This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2874635, IEEE Wireless Communications Letters

A Dynamic Pricing Strategy for Vehicle Assisted Mobile Edge Computing Systems

Di Han, Wei Chen, *Senior Member, IEEE*, and Yuguang Fang, *Fellow, IEEE*

***Abstract*—The idle computing resources of parked vehicles could be utilized to improve performance by assisting task executions in mobile edge computing (MEC) systems. As a result, the owner of a vehicle could be compensated, resulting in a win-win situation. A dynamic pricing strategy is proposed to minimize the average cost of the MEC system under the constraints on Quality of Service (QoS) by adjusting the price constantly based on the current system state. To do so, a cost minimization problem is solved to obtain the optimal dynamic pricing strategy efficiently. Finally, the optimization results are validated with extensive simulations.**

***Index Terms*—Mobile edge computing, dynamic pricing strategy, autonomous vehicle, Markov chain.**

I. Introduction

In future smart city, vehicles equipped with power capability in communication, computing, and storage, e.g., autonomous vehicles, could be viewed as important networking resources to handle the explosively growing wireless data traffic [1]. To ensure powerful performance, an expensive communication and computing unit must be installed in these vehicles. Nowadays, existing computing solutions for level 4 autonomous driving often cost tens of thousands of dollars [2]. However, the current utilization rate of vehicles is not very high, e.g., 2016 average driving time per day in US is only 50.6 minutes according to the survey of AAA Foundation for Traffic Safety. Thus, these computing units will be left idle most of time. To make full use of these idle computing resources, we could incentivize owners of these vehicles to allow their vehicles to be used for processing computing tasks..

MEC is a promising paradigm to enable mobile devices to enjoy resourceful computing power with lower latency [3] and dynamic allocation of the computing resources in the MEC is an interesting research issue to be addressed in the future [4]. It could be a potential scenario in future smart cities where the vehicles with computing units could be utilized as temporary servers of an MEC system, particularly when the computing resources owned by the MEC system itself is not sufficiently enough to guarantee QoS. A reasonable solution is to allow computing units of parked vehicles, e.g., autonomous

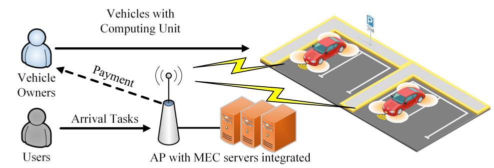

Fig. 1. A vehicle-assisted MEC system.

vehicles, be leased to the MEC system to execute computing tasks and exchange data with the MEC system through the vehicle-to-infrastructure (V2I) communication, as shown in Fig. 1. This will be a win-win situation where not only the MEC system could achieve better performance but also the owners of these vehicles could gain economic benefits from the operator of the MEC system, especially when these vehicles are not energy-hungry, e.g., electric vehicles equipped with large battery packs.

However, the arrival of computing tasks and locations of vehicles, i.e., entries to and exits from the coverage range of the MEC system, are greatly stochastic and uncertain, which are hard to predict and control accurately. Thus, the performance of a fixed price strategy is often very poor since it does not take the real-time dynamic change into consideration, e.g., the number of tasks in execution and the number of parked vehicles in the coverage range of the MEC system. Dynamic pricing strategy could provide a more attractive approach by adjusting the price constantly, which has attracted great attention in both academia and industries. In [5], by implementing dynamic parking pricing strategy, travel delay of cruising and the generic congestion can be effectively curtailed in urban networks. Moreover, time-varying pricing strategies are widely used in electricity use, which charge more for energy use on peak to reduce peak demand [6]. Similarly, we could raise price to attract more parked vehicles when the servers are not sufficient to support computation tasks, and vice versa. Thus, there exists a tradeoff between the average cost, i.e., the average reward paid by the MEC system, and QoS of the MEC system.

In this paper, a dynamic pricing strategy is proposed to minimize the average cost with the constraints on QoS based on probabilistic scheduling approach. Each task is assumed in outage state and will be dropped if it cannot be served when it arrives at or has to be dropped before its execution is completed, and the packet loss rate is considered as the performance metric. The system could be modeled by a two-dimensional Markov chain whose state is determined by the system state, i.e., the number of tasks in execution and the

This research was supported in part by the National Science Foundation of China under Grant Nos. 61671269 and 61621091, and the national program for special support for eminent professionals (10000-talent program). The work of Y. Fang was supported in part by US National Science Foundation under grant CNS-1717736.

