

Task Replication for Deadline-Constrained Vehicular Cloud Computing: Optimal Policy, Performance Analysis, and Implications on Road Traffic

Zhiyuan Jiang¹, Member, IEEE, Sheng Zhou¹, Member, IEEE, Xueying Guo², Member, IEEE, and Zhisheng Niu¹, Fellow, IEEE

Abstract—In vehicular cloud computing (VCC) systems, the computational resources of moving vehicles are exploited and managed by infrastructures, e.g., roadside units, to provide computational services. The offloading of computational tasks and collection of results rely on successful transmissions between vehicles and infrastructures during encounters. In this paper, we investigate how to provide timely computational services in VCC systems. In particular, we seek to minimize the deadline violation probability given a set of tasks to be executed in vehicular clouds. Due to the uncertainty of vehicle movements, the task replication methodology is leveraged which allows one task to be executed by several vehicles, and thus trading computational resources for delay reduction. The optimal task replication policy is of key interest. We first formulate the problem as a finite-horizon sampled-time Markov decision problem and obtain the optimal policy by value iterations. To conquer the complexity issue, we propose the balanced-task-assignment (BETA) policy which is proved optimal and has a clear structure: it always assigns the task with the minimum number of replicas. Moreover, a tight closed-form performance upper bound for the BETA policy is derived, which indicates that the deadline violation probability follows the Rayleigh distribution approximately. Applying the vehicle speed-density relationship in the traffic flow theory, we find that vehicle mobility benefits VCC systems more compared with road traffic systems, by showing that the optimum vehicle speed to minimize the deadline violation probability is larger than the critical vehicle speed in traffic theory which maximizes traffic flow efficiency.

Index Terms—Cloud computing, hard deadline, Markov decision process, task replication, vehicular networks.

I. INTRODUCTION

WITH the rapid development of autonomous driving technologies and a wide variety of on-board infotainment services, the future vehicles are equipped with advanced

and sufficient computational resources to facilitate sophisticated artificial-intelligence (AI) based algorithms. Wireless communication modules are also indispensable, which enable various types of communications including vehicle-to-vehicle, vehicle-to-infrastructure, and vehicle-to-everything over different protocols, e.g., dedicated short range communication [1], long-term-evolution-vehicle [2], and future 5G technologies [3]. Armed with these components, the vehicular network is evolving into a connected group of intelligent individuals which has great potentials. Consequently, the vehicle-as-a-resource concept has been proposed recently which exploits the enormous resources of vehicles and the vehicular network to facilitate new types of services, such as sensing, vehicle-bearing data transfer, cloud computing, and localization [4]–[14].

Among the resources that the vehicular network can offer, the computational resources are considered to be the most desirable one due to the growing computing demand of AI systems and the hardware limitation of user equipments [7], [8]. The vehicular cloud computing (VCC) [4], [9] is therefore proposed to allow the computing tasks to be collected at the task assignment station, e.g., a roadside unit (RSU), and offloaded to vehicles. A pictorial description of the VCC architecture is given in Fig. 1, in which the RSU controls the vehicular cloud through communications with vehicles inside its coverage area [15]. The computational resources of vehicles are abstracted based on the network function virtualization technology [16], such that it is transparent to users. The users can be pedestrians, sensors, and vehicles with insufficient computation capability. The task execution procedure is as follows. An RSU, denoted by task-RSU, collects tasks from users and offloads them to its associated vehicles based on a given task assignment policy during meetings between vehicles and the RSU. After the task computation, the output of the task is fed back to the task-RSU through the backhaul when the vehicle meets with any RSU in the network. Since the tasks can only be offloaded (or collected) to (or from) the vehicles inside the limited coverage area by successful transmissions during encounters [15], such a VCC system is referred to as an encounter-based VCC (eVCC) system [17]–[20]. One can immediately identify that the most intriguing challenge in the eVCC system is that the service process exhibits large uncertainty due to vehicle mobility. The meetings between the task-RSU and service-vehicles (SVs) and the meetings

Manuscript received August 21, 2017; revised October 9, 2017; accepted October 26, 2017. Date of publication November 8, 2017; date of current version February 9, 2018. This work was supported in part by the Nature Science Foundation of China under Grant 61701275, Grant 91638204, Grant 61571265, and Grant 61621091, in part by the China Post-Doctoral Science Foundation, and in part by the Intel Collaborative Research Institute for Mobile Networking and Computing. (Corresponding author: Sheng Zhou.)

Z. Jiang, S. Zhou, and Z. Niu are with the Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: zhiyuan@tsinghua.edu.cn; sheng.zhou@tsinghua.edu.cn; niuzhs@tsinghua.edu.cn).

X. Guo is with the Department of Computer Science, University of California at Davis, Davis, CA 95616 USA (e-mail: guoxueying@outlook.com).

Digital Object Identifier 10.1109/JIOT.2017.2771473

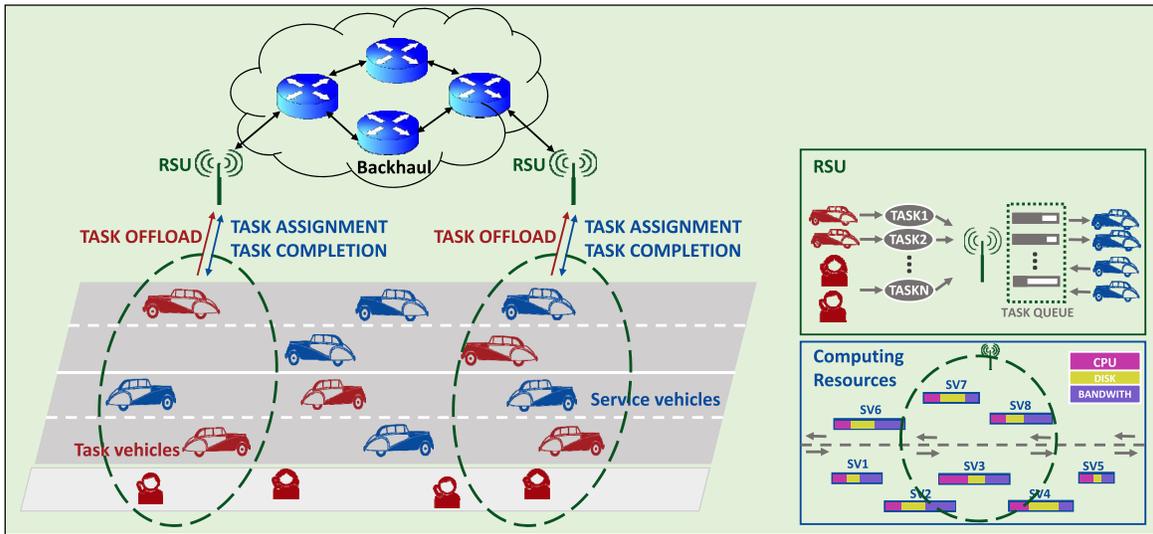


Fig. 1. VCC system overview.

afterwards when the outputs of the tasks are collected are both subject to the random behavior of moving vehicles. Consequently, the timeliness of the task computation is difficult to guarantee. Therefore, how to provide timely cloud computing services in eVCC systems becomes a challenging problem.

We propose to exploit the task replication technique to overcome this problem, which allows one task to be offloaded to multiple vehicles and the task is executed as long as one of the vehicles executes the task and feeds back the output. In this way, the mobility and high volume of vehicles are, to some extent, exploited to provide low-latency services. Toward this end, the key questions include how to assign (replicate) tasks to minimize the deadline violation probability of tasks given stochastic vehicle movements, and how to analyze and optimize the performance.

In the literature, there are some recent efforts dedicated to VCC systems. Eckhoff *et al.* [21] and Liu *et al.* [22] proposed to exploit parked cars for task offloading and cooperative sensing. On the contrary, since parked cars can be regarded as part of the static cloud servers and the corresponding task offloading policy has been well investigated (see [23], [24], and references therein), we focus on utilizing vehicles on the move. The eVCC task scheduling and resource allocation problem are formulated based on a semi-Markov-decision-process (SMDP) approach in [6] and [25]. However, the SMDP-based approach suffers from the curse of dimensionality and hence cannot be applied in high vehicle density scenarios. A large body of work has focused on the task replication policy and performance analysis in, e.g., data retrieval, multiserver data processing systems [26]–[31]. The task replication technique is usually leveraged to deal with the straggler problem [29], [31], where the service process has a heavy-tail distribution. Gardner *et al.* [26] focused on analyzing the performance of task replication policies based on the queuing theory. In [27] and [30], the optimal replication degree, i.e., the number of replicas, is investigated. However,

it is not applicable to the eVCC system where the servers (vehicles) are not always present. In [28], the optimal data retrieval replication policy is proved to be an shortest expected remaining processing time policy, which essentially schedules the task with the shortest time to go. Interestingly, although in different contexts, we will show that the optimal policy to minimize the deadline violation probability in VCC systems actually schedules the task with the longest time to go (the task with least number of replicas has the longest expected time to go).

In this paper, the optimal multitask replication policy to minimize the deadline violation probability and its performance in the eVCC system are investigated. The main contributions include the following.

- 1) We first formulate the task assignment problem as a sampled-time MDP, by which the optimal policy and deadline violation probability can be computed by finite-horizon value iteration. This result is used as the benchmark to validate optimality. To conquer the computational complexity issue, we design the balanced-task-assignment (BETA) policy by which the resultant task assignment is as balanced as possible, and prove its optimality.
- 2) A tight upper bound for the deadline violation probability following the optimal BETA policy is derived in closed-form, which implies that the optimum deadline violation probability follows the Rayleigh distribution approximately. We further apply the results from the vehicular traffic flow theory and demonstrate that there exists an interesting relationship between computation delay and vehicle density (or speed). A larger vehicle density leads to more vehicles executing the tasks, but meanwhile slows down the average vehicle speed due to limited human reaction time [32] which results in larger computation delay considering the task output has to be collected after offloaded. We show that the optimum vehicle density to minimize the deadline violation

probability is in general lower, and the corresponding vehicle speed is higher, than the critical vehicle density and speed in traffic theory to maximize traffic efficiency, respectively.

The remainder of this paper is organized as follows. In Section II, the system model is presented and the problem is formulated. In Section III, a sampled-time MDP-based solution is presented and the results are utilized as a benchmark to validate policy optimality. In Section IV, the optimal BETA policy is described and proved for optimality. In Section V, a closed-form performance bound is derived for the deadline violation probability given the optimal BETA policy and the optimum vehicle density is analyzed by applying the results from the vehicular traffic theory in Section VI. The simulation results are conveyed in Section VII. Finally, we conclude this paper in Section VIII.

