible centralized control for network resources, which is critical for achieving reliable V2X communication for fast-moving vehicles. The 3rd Generation Partnership Project (3GPP) group has defined cellular-based vehicle-to-everything (V2X) communications which include vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-pedestrian (V2P) communications [3]. In general, V2X communications support both safety and non-safety services. Targeting at decreasing traffic accidents, safety services have a more stringent requirement on delay and reliability. System latency includes both signal transmission time and the delay caused by associated signalling process, e.g., resource request and allocation.

Resource management in cellular vehicular networks is an important and challenging problem. In [4]–[6], theoretical analysis on resource management for vehicular device-to-device (V-D2D) communications underlying cellular networks is provided. In [7], a joint power control and mode selection scheme is proposed for the same networks, where V-D2D users are allowed to reuse the cellular uplink resources. However, these resource management schemes have the following limitations: (1) They rarely take the signalling latency into consideration when conducting the latency analysis; (2) frequent and numerous resource allocation requests due to high mobility can become heavy load for a central base station and cause signalling congestion and large latency. Such heavy and congested network load can significantly counteract the benefits of ultra latency optimization in the 5G standard and jeopardize the effectiveness of safety applications.

Heterogeneous vehicular network is efficient for offloading centralized traffic, improving the sum-rate capacity and supporting low-latency communications [8]–[12]. A heterogeneous vehicular network is typically a two-tier network with a central MBS and multiple distributed RSUs. In [8], [9], network performance is characterized, with particular concern on frequent handoff between RSUs. In [10], an analytical model based on clustering is proposed to characterize the network performance. In [11], [12], cognitive radio is proposed for sharing spectrum between macrocell and RSUs. These studies demonstrate the feasibility and great potentials of heterogeneous vehicular networks.

In this paper, based on the two-tier heterogeneous vehicular network, we propose a semi-persistent resource allocation scheme. This scheme aims to address the two aforementioned limitations in resource management, and can significantly reduce signalling latency while optimizing resource allocation. In the proposed scheme, the MBS provides centralized control over resource allocation for RSUs and each RSU provides direct short-range communication as well as resource allocation to vehicles within its coverage. The traffic of vehicles is regular and predictable, so is their resource requirement for communications. This scheme first pre-allocates persistent resource to each RSU based on predicted traffic, and then provides additional real-time dynamic resource allocation to RSUs with insufficiently allocated resource during pre-allocation. With pre-allocated resource, each RSU can directly provide initial resource allocation to vehicles when receiving their requests, without having to wait for the resource being assigned by the MBS. This can significantly reduce the signalling latency and mitigate congestion. By combining pre-allocation and real-time dynamic allocation, the proposed scheme is expected to achieve an excellent balance between spectrum efficiency and latency.

Performance of the proposed scheme is closely related to the accuracy of predicted traffic, the ratio between allocated

This work was supported by National Key R&D Program of China No.2018YFC 1504502. Corresponding author: Xiaoxiang Wang.

Ping Chu is with Beijing University of Posts and Telecommunications & University of Technology Sydney, Australia(Email: ping.chu@student.uts.edu.au).

J. Andrew Zhang and Gengfa Fang are with the Global Big Data Technology Centre, University of Technology Sydney, Australia (Email: {Andrew.Zhang; gengfa.fang}@uts.edu.au).

Xiaoxiang Wang and Dongyu Wang are with Beijing University of Post and Telecommunication(Email: {cpwang; dy_wang}@bupt.edu.cn).

persistent and dynamic resources, and the actual resource optimization algorithm. In this paper, we consider a segment-based road model, where a road is divided into multiple segments and the resource request and allocation are per-segment based. We first develop a cost function for the overall latency, including both communication and signalling latency, subject to the bandwidth constraints. We then develop a k-nearest neighbour (kNN) algorithm for predicting the vehicle traffic and then the resource requirement, by assuming a linear relationship between them. Based on the predicted value, we then propose an improved water-filling algorithm to generate near-optimal solution to the constrained delay minimization problem. The main contributions of this paper are summarized as follows:

- *System framework*: We propose a semi-persistent resource allocation strategy, on top of the two-tier heterogeneous architecture, for vehicular communications. It can efficiently solve the potential network congestion problem, and significantly reduce the signalling delay.
- *Traffic Prediction Algorithm*: We propose a low-complexity kNN algorithm with a weighted prediction function based on latest and historical data sets in both windowed space and time domains. The algorithm does not make any assumption on the statistical distribution of the data. It is shown to achieve excellent prediction performance when tested with real traffic data.
- *Semi-persistent resource allocation scheme*: We introduce a cost function for minimizing the relative latency, and then propose an optimization scheme for persistent and dynamic resource allocations using a geometric water-filling algorithm with individual upper bound constraints.

Extensive simulation results using practically measured traffic data demonstrate the superiority of our proposed scheme in latency reduction. Compared to benchmark schemes, our scheme is shown to achieve latency reduction up to 11.6%.

The rest of this paper is organized as follows. Section II formulates the research problem and introduces the semi-persistent resource allocation scheme. Section III presents the short-term vehicular traffic prediction algorithm using the windowed kNN method. Section IV presents the improved water-filling optimization method for resource allocation. Simulation results are provided in Section V and Section VI concludes the paper.

## II. Problem Formulation for Semi-persistent Resource Allocation

In this section, we present the system model, propose the semi-persistent resource allocation scheme, and then formulate the cost function of delay that will be optimized later.

### A. System Model

We consider an urban free-way fully covered by the LTE cellular network, as shown in Fig. 1, where a straight road with two lanes is exemplified. We assume that the road is equipped with RSUs, spaced at distances based on individual coverage. We divide the road of interest into $N$ segments, where each RSU supports the communication of vehicles in each segment. Road segmentation enables efficient spectral resource allocation and management, and offers potentials for low-latency communications. Denote the segment set as $\mathcal{N} = \{1, 2, ..., N\}$. Given two-way traffic, the density of vehicles in different segments can be different, and can also be correlated over space and time.

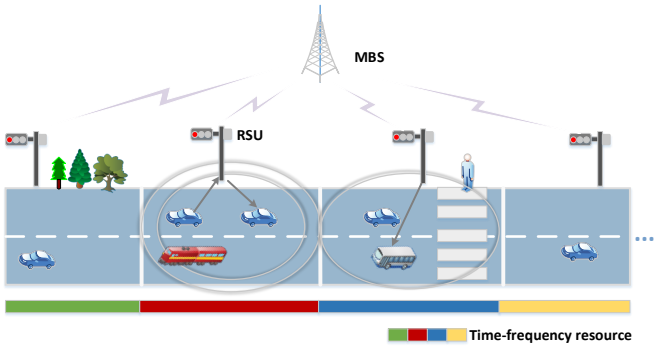



Fig. 1. RSU-cellular architecture and the segmented road model.

For vehicular communications, we consider a two-tier heterogeneous network, where a MBS provides centralized control on network resource for RSUs, and each RSU is directly responsible for providing access to vehicles in the V2I communication mode, and conducting local resource allocation for direct V2V communications within its segment. Since the communications between vehicles are mainly limited to be within a segment, they are directly managed by the corresponding RSU. When resource allocation is needed, each RSU collects requests from vehicles and lodges a request to the MBS. Once the resource is allocated, the RSU will then confirm with the vehicles.

The required network resource for communications is assumed to be linearly proportional to the vehicle traffic within the segment. We represent the requested resource as bandwidth, but it can be easily extended to general time-frequency resource blocks. Hence for each segment, the required bandwidth is linearly proportional to the vehicle traffic with a scalar $\varepsilon$.