D. Han and W. Chen are with the Department of Electronic Engineering, Tsinghua University, Beijing 100084, China. (e-mail: hd15@mails.tsinghua.edu.cn, wchen@tsinghua.edu.cn).

Y. Fang is with the Department of Electrical and Computer Engineering and University of Florida, Gainesville, FL 32611, USA. (e-mail: fang@ece.ufl.edu).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2874635, IEEE Wireless Communications Letters

number of parked vehicles in the MEC system. Our objective is to find the optimal pricing strategy and, based on this objective, the optimization problem can be converted into a linear programming so that the optimal dynamic pricing strategy can be obtained efficiently.

II. System Model

As shown in Fig. 1, we consider an MEC system that consists of an AP node (integrated with n_0 MEC servers) and a parking lot where at most N_0 vehicles equipped with computing units could park. Assume that the parked vehicles could be unitized as the temporary MEC servers. To simplify the analysis, we assume that each vehicle has the identical computing capability [7] with an MEC server.

Time is divided into time slots. Let $N[t]$ and $M[t]$ denote the number of parked vehicles and computing tasks be executing in the MEC system at the beginning of the t-th time slot. Let $a_{\mathrm{v}}[t]$ and $d_{\mathrm{v}}[t]$ denote the number of vehicles arriving and departing in the t-th time slot. Likewise, $a_{\mathrm{u}}[t]$ and $d_{\mathrm{u}}[t]$ denote the number of tasks arriving and departing in the t-th time slot. Then the dynamic system state can be expressed as

$$\begin{cases} N[t+1] = \left(\min\left\{N[t] + a_{\mathrm{v}}[t] - d_{\mathrm{v}}[t], N_0\right\}\right)^+, \\ M[t+1] = \left(\min\left\{N[t+1] + n_0, M[t] + a_{\mathrm{u}}[t] - d_{\mathrm{u}}[t]\right\}\right)^+, \end{cases} \tag{1}$$

where the superscript '+' denotes nonnegative, i.e., $a^+ = \max\{a, 0\}$. When the available servers cannot support all tasks in the system, the packet loss occurs. The number of tasks dropped at the beginning of the t-th time slot is given by

$$l[t] = \max\left\{0, M[t] - N[t] - n_0\right\}. \tag{2}$$

We model task arrivals and departures as Bernoulli Process [8] and [9]. A new computing task arrives at the MEC system at the beginning of the t-th time slot with arrival rate λ_{u}, i.e.,

$$\begin{cases} \Pr\{a_{\mathrm{u}}[t] = 1\} = \lambda_{\mathrm{u}}, \\ \Pr\{a_{\mathrm{u}}[t] = 0\} = 1 - \lambda_{\mathrm{u}}. \end{cases} \tag{3}$$

At the end of each time slot, each task could be completed and departs from the system with departure rate μ_{u}. The departure of each task is independent with each other. Thus, we have

$$\Pr\{d_{\mathrm{u}}[t] = d\} = C(m, d)\mu_{\mathrm{u}}^d(1 - \mu_{\mathrm{u}})^{m-d}, \forall 0 \le d \le m, \tag{4}$$

where $m = M[t] > 0$ and $C(m, d)$ is the binomial coefficient. Otherwise, if $M[t] = 0$, we have $\Pr\{d_{\mathrm{u}}[t] = 0\} = 1$. Likewise, the distribution of $a_{\mathrm{v}}[t]$ and $d_{\mathrm{v}}[t]$ are dependent on the arrival rate $\lambda_{\mathrm{v}}[t]$ and departure rate $\mu_{\mathrm{v}}[t]$ of the parked vehicles, which is given by

$$\begin{cases} \Pr\{a_{\mathrm{v}}[t] = 1\} = \lambda_{\mathrm{v}}[t], \\ \Pr\{a_{\mathrm{v}}[t] = 0\} = 1 - \lambda_{\mathrm{v}}[t], \end{cases} \tag{5}$$