Throughout this paper, we use boldface uppercase letters, boldface lowercase letters and lowercase letters to designate matrices, column vectors and scalars, respectively. Denote by $\mathbb{E}(\cdot)$ as the expectation operation. Denote by $\mathbb{1}(\cdot)$ as the indicator function. The exponential distribution with mean of λ is denoted as $\text{Exp}(\lambda)$, and $\exp(\cdot)$ denotes the exponential function. The cardinality of a set is denoted by $|\cdot|$. The probability of an event is denoted by $\Pr\{\cdot\}$. Denote by $k \bmod N$ as the remainder of the Euclidean division of k by N .

II. SYSTEM MODEL AND PROBLEM FORMULATIONS

We consider a Poisson vehicular network model with exponential vehicle-meeting time (PV-E). The following two model assumptions are made.

- 1) *Poisson Vehicle Spatial Distribution*: The number of vehicles in the system at time t , denoted by M_t , follows the Poisson distribution.¹ Denoting the length of roads in the system as S , the average vehicle density is λ , then

$$\Pr\{M_t = k\} = \frac{\lambda^k S^k e^{-\lambda S}}{k!}. \quad (1)$$

The vehicle number is assumed to be the same during the task offloading since the deadline is usually much smaller than the vehicle number variation time in vehicular networks.

- 2) *Homogeneous Memoryless Vehicle Meeting*: At any time and any RSU in the system, the time remaining before vehicle- m meets² with the RSU follows an independent exponential distribution with mean $\mu_m = \mu$, $\forall m$,³ that is

$$\Pr\{t < x\} = 1 - e^{-\mu x}. \quad (2)$$

¹The Poisson point process is widely used to model the stochastic spatial geometry of base stations and vehicles [33].

²In this paper, we define a meeting between a vehicle and an RSU as an event that a transmission (task offloading or output collection) is successful between them during the passage of the vehicle through the coverage area of the RSU. This definition excludes the encounters during which the transmissions are unsuccessful due to bad wireless channel conditions, insufficient passage time, etc.

³The exponential meeting time of vehicles is validated and widely used to model the vehicle behavior [17], [18], [34], [35].

Consider a task-RSU with N tasks that need to be offloaded and executed before deadline D_n , $n \in \{1, \dots, N\}$, where n is the task index. The task size is normalized such that each task can be offloaded to one vehicle at a time. In this paper, we consider the tasks are the with same type, i.e., the delay deadline is the same for all tasks and hence hereinafter $D_n = D$.⁴ The tasks are executed in the eVCC system as follows. For each task, it has to be first offloaded to a vehicle when the vehicle meets with the task-RSU. After the meeting, the task is assumed to be successfully offloaded and the vehicle uses on-board computational resources to compute the task. After that, when the vehicle meets with any RSU in the system, the computing result is fed back to the task-RSU through backhaul and the task is considered as successfully executed. Moreover, we denote

$$\omega_n = \begin{cases} 1, & \text{if the task-}n \text{ is executed before deadline} \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

In this paper, we focus on the scenario where the task computation time is negligible compared with vehicle meeting time as shown in [6].

We consider the task assignment problem that upon the meeting between a vehicle and the task-RSU, which task should be offloaded to the vehicle so as to minimize the deadline violation ratio, which is defined as

$$p_v = 1 - \sum_{n=1}^N \frac{\omega_n}{N}. \quad (4)$$

Moreover, in the eVCC system, the task service time fluctuates dramatically due to the unpredictable and diversified vehicle movements. Therefore, in order to counteract this so-called straggler effect [31], the tasks are replicated and executed independently. One task can be offloaded multiple times to different vehicles. Once one of the vehicles finishes the task (the output is fed back), the task is considered to be successfully executed. Since the SVs are out of coverage area when not meeting with RSUs in the eVCC system, they cannot be assigned new tasks even if the task at hand has been finished by other vehicles, until they meet with an RSU again. This is referred to as the nonpurging model in [36]. In a purging model for, e.g., multiprocessor computing, the processor begins a new task as soon as the current task is finished (maybe by other processors).

A. Task Execution Procedure in eVCC Systems

The detailed task execution procedure in eVCC systems is described in the following.

- 1) *Task Collection*: Define the task execution interval as T . Within the interval, the task collection and assignment are executed simultaneously. The task requests from users are collected at each RSU. Denote the number of collected task requests at the t th interval as N_t .

⁴The following MDP formulation can be immediately generalized to the multitype task scenario, but the analysis for the optimal policy would be considerably more complicated and is left for future work.

2) *Task Assignment*: In the $(t + 1)$ th interval, the N_t tasks collected in the previous interval at each RSU are assigned based on the proposed algorithm described in Section IV. Specifically, upon the arrival of a vehicle, the n th task is assigned based on the algorithm. If the assignment is successful subject to wireless transmission failure, the number of replications for the n th task increases by one.⁵

3) *Task Output Feedback*: At the end of each interval, each RSU collects the task output feedback from assigned vehicles through backhaul links. The assigned vehicles may finish computing the tasks and feed back the task outputs when meeting with another RSU. In this case the output is transmitted to the original RSU through backhaul links. Afterwards, each RSU transmits the outputs to corresponding users and the task execution is finished. The above procedure then repeats itself.

III. SAMPLED-TIME MARKOV CHAIN-BASED APPROACH

First, the continuous time model is converted to a discrete-time approximation. Consider some given small increment of time δ , a sampled time approximation to the continuous time Poisson vehicle meeting for vehicle- m is a discrete Bernoulli process whose meeting probability is $\mu_m \delta$ [37]. It is straightforward to derive that the meeting rate for a total of K vehicles is $\sum_{m=1}^K \mu_m$, and thus the meeting probability is $K\mu\delta$ due to the second assumption in Section II.

On account of the task replication, the system state space is characterized by

$$\mathcal{S} = \{(r_1, \dots, r_N, d) | 1 \leq d \leq D, r_i \in \Gamma, \forall i = 1, \dots, N\} \quad (5)$$

where d is the number of elapsed time slots, i.e., the computation delay. Note that the problem is with finite time horizon of $1 \leq d \leq D$, $d \in \mathbb{Z}$, for deadline violation consideration. The number of current replicas of task- n which are being processed in the eVCC system is denoted by r_n , which is sufficient to describe the system state due to the memoryless vehicle meeting assumption. The time slot each replica is made is irrelevant based on the assumption. The state space is finite since the number of replicas cannot exceed the number of vehicles in the system. For ease of exposition, with slight abuse of notations we denote $r_n = F$ as that task- n is finished. The set

$$\Gamma = \{1, \dots, M, F\} \quad (6)$$

denotes the set of values of r_n , $\forall n$. For each state with $d = D$, it denotes a terminal state. Since $1 \leq d \leq D$ and the terminal states are mutually exclusive, there exists a unique terminal state where the system terminates in. The reward is defined as the number of finished tasks at time D . Based on this definition and (4), maximizing the reward is equivalent to minimizing the deadline violation ratio. Also note that the initial state is always

$$s_0 = \left(\underbrace{0, \dots, 0}_N, 0 \right). \quad (7)$$

⁵Note that each RSU carries out the task assignment independently, which means that the recourse constraints of vehicle are ignored in this paper.

It is without loss of generality to assume there is a vehicle meeting at the time slot zero, since it is equivalent to consider the delay deadline as $D - D_0$ if the first time when a vehicle meets with the task-RSU is D_0 .

The system state is reviewed and the decisions are made at the meeting time⁶ between vehicles and the task-RSU, expect the time slot zero. Note that based on the discrete time conversion, a sufficiently small time increment δ is chosen such that at any given time slot the probability of multiple vehicle meetings is negligible. It is easy to validate the fact that based on the system model and system state description in (5), the following Markovian properties are satisfied: if an action a is chosen in state x at a decision time slot, then the time till the next decision time slot, i.e., τ , is determined by the memoryless vehicle meetings with the RSU. Moreover, the system state at the next decision time slot $s_{t+\tau}$ depends on the task execution during time interval $(t, t + \tau)$ which is also memoryless. Hence, the time till next decision time slot and the next system state depend only on the present state x and the chosen action a , that is,

$$\begin{aligned} \Pr\{s_{t+\tau} = y | (s_t = x, a_t = a), \dots, (s_0, a_0)\} \\ &= \Pr\{s_{t+\tau} = y | (s_t = x, a_t = a)\} \\ &\quad \Pr\{\tau = z | (s_t = x, a_t = a), \dots, (s_0, a_0)\} \\ &= \Pr\{\tau = z | (s_t = x, a_t = a)\}. \end{aligned} \quad (8)$$

Also the reward incurred only depends on the terminal state, and thus independent of the past.

A. Transition Probability

At the decision time slot, given an action that the n th task (unfinished) is offloaded to the meeting vehicle, the state transition probability is

$$\begin{aligned} \Pr\{((r_i | i \in \mathcal{M}, i \neq n), r_n, (r_j | j \notin \mathcal{M}, j \neq n), d) \rightarrow \\ \times ((F | i \in \mathcal{M}, i \neq n), r_n + 1, (r_j | j \notin \mathcal{M}, j \neq n)), d + \tau\} \\ &= \begin{cases} (1-s)^{\tau-1} s \prod_{n \notin \mathcal{M}} p_n^\tau \prod_{m \in \mathcal{M}} (1-p_m^\tau), & d + \tau < D \\ (1-s)^{\tau-1} \prod_{n \notin \mathcal{M}} p_n^\tau \prod_{m \in \mathcal{M}} (1-p_m^\tau), & d + \tau = D \end{cases} \end{aligned} \quad (9)$$

where p_i denotes the probability that the i th task is not finished until the next decision time slot. The set \mathcal{M} contains the tasks that have not been finished before time d and are finished during time slot d to the next decision time slot. It follows that:

$$p_i = \max\{1 - q_i, \mathbb{1}(r_i = F)\}, \quad i \in \{1, \dots, N\} \quad (10)$$

and

$$q_i = (r_i + \mathbb{1}(i = n))B\mu\delta, \quad i \in \{1, \dots, N\} \quad (11)$$

where q_i denotes the probability that at least one of the vehicles which are executing task- i meets with one of the B RSUs. Note that there are $(r_i + \mathbb{1}(i = n))B$ different vehicle-RSU combinations that can lead to the execution of the i th task.

⁶Although the meeting is defined as an event that a successful transmission occurs during the passage of the vehicle through the coverage area of an RSU, the meeting time is defined as a time point instead of a period of time at which the transmission is completed.