We do not consider frequency reuse here and the frequency channels being allocated across segments are different. For frequency reuse, in the considered road model we may divide the roads into blocks, each block containing a sequence of continuous segments. The same set of frequencies can then be allocated in an increasing order to the segments in each block, so that segments in different blocks using the same frequencies are sufficiently spaced with negligible interference. Our proposed scheme can then be applied to each block straightforwardly.

### B. Semi-persistent Resource Allocation

In a conventional persistent resource allocation scheme, the MBS only allocates frequency resources to RSUs in advance based on traffic prediction. On the opposite, in a dynamic scheme, the MBS only allocates resources to RSUs upon their real-time requests, without doing pre-allocation. These two allocation schemes have respective advantages and

disadvantages. For persistent allocation, its performance largely depends on the accuracy of predicted resource demands. Although it avoids the real-time signalling loading and delay, inaccurate and insufficient prediction and resource allocation can lead to low spectrum efficiency and even longer delay. For dynamic allocation, each request and confirmation for resource allocation incurs delay, and it could also be impractical due to the instantaneous very-high loading resulted from fast-varying traffic flow.

Our semi-persistent scheme combines persistent and dynamic resource allocation, which can achieve an excellent balance between reducing the delay and improving the bandwidth efficiency. In the proposed semi-persistent scheme, the total available resource is divided into persistent and dynamic resource pools. Using predicted mean traffic and then the mean bandwidth needs, the MBS pre-allocates some resource to each RSU in a segment from the persistent pool in advance, as will be detailed in Section III. In real-time, the MBS then further allocates resources to the segments that only need additional resources upon requests from the dynamic resource pool, as will be investigated in Section IV. Our scheme can be applied to either vehicle density based prediction by assuming a linear relationship between the bandwidth requirement and the number of vehicles in one segment, or the prediction for communication bandwidth/capacity directly.

Assume that we are at time $tT_s$ where $t$ is an integer and $T_s$ is the observation interval, and we are now processing the resource allocation problem for the next time period from $tT_s$ to $(t+1)T_s$. For simplicity, we use $t$ to represent either the time $tT_s$ or the period from $(t-1)T_s$ to $tT_s$ hereafter. We define some symbols and rules as follows:

- The total available bandwidth, total allocated persistent bandwidth and total remained dynamic bandwidth are denoted as $B$, $B^P$, and $B^D$ respectively. We have $B^D + B^P \leq B$, as there could be unallocated bandwidth due to constraints applied in our optimization problem. The current time is $t$;
- The persistent bandwidth allocated to the $n$-th segment for $(t+1)$ is $B_n^p(t+1), n \in \mathcal{N}$. Therefore $B^P = \sum_{n=1}^{N} B_n^p(t+1)$.
- The dynamic bandwidth allocated to each segment for $(t+1)$ is $B_n^d(t+1)$, where $\sum_{n=1}^{N} B_n^d(t+1) \leq B^D$. In addition, the actual bandwidth requirement at $t+1$ is denoted as $B_n^{\text{req}}(t+1), n \in \mathcal{N}$ and $B_n^d(t+1)$ needs to satisfy $B_n^d(t+1) \leq B_n^{\text{req}}(t+1) - B_n^p(t+1)$.
- The total allocated bandwidth $B_n(t+1)$ to the $n$-th segment at $t+1$ is then given by $B_n(t+1) = B_n^p(t+1) + B_n^d(t+1)$.
- When $B_n^p(t+1) < B_n^{\text{req}}(t+1)$, the $n$-th RSU will request dynamic resources from the MBS. Let the signalling and processing delay for the dynamic resource request and allocation be $t^s$. In this paper, we assume that $t^s$ is a constant for all segments.
- The ground-truth delay $\tau_n^{\text{exp}}(t+1)$ denotes the "true" delay if the requested bandwidth $B_n^{\text{req}}(t+1)$ is fully allocated.

Referring to Fig. 2, the system operation and the actual latency in the proposed scheme are described as follows. We divide the communication process within each segment into two stages. In the first stage, a RSU schedules vehicle communications using the pre-allocated persistent bandwidth, meanwhile it will start new resource request with the MBS upon receiving the actual bandwidth requests from vehicles. After communication for a period $t^s$ that corresponds to the signalling and processing delay for the new bandwidth request, the RSU receives updated bandwidth allocation and the second stage starts. The RSU now schedules the communication with the combined persistent and dynamic bandwidth. If no dynamic bandwidth is allocated, the segment continues using only the persistent bandwidth. We ignore the signalling delay within a segment during this process because it has little impact on our proposed scheme. Note that the prediction and persistent bandwidth allocation is assumed to be done in a relatively long interval of tens to hundreds of timeslots, while dynamic allocation is done within each timeslot, with a period of typically 100 ms. For vehicle density based prediction, the interval can be tens of seconds as the density will not change significantly during a short period; and for network-bandwidth-demand based prediction, it can be smaller.

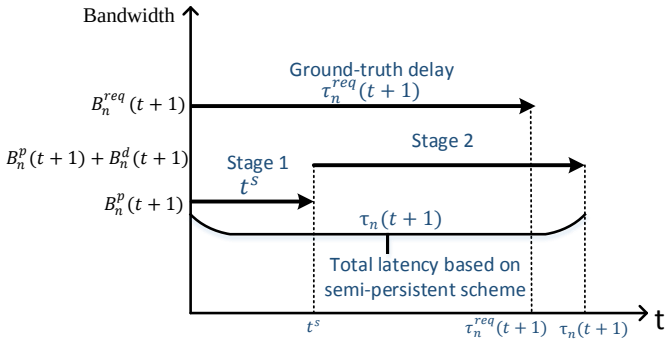



Fig. 2. Illustration of latency for each timeslot in the $n$-th segment for the period $(t, t+1)T_s$. Each period contains tens to hundreds of such timeslots.

### C. Formulation of the Cost Function

We now formulate an average relative latency as the cost function for our optimization problem. We aim to minimize the average relative latency by optimizing the allocation of persistent and dynamic bandwidth.

Based on Shannons capacity formula, the data rate is linearly proportional to the bandwidth. Here, we approximate the data rate as the product of the bandwidth and a segment-dependent constant $c_n$ accounting for possibly different signal-to-noise ratios in different segments. Let the total number of bits to be transmitted be $S_n(t+1)$ in time $t+1$ and in the $n$-th segment. Then the bits being transmitted during the first stage will be $c_n B_n^p(t+1)t^s$ and the transmission latency in the second stage will be

$$\tau_n^a(t+1) = \frac{S_n(t+1) - c_n B_n^p(t+1)t^s}{c_n(B_n^p(t+1) + B_n^d(t+1))}. \quad (1)$$

The total latency is then given by

$$\tau_n(t+1) = \tau_n^a(t+1) + t^s. \quad (2)$$

Therefore, the ground-truth expected latency that corresponds to the actually required bandwidth is

$$\tau_n^{\exp}(t+1) = \frac{S_n(t+1)}{c_n B_n^{\text{req}}(t+1)}. \quad (3)$$

Note that $\tau_n^{\exp}(t+1)$ is typically specified in a standard, for example, $\tau_n^{\exp}(t+1) = 100$ ms according to 3GPP TR36.885.