and

$$\Pr\{d_{\mathrm{v}}[t] = d\} = C(n, d)\mu_{\mathrm{v}}[t]^d(1 - \mu_{\mathrm{v}}[t])^{n-d}, \forall 0 \le d \le n, \tag{6}$$

where $n = N[t] > 0$, and $\lambda_{\mathrm{v}}[t]$ and $\mu_{\mathrm{v}}[t]$ are dependent on the price $c[t]$ at the current time slot. The price $c[t]$ is defined as the payment obtained by each vehicle in t-th time slot. Consider that K price standards are available, whose set is denoted by $\mathcal{C} = \{c_1, c_2, \ldots, c_K\}$ and $c_i < c_j$ for any $i < j$. Let λ_k and μ_k denote the arrival rate and departure rate of vehicle with a given price standard c_k, i.e., $\lambda_{\mathrm{v}}[t] = \lambda_k$ and $\mu_{\mathrm{v}}[t] = \mu_k$ when $c[t] = c_k$. It is reasonable to assume that higher price will attract vehicle to arrive and park with a higher probability and for a longer time. Thus, we have $\lambda_i < \lambda_j$ and $\mu_i < \mu_j$ for any $c_i < c_j$.

III. Dynamic Pricing Strategy

According to Eq. (1), the system state $M[t+1]$ and $N[t+1]$ in the next time slot only depend on the current state $M[t]$ and $N[t]$, and not on the state at the previous slots. Therefore, the system state can be formulated as a two-dimensional Markov chain with $M[t]$ and $N[t]$. We denote the state (m, n) as the system state $M[t] = m$ and $N[t] = n$.

At the beginning of each time slot, the state of the Markov chain could make transition to other states. For the ease of understanding, an instance with transition diagram of one state $(1, 1)$ is given in Fig. 2. To keep the figure legible, we denote $(a_{\mathrm{u}}[t], d_{\mathrm{u}}[t], a_{\mathrm{v}}[t], d_{\mathrm{v}}[t])$ as (a_1, d_1, a_2, d_2) for each link. The state $(1, 1)$ cannot make transition to the states in the next time slot that do not have a link with it, e.g., $(3, 2)$. Since packet drop occurs when $M[t] > N[t] + n_0$, the MEC system will have to drop one packet when state $(1, 1)$ transfers to state $(2, 0)$, which is illustrated in Fig. 2.

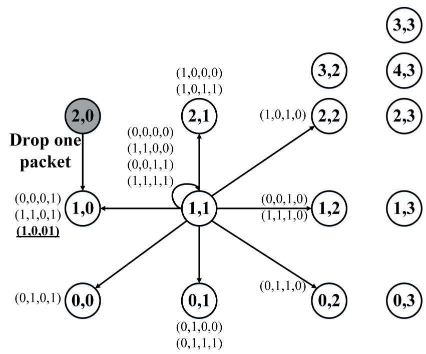

Fig. 2. Transition diagram of state $(1, 1)$ with $N_0 = 3$ and $n_0 = 1$.

Dynamic pricing strategy is to adjust price at the beginning of each time slot, which is determined by the probability $f^k_{m,n}$ of choosing price c_k given state (m, n), i.e.,

$$f^k_{m,n} = \Pr\left\{c[t] = c_k | M[t] = m, N[t] = n\right\}. \tag{7}$$

The normalization condition always holds for each $0 \le m \le N + n_0$ and $0 \le n \le N$,

$$\sum_{k=1}^{K} f^k_{m,n} = 1. \tag{8}$$

Let $R^k_{m,n}$ denote the average number of packets dropped in the current time slot at state (m, n) with price standard c_k, which is given by

$$R^k_{m,n} = \mathbb{E}\{l[t] | M[t] = m, N[t] = n, c[t] = c_k\}. \tag{9}$$

Denote $q_{m,n}(\Delta m, \Delta n)$ as the transition probability from the state (m, n) to state $(m + \Delta m, n + \Delta n)$ in the Markov