Based on the homogeneous memoryless vehicle meeting assumption, the event that task- i is finished happens with a rate of $(r_i + \mathbb{1}(i = n))B\mu$, due to the fact that the distribution of the minimum of several exponentially distributed random variables is also exponential with a rate that is the summation of the rates of the variables. Denote the probability that a vehicle meets with the task-RSU as s . Based on the same arguments

$$s = M\mu\delta \quad (12)$$

where M denotes the number of vehicles in the system during this task computation. Based on the first assumption in Section II, M is Poisson distributed but fixed during one task computation. Therefore, the time index is omitted for simplicity and its stochastic nature will be considered in the performance analysis in Section V. Note that δ is sufficiently small such that s is also small due to the discrete time approximation formulation. Assuming the current decision time slot is d , the probability that the next vehicle meets with the task-RSU at time $d + \tau$ is

$$\Pr\{\text{next meeting at time } d + \tau\} = (1 - s)^{\tau-1}s \quad (13)$$

when $d + \tau < D$. Since a finite horizon of $1 \leq d \leq D$ is sufficient, the transition probability to a terminal state, which is denoted by the time index of D , is

$$\Pr\{\text{no more meeting before deadline}\} = (1 - s)^{D-d-1}. \quad (14)$$

We assume that the vehicle meetings are independent. Therefore, the task completion and the new vehicle meetings are independent.

B. Finite-Horizon MDP-Based Solution

The system reward is expressed by a terminal reward vector \mathbf{h} , where the entry of \mathbf{h} is a function of a terminal state $s_t \in \mathcal{S}$, that is,

$$h(s_t) = \sum_{i=1}^N \mathbb{1}(s_t(i) = F) \quad (15)$$

which denotes the number of finished tasks before deadline. Consequently, the solution to the formulated MDP minimizes the deadline violation ratio in (4).

Since this is a finite horizon MDP problem with finite state space, the backwards recursive iteration algorithm can be applied to find the optimal dynamic policy. Note that the optimal policy of a finite horizon problem is, in general, a dynamic policy, i.e., not a stationary policy as often observed in the infinite-horizon MDP, meaning that given the current system state, the policy also depends on the current time. Applying the backwards recursive iteration, we start from the terminal stage, denoted by stage K

$$J_k(s_k) = \max_{a_k \in \{1, \dots, N\}} \mathbb{E}\{J_{k+1}(s_{k+1}) + g_k(s_k)\} \quad (16)$$

where $J_K(s_K) = g_K(s_K)$ is the terminal reward. Based on [38], the optimal reward

$$J^*(s_0) = J_0(s_0) \quad (17)$$

and the optimal dynamic policy is given by $a^* = \{a_0^*, \dots, a_{K-1}^*\}$, where a_k^* 's are derived from (16).

IV. OPTIMAL BALANCED-TASK-ASSIGNMENT POLICY

The finite-horizon MDP-based solution described above is guaranteed to find the optimum. However, as in many application scenarios, it suffers from the curse of dimensionality. One can easily find that the state space grows exponentially with the number of tasks. In what follows, we prove that the BETA policy, which is also stationary, is the optimal policy. First, the definition of the BETA policy is given.

Definition 1: The BETA policy is defined as at every decision time slot when a vehicle meets with the task-RSU, given the current task status (r_1, \dots, r_N) , offload the n th task to the vehicle, where

$$n = \arg \min_{i \in \{1, \dots, N\}} \{r_i | r_i \neq F\} \quad (18)$$

and ties are broken arbitrarily.

Remark 1: It is not difficult to figure out that the task status following the BETA policy is as balanced as possible, in the sense that given the total number of computing vehicles as $\gamma = \sum_{r_i \neq F} r_i$, and the number of unfinished tasks as $u = |\{r_i | r_i \neq F\}|$

$$r_i^* \in \{\lfloor \gamma/u \rfloor, \lceil \gamma/u \rceil\}, i \in \{1, \dots, N\} \quad (19)$$

where r_i^* denotes the resultant task assignment by the BETA policy, and $\lfloor x \rfloor$ and $\lceil x \rceil$ denote the nearest integers below and above x , respectively.

Theorem 1: Given the homogeneous memoryless vehicle meeting assumption (2), the BETA policy is optimal in minimizing the deadline violation ratio in (4).

Proof: The proof is based on backwards induction on time. See Appendix A for details. ■

Remark 2: A high-level intuition to explain the optimality of BETA policy is as follows. A stochastic coupling argument (the details are omitted for brevity) can be used to show that a necessary and sufficient condition for optimality is to avoid unnecessary service, which is defined as the event that a vehicle executes a task that has already been finished by vehicles before it. Based on this observation, the tasks should be assigned as balanced as possible, assuming the service occurs uniformly among tasks, to reduce the potential number of unnecessary services. The fundamental reason for the optimality of the BETA policy is that the tasks are treated equally, both from the deadline and the violation ratio perspectives.

V. PERFORMANCE ANALYSIS FOR BETA POLICY

The BETA policy is proved to be optimal, as long as the homogeneous memoryless vehicle meeting assumption described in Section II is upheld. The vehicle meeting process at the task-RSU is irrelevant, which can be observed by the proof of Theorem 1. Despite that the optimality of the BETA policy is shown, it is difficult to obtain a closed-form formula for the performance, i.e., concretely the deadline violation probability (average violation ratio) of the system. The difficulty stems from deriving the static distribution of a high-dimensional Markov chain described by the formulated MDP given the BETA policy. To address this issue and gain more insights into the performance, we derive a closed-form

upper bound for the deadline violation probability of the eVCC system, which is shown to be quite tight, by noticing the fact that the BETA policy evenly replicates the tasks and therefore the performance of the BETA policy should be similar with an equivalent system where there is only one task with one N th of the vehicles. The following theorem formalizes this finding.

Theorem 2: The deadline violation probability of the system is upper bounded by

$$\Pr\{t_D > D\} \leq \exp\left\{-\frac{\lambda S}{N}\left(1 + \frac{\lambda_1}{\lambda_2 - \lambda_1}e^{-\lambda_2 D} - \frac{\lambda_2}{\lambda_2 - \lambda_1}e^{-\lambda_1 D}\right)\right\} \quad (20)$$

where $\lambda_2 = \mu B$ and $\lambda_1 = \mu$.

Proof: See Appendix B. ■

Corollary 1 (Short Deadline Regime): When $D \rightarrow 0$, the task computation time lower bound derived in Theorem 2, i.e., t_D , follows the Rayleigh distribution, that is:

$$\Pr\{t_D < D\} = 1 - e^{-\frac{\lambda S}{2N}B\mu^2 D^2} \quad (21)$$

and the deadline violation probability upper bound is $1 - \Pr\{t_D < D\}$ given deadline D . The mean computation delay is

$$\mathbb{E}[t_D] = \sqrt{\frac{N\pi}{2\lambda BS\mu^2}}. \quad (22)$$

Proof: See Appendix C. ■

Remark 3: Since Corollary 1 is the Taylor expansion approximation of Theorem 2, the short deadline regime is there when

$$\lambda_1 D \rightarrow 0 \text{ and } \lambda_2 D \rightarrow 0 \quad (23)$$

which denote the average numbers of meetings between a vehicle and one and B RSUs before deadline, respectively. The numbers are usually small in typical scenarios, making the approximation accurate in most cases. Therefore, Corollary 1 is used in the following section for ease of exposition.

VI. IMPLICATIONS ON ROAD TRAFFIC: OPTIMUM VEHICLE DENSITY FOR EVCC

In this section, some insights are given by combining the performance analysis of the optimal BETA policy and the traffic engineering theory [32]. It is observed that the average vehicle speed decreases with the vehicle density in the real world due to limited human reaction time [32]. Therefore, to minimize the deadline violation probability, the impacts of vehicle density and vehicle speed, which reflect on the number of SVs and the output collection rate, respectively, need to be jointly considered to find the optimum tradeoff.

Based on the traffic theory, there are three macroscopic parameters that describe the traffic flow state on the road: the average (space mean) speed of traffic stream V , the rate of flow F , and the vehicle density L . With the unified units, the fundamental equation of the speed-flow-density relationships is

$$F = VL. \quad (24)$$

The general relationship of the basic parameters can be well illustrated by the approximate linear relationship between the speed of vehicles and the vehicle density [17], that is,

$$V(L) = V_{\max}\left(1 - \frac{L}{L_{\max}}\right) \quad (25)$$

where V_{\max} is the speed limit, and L_{\max} is the traffic jam vehicle density. Later in the section, the linear relationship will be generalized. In the traffic theory, there is a concept named *critical vehicle density*, which is defined as the optimum vehicle density given a stationary traffic flow that maximizes the traffic efficiency, i.e., the traffic flow rate. It is straightforward to see that the critical vehicle density to maximize the vehicle flow rate given (25) is

$$L^* = \frac{L_{\max}}{2}. \quad (26)$$

Essentially, it states the simple fact that when there is too much traffic wherein the vehicles cannot move or when there is too little traffic wherein the vehicles are too few, the traffic flow throughput is not satisfactory.

The question is whether the critical vehicle density is also optimal for the eVCC system. Toward this end, we have the following theorem.

Theorem 3: When the speed-density relationship, i.e., $V(L)$, satisfies the following conditions, the optimum vehicle density for eVCC system L^\dagger is smaller than the critical vehicle density L^* .

- 1) $V(L)$ is monotonically nonincreasing, and $V(L) \geq 0$ for $0 \leq L \leq L_{\max}$.
- 2) $V(L)$ is a concave function of L , or it satisfies

$$V(L) = G - \sum_{k=1}^K c_k L^{\alpha_k} \quad (27)$$

where $\alpha_k \in (0, +\infty) \cup (-1, -(1/2))$, $c_k \geq 0$, $\forall k$ and $G \geq 0$.

Proof: See Appendix D. ■

Remark 4: Note that the first condition in Theorem 3 is almost always met in practice, due to the fact that people would drive more slowly with more vehicles on the road [40]. The second condition allows several well-recognized speed-density models to be applicable in [41, Ch. 12] and [42].

Remark 5: Theorem 3 establishes that under most circumstances, the optimum vehicle density for the eVCC system is smaller than the critical vehicle density. Therefore, the corresponding optimum speed comparison is the opposite, which indicates that vehicle mobility actually benefits the eVCC system more than the traditional traffic system.

Corollary 2: In the short deadline regime, if the linear speed-density relationship is satisfied, the optimum vehicle density to minimize the average deadline violation probability is

$$L^\dagger = \frac{L_{\max}}{3}. \quad (28)$$

Proof: See Appendix E. ■

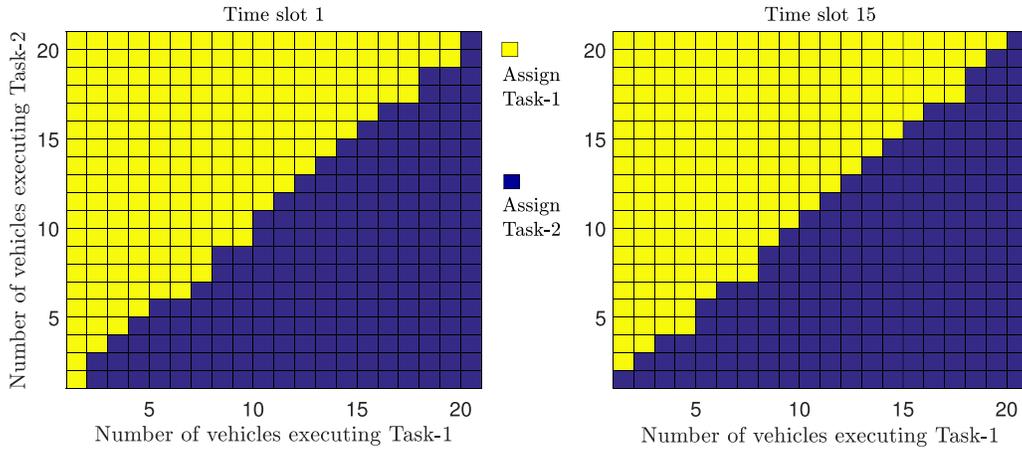


Fig. 2. Optimal task assignment policy given by solving MDP-1. The total number of vehicles is $M = 20$, the number of RSUs is $B = 5$, the delay deadline is $D = 20$, and only two tasks are considered.