If the allocated persistent bandwidth exceeds the actually required bandwidth in one segment, i.e., $B_n^p(t+1) \geq B_n^{\text{req}}(t+1)$, the RSU does not need to request more bandwidth from MBS. Hence, we let $\mathcal{U}$ and $\mathcal{V}$ denote the index sets of the RSUs that do not and do request dynamic bandwidth, respectively. The size of the sets are $U$ and $V$. We can then define the averaged total relative latency $T(t+1)$ as

$$\begin{aligned} T(t+1) &= \frac{1}{N}\sum_{n=1}^{N}\frac{\tau_n(t+1)}{\tau_n^{\exp}(t+1)} \\ &= \frac{1}{N}\left\{\sum_{u\in\mathcal{U}}\frac{B_u^{\text{req}}(t+1)}{\min\left\{B_u^p(t+1), B_u^{\text{req}}(t+1)\right\}} + \right. \\ &\left. \sum_{v\in\mathcal{V}}\left[\frac{B_v^{\text{req}}(t+1) - B_v^p(t+1)t_v^s}{B_v^p(t+1)+B_v^d(t+1)} + t_v^s\right]\right\} \\ &= \frac{1}{N}\left\{U + \sum_{v\in\mathcal{V}}\left[\frac{B_v^{\text{req}}(t+1) - B_v^p(t+1)t_v^s}{B_v^p(t+1)+B_v^d(t+1)} + t_v^s\right]\right\}, \quad (4) \end{aligned}$$

where $N = U + V$ and $t_v^s = t^s/\tau_v^{\exp}(t+1)$ is the relative signalling latency. Note that both $U$ and $V$ depend on the allocated persistent resource.

Our objective is to determine optimal allocations for persistent and dynamic resources so that $T(t+1)$ is minimized. The optimization problem is formulated as

$$\min_{\{B_v^d(t+1)\}_{v=1}^{V}} \quad T(t+1) \quad (5a)$$

$$\text{subject to} \quad 0 < \sum_{v=1}^{V} B_v^d(t+1) \leq B^D; \quad (5b)$$

$$0 \leq B_v^d(t+1) \leq B_v^{d_u}(t+1), \ \forall v; \quad (5c)$$

where

$$B_v^{d_u}(t+1) = B_v^{\text{req}}(t+1) - B_v^p(t+1) \quad (6)$$

is the upper bound of $B_v^d(t+1)$. Equation (5b) represents a general constraint on the total available bandwidth for dynamic allocation. The constraint in (5c) ensures that no more bandwidth than the actually needed will be allocated to one RSU. It guarantees the fairness during resource allocation and enables the overall optimality attained.

## III. Vehicular Traffic Prediction based on kNN

Accurate short-term traffic prediction is important for efficient resource management. Existing techniques can be classified into two main classes: parametric model-based and non-parametric based predictions [13]. The former adopts explicit mathematical models such as auto-regressive-integrated-moving-average (ARIMA) and its variants [14], [15]. The latter does not assume a model, and it predicts the traffic using machine learning techniques such as artificial neural networks (ANN) [16] and kNN [17]–[19]. In this paper, we choose kNN for traffic prediction for its simplicity and efficiency in solving non-linear problems which typically exist in traffic prediction [20]. Its efficiency can be seen from our simulation results using practically measured traffic data as will be presented in Section V. It is noted that although our prediction scheme is presented by referring to the vehicular traffic prediction, it can be readily extended to direct network traffic prediction.


Fig. 3. Processing flow of the ST-kNN algorithm.

In this paper, we focus on short time traffic prediction (30 seconds) using a space-time kNN (ST-kNN) algorithm. Previous works on the kNN method mainly focused on the whole road, which did not consider the difference and correlation between different parts on the road. In order to apply it into our problem where the traffic for each segment needs to be predicted, we introduce a space-time windowing method to exploit the data correlation, which can achieve better prediction performance. Our proposed kNN method predicts the future traffic using a weighted prediction function based on latest and historical data sets in both windowed space and time domains.

Fig. 3 depicts the processing flow of the proposed kNN method. The search procedure finds the nearest neighbours of each segment in the time and spatial domains from historical observations that are most similar to the current conditions. The nearest neighbors are then used as the input to a linear function to generate the prediction. The main process of our improved ST-kNN algorithm is summarized below and described in detail next.

1) Define an appropriate state space;
2) Decide the window size;
3) Define a distance metric to determine nearness of historical data to the current conditions; and
4) Select a prediction method given a collection of nearest neighbours.

### A. State Space

*State vector* is a standard for comparing the current data and the historical database. It can have a significant impact on the prediction accuracy. Here the state vector contains the traffic values measured over a continuous period $(t, t-1, ..., t-q)$, where $q$ needs to be selected carefully to avoid resulting in excessive similar values when comparing the current observation data and the historical database. We will show the impact of the value $q$ on system performance in Section V. For Segment $n (n \in \mathcal{N})$, the current traffic state vector is given by

$$\mathbf{v}_n(t) = [V_n(t),\ V_n(t-1),\ V_n(t-2), ...,\ V_n(t-q)] \quad (7)$$

where $V_n(t)$ is the traffic value at time $t$.

The state vector $\mathbf{v}_n(t)$ is then compared with neighbouring blocks in both space and time domains from the historical database. Selection of these blocks will be discussed in next subsection.

### B. Choice of the Window Size $\nu$

Considering one traffic direction on the road, the traffic flow in different segments can be highly correlated over both the spatial and temporal domains. Fig. 4. illustrates the correlation of the traffic volume between different segments. The same color stands for the mostly correlated traffic, e.g., $V_3(t)$ in Segment 3 at time $t$ is strongly correlated with $V_1(t-2)$, $V_2(t-1)$, $V_4(t+1)$, and $V_5(t+2)$.

In this paper, the window size is selected according to the correlation between different segments dependent on the movement speed of vehicles. The vehicle speeds could vary significantly between different time period of a day such as peak and off-peak time. The time periods can be classified according to the variation of the speed or the average speed using e.g., an unsupervised classification algorithm [21]. In this paper, we use the average speed $\overline{v}$.

For predicting the traffic in Segment $n$, we apply windowing across segments of distance $L = \overline{v}T_p$ in the spatial domain and over a period of $T_p$ in the time domain, where $T_p$ is the prediction interval. During this period, vehicles at Segment $m$, the furthest to Segment $n$, will travel to Segment $n$. In other words, the current traffic flow rate at time $t$ in Segment $n$ is strongly correlated with those up to Segment $m$ at time $t - T_p$. The actual number of blocks will be the quantized values for $L$ and $T_p$ with respect to the length of segments and $T_s$, respectively. Let $\nu_{nm}$ be the relative index of samples and $0 < \nu_{nm} \leq \lceil T_p/T_s \rceil$ where $\lceil T_p/T_s \rceil$ denotes the least integer larger than $T_p/T_s$. Note that the selected state vector in the neighbouring blocks can be represented as

$$\begin{cases} \mathbf{v}_m(t+\nu_{nm}) = [V_m(t+\nu_{nm}),\ V_m(t+\nu_{nm}-1), \ldots, \\ \qquad\qquad V_m(t+\nu_{nm}-q)]\,, \text{ when } m > n, \\ \mathbf{v}_m(t-\nu_{nm}) = [V_m(t-\nu_{nm}),\ V_m(t-\nu_{nm}-1), \ldots, \\ \qquad\qquad V_m(t-\nu_{nm}-q)]\,, \text{ when } m < n. \end{cases} \tag{8}$$

When $m = n$, then $\nu_{nm} = 0$. Here, the window size $\nu$ based on speed and distance is only suitable for one-way road.

### C. Distance Metric of Traffic Data

Learning a good distance metric in feature space is crucial in real-world application [22]. Here we use Euclidean distance, a common distance metric between real-time and historical traffic data used in kNN prediction [17], [23]. It is given by

$$l_i = \frac{\sum_{i_q=1}^{q} \sqrt{(V_n(t-i_q) - V_m(t \pm \nu_{nm} - i_q))^2}}{q}, \tag{9}$$

where $V_n(t-i_q)$ is the traffic flow of segment $n$ at the time $(t-i_q)$ and $V_m(t \pm \nu_{nm} - i_q)$ is the traffic volume of segment $m$ which is strongly correlated with $V_n(t-i_q)$ in the historical database, $l_i$ represents the Euclidean distance of the two traffic state vectors.