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LWC.2018.2874635, IEEE Wireless Communications Letters

chain. $R^k_{m,n}$ and $q_{m,n}(\Delta m, \Delta n)$ could be obtained according to Eqs. (1-4) in the Section II and details are omitted due to space limit. Based on this, the transition matrix of the Markov chain $\boldsymbol{H}$ could be obtained. Denote the steady-state distribution of this Markov chain by $\boldsymbol{\pi} = \{\pi_{0,0}, \pi_{0,1}, \ldots, \pi_{0,N_0}, \ldots, \pi_{n+n_0,n} \ldots, \pi_{N_0+n_0,N_0}\}$ which satisfies

$$\begin{cases} \boldsymbol{1}^T\boldsymbol{\pi} = 1, \\ \boldsymbol{H}\boldsymbol{\pi} = \pi. \end{cases} \tag{10}$$

In the t-th time-slot, when system state $(M[t], N[t]) = (m, n)$, the cost and expectation of packets dropped are given by nc_k and $R^k_{m,n}$ with probability $f^k_{m,n}$. Thus, the average cost and average expectation of tasks dropped are given by

$$C_{\text{ava}} = \sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} nc_k\pi_{m,n}f^k_{m,n} \tag{11}$$

and

$$E_{\text{los}} = \sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} R^k_{m,n}\pi_{m,n}f^k_{m,n}. \tag{12}$$

Furthermore, the expectation of number of the tasks arriving in the t-th time-slot is the arrival rate λ_{u}. Thus, the packet loss rate, which is the probability that a computing task is dropped before it is completed, is given by

$$P^{\text{ava}}_{\text{los}} = \frac{E_{\text{los}}}{\lambda_{\text{u}}}. \tag{13}$$

IV. Optimal Cost-QoS Tradeoff

In typical systems, the QoS of users should be guarantee firstly, i.e., the packet loss rate $P^{\text{ava}}_{\text{los}}$ cannot exceed the tolerance $P^{\text{th}}_{\text{los}}$. Furthermore, it is necessary to reduce the cost of the MEC system, i.e., average cost C_{ava}, as much as possible. Thus, we have the following optimization problem:

$$\min_{\boldsymbol{\pi}, f^k_{m,n}} \quad \sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} nc_k\pi_{m,n}f^k_{m,n} \tag{14.a}$$

$$\text{s.t} \quad \frac{1}{\lambda_{\text{u}}}\sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} R^k_{m,n}\pi_{m,n}f^k_{m,n} \le P^{\text{th}}_{\text{los}} \tag{14.b}$$

$$\boldsymbol{1}^T\boldsymbol{\pi} = 1 \tag{14.c}$$

$$\boldsymbol{H}\boldsymbol{\pi} = \pi \tag{14.d}$$

$$\sum_{k=1}^{K} f^k_{m,n} = 1 \quad \forall m, n \tag{14.e}$$

$$f^k_{m,n} \ge 0 \quad \forall m, n, k \tag{14.f}$$

$$\pi_{m,n} \ge 0 \quad \forall m, n, \tag{14.g}$$

where constraints (14.b) and (14.c-d) denote the constraints on QoS tolerance and the steady-state condition, respectively.

The objective function and constraints in optimization (14) are linear combinations of $\{\pi_{m,n}f^k_{m,n}\}$, $\{f^k_{m,n}\}$, or $\{\pi_{m,n}\}$. By recalling the normalization condition of $\{f^k_{m,n}\}$ in Eq. (8), $\pi_{m,n}$ can also be expressed as

$$\pi_{m,n} = \sum_{k=1}^{K}\pi_{m,n}f^k_{m,n} = \sum_{k=1}^{K} y^k_{m,n}. \tag{15}$$

By substituting Eq. (15) into problem (14), the steady-state condition can be expressed as constraint (16.c) and a matrix equation $\mathbf{Qy} = 0$ with $y^k_{m,n} = \{\pi_{m,n}f^k_{m,n}\}$ as variables, where the constant matrix is denoted by $\mathbf{Q}$ and can be derived from $\mathbf{H}$. In this way, the optimization (14) is converted into a linear programming which is summarized as follow:

$$\min_{\boldsymbol{\pi}, f^k_{m,n}} \quad \sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} nc_k y^k_{m,n} \tag{16.a}$$

$$\text{s.t} \quad \frac{1}{\lambda_{\text{u}}}\sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} R^k_{m,n}y^k_{m,n} \le P^{\text{th}}_{\text{los}} \tag{16.b}$$