Remark 6: This result corresponds to the critical vehicle density with linear speed-density model in (26). It demonstrates, in closed-form, the comparisons between the optimal vehicle densities in VCC systems and traffic systems.

A. Asymptotic Analysis

1) *Large City Regime:* In the large city regime, the length of the road is approaching infinity, i.e., $S \rightarrow \infty$. Since the number of available vehicles is increased whereas the distance of traveling is also increased, it is interesting to investigate the performance in the regime. Based on Theorem 2, it follows that when $S \rightarrow \infty$, $\mu \rightarrow 0$, then the deadline violation probability approaches the following based on the linear speed-density relationship (25):

$$\begin{aligned} \Pr(t_D > D) &\xrightarrow{S \rightarrow \infty} \exp\left(-\frac{LS}{2N}B\mu^2D^2\right) \\ &\xrightarrow{S \rightarrow \infty} \exp\left(-\frac{LV_{\max}^2B}{2SN}\left(1 - \frac{L}{L_{\max}}\right)^2D^2\right). \end{aligned} \quad (29)$$

It can be observed that as long as the RSU density, i.e., B/S , and the vehicle density remains unchanged, the performance is irrelevant with the road length.

2) *High RSU Density Regime:* When the RSU density goes to infinity, i.e., $B/S \rightarrow \infty$, combining with Theorem 2, we obtain

$$\Pr(t_D > D) \xrightarrow{\frac{B}{S} \rightarrow \infty} \exp\left(-\frac{LS}{N}(1 - e^{-\mu D})\right). \quad (30)$$

If we continue to assume the short deadline regime as in Corollary 1, then (30) becomes

$$\Pr(t_D > D) \xrightarrow{\frac{B}{S} \rightarrow \infty, D \rightarrow 0} \exp\left(-\frac{LV_{\max}}{N}\left(1 - \frac{L}{L_{\max}}\right)D\right). \quad (31)$$

It turns out that the optimum vehicle density in this regime is identical with the critical vehicle density which maximizes the traffic flow rate. This is explained by the fact that when

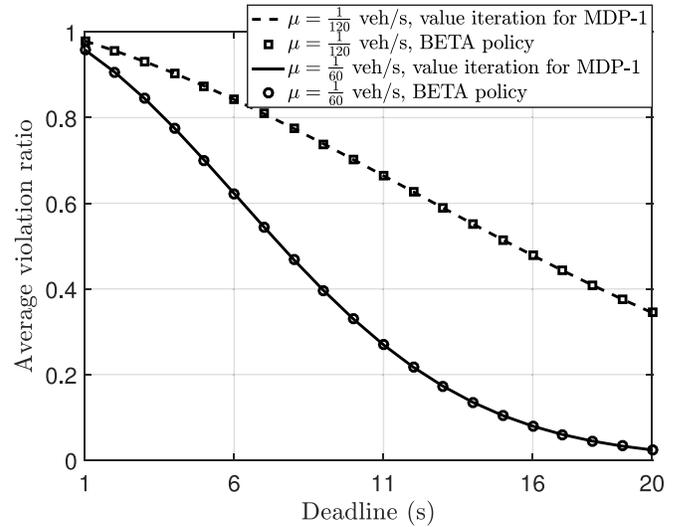


Fig. 3. Optimal average violation ratio given by value iteration to solve MDP-1, in comparison with the achieved average violation ratio by the BETA policy. The total number of vehicles is $M = 20$, the number of RSUs is $B = 5$, and only two tasks are considered.

RSUs are everywhere, the tasks would be immediately finished after they are offloaded, and hence the task computation time is determined by the vehicle meeting rate at the task-RSU, which is equivalent with the traffic flow rate assuming homogeneous vehicle meetings.

VII. NUMERICAL RESULTS

In this section, numerical results based on computer simulation are given to demonstrate the optimality of the BETA policy, to validate the tightness of the performance upper bound given in Theorem 2 and Corollary 1, and to show the impacts of several key system parameters. The vehicle meeting rate μ can be derived from (25), where the speed limit is $V_{\max} = 100$ km/h and the traffic jam vehicle density is $L_{\max} = 140$ veh/km assuming two lanes [32].

Figs. 2 and 3 are shown to demonstrate the optimality of the BETA policy. The optimal policy is obtained by value iteration

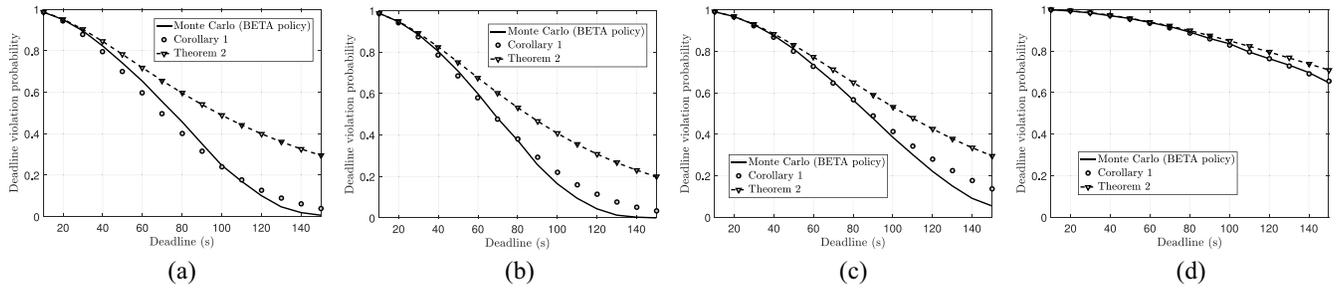


Fig. 4. Optimal average violation ratio versus the deadline given by Monte Carlo simulations based on the BETA policy, in comparison with the violation probability upper bound given in Theorem 2 and Corollary 1. The number of RSUs is $B = 10$, the road length is 10 km, and the number of tasks is 50. (a) $L = 30$ veh/km. (b) $L = 60$ veh/km. (c) $L = 90$ veh/km. (d) $L = 120$ veh/km.

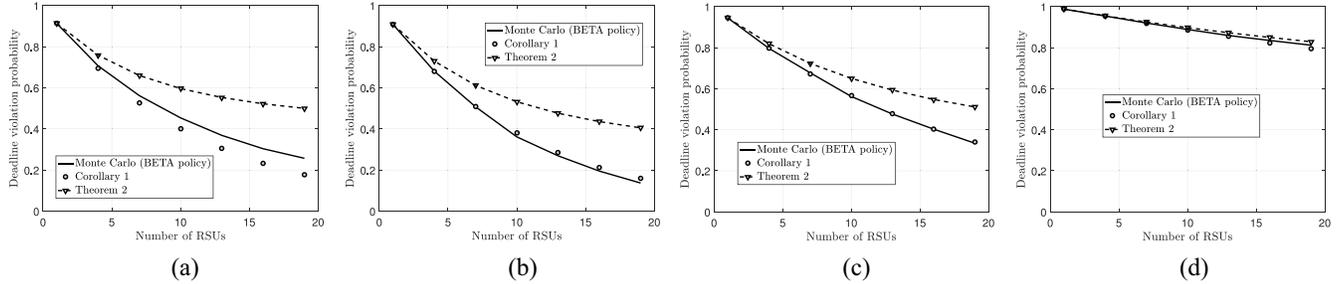


Fig. 5. Optimal average violation ratio versus the number of RSUs on the road given by Monte Carlo simulations based on the BETA policy, in comparison with the violation probability upper bound given in Theorem 2 and Corollary 1. The road length is 10 km, the number of tasks is 50, and the deadline is 80 s. (a) $L = 30$ veh/km. (b) $L = 60$ veh/km. (c) $L = 90$ veh/km. (d) $L = 120$ veh/km.

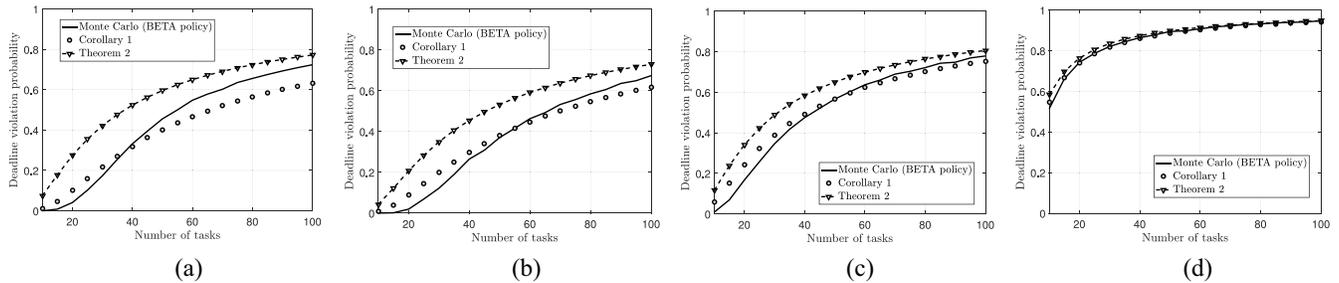


Fig. 6. Optimal average violation ratio versus the number of tasks given by Monte Carlo simulations based on the BETA policy, in comparison with the violation probability upper bound given in Theorem 2 and Corollary 1. The road length is 10 km, the number of RSUs is 10, and the deadline is 80 s. (a) $L = 30$ veh/km. (b) $L = 60$ veh/km. (c) $L = 90$ veh/km. (d) $L = 120$ veh/km.

to solve MDP-1. It is observed in Fig. 2 that the optimal policy has the same structure with the BETA policy, which always assigns the task with fewer replicas. It is worth mentioning that in general the optimal policy for a finite-horizon MDP is not stationary, in the sense that it may change as the time goes on. However, it is shown that the BETA policy is optimal at any time slot (time slots 1 and 15 are shown). Also note that the ties are broken arbitrarily, and therefore the policy on the diagonal can go either way. Fig. 3 shows that optimum value of the initial state equals the average violation ratio obtained by simulating the BETA policy, with two distinct departure rates.