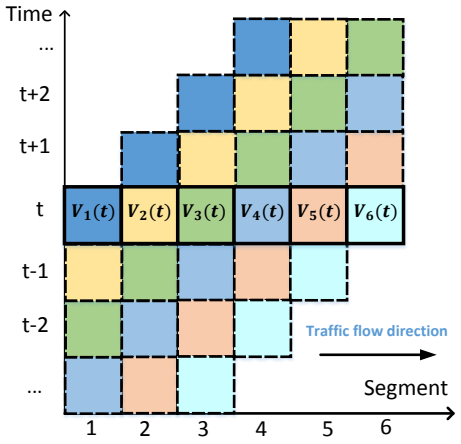



Fig. 4. Illustration of traffic correlation and the windowing concept.

### D. Prediction Function

Based on the Euclidean distance values computed for the blocks in the windowed spatial and temporal domains, $k$ nearest neighbours with the least distance values are found from the historical database. We then use a weighted linear function [24] [page 2-3] to predict the traffic. The predicted traffic flow $V_n(t+1)$ in segment $n$ at time $(t+1)$ is given by

$$V_n(t+1) = \sum_{i=1}^{k} a_i V_m(i, t \pm \nu_{nm} + 1), \tag{10}$$

where the weight $a_i$ is defined as

$$a_i = \frac{l_i^{-1}}{\sum_{i=1}^{k} l_i^{-1}}. \tag{11}$$

Assuming a linear relationship between the vehicular and network traffic, the predicted bandwidth for time $t+1$ can be obtained as $\hat{b}_n^p(t+1) = \varepsilon V_n(t+1), n \in \mathcal{N}$, where $\varepsilon$ is the mapping coefficient.

### E. Application of a Scaling Coefficient

Instead of directly using the predicted value to allocate the bandwidth to segments, we apply a scaling factor $\theta$ to it and the pre-allocated persistent bandwidth for the $n$-th segment is given by

$$B_n^p(t+1) = \theta \hat{b}_n^p(t+1), \tag{12}$$

with

$$0 < \theta \leq \frac{B}{\sum_{n=1}^{N} \hat{b}_n^p(t+1)}. \tag{13}$$

There are two reasons that we introduce $\theta$ here. Firstly, the total available bandwidth could be smaller than the sum of the predicted bandwidth for all segments, and hence we need to do a scaling. Secondly, even when the total bandwidth is sufficient, allocating less resource than the predicted value may achieve overall better performance given the fluctuation of the instantaneous traffic. We will investigate and propose a rule-of-thumb for selecting the values of $\theta$ for a given ratio between the total available bandwidth and the statistical mean of the required bandwidth in Section V.

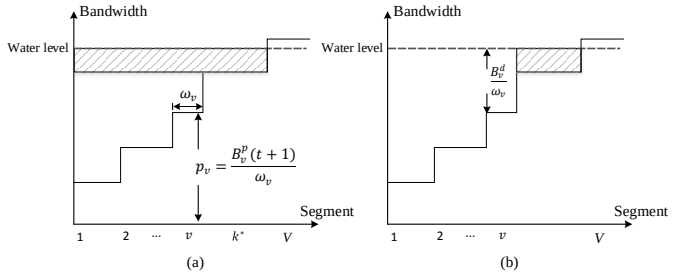



Fig. 5. Illustration of the BC-WFA water-filling for dynamic resource allocation.

## IV. Minimization of Average Relative Latency

Assume that the total available bandwidth $B$ is known. After allocating the persistent bandwidth using (12), we now get the remaining bandwidth $B^D(t+1)$ as

$$B^D(t+1) = B - \theta \sum_{n=1}^{N} \hat{b}_n^p(t+1), \tag{14}$$

which is available for dynamic allocation based on the actual requests $B_n^{\text{req}}(t+1)$ from RSUs.

Our goal is now to find the optimal dynamic allocation $B_n^d(t+1)$ for (5), with a given $\theta$. Note that strictly speaking, an optimal solution will require optimization for both $\theta$ and $B_n^d(t+1)$. Since there exists no closed-form statistical distribution for the required bandwidth, it is hardly possible to find a closed-form expression for the optimal $\theta$. Therefore, we only test a series of values for $\theta$ in the simulation in Section V, and disclose the optimal range of $\theta$ based on practically measured channels. Note that in the case of $B_n^p(t+1) \geq B_n^{\text{req}}(t+1)$ when the allocated persistent bandwidth is sufficient, a RSU will not request for dynamic bandwidth.

For the optimization problem in (5), we can separately consider two situations. In the first situation, $B^D \geq \sum_{v=1}^{V} B_v^{d_u}(t+1)$, which means there is sufficient bandwidth that can satisfy every RSU's request. In this case, the MBS can just allocate the bandwidth to each RSU as being requested. In the second situation, $B^D < \sum_{v=1}^{V} B_v^{d_u}(t+1)$, which implies that $\sum_{v=1}^{V} B_v^d(t+1) = B^D$ and not all requests can be satisfied. An optimization algorithm is needed for allocating the dynamic bandwidth in this situation.

It is easy to verify that $T(t+1)$ in (5) is a convex function of $B_v^d(t+1)$. Hence the optimization problem meets the Karush-Kuhn-Tucker (KKT) conditions and can be generally solved by convex optimization algorithms such as linear programming. To provide a closed-form solution and shed more insights on the design, we propose a bandwidth-constrained water-filling algorithm (BC-WFA) next.

Under the conditions of $B^D < \sum_{v=1}^{V} B_v^{d_u}(t+1)$, and $\sum_{v=1}^{V} B_v^d(t+1) = B^D$, the BC-WFA algorithm optimizes the dynamic bandwidth allocation, in order to minimize the relative latency in (5). The algorithm is developed based on the geometric water-filling method in [25]. The detailed steps are presented in Algorithm 1, with the concept illustrated in Fig. 5, where the dashed line denotes the water level.

**Algorithm 1** BC-WFA Algorithm.

**Input:** vector $\{p_v\}$, $\{w_v\}$,$\left\{B_v^{d_u}\right\}$ for $v = 1, 2, ..., V$, the set $E = \{1, 2, ..., V\}$, and $B^D$.
**Output:** $\left\{B_v^d\right\}$

1: **while** $E \neq \emptyset$ **do**
2: Use (18)-(20) to compute $\left\{B_v^d\right\}$.
3: $\Lambda \leftarrow \left\{v \mid B_v^d > B_v^{d_u}, v \in E\right\}$.
4: **if** $\Lambda \neq \emptyset$ **then**
5: **if** $v \in \Lambda$ **then**
6: $B_v^d = B_v^{d_u}$.
7: **end if**
8: $E \leftarrow E \backslash \Lambda$, $B^D = B^D - \sum_{v \in \Lambda} B_v^{d_u}$.
9: **else**
10: Output $\left\{B_v^d\right\}$ as $v \in E$.
11: **end if**
12: **end while**

Firstly, let $\{B_v^p(t+1)\}_{v=1}^{V}$ be a sorted sequence, which is positive and monotonically increasing. Let $p_v$ denote the "step depth" of the $v$th stair and $w_v$ represents the weighted coefficient as shown in Fig. 5(a). They are given by

$$w_v = \sqrt{B_v^{\text{req}}(t+1) - B_v^p(t+1)t_v^s}, \text{ for } v = 1, 2, ..., V. \tag{15}$$

$$p_v = \frac{B_v^p(t+1)}{w_v}, \text{for } v = 1, 2, ..., V. \tag{16}$$

Let $B_2^D(k)$ represent the water volume (dynamic bandwidth) above the $k$th step, as shown in the shadowed area in Fig. 5(a). It is given by

$$B_2^D(k) = \left[B^D - \sum_{v=1}^{k-1}(p_k - p_v)w_v\right]^+, \text{for } k = 1, ...., V \tag{17}$$

where $(x)^+ = \max\{0, x\}$.