$$\sum_{n=0}^{N_0}\sum_{m=0}^{n+n_0}\sum_{k=1}^{K} y^k_{m,n} = 1 \tag{16.c}$$

$$\boldsymbol{Q}\boldsymbol{y} = 0 \tag{16.d}$$

$$y^k_{m,n} \ge 0 \quad \forall m, n, k \tag{16.e}$$

This problem can be solved efficiently in polynomial time using interior-point method [10]. After the optimal solution ${y^k_{m,n}}^*$ of the linear programming (16) is obtained, the corresponding steady-state distribution can be represented as

$$\pi^*_{m,n} = \sum_{k=1}^{K} {y^k_{m,n}}^*. \tag{17}$$

To obtain the cost-optimal strategy, we can derive ${f^k_{m,n}}^*$ from ${y^k_{m,n}}^*$, which is given below.

Case 1 When $\pi^*_{m,n} \neq 0$, the optimal strategy is given by

$${f^k_{m,n}}^* = \frac{{y^k_{m,n}}^*}{\pi^*_{m,n}}. \tag{18}$$

Case 2 When $\pi^*_{m,n} = 0$, which means that the state (m, n) is a transient state. Then, a simple strategy can be used, i.e.,

$${f^k_{m,n}}^* = \frac{1}{k}. \tag{19}$$

The time complexity of deriving ${f^k_{m,n}}^*$ from ${y^k_{m,n}}^*$ is also polynomial. In conclusion, the cost-optimal dynamic pricing strategy could be obtained in polynomial time. Based on this result, the optimal cost-QoS tradeoff can be achieved.

Remark 1 *Our proposed approach could be extended and applied to a more generalized scenario where the priority of tasks and vehicles with different computing capability are considered. By modeling each type of task and vehicle into a queue, the system could be formulated into a Markov chain with more dimensions. Then, the cost-optimal dynamic pricing strategy could be obtained using the proposed approach.*

V. Numerical Results

In this section, we validate our theoretical results via simulation studies, and explain the outcomes in a more comprehensive way. Throughout this section, we set $n_0 = 3$, $K = 7$, $\mu_{\text{u}} = 0.2$, and other parameters are summarized in Table I. By solving the optimization problem (16), the optimal dynamic pricing strategy can be obtained.

TABLE I
PARAMETERS USED FOR SIMULATION

k	1	2	3	4	5	6	7
c_k	0	1	2	4	8	16	32
λ_k	0	0.2	0.3	0.4	0.5	0.6	0.7
μ_k	1	0.3	0.25	0.2	0.15	0.1	0.05

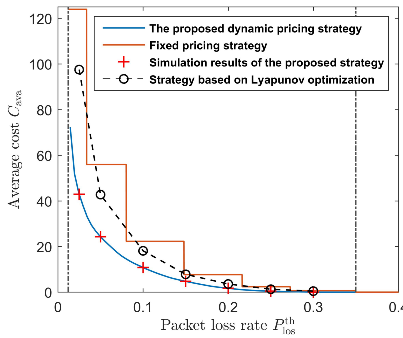

Fig. 3. Optimal cost-QoS tradeoffs between different strategies.

Fig. 3 compares the numerical results between the proposed dynamic pricing strategy, one another dynamic pricing strategy based on Lyapunov optimization, and fixed pricing strategy with $N_0 = 4$ and $\lambda_{\rm u} = 0.8$. The basic idea of the Lyapunov optimization [11] is to minimize its Lyapunov drift-plus-penalty function, whose objective is to stabilize the virtual queue $N[t] + n_0 - M[t]$ while optimizing the average cost.

In Fig. 3, simulation results of the proposed dynamic pricing strategy are given by Monte-Carlo simulation, which match perfectly well with the optimization results. It can be seen that with the decrease of packet loss rate constraint $P_{\rm los}^{\rm th}$, i.e., higher QoS requirement, the required cost rises in all strategies. Additionally, when the cost is large enough, the packet loss rate in all strategies will decrease to a minimum, which verifies the performance improvement by utilizing the computing units of parked vehicles as temporary MEC servers. Moreover, for any $P_{\rm los}^{\rm th}$, the average cost in the fixed pricing strategy is always higher than that in the proposed dynamic pricing strategies. The average cost of the strategy based on Lyapunov optimization is also higher since it does not take the distribution of arrival and departure of the task and vehicle into consideration. Thus, the performance improvement of the proposed dynamic pricing strategy has been verified.