Figs. 4–8 are shown to reflect the impact of several key parameters, and to compare the simulation results with the closed-form expressions in Theorem 2 and Corollary 1. The simulation results are obtained by running the BETA policy for 1000 iterations and calculating the average, where in each iteration the number of vehicles is generated based on the

Poisson distribution as assumed in Section II. In general, it is shown that the deadline violation probability performance can be well characterized by the closed-form upper bounds given in Theorem 2 and Corollary 1. In particular, the result in Corollary 1 is very close to the simulation results. The optimum vehicle density given in Proposition 2, which minimizes the deadline violation probability is validated in Fig. 7. It is shown that the promptness of the eVCC system with the optimum vehicle density by Proposition 2 is significantly better than the traditional critical vehicle density in the traffic engineering theory, demonstrating that vehicle mobility benefits the eVCC system more compared with the traffic system. It also indicates that we need to rethink the desired traffic conditions in the future when vehicular network-based applications such as eVCC, are widely implemented. In Fig. 8, the performance under a widely used speed-density model, which is validated based on real traffic measurements collected by using 3 A.M. peak sensor data from a section of the M27

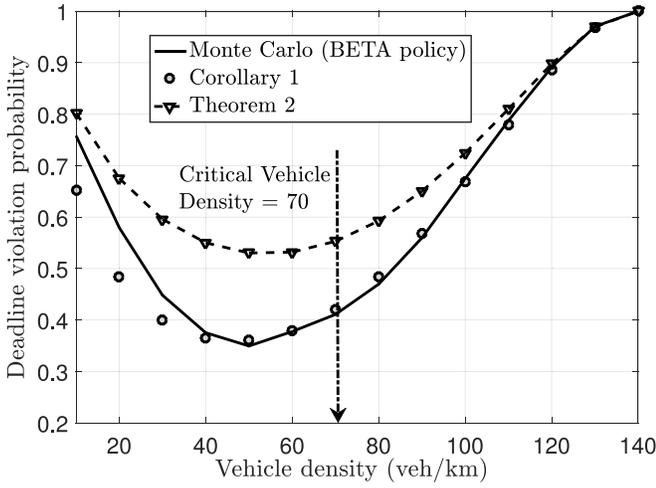


Fig. 7. Optimal average violation ratio versus the vehicle density on the road given by Monte Carlo simulations based on the BETA policy, in comparison with the violation probability upper bound given in Theorem 2 and Corollary 1. The road length is 10 km, the number of RSUs is 10, the number of tasks is 50, and the deadline is 80 s.

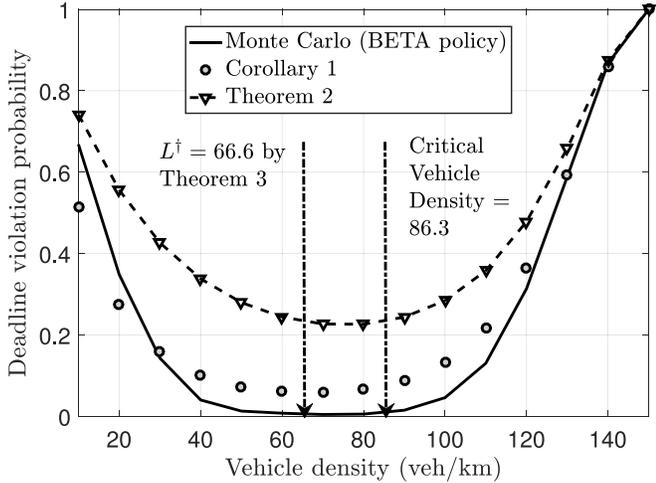


Fig. 8. Based on measurement validated model in [39] and [42], the optimal average violation ratio versus the vehicle density on the road given by Monte Carlo simulations based on the BETA policy, in comparison with the violation probability upper bound given in Theorem 2 and Corollary 1. The road length is 10 km, the number of RSUs is 10, the number of tasks is 50, and the deadline is 80 s.

freeway in Southampton, U.K. [39]. Based on the model, the speed-density relationship is characterized by

$$V = 116.4 \left[1 - \left(\frac{L}{149.797} \right)^{1.964} \right] \quad (32)$$

which satisfies the second condition in Theorem 3.

A. Tradeoff Between eVCC Efficiency and Traffic Efficiency

Define the eVCC efficiency as the average number of executed tasks in unit time, that is,

$$\eta_{CE} = \frac{N \Pr\{t < D\}}{T} \quad (33)$$

where N/T denotes the number of generated tasks in unit time. The traffic efficiency is the traffic flow rate, i.e., the average

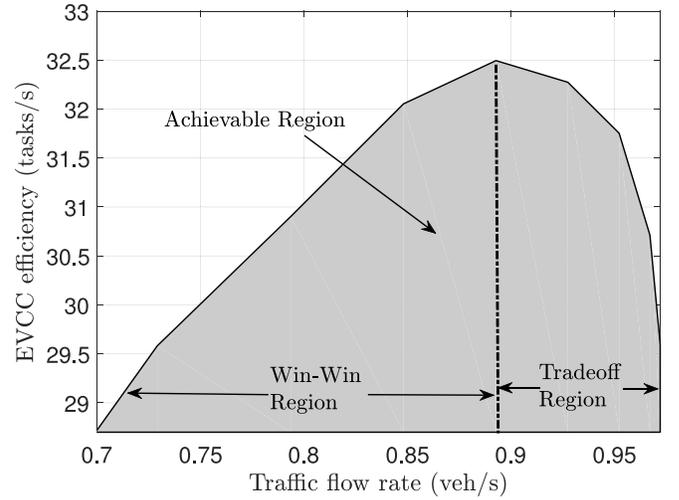


Fig. 9. Tradeoff between eVCC efficiency and traffic efficiency. The road length is 10 km, the number of RSUs is 10, the number of tasks is 50, and the deadline is 80 s.

number of vehicles passing through in unit time, which is

$$\eta_{TE} = VL. \quad (34)$$

To study the tradeoff between η_{CE} and η_{TE} , Fig. 9 is depicted. In the win-win region, the eVCC and traffic efficiency both go up with the increased vehicle density. It is observed that after some point, which actually corresponds to L^\dagger in Corollary 2, the eVCC efficiency drops with the traffic efficiency, which is denoted by the tradeoff region in Fig. 9. The finding indicates that the future intelligent transportation system should strike a good balance between traffic efficiency and eVCC efficiency.

VIII. CONCLUSION

In this paper, in order to minimize the deadline violation probability in the eVCC system, we design optimal multitask replication policy by finite horizon MDP. Further, it is proved that the optimal policy given homogeneous memoryless vehicle meeting and identical deadline of tasks is the BETA policy. The BETA policy always assigns the task with the least number of replicas, hence the longest expected time to go, which happens to be the opposite of several existing deadline-driven algorithms [28], [43], [44] for applications with different contexts.

A tight, closed-form upper bound for the deadline violation probability by the BETA policy is derived which shows that the execution time is Rayleigh distributed with variance scaling with λV^2 . It is intriguing to find out that, combining with the traffic engineering theory, the optimum vehicle density to minimize the deadline violation probability is lower than the traditional critical vehicle density which maximizes the traffic flow rate. Given that the average vehicle speed decreases as the vehicle density grows, the result implies that the increased vehicle mobility benefits the eVCC system more compared to the traditional traffic system attributing to higher vehicle meeting rate, until the point that there are too few vehicles on the road due to decreased vehicle density. Such a tradeoff is thoroughly described by the presented analysis. A general and

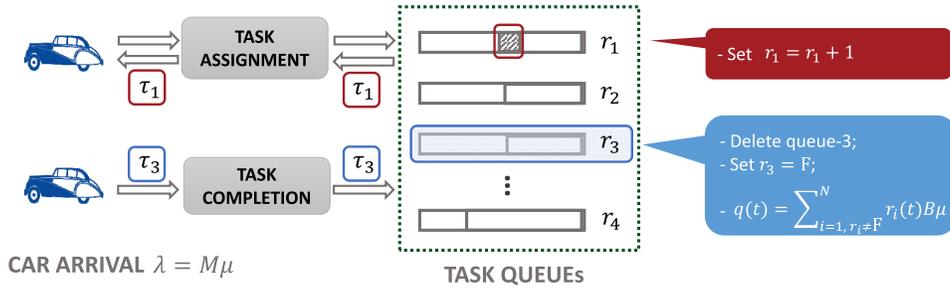


Fig. 10. Pictorial description of MDP-2. The discrete meeting probability is denoted by p , and the departure probability is denoted by q which are both i.i.d. over time and follows Bernoulli distribution.

sufficient condition under which the optimum vehicle speed for the eVCC system is larger than that for the traffic system is given. Combining with the linear speed-density relationship and the asymptotic analysis, it is shown that the optimum vehicle density for eVCC is one-third of the traffic jam vehicle density. Simulation results show that the derived closed-form expressions can fully characterize the impacts of several key parameters.

The future directions include extensions to nonexponential service time distributions and nonidentical task deadlines. With task service time no longer being memoryless, the task with a longer waiting time currently should expect the waiting time to be shorter and even longer, with light-tail and heavy-tail service time distributions, respectively. The task replication policy is hence affected. When the tasks have different deadlines, it is natural that they should be treated differently, which deviates from the BETA policy. In either case, it is conjectured that the BETA policy can be extended by leveraging a utility function [45] to characterize those factors.

APPENDIX A PROOF OF THEOREM 1

The sketch of the proof is as follows. First, we will derive an equivalent MDP which has the same optimal policy as the above-mentioned MDP. Then we will show that the BETA policy is optimal based on the equivalent MDP. Toward developing the equivalent MDP, we first prove the following lemma.

Lemma 1: Define the total service rate under policy π as

$$R(\pi) = \sum_{t=0}^{D-1} \sum_{i=1, r_i(t) \neq F}^N r_{i,\pi}(t) \mu B. \quad (35)$$

If $R(\pi_1) \geq R(\pi_2)$, the distribution of $p_v(\pi_1)$ dominates that of $p_v(\pi_2)$, that is

$$\Pr\{p_v(\pi_1) < x\} \geq \Pr\{p_v(\pi_2) < x\}. \quad (36)$$

Consequently, the average violation ratio under π_1 is no more than that under π_2 .

Proof: Lemma 1 simply implies that if a policy has a stronger serving ability (higher service rate), it has better violation ratio. To make it rigorous, a coupling argument is adopted. Consider a given total service rate of $R(\pi)$. The service process is with a time varying service rate of $r_i(t)B$, $i \in \{1, \dots, N\}$.

$r_i(t) \neq F$, and the serving time is from 0 to D . Given that the service process is memoryless and independent with the policy once the service rate is given. The considered service process is equivalent with a system with $R(\pi)$ servers, and service rates are all 1, serving for one time slot. The equivalence stem from the fact that the distribution of the number of service tasks are the same.