According to the GWF algorithm without the individual upper bond constraint (5c) in [25], the explicit solution to (5) is given by

$$\begin{cases} B_v^d = \left[\frac{B_{k^*}^d}{w_{k^*}} + (p_{k^*} - p_v)\right] w_v, 1 \leq v \leq k^*, \\ B_v^d = 0, k^* < v \leq V, \end{cases} \tag{18}$$

where

$$k^* = \max\left\{k \mid B_2^D(k) > 0, 1 \leq k \leq V\right\}. \tag{19}$$

Here $k^*$ can be treated as the highest step under water. The allocated dynamic bandwidth for this step is

$$B_{k^*}^d = \frac{w_{k^*}}{\sum_{v=1}^{k^*} w_v} B_2^D(k^*), \tag{20}$$

which is illustrated by the shadowed area in Fig. 5(b).

In Algorithm 1, the constraints (5c) are checked in Steps 4-7. Allocations do not satisfy the constraints are set as the individual upper bound and then removed in Step 8. The process repeats until all allocations are completed with constraints satisfied.

## V. Simulation Results

In this section, we present simulation results for the proposed BC-WFA semi-persistent resource allocation scheme. As mentioned before, we would like to use real data for the simulation. Since the network model for vehicular communications is not available yet, we base our simulation on real vehicle traffic, and assume a linear mapping with $\varepsilon = 1$ between the network traffic (required bandwidth) and the vehicle traffic. Actually, we can see that $\varepsilon$ does not have a direct impact on the objective function in (4) only if the linear mapping holds. In the case when this relationship does not hold, for example, when there are a burst of bandwidth requests, the proposed semi-persistent scheme can still cope with this via dynamically allocating bandwidth in real time using the proposed water-filling algorithm. This is because the vehicular traffic prediction is only used for allocating persistent bandwidth. However, the burst will translate into increased variance and the overall performance may be degraded.

We use the vehicular traffic dataset collected from a section of Interstate 80 (I-80) freeway located in Emeryville, California [26]. The dataset includes both traffic density and average vehicle speed. The study area was approximately 500 meters (1,640 feet) in length and consisted of six freeway lanes, including a high-occupancy vehicle (HOV) lane. The section includes six traffic stations and the time interval of the collected data is 30s within 10 days. Traffic flow data from the first nine days is selected as the historic database and data collected from the last day is used as the test data. Assuming that RSUs co-locate with these traffic stations. The road is accordingly divided into six segments numbered as 1 to 6. We consider a one-way traffic with direction from Segment 1 to 6. With such data, we are capable of doing prediction every 30s. As discussed before, there is no particular requirement on the interval of prediction, and hence the interval here is indicative only.

Our simulation is based on a setup where the 30s interval is divided into many timeslots (200 in this paper). During the interval, the required bandwidth in each timeslot in each segment could be varying and is assumed to follow a Gaussian distribution. Let the measured real freeway traffic over the 30s interval in [26] be the average of this distribution, denoted as $\overline{B}_n^{\text{req}}(t+1)$ for the time interval $[tTs, (t+1)T_s]$, with $T_s = 30$s. The actual bandwidth requirement is then assumed to follow the Gaussian distribution with mean $\overline{B}_n^{\text{req}}(t+1)$ and variance $0.2\overline{B}_n^{\text{req}}(t+1)$, i.e., $f(B_n^{\text{req}}(t+1)) \rightarrow \mathcal{N}\left(\overline{B}_n^{\text{req}}(t+1), 0.2\overline{B}_n^{\text{req}}(t+1)\right)$. Based on the predicted average traffic, persistent bandwidth will be requested and pre-allocated; the actually required bandwidth in each timeslot is then generated following the Gaussian distribution, and dynamic bandwidth is then allocated if requested. Note that our scheme does not exploit and hence does not rely on the actual traffic models. But its performance may be affected by the models indirectly. For example, a distribution with larger variance can lead to lower efficiency of the proposed scheme.

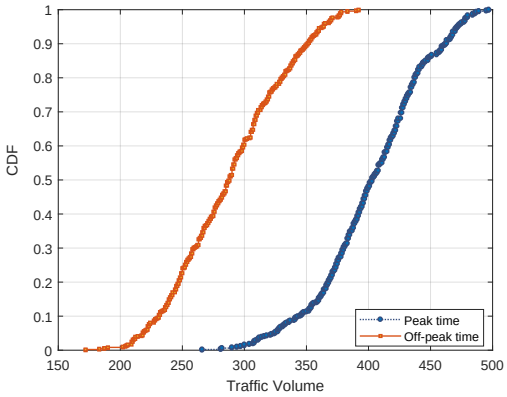



Fig. 6. The traffic volume during off-peak and peak time period, respectively.

TABLE I
Obtained best values for parameters $k$ and $q$.

| Segment n | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| k | 15 | 15 | 18 | 15 | 15 | 15 |
| q | 5 | 5 | 4 | 5 | 5 | 5 |

### A. Prediction Performance

In order to evaluate the performance of the proposed prediction algorithm, we define and use Mean Absolute Percent Error (MAPE) [27] as a performance metric

$$\text{MAPE} = \frac{1}{n}\sum_{j=1}^{n}\left|\frac{V_j^p - V_j^r}{V_j^r}\right|, \tag{21}$$

where $V_j^p$ is the predicted traffic flow and $V_j^r$ is the real traffic flow. $n$ is the number of predictions. Larger MAPE means worse prediction performance.

We first look for the best parameter values for the ST-kNN algorithm. To determine a proper window size $\nu_{nm}$ which is related to the moving speed, we adopt an ISODATA algorithm and classify a day into off-peak and peak time periods based on vehicle moving speed [21]. Fig. 6 shows the cumulative distribution function (CDF) of the average traffic for peak and off-peak periods using the real traffic data. The window size $\nu_{nm}$ can then be calculated for different time periods, and may vary across segments. Using segment 5 as an example at peak time (6:30-10 am), we get $\nu_{56} = 1, \nu_{54} = 1, \nu_{53} = 2, \nu_{52} = 3, \nu_{51} = 4$, based on the average speed and distance between segments [26].

We then determine the values for $k$ and $p$. Fig. 7 shows the MAPE value of the prediction for Segment 5 using ST-kNN with different $k$ and $q$ values. When $k = 15$ and $q = 5$, the algorithm is found to achieve the best accuracy, with the averaged MAPE of 0.098 for the whole day. The best values for parameters $k$ and $q$ for all segments are shown in Table I. In practical implementation, a trial-and-update process can be applied regularly to decide their best values for the next time period. Since the characteristics of the traffic flow in a certain area vary slowly, such updates can be done slowly and at a low computational cost due to the simplicity of the ST-kNN algorithm.