Fig. 4 presents the optimal cost versus the size of the parking lot with different task arrival rates $\lambda_{\rm u}$.The packet loss rate constraint $P_{\rm los}^{\rm th}$ is set to 0.15. As expected, the cost decreases when the size of parking lot N_0 increases and then approaches to different asymptotes, where the line with higher $\lambda_{\rm u}$ exhibits higher average cost. For a given N_0, the system with higher $\lambda_{\rm u}$ requires higher cost. Therefore, for a busy MEC system with higher $\lambda_{\rm u}$, a larger parking lot is required or otherwise higher cost has to be incurred.

VI. CONCLUSION

In this paper, we have investigated a dynamic pricing strategy for vehicle assisted MEC systems. By adjusting the price dynamically to control the arrival and departure rates of vehicles, the cost of the MEC system will be minimized under a given constraint on QoS, which is evaluated by the packet drop rate. The system is modeled as a two-dimensional Markov chain. Then the average cost and QoS could be obtained by analyzing the steady-state distribution of Markov chain. Based on these, the optimization problem is formulated and solved. Moreover, the cost-optimal dynamic pricing strategy could be obtained to minimize the cost of the MEC system and achieve optimal cost-QoS tradeoff, which has a significant performance improvement compared with the fixed pricing strategy. Finally, our theoretical results is validated by comprehensive simulations.

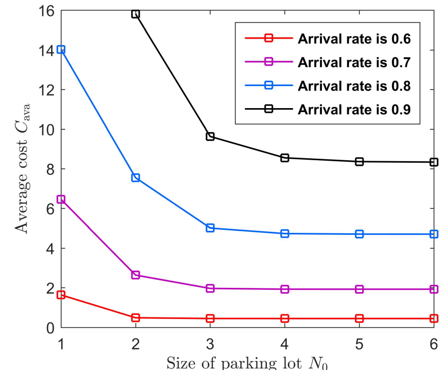

Fig. 4. Optimal cost versus parking lost size N.

REFERENCES

[1] H. Ding, C. Zhang, Y. Cai, and Y. Fang, "Smart cities on wheels: A newly emerging vehicular cognitive capability harvesting network for data transportation," *IEEE Wireless Communications*, vol. 25, no. 2, pp. 160–169, Apr. 2018.

[2] S. Liu, J. Tang, Z. Zhang, and J. Gaudiot, "Computer architectures for autonomous driving," *IEEE Computer*, vol. 50, no. 8, pp. 18–25, Aug. 2017.

[3] M. T. Beck, M. Werner, S. Feld, and T. Schimper, "Mobile edge computing: A taxonomy," in *Proc. International ConferenFce on Advances in Future Internet*, Nov. 2018.

[4] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628–1656, Mar. 2017.

[5] N. Zheng and N. Geroliminis, "Modeling and optimization of multimodal urban networks with limited parking and dynamic pricing," *Transportation Research Part B: Methodological*, vol. 83, pp. 36–58, Jan. 2016.

[6] G. R. Newsham and B. G. Bowker, "The effect of utility time-varying pricing and load control strategies on residential summer peak electricity use: A review," *Energy Policy*, vol. 38, no. 7, pp. 3289–3296, July 2010.

[7] X. Hou, Y. Li, M. Chen, D. Wu, and S. Chen, "Vehicular fog computing: A viewpoint of vehicles as the infrastructures," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 3860–3873, Feb. 2016.

[8] J. Liu, Y. Mao, J. Zhang, and K. B. Letaief, "Delay-optimal computation task scheduling for mobile-edge computing systems," in *Proc. IEEE ISIT*, Aug. 2016.

[9] M. Jia, W. Liang, Z. Xu, and M. Huang, "Cloudlet load balancing in wireless metropolitan area networks," in *Proc. IEEE INFOCOM*, Apr. 2016.

[10] S. P. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge Unviersity Press, 2004.

[11] W. Sun, J. Liu, Y. Yue, and H. Zhang, "Double auction-based resource allocation for mobile edge computing in industrial internet of things," *IEEE Transactions on Industrial Informatics*, DOI: 10.1109/TII.2018.2855746, 2018.