Due to the independent identically distributed (i.i.d.) assumption, we can consider that the servers under π_2 are a subset of the servers under π_1 , given that $R(\pi_1) \geq R(\pi_2)$. Based on a coupling argument, assuming the occurrence of an arbitrary, and identical subset of servers with task completion in the time slot, the number of task completion under π_1 is higher than that under π_2 , since the servers under π_2 is a subset of π_1 . This coupling argument results in the distribution dominance. The average results follow immediately. With this, we complete the proof. ■

Next, we develop an equivalent MDP (denoted by MDP-2), based on the above-mentioned MDP (denoted by MDP-1).

Lemma 2: The optimal policy of MDP-1 is the same with the following MDP-2.

The system is still time-slotted as previously explained in Section II. After the task assignment, the vehicle is placed in one of the N queues (see Fig. 10 for an illustration), where each queue denotes a task with r_i task replicas. The N queues are served by a common server with a service rate of $\sum_{i=1, r_i(t) \neq F}^N r_i(t)B\mu$. Whenever a task completion occurs at time t_m , the corresponding queue is deleted, and we set $r_i(t) = F, \forall t > t_m$. The total reward is defined as

$$\mathbb{E} \left[\sum_{t=0}^{D-1} \sum_{i=1, r_i(t) \neq F}^N r_i(t) \right]. \quad (37)$$

Proof: By construction of MDP-2 and Lemma 2, it is straightforward to see that the optimal policies of MDP-1 and MDP-2 are the same. ■

Based on Lemma 2, in order to show that the BETA policy is optimal for MDP-1, it is sufficient to show that it is optimal for MDP-2. Toward this end, we will first show that the BETA policy is the myopic policy for MDP-2, meaning that the BETA policy maximizes the immediate reward for MDP-2. Then, we will show that the myopic policy is indeed the optimal policy, and hence the BETA policy is optimal for MDP-1. Similar techniques have been used in [46], but in a different scenario.

Define $g_t(\mathbf{r}(t), a)$ as the immediate reward, given the queue status $\mathbf{r}(t)$, and the action of assigning to the a th queue. Denote by $c(t) = 1$ as that there is a vehicle meeting with the task-RSU, and $c(t) = 0$ otherwise. Then, based on the law of total probability

$$\begin{aligned} \mathbb{E}[g_t(\mathbf{r}(t), a)] &= \left(1 - \sum_{i=1, r_i(t) \neq F}^N p_i\right) \left(\sum_{i=1, r_i(t) \neq F}^N r_i(t) + c(t)\right) \\ &\quad + \sum_{i=1, r_i(t) \neq F}^N p_i \left(\sum_{n=1, r_n(t) \neq F}^N r_n(t) + c(t) \right. \\ &\quad \left. - r_i(t) - \mathbb{1}(a = i)c(t)\right) \\ &= \sum_{i=1, r_i(t) \neq F}^N r_i(t) + c(t) - \sum_{i=1, r_i(t) \neq F}^N p_i r_i(t) \\ &\quad + \mathbb{1}(a = i)c(t) \end{aligned} \quad (38)$$

where p_i denotes the probability that a task which belongs to queue- n is executed in this time slot

$$p_i = (r_i(t) + \mathbb{1}(i = a)c(t))\mu B\delta \quad \forall 1 \leq i \leq N, r_i(t) \neq F. \quad (39)$$

After a task completion from queue- n , the queue- n is deleted. Based on (38) and (39), it is easy to see that the myopic policy which maximizes $\mathbb{E}[g_t(\mathbf{r}(t), a)]$ is to assign the task- n_m to the vehicle, where

$$n_m = \arg \min_{i \in \{1, \dots, N\}, r_i(t) \neq F} r_i(t). \quad (40)$$

This coincides with the BETA policy.

In what follows, we will show the myopic, i.e., BETA policy, is optimal. The proof is based on backwards induction on time t . Define the total reward from time t on based on the BETA policy given the current $\mathbf{r}(t)$ as $W_t(\mathbf{r}(t))$. The following statements will be shown to be valid based on backwards induction.

Statement I: $W_t(\mathbf{r}(t))$ is the optimal total reward from t on, i.e., the BETA policy is optimal.

Statement II: For each $t \geq 0$, if $r_i(t) \leq r_j(t)$

$$\begin{aligned} W_t(\mathbf{r}(t) \text{ with } r_i(t) = F) \\ \geq W_t(\mathbf{r}(t) \text{ with } r_j(t) = F). \end{aligned} \quad (41)$$

For notation simplicity, denote $W_t(\mathbf{r}(t)$ with $r_i(t) = F$ as $\hat{W}_{t,i}(\mathbf{r}(t))$. Given that statements I and II are both valid from time $t + 1$ to D . We want to show that they are also valid at

time t . Define the total reward from time t on, given the action a , and followed by actions based on the BETA policy from $t + 1$ on, as $q_t(\mathbf{r}(t), a)$. Then statement I is equivalent to

$$W_t(\mathbf{r}(t)) = \max_a q_t(\mathbf{r}(t), a) \quad (42)$$

where $q_t(\mathbf{r}(t), a)$ is given in (43), shown at the bottom of this page. It follows that:

$$\begin{aligned} a_m &= \arg \max_a q_t(\mathbf{r}, a) \\ &= \arg \max_a \left(-2r_a(t) + \hat{W}_{t+1,a}(\mathbf{r}(t))\right). \end{aligned} \quad (44)$$

Based on the induction hypothesis in statement II, it follows that:

$$\begin{aligned} a_m &= \arg \max_a \left(\hat{W}_{t+1,a}(\mathbf{r}, a)\right) \\ &= \arg \min_a [r_a(t)]. \end{aligned} \quad (45)$$

With this, we establish the fact that the BETA policy is optimal at time t , given the induction hypothesis in statements I and II.

To show statement II is also valid at time t , it follows that given $r_i \leq r_j$ and $m = \arg \min_{1 \leq n \leq N} r_n(t)$, $\forall i, j \in \{1, \dots, N\}$:

$$\begin{aligned} \hat{W}_{t,i}(\mathbf{r}(t)) - \hat{W}_{t,j}(\mathbf{r}(t)) \\ = r_j - r_i + \left(\left(r_j + \mathbb{1}(j = m)c(t)\right)^2 - \left(r_i + \mathbb{1}(i = m)c(t)\right)^2\right) \end{aligned} \quad (46)$$

$$\begin{aligned} + p_j \hat{W}_{t+1,i}(\mathbf{r}(t)) - p_i \hat{W}_{t+1,j}(\mathbf{r}(t)) \\ + \sum_{i=1, r_i(t) \neq F}^N \left(\hat{W}_{t+1,i}(\mathbf{r}(t)) - \hat{W}_{t+1,j}(\mathbf{r}(t))\right) \end{aligned} \quad (47)$$

$$+ \sum_{n \neq i} p_n \hat{W}_{t+1,n,i}(\mathbf{r}(t)) - \sum_{n \neq j} p_n \hat{W}_{t+1,n,j}(\mathbf{r}(t)). \quad (48)$$

It is easy to see that (46) ≥ 0 , since $r_j \geq r_i$.

To show (47) ≥ 0 , two cases are considered. If $r_i = r_j$, then (47) is trivially true. Otherwise given $r_j \geq r_i$, $r_i + 1 \leq r_j$, then

$$\begin{aligned} p_j &= (r_j + \mathbb{1}(m_j = j)c(t))\mu B\delta \geq (r_i + 1)\mu B\delta \\ &\geq (r_i + \mathbb{1}(m_i = i)c(t))\mu B\delta = p_i \end{aligned} \quad (49)$$

where

$$\begin{aligned} m_j &= \arg \min_{n \neq j} r_n(t) \\ m_i &= \arg \min_{n \neq i} r_n(t). \end{aligned} \quad (50)$$

Combining with the induction hypothesis statement II, we obtain (47) ≥ 0 .

$$\begin{aligned} q_t(\mathbf{r}(t), a) &= \mathbb{E}[g_t(\mathbf{r}(t), a)] + \sum_{i=1, r_i(t) \neq F}^N p_i W_{t+1}(\mathbf{r}(t)) + \sum_{i=1, r_i(t) \neq F}^N p_i \hat{W}_{t+1,i}(\mathbf{r}(t)) \\ &= \sum_{i=1, r_i(t) \neq F}^N r_i(t) + c(t) - \sum_{i=1, r_i(t) \neq F}^N (r_i(t) + \mathbb{1}(a = i)c(t))^2 \mu B\delta + \sum_{i=1, r_i(t) \neq F}^N p_i W_{t+1}(\mathbf{r}(t)) \\ &\quad + \sum_{i=1, r_i(t) \neq F}^N (r_i(t) + \mathbb{1}(a = i)c(t))\mu B\delta \hat{W}_{t+1,i}(\mathbf{r}(t)) \end{aligned} \quad (43)$$

To show that (53) ≥ 0 , let us consider the following four cases.

Case A: $m_i \neq j, m_j \neq i$. Then, $m_i = m_j$

$$(48) = r_j(t)\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) - r_i(t)\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) + \sum_{n \neq i,j} p_n \left(\hat{W}_{t+1,n,i}(\mathbf{r}(t)) - \hat{W}_{t+1,n,j}(\mathbf{r}(t)) \right) \geq 0 \quad (51)$$

based on $r_j(t) \geq r_i(t)$ and induction hypothesis statement II.

Case B: $m_i = j, m_j \neq i$. Then, $r_i(t) \leq r_j(t) = r_{m_i}(t) \leq r_n, \forall n \neq i, j$. It follows that $m_j = i$, and this case is infeasible.

Case C: $m_i \neq j, m_j = i$

$$(48) = (r_{m_i}(t) + c(t))\mu B\delta \hat{W}_{t+1,m_i,j}(\mathbf{r}(t)) + r_j(t)\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) \times (\mathbf{r}(t)) - (r_i(t) + c(t))\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) - r_{m_i}(t)\mu B\delta \hat{W}_{t+1,m_i,j}(\mathbf{r}(t)) \geq 0 \quad (52)$$

based on the same technique to prove (47) ≥ 0 .

Case D: $m_i = j, m_j = i$

$$(48) = (r_j(t) + c(t))\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) - (r_i(t) + c(t))\mu B\delta \hat{W}_{t+1,i,j}(\mathbf{r}(t)) \geq 0. \quad (53)$$

With this, we complete the induction proof, and hence the theorem is proved.

APPENDIX B PROOF OF THEOREM 2

We prove the theorem by first showing that the deadline violation probability is upper bounded by a system with one task and $\lambda S/N$ vehicle meeting rate. Then the performance of the task system is derived in a closed-form.

Lemma 3: The deadline violation probability is upper bound by a system with one task and a vehicle meeting rate of $\lambda S\mu/N$.