The predicted traffic flow, as well as the actual data, are shown in Fig. 8(a). For comparison, we also presented the

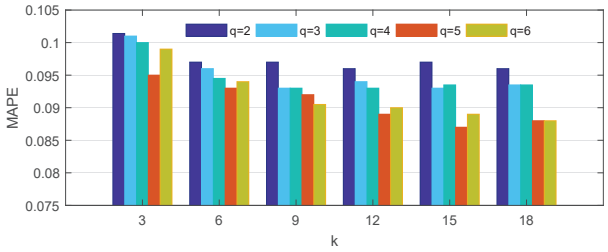
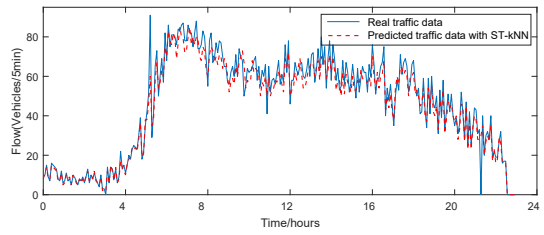
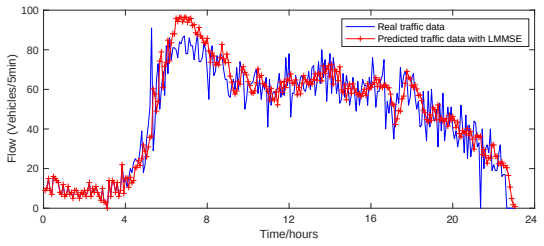



Fig. 7. Prediction results with different $k$ and $q$ values in Segment 5.




(a)




(b)

Fig. 8. Predicted and real traffic flow in a day in Segment 5 using (a) ST-kNN and (b) LMMSE algorithm. Time interval between the plotted samples is 5 minutes. The average MAPE values for ST-kNN and LMMSE are 0.098 and 0.1063, respectively.

prediction results in Fig. 8(b) for the least minimum mean square error (LMMSE) algorithm, which is widely used for channel and traffic prediction [28], [29]. The size of the adopted prediction correlation matrix in LMMSE is $10 \times 10$, and its complexity is much higher than the proposed ST-kNN due to the matrix inversion operation. For LMMSE, the averaged MAPE is 0.1063, comparable to that of the ST-kNN algorithm. The accuracy of LMMSE predictor can be improved with increasing the size of the prediction correlation matrix, at a higher complexity. The figure shows that LMMSE also intends to smooth the output of prediction, and hence is not as accurate as ST-kNN for predicting small-scale fluctuations.

### *B. Latency and Bandwidth Efficiency*

Based on requirements specified by 3GPP TR 36.885 [3], we adopt $\tau_n^{\exp}(t+1) = 100$ ms and the signalling delay $t^s$ is set to vary from 5ms to 25ms. Note that different latency values can be set for different segments in our scheme, for example, more stringent latency values such as $\tau_n^{\exp}(t+1) = 20$ ms can be used in certain safety critical cases like truck platooning. The relative latency $T(t+1)$ is obtained for each timeslot and is then averaged over all timeslots.

For comparison, we use the following two benchmark schemes. One is the conventional purely dynamic allocation scheme (DS) where water-filling algorithm is used to allocate the total available bandwidth to each RSU in real time, with the goal of minimizing average relative latency. This corresponds to the case of $\theta = 0$ in the proposed semi-persistent scheme (SPS) and we can denote the delay as $T(t+1)|_{\theta=0, n\in\mathcal{N}}$. The other one is a purely persistent scheme (PS) where the total bandwidth is all allocated to RSUs based on predicted traffic without the following dynamic allocation. The optimal relative latency for the persistent scheme is given by

$$T(t+1) = \frac{1}{N}\sum_{n=1}^{N} \frac{B_n^{\text{req}}(t+1)}{\min\left\{\theta_{\max}\hat{b}_n^p(t+1), B_n^{\text{req}}(t+1)\right\}}, \quad (22)$$

where $\theta_{\max} = \dfrac{B}{\sum_{n=1}^{N}\hat{b}_n^p(t+1)}$.

The total bandwidth $B$ may also be optimized, which is beyond the scope of this paper. Here, we simply set it as a scaled value of the statistical mean of the total requested bandwidth, i.e., $B = a\mathrm{E}_t(\sum_n B_n^{\text{req}}(t))$, where $a$ is the scalar close to 1, and $\mathrm{E}_t(\cdot)$ denotes the averaging operation over time. We will study its impact on the performance of the proposed scheme and on the coefficient $\theta$ shortly.

We also define another performance metric, the bandwidth efficiency as

$$\rho = \frac{\sum_{n=1}^{N}\min\left\{B_n^p(t+1) + B_n^d(t+1), B_n^{\text{req}}(t+1)\right\}}{B}, \quad (23)$$

which characterizes the efficiency of bandwidth usage.

We first study whether there is an optimal $\theta$ that minimizes the overall latency for the proposed scheme. Figs. 9 and 10 present the average relative latency and bandwidth efficiency for the proposed SPS and DS, respectively, with $a = 0.9$ and $t^s = 15$ms. Note that the curves for DS are level straight line as the scheme is unrelated to $\theta$. For PS, the value of $\theta_{\max}$ is fixed in each segment. Hence we directly provide the averaged values across segments, which are $1.1650$ and $1.1404$ for the relative latency and $85.45\%$ and $87.91\%$ for the bandwidth efficiency, for off-peak and peak time respectively. We can see that the proposed SPS provides the lowest latency in the three schemes for most of the $\theta$ values ($\theta > 0.7$). Its bandwidth efficiency is always better than PS, and is close to DS when $\theta < 0.9$ and then the gap increases after that. The increased gap is due to the fact that in the proposed scheme, the allocated persistent bandwidth grows with $\theta$ increasing, which causes $B_n^p(t+1) > B_n^{\text{req}}(t+1)$ for some segments. We can also observe that the latency curves for the proposed scheme are somewhat convex, with optimal values of $\theta$ in the range of approximately $[1.04, 1.1]$ and $[0.98, 1.2]$, respectively.

In Fig. 11, we further show the bar-plot for the optimal values of $\theta$ for different values of $B$ through varying the scalar $a$. It is clear that the ranges of optimal $\theta$ increase consistently with $a$ increasing.

In Fig. 12, we demonstrate how the average relative delay varies with the total bandwidth. For the proposed semi-persistent scheme, we present the results for the cases where both the optimal $\theta$ and $\theta = a$ are used. The proposed SPS achieves consistently lower latency than the two benchmark schemes. With $a = 1$, the achieved average latency is only

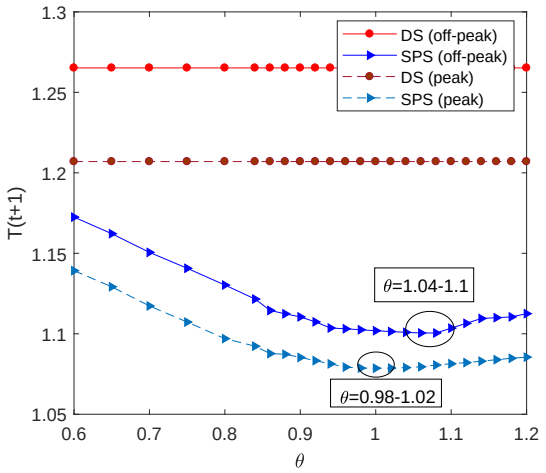



Fig. 9. Average relative latency versus $\theta$ for DS and SPS during off-peak and peak time. For PS, the averaged latencies are 1.1650 and 1.1404, respectively. Signalling latency is $t^s = 15$ms and $a = 0.9$.

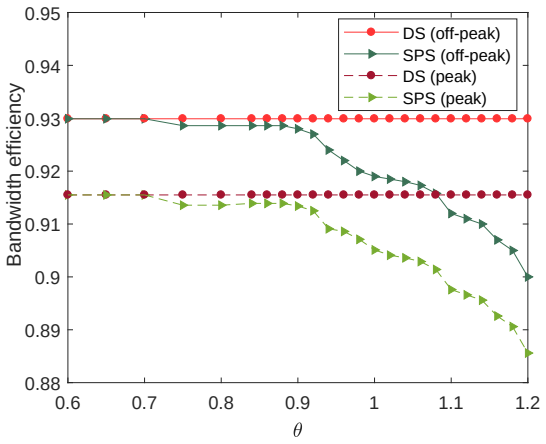



Fig. 10. Bandwidth efficiency versus $\theta$ for DS and SPS during off-peak and peak time. The averaged bandwidth efficiencies for PS are 85.45% and 87.91%, respectively.