Proof: Consider a round-Robin task assignment policy which assigns the $(k \bmod N)$ th task to the k th meeting vehicle, even if the task has been finished. Since the division of a Poisson process is also a Poisson process, the resultant meeting rate in MDP-2 for the k th queue is $\lambda S/N$. It is straightforward to see the performance of the BETA policy is upper bounded by this round-Robin policy by noticing that the BETA policy is the same with the round-Robin policy, except that it does not assign finished tasks. ■

Lemma 4: For a system with one task and a vehicle meeting rate of $\lambda S\mu/N$, the deadline violation probability is identical to a system with $\lambda S/N$ servers each of which has an i.i.d. service time distribution of

$$F(x) = 1 + \frac{\lambda_1}{\lambda_2 - \lambda_1} e^{-\lambda_2 x} - \frac{\lambda_2}{\lambda_2 - \lambda_1} e^{-\lambda_1 x}. \quad (54)$$

The task is completed once one of the servers finishes.

Proof: Since there is only one task, all the meeting vehicles are carrying the same task. The total number of vehicles is $\lambda S/N$ (ignoring the remainder of division), and each vehicle needs an extra service time of t_s after meeting at the task-RSU, where

$$t_s \sim \text{Exp}(B\mu). \quad (55)$$

Therefore, the service process is equivalent to one with $\lambda S/N$ servers, and each with a service time

$$t = t_a + t_s \quad (56)$$

where

$$t_a \sim \text{Exp}(\mu) \quad (57)$$

denotes the meeting delay distribution, t_a and t_s are independent.

The natural way to establish the distribution of the sum of two independent random variables is to leverage the characteristic function (CF). Given the CF of a random variable $x \sim \text{Exp}(\lambda)$ as

$$\varphi_x(z) = \lambda(\lambda - jz)^{-1} \quad (58)$$

where $j = \sqrt{-1}$ here, and the CF of t is, hence,

$$\begin{aligned} \varphi_t(z) &= \varphi_{t_a}(z) \cdot \varphi_{t_s}(z) \\ &= \lambda_1(\lambda_1 - jz)^{-1} \lambda_2(\lambda_2 - jz)^{-1} \\ &= \left(\frac{1}{\lambda_1 - jz} - \frac{1}{\lambda_2 - jz} \right) \frac{\lambda_1 \lambda_2}{\lambda_2 - \lambda_1} \\ &= \lambda_1(\lambda_1 - jz)^{-1} \cdot \frac{\lambda_2}{\lambda_2 - \lambda_1} + \lambda_2(\lambda_2 - jz)^{-1} \cdot \frac{\lambda_1}{\lambda_1 - \lambda_2}. \end{aligned} \quad (59)$$

It follows that the density function of t is:

$$f(t) = \frac{\lambda_2}{\lambda_2 - \lambda_1} \lambda_1 e^{-\lambda_1 t} + \frac{\lambda_1}{\lambda_1 - \lambda_2} \lambda_2 e^{-\lambda_2 t} \quad (60)$$

and hence the distribution function follows immediately. ■

Based on Lemmas 3 and 4, the violation happens when all servers fail to execute the task on time, that is,

$$t_m \triangleq \min_{i \in \{1, \dots, M\}} [t_i] > D \quad (61)$$

where M is a Poisson distributed random variable with mean of $\lambda S/N$ to follow the assumption 1) in Section II. The deadline violation probability follows, which reads:

$$\Pr\{t_m > D\} = \sum_{m=0}^{\infty} \Pr\{t_m > D | M = m\} \Pr\{M = m\}. \quad (62)$$

To proceed, we have the following lemma.

Lemma 5: Given a distribution function $F(x)$

$$v(x) \triangleq \sum_{k=0}^{\infty} \frac{1}{k!} \alpha^k e^{-\alpha} (1 - F(x))^k = e^{-\alpha F(x)}. \quad (63)$$

Proof:

$$v(x) = e^{-\alpha} + \sum_{k=1}^{\infty} \frac{1}{k!} e^{-\alpha} \alpha^k (1 - F(x))^k \quad (64)$$

$$\begin{aligned} \frac{dv(x)}{dx} &= - \sum_{k=1}^{\infty} \frac{1}{k!} e^{-\alpha} \alpha^k (1 - F(x))^{k-1} \frac{dF(x)}{dx} \\ &= -\alpha \sum_{k=0}^{\infty} \frac{1}{k!} e^{-\alpha} \alpha^k (1 - F(x))^k \frac{dF(x)}{dx} \\ &= -\alpha \frac{dF(x)}{dx} v(x). \end{aligned} \quad (65)$$

Therefore,

$$\frac{dv(x)}{v(x)} = -\alpha dF(x) \quad (66)$$

$$v(x) = Ce^{-\alpha F(x)}. \quad (67)$$

Since when $x = 0$, $F(0) = 0$, then $v(0) = 1$, $C = 1$, and $v(x) = e^{-\alpha F(x)}$. ■

Based on Lemma 5

$$\Pr\{t_m > D\} = e^{-\frac{\lambda S}{N} F(x)} \quad (68)$$

where $F(x)$ is given in (54), which concludes the proof. ■

APPENDIX C

PROOF OF COROLLARY 1

Consider the Taylor series

$$e^{-\lambda x} = 1 - \lambda x + \frac{1}{2}\lambda^2 x^2 + o(x^2) \quad (69)$$

where $o(x^n)$ denotes higher order infinitesimals of x^n . When $\lambda_1 D$ and $\lambda_2 D$ approach zeros, the upper bound in Theorem 2 is asymptotically

$$\begin{aligned} & \exp\left\{-\frac{\lambda S}{N}\left(1 + \frac{\lambda_1}{\lambda_2 - \lambda_1}e^{-\lambda_2 D} - \frac{\lambda_2}{\lambda_2 - \lambda_1}e^{-\lambda_1 D}\right)\right\} \\ & \xrightarrow{D \rightarrow 0} \exp\left\{-\frac{\lambda S}{N}\left(1 + \frac{\lambda_1}{\lambda_2 - \lambda_1}\left(1 - \lambda_2 x + \frac{1}{2}\lambda_2^2 D^2\right) - \frac{\lambda_2}{\lambda_2 - \lambda_1}\left(1 - \lambda_1 D + \frac{1}{2}\lambda_1^2 D^2\right) + o(D^2)\right)\right\} \\ & = \exp\left\{-\frac{\lambda S}{2N}\lambda_1 \lambda_2 D^2 + o(D^2)\right\}. \end{aligned} \quad (70)$$

The conclusion follows immediately by noticing $\lambda_1 = \mu$ and $\lambda_2 = B\mu$, which completes the proof.

APPENDIX D

PROOF OF THEOREM 3

First, it is easy to see that

$$L^* = \arg \max_{0 \leq L \leq L_{\max}} [V(L)L] \quad (71)$$

$$L^\dagger = \arg \max_{0 \leq L \leq L_{\max}} [V^2(L)L]. \quad (72)$$

It is obvious that both L^* and L^\dagger are positive. If $L^* = L_{\max}$, then $L^\dagger \leq L^*$ is trivially true. Otherwise, we have, based on the first-order Karush–Kuhn–Tucker (KKT) condition

$$V(L^*) + L^* V'(L^*) = 0 \quad (73)$$

$$V(L^\dagger) + 2L^\dagger V'(L^\dagger) = 0. \quad (74)$$

It is easy to find out that if $V(L)$ is concave, then $V'(L)$ is non-increasing, and therefore $L^* \geq L^\dagger$. Otherwise if $V(L)$ satisfies the condition in (27), wherein $V(L)$ has a polynomial form, it is sufficient to show for each monomial term, $L^* \geq L^\dagger$. Specifically, we consider

$$V(L) = g - cL^\alpha \quad (75)$$

where $\alpha \in (0, +\infty) \cup (-1, -(1/2))$, $c \geq 0$, $\forall k$ and $g \geq 0$. It follows based on the KKT condition that:

$$L^* = \left(\frac{g}{c(1+\alpha)}\right)^{\frac{1}{\alpha}} \quad (76)$$

$$L^\dagger = \left(\frac{g}{c(1+2\alpha)}\right)^{\frac{1}{\alpha}}. \quad (77)$$

Hence $L^* \geq L^\dagger$ is equivalent to

$$\frac{g\alpha}{c(1+2\alpha)(1+\alpha)} \geq 0. \quad (78)$$

The condition in (27) follows immediately, which concludes the proof. ■

APPENDIX E

PROOF OF COROLLARY 2

In our model, the vehicle density is λ . The rate of traffic flow is the reciprocal of the intervehicle-meeting time at the task-RSU, that is,

$$F = \mathbb{E}\left[\frac{1}{\min[t_1, \dots, t_M]}\right] = \lambda S \mu \quad (79)$$

where t_i denotes the intermeeting time of the i th vehicle. Therefore, the average speed is

$$V = \frac{F}{\lambda} = S \mu. \quad (80)$$

Combining with (25), we obtain

$$\mu = \frac{V_{\max}}{S} \left(1 - \frac{\lambda}{L_{\max}}\right). \quad (81)$$

Given that in the short deadline regime, the deadline violation probability follows the Rayleigh distribution given in Corollary 1, the optimum vehicle density is, therefore,

$$L^\dagger = \arg \max_{0 \leq \lambda \leq L_{\max}} [\lambda \mu^2]. \quad (82)$$

Combining with (81), the conclusion of the proposition follows immediately based on the Cauchy–Schwarz inequality.

REFERENCES

- [1] J. B. Kenney, “Dedicated short-range communications (DSRC) standards in the United States,” *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.
- [2] S. Chen, J. Hu, Y. Shi, and L. Zhao, “LTE-V: A TD-LTE-based V2X solution for future vehicular network,” *IEEE Internet Things J.*, vol. 3, no. 6, pp. 997–1005, Dec. 2016.
- [3] C.-X. Wang *et al.*, “Cellular architecture and key technologies for 5G wireless communication networks,” *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.
- [4] X. Hou *et al.*, “Vehicular fog computing: A viewpoint of vehicles as the infrastructures,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 3860–3873, Jun. 2016.
- [5] C.-C. Lin, D.-J. Deng, and C.-C. Yao, “Resource allocation in vehicular cloud computing systems with heterogeneous vehicles and roadside units,” *IEEE Internet Things J.*, to be published.
- [6] K. Zheng, H. Meng, P. Chatzimisios, L. Lei, and X. Shen, “An SMDP-based resource allocation in vehicular cloud computing systems,” *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7920–7928, Dec. 2015.
- [7] I. Jang, S. Choo, M. Kim, S. Pack, and G. Dan, “The software-defined vehicular cloud: A new level of sharing the road,” *IEEE Veh. Technol. Mag.*, vol. 12, no. 2, pp. 78–88, Jun. 2017.

- [8] S. Bitam, A. Mellouk, and S. Zeadally, "VANET-cloud: A generic cloud computing model for vehicular ad hoc networks," *IEEE Wireless Commun.*, vol. 22, no. 1, pp. 96–102, Feb. 2015.
- [9] S. Abdelhamid, H. S. Hassanein, and G. Takahara, "Vehicle as a resource (VaaR)," *IEEE Netw.*, vol. 29, no. 1, pp. 12–17, Jan./Feb. 2015.
- [10] M. Gerla, "Vehicular cloud computing," in *Proc. 11th Annu. Mediterr. Ad Hoc Netw. Workshop (Med-Hoc-Net)*, Jun. 2012, pp. 152–155.
- [11] E. Lee, E.-K. Lee, M. Gerla, and S. Y. Oh, "Vehicular cloud networking: Architecture and design principles," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 148–155, Feb. 2014.
- [12] M. Whaiduzzaman, M. Sookhak, A. Gani, and R. Buyya, "A survey on vehicular cloud computing," *J. Netw. Comput. Appl.*, vol. 40, pp. 325–344, Apr. 2014.
- [13] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh, "Rethinking vehicular communications: Merging VANET with cloud computing," in *Proc. 4th IEEE Int. Conf. Cloud Comput. Technol. Sci.*, Taipei, Taiwan, Dec. 2012, pp. 606–609.
- [14] S. Olariu, I. Khalil, and M. Abuelela, "Taking VANET to the clouds," *Int. J. Pervasive Comput. Commun.*, vol. 7, no. 1, pp. 7–21, 2011.
- [15] K. Mershad and H. Artail, "Finding a STAR in a vehicular cloud," *IEEE Intell. Transp. Syst. Mag.*, vol. 5, no. 2, pp. 55–68, Apr. 2013.
- [16] S. Han, C. L. I, Z. Xu, and C. Rowell, "Large-scale antenna systems with hybrid analog and digital beamforming for millimeter wave 5G," *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 186–194, Jan. 2015.
- [17] S. Yousefi, E. Altman, R. El-Azouzi, and M. Fathy, "Analytical model for connectivity in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3341–3356, Nov. 2008.
- [18] L. Zhou, Y. Zhang, K. Song, W. Jing, and A. V. Vasilakos, "Distributed media services in P2P-based vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 2, pp. 692–703, Feb. 2011.
- [19] L. Zhou, R. Q. Hu, Y. Qian, and H.-H. Chen, "Energy-spectrum efficiency tradeoff for video streaming over mobile ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 5, pp. 981–991, May 2013.
- [20] L. Zhou, "Mobile device-to-device video distribution: Theory and application," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 12, no. 3, pp. 1–23, Jun. 2016.
- [21] D. Eckhoff, C. Sommer, R. German, and F. Dressler, "Cooperative awareness at low vehicle densities: How parked cars can help see through buildings," in *Proc. IEEE GLOBECOM*, Kathmandu, Nepal, Dec. 2011, pp. 1–6.
- [22] N. Liu, M. Liu, W. Lou, G. Chen, and J. Cao, "PVA in VANETs: Stopped cars are not silent," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011, pp. 431–435.
- [23] K. Kumar and Y.-H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?" *Computer*, vol. 43, no. 4, pp. 51–56, Apr. 2010.
- [24] C. You, K. Huang, H. Chae, and B.-H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.
- [25] H. Liang, L. X. Cai, D. Huang, X. Shen, and D. Peng, "An SMDP-based service model for interdomain resource allocation in mobile cloud networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 5, pp. 2222–2232, Jun. 2012.
- [26] K. Gardner, S. Zbarsky, S. Doroudi, M. Harchol-Balter, and E. Hyttia, "Reducing latency via redundant requests: Exact analysis," *SIGMETRICS Perform. Eval. Rev.*, vol. 43, no. 1, pp. 347–360, Jun. 2015.
- [27] N. B. Shah, K. Lee, and K. Ramchandran, "When do redundant requests reduce latency?" *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 715–722, Feb. 2016.
- [28] Y. Sun, Z. Zheng, C. E. Koksal, K.-H. Kim, and N. B. Shroff, "Provably delay efficient data retrieving in storage clouds," in *Proc. IEEE INFOCOM*, Hong Kong, Apr. 2015, pp. 585–593.
- [29] D. Wang, G. Joshi, and G. Wornell, "Using straggler replication to reduce latency in large-scale parallel computing," *SIGMETRICS Perform. Eval. Rev.*, vol. 43, no. 3, pp. 7–11, Dec. 2015.
- [30] Y. Kim, R. Righter, and R. Wolff, "Job replication on multi-server systems," *Adv. Appl. Probab.*, vol. 41, no. 2, pp. 546–575, 2009.
- [31] J. Dean and L. A. Barroso, "The tail at scale," *ACM Commun.*, vol. 56, no. 2, pp. 74–80, Feb. 2013.
- [32] R. P. Roess, E. S. Prassas, and W. R. McShane, *Traffic Engineering*. Upper Saddle River, NJ, USA: Prentice-Hall, 2011.
- [33] S. N. Chiu, D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic Geometry and Its Applications*. Chichester, U.K.: Wiley, 2013.
- [34] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: A survey," *IEEE Commun. Mag.*, vol. 46, no. 9, pp. 59–67, Sep. 2008.
- [35] A. O. Allen, *Probability, Statistics, and Queueing Theory*. New York, NY, USA: Academic, 2014.
- [36] F. Poloczek and F. Ciucu, "Contrasting effects of replication in parallel systems: From overload to underload and back," in *Proc. ACM SIGMETRICS*, Juan-les-Pins, France, Jun. 2016, pp. 375–376.
- [37] R. G. Gallager, *Discrete Stochastic Processes*, vol. 321. New York, NY, USA: Springer, 2012.
- [38] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 1. Belmont, MA, USA: Athena Sci., 1995.
- [39] T. Toledo, M. E. Ben-Akiva, D. Darda, M. Jha, and H. N. Koutsopoulos, "Calibration of microscopic traffic simulation models with aggregate data," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 1876, no. 1, pp. 10–19, 2004.
- [40] M. Van Aerde, "Single regime speed-flow-density relationship for congested and uncongested highways," in *Proc. 74th TRB Annu. Conf.*, 1995.
- [41] D. R. Drew, *Traffic Flow Theory and Control*. New York, NY, USA: McGraw-Hill, 1968.
- [42] L. A. Pipes, "Car following models and the fundamental diagram of road traffic," *Transp. Res.*, vol. 1, no. 1, pp. 21–29, 1967.
- [43] X. Guo, R. Singh, P. R. Kumar, and Z. Niu, "A high reliability asymptotic approach for packet inter-delivery time optimization in cyber-physical systems," in *Proc. MobiHoc*, Hangzhou, China, 2015, pp. 197–206.
- [44] L. Schrage, "A proof of the optimality of the shortest remaining processing time discipline," *Oper. Res.*, vol. 16, no. 3, pp. 687–690, 1968.
- [45] I.-H. Hou and P. R. Kumar, "Queueing systems with hard delay constraints: A framework for real-time communication over unreliable wireless channels," *Queueing Syst.*, vol. 71, nos. 1–2, pp. 151–177, Jun. 2012.
- [46] Q. Zhao, B. Krishnamachari, and K. Liu, "On myopic sensing for multi-channel opportunistic access: Structure, optimality, and performance," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 5431–5440, Dec. 2008.



Zhiyuan Jiang (S'12–M'16) received the B.E. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2010 and 2015, respectively.

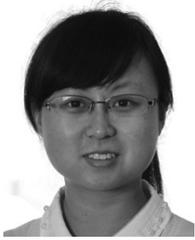
From 2013 to 2014, he was a Visiting Student with the WiDeS Laboratory, University of Southern California, Los Angeles, CA, USA. From 2015 to 2016, he was with Ericsson Research, Pisa, Italy. He is currently a Post-Doctoral Fellow with the Department of Electronic Engineering, Tsinghua University. His current research interests include

massive MIMO systems, stochastic optimization, and information theory.



Sheng Zhou (S'06–M'12) received the B.E. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2005 and 2011, respectively.

In 2010, he was a Visiting Student with the Wireless System Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA, USA, for five months. From 2014 to 2015, he was a Visiting Researcher with the Central Research Laboratory, Hitachi Ltd., Tokyo, Japan. He is currently an Associate Professor with the Department of Electronic Engineering, Tsinghua University. His current research interests include cross-layer design for multiple antenna systems, mobile edge computing, and green wireless communications.



Xueying Guo (S'14–M'17) received the B.E. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2011 and 2017, respectively.

From 2013 to 2014, she was a Visiting Scholar with the Computer Engineering and System Group, Department of Electrical and Computer Engineering, Texas A&M University, College Station, TX, USA. She is a Post-Doctoral Researcher with the Department of Computer Science, University of California at Davis, Davis, CA, USA. Her current

research interests include machine learning, wireless networking, and edge cloud systems.



Zhisheng Niu (M'98–SM'99–F'12) received the Graduation degree from Beijing Jiaotong University, Beijing, China, in 1985, and the M.E. and D.E. degrees from the Toyohashi University of Technology, Toyohashi, Japan, in 1989 and 1992, respectively.

From 1992 to 1994, he was with Fujitsu Laboratories Ltd., Kawasaki, Japan. In 1994, he joined Tsinghua University, Beijing, where he is currently a Professor with the Department of Electronic Engineering. His current research interests include

queuing theory, traffic engineering, mobile Internet, radio resource management of wireless networks, and green communication and networks.

Dr. Niu was a recipient of the Outstanding Young Researcher Award from the Natural Science Foundation of China in 2009 and the Best Paper Award from the IEEE Communication Society Asia–Pacific Board in 2013. He was a co-recipient of the Best Paper Awards from the 13th, 15th, and 19th Asia–Pacific Conference on Communication in 2007, 2009, and 2013, respectively, the International Conference on Wireless Communications and Signal Processing, and the Best Student Paper Award from the 25th International Teletraffic Congress. He was an active volunteer for various academic societies, including the Director of Conference Publications from 2010 to 2011, the Director of the Asia–Pacific Board of the IEEE Communication Society from 2008 to 2009, a Membership Development Coordinator of the IEEE Region 10 from 2009 to 2010, the Councillor of the IEICE Japan from 2009 to 2011, and a Council Member of the Chinese Institute of Electronics from 2006 to 2011. He was a Distinguished Lecturer from 2012 to 2015, the Chair of the Emerging Technology Committee of the IEEE Communication Society from 2014 to 2015, a Distinguished Lecturer of the IEEE Vehicular Technologies Society, from 2014 to 2016, a member of the Fellow Nomination Committee of the IEICE Communication Society from 2013 to 2014, a Standing Committee member of the Chinese Institute of Communications from 2012 to 2016, and an Associate Editor-in-Chief of the IEEE/CIC joint publication *China Communications*. He is a Fellow of the IEICE.