5% more than the expected one. The latency gaps between the two values of $\theta$ are also small, which indicates that $\theta = a$ can be simply used as a rule-of-thumb for the proposed scheme.

Finally, we evaluate how the latency varies with the relative signalling delay $t_n^s$ for the three schemes. The simulation results are presented in Fig. 13, where the optimal value of $\theta$ is used in the SPS. As expected, the relative latency increases linearly with $t_n^s$ for the DS and it remains as a constant for the PS. Across the simulated range of $t_n^s$, the proposed SPS grows slowly with $t_n^s$ increasing, and always achieves the lowest latency. The bandwidth efficiency is unrelated to the signalling delay and therefore is not presented.

## VI. Conclusions

We proposed a novel semi-persistent resource allocation scheme on top of a two-tier heterogeneous network for vehicular communications. The scheme can improve bandwidth efficiency, avoid network congestion, and reduce the processing latency significantly. We proposed a simple and effective ST-kNN method for predicting the short-term traffic

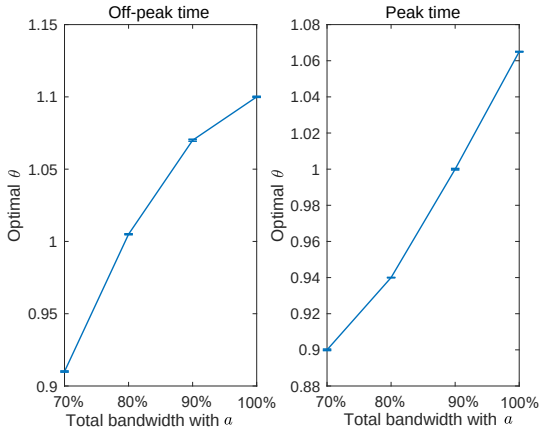



Fig. 11. Optimal $\theta$ values for different total bandwidth $B = a\mathrm{E}_t(\sum_n B_n^{\mathrm{req}}(t))$ for SPS.

flow by considering the correlation window in both time and spatial domains according to the vehicle moving speed. Based on the total available bandwidth and the predicted resource needs mapped from the predicted vehicle traffic, an improved water-filling algorithm is proposed to optimally allocate the resource to each RSU. By combining pre-allocation of persistent resource and dynamic resource allocation in real time, significant delay linked to resource allocation can be reduced, with negligible degradation on spectrum efficiency. Supported by simulation results with real traffic data, the proposed semi-persistent scheme over the RSU-cellular architecture is effective and promising for vehicular communications.

Note that although our scheme was presented by referring to a free-way road model, with some minor adaptation it can also be applied to more complex environment such as dense urban area with crossroads. One major adaptation would be setting up the window size $\nu$ in the ST-kNN algorithm with a reasonable value but not directly through the average vehicular speed. With rigorous network traffic models, it is also possible to derive the optimal value of $\theta$ analytically to replace the current rule-of-thumb.

## References


[1] G. Karagiannis, O. Altintas, E. Ekici, G. Heijenk, B. Jarupan, K. Lin, and T. Weil, "Vehicular networking: A survey and tutorial on requirements, architectures, challenges, standards and solutions," *IEEE communications surveys & tutorials*, vol. 13, no. 4, pp. 584–616, 2011.

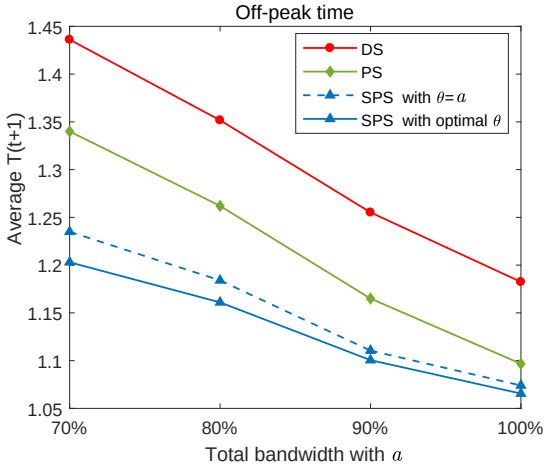
[2] H. Vahdat-Nejad, A. Ramazani, T. Mohammadi, and W. Mansoor, "A survey on context-aware vehicular network applications," *Vehicular Communications*, vol. 3, pp. 43–57, 2016.

[3] 3rd Generation Partnership Project, "Study on LTE-based V2X services," Technical Specification Group Radio Access Network, Tech. Rep. 3GPP TR 36.885 V2.0.0, 2016.

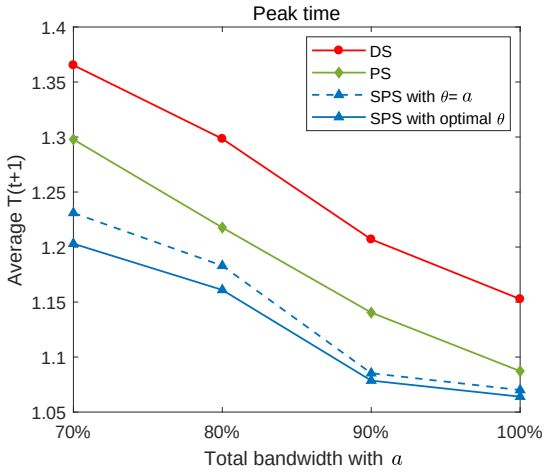
[4] L. Liang, G. Y. Li, and W. Xu, "Resource allocation for D2D-enabled vehicular communications," *IEEE Transactions on Communications*, vol. 65, no. 7, pp. 3186–3197, 2017.

[5] W. Sun, E. G. Ström, F. Brännström, K. C. Sou, and Y. Sui, "Radio resource management for D2D-based V2V communication," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6636–6650, 2016.

[6] J. Sun, Z. Zhang, C. Xing, and H. Xiao, "Uplink resource allocation for relay-aided device-to-device communication," *IEEE Transactions on Intelligent Transportation Systems*, 2018.




(a)




(b)

Fig. 12. Variation of average relative latency with the total bandwidth. Signalling delay is set as 15 ms.

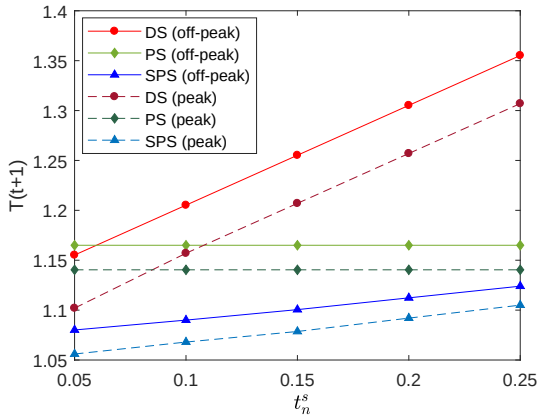



Fig. 13. The average relative time with varying relative signalling delay for three resource allocation schemes, where $\theta = 1.06$ and $\theta = 0.98$ for off-peak and peak time, respectively.


[7] N. Cheng, H. Zhou, L. Lei, N. Zhang, Y. Zhou, X. Shen, and F. Bai, "Performance analysis of vehicular device-to-device underlay communication," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 6, pp. 5409–5421, 2017.

[8] X. Ge, H. Cheng, G. Mao, Y. Yang, and S. Tu, "Vehicular communications for 5g cooperative small-cell networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 7882–7894, 2016.

[9] E. Bernal-Mor, V. Pla, J. Martinez-Bauset, and L. Guijarro, "Performance analysis of two-tier wireless networks with dynamic traffic, backhaul constraints, and terminal mobility," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 241–250, 2016.

[10] Q. Zheng, K. Zheng, L. Sun, and V. C. Leung, "Dynamic performance analysis of uplink transmission in cluster-based heterogeneous vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 64, no. 12, pp. 5584–5595, 2015.

[11] H. He, H. Shan, A. Huang, and L. Sun, "Resource allocation for video streaming in heterogeneous cognitive vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 10, pp. 7917–7930, 2016.

[12] Z. Xiao, X. Shen, F. Zeng, V. Havyarimana, D. Wang, W. Chen, and K. Li, "Spectrum resource sharing in heterogeneous vehicular networks: A noncooperative game-theoretic approach with correlated equilibrium," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9449–9458, 2018.

[13] S. Oh, Y.-J. Byon, K. Jang, and H. Yeo, "Short-term travel-time prediction on highway: a review of the data-driven approach," *Transport Reviews*, vol. 35, no. 1, pp. 4–32, 2015.

[14] Y.-j. Kim, J.-s. Hong *et al.*, "Urban traffic flow prediction system using a multifactor pattern recognition model," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2744–2755, 2015.

[15] D. Miao, X. Qin, and W. Wang, "The periodic data traffic modeling based on multiplicative seasonal ARIMA model," in *Wireless Communications and Signal Processing (WCSP), 2014 Sixth International Conference on*. IEEE, 2014, pp. 1–5.

[16] V. Alarcon-Aquino and J. A. Barria, "Multiresolution FIR neural-network-based learning algorithm applied to network traffic prediction," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 36, no. 2, pp. 208–220, 2006.

[17] B. Sun, W. Cheng, P. Goswami, and G. Bai, "Short-term traffic forecasting using self-adjusting k-nearest neighbours," *IET Intelligent Transport Systems*, vol. 12, no. 1, pp. 41–48, 2017.

[18] G. A. Davis and N. L. Nihan, "Nonparametric regression and short-term freeway traffic forecasting," *Journal of Transportation Engineering*, vol. 117, no. 2, pp. 178–188, 1991.

[19] L. Zhang, Q. Liu, W. Yang, N. Wei, and D. Dong, "An improved k-nearest neighbor model for short-term traffic flow prediction," *Procedia-Social and Behavioral Sciences*, vol. 96, pp. 653–662, 2013.

[20] N. Bhatia and Vandana, "Survey of nearest neighbor techniques," *International Journal of Computer Science and Information Security*, vol. 8, no. 2, 2010.

[21] P. Duan, G. Mao, C. Zhang, and S. Wang, "Starima-based traffic prediction with time-varying lags," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2016, pp. 1610–1615.

[22] L. Yang and R. Jin, "Distance metric learning: A comprehensive survey," *Michigan State Universiy*, vol. 2, no. 2, p. 4, 2006.

[23] X. Wang, K. An, L. Tang, and X. Chen, "Short term prediction of freeway exiting volume based on svm and knn," *International Journal of Transportation Science and Technology*, vol. 4, no. 3, pp. 337 – 352, 2015, special Issue on Travel Demand Management. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S2046043016301319

[24] F. Guo, R. Krishnan, and J. Polak, "Short-term traffic prediction under normal and incident conditions using singular spectrum analysis and the k-nearest neighbour method," *IET and ITS Conference on Road Transport Information and Control*, 2012.

[25] P. He, L. Zhao, S. Zhou, and Z. Niu, "Water-filling: A geometric approach and its application to solve generalized radio resource allocation problems," *IEEE transactions on Wireless Communications*, vol. 12, no. 7, pp. 3637–3647, 2013.

[26] NGSIM, "Next generation simulation," *http://ngsim-community.org/*, 2010.

[27] J. S. Armstrong and F. Collopy, "Error measures for generalizing about forecasting methods: Empirical comparisons," *International journal of forecasting*, vol. 8, no. 1, pp. 69–80, 1992.

[28] J. Zhang, D. Smith, and Z. Chen, "Linear finite state markov chain predictor for channel prediction," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2012 IEEE 23rd International Symposium on*. IEEE, 2012, pp. 2085–2089.

[29] Y. Gao, G. He, and J. C. Hou, "On exploiting traffic predictability in active queue management," in *INFOCOM 2002. Twenty-First Annual*

*Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3. IEEE, 2002, pp. 1630–1639.

**Ping Chu** received the B.E. degree from Qufu Normal University, Shandong, China, in 2013. She is currently working toward the cotutelle Ph.D. degree with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing, China, and the School of Electrical and Data Engineering, University of Technology Sydney, Sydney, Australia. Her research interests include Vehicle-to-Vehicle communication, resoure optimization for C-V2X communication, and Device-to-Device communication.

**J. Andrew Zhang** received the B.Sc. degree fromXian JiaoTong University,Xian,China, in 1996, the M.Sc. degree from Nanjing University of Posts and Telecommunications, Nanjing, China, in 1999, and the Ph.D. degree from the Australian National University, Canberra, ACT, Australia, in 2004. He is currently an Associate Professor with the School of Computing and Communications, University of Technology Sydney, Ultimo NSW, Australia. He was a Researcher with Data61, CSIRO, Australia from 2010 to 2016, the Networked Systems, NICTA, Australia from 2004 to 2010, and ZTE Corp, Nanjing, China from 1999 to 2001. His research interests include the area of signal processing for wireless communications and sensing, and autonomous vehicular networks. He has published more than 120 papers in leading international journals and conference proceedings, and has won four Best Paper Awards for his work. He is a recipient of CSIRO Chairmans Medal and the Australian Engineering Innovation Award in 2012 for exceptional research achievements in multigigabit wireless communications.

**Xiaoxiang Wang** received the B.S degree in physics from Qufu Normal University, Qufu, China, in 1991, the M.S. degree in information engineering from East China Normal University, Shanghai, China, in 1994, and the Ph.D. degree in electronic engineering from Beijing Institute of Technology, Beijing, China, in 1998. In 1998, she joined the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. From August 2010 to February 2011, she was a Visiting Fellow with the Department of Electrical and Computer Engineering, North Carolina State University, Raleigh. Her research interests include communications theory and signal processing, with specific interests in cooperative communications, multiple-input-multiple-output systems, multimedia broadcast/multicast service systems, and resource allocation.

**Gengfa Fang** received the masters degree in telecommunications from Zhejiang University in 2002 and the Ph.D. degree in wireless communications from the Institute of Computing Technology, Chinese Academy of Sciences in 2007. From 2007 to 2009, he was a Researcher with the Canberra Research Laboratory, National ICT Australia on WCDMA Femtocell Project. In 2010, he was researching on Rural Broadband Access project at CSIRO as a Research Scientist for six months. From 2009 to 2015, he was with the Department of Engineering Macquarie University. In 2016, he moved to the University of Technology Sydney, where he is currently a Senior Lecturer.

**Dongyu Wang** received the B.S. degree in 2008 and M.S. degrees in 2011, respectively, both from Tianjin Polytechnic University, China, and the Ph.D. degree in 2014 from Beijing University of Posts & Telecommunications, China. From September 2014 to the present, he is a Post Ph.D with Department of Biomedical Engineering, Chinese PLA General Hospital, Beijing. His research interests include device to device communication, multimedia broadcast/multicast service systems, resource allocation, theory and signal processing, with specific interests in cooperative communications, and MIMO systems.