Partial Computation Offloading and Adaptive Task Scheduling for 5G-Enabled Vehicular Networks

Zhaolong Ning, *Senior Member, IEEE*, Peiran Dong, Xiaojie Wang, Xiping Hu, Jiangchuan Liu, *Fellow, IEEE*, Lei Guo, Bin Hu, *Senior Member, IEEE*, Ricky Y. K. Kwok, *Fellow, IEEE*, and Victor C. M. Leung, *Fellow, IEEE*

Abstract—A variety of novel mobile applications are developed to attract the interests of potential users in the emerging 5G-enabled vehicular networks. Although computation offloading and task scheduling have been widely investigated, it is rather challenging to decide the optimal offloading ratio and perform adaptive task scheduling in high-dynamic networks. Furthermore, the scheduling policy made by the network operator may be violated, since vehicular users are rational and selfish to maximize their own profits. By considering the incentive compatibility and individual rationality of vehicular users, we present POETS, an efficient partial computation offloading and adaptive task scheduling algorithm to maximize the overall system-wide profit. Specially, a two-sided matching algorithm is first proposed to derive the optimal transmission scheduling discipline. After that, the offloading ratio of vehicular users can be obtained through convex optimization, without any information of other users. Furthermore, a non-cooperative game is constructed to derive the payoff of vehicular users that can reach the equilibrium between users and the network operator. Theoretical analyses and performance evaluations based on real-world traces of taxies demonstrate the effectiveness of our proposed solution.

Index Terms—5G-enabled vehicular networks, partial offloading, adaptive scheduling, Nash equilibrium

1 Introduction

5G-enabled vehicular networks connect billions of vehicles, based on which a variety of novel applications, such as interactive gaming, face recognition, and augmented reality, are developed. To attract the interests of potential users, mobile applications are always rich and live, consuming tremendous computation resources and battery energy, which raises significant challenges for resource-constrained facilities [1], [2]. Without stable power supply, vehicles cannot tolerate the consumption of tremendous computation resources and battery energy. To overcome the above obstacles, Mobile Edge Computing (MEC) is envisioned as a promising paradigm by enabling users to offload computation-intensive tasks to resource-sufficient servers. In proximity of users, MEC can provide pervasive and agile services with the assistance of ubiquitous wireless networks (4G/5G macro-cells or RoadSide Units (RSUs)).

Many existing researches leverage cellular spectrum for computation offloading by either Time Division Multiple Access (TDMA) or Orthogonal Frequency Division Multiple Access (OFDMA) [3]. However, limited channel resources are increasingly difficult to satisfy the growing service requirements of users [4]. The fierce competitions on channel resources lead to an unbearable communication delay, and the decline of users' quality of experience. Thus, 5G technology is developed to satisfy the stringent requirements of vehicular networks, such as ultra-low latency, continuity of experience and high reliability [5]. Nevertheless, there are still some critical challenges to be addressed.

- Scarce wireless spectrum resources cannot satisfy the requirements of excessive vehicular users. The transmission scheduling in 5G-enabled vehicular networks is more complicated than that in OFDMA and TDMA networks.

- *Zhaolong Ning is with the Chongqing Key Laboratory of Mobile Communications Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China, and also with the School of Software, Dalian University of Technology, Dalian 116620, China. E-mail: z.ning@ieee.org.*
- *Peiran Dong is with the School of Software, Dalian University of Technology, Dalian 116620, China. E-mail: peiran_dong@outlook.com.*
- *Xiaojie Wang is with the Chongqing Key Laboratory of Mobile Communications Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China, and also with the Department of Computing, the Hong Kong Polytechnic University, Hong Kong 999077, China. E-mail: xiaojie.kara.wang@ieee.org.*
- *Xiping Hu and Bin Hu are with the School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China. E-mail: {huxp, bh}@lzu.edu.cn.*
- *Jiangchuan Liu is with the School of Computing Science, Simon Fraser University, British Columbia V5A 1S6, Canada. E-mail: jcliu@cs.sfu.ca.*
- *Lei Guo is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China. E-mail: guolei@cqupt.edu.cn.*
- *Ricky Y. K. Kwok is with the Department of Electrical and Electronic Engineering, University of Hong Kong, Hong Kong 999077, China. E-mail: Ricky.Kwok@hku.hk.*
- *Victor C. M. Leung is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 51806, China, and also with the Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, British Columbia V6T 1Z4, Canada. E-mail: vleung@ieee.org.*

Manuscript received 22 Jan. 2020; revised 18 July 2020; accepted 14 Sept. 2020. Date of publication 18 Sept. 2020; date of current version 4 Mar. 2022. (Corresponding authors: Xiaojie Wang, Xiping Hu, Lei Guo, and Bin Hu.) Digital Object Identifier no. 10.1109/TMC.2020.3025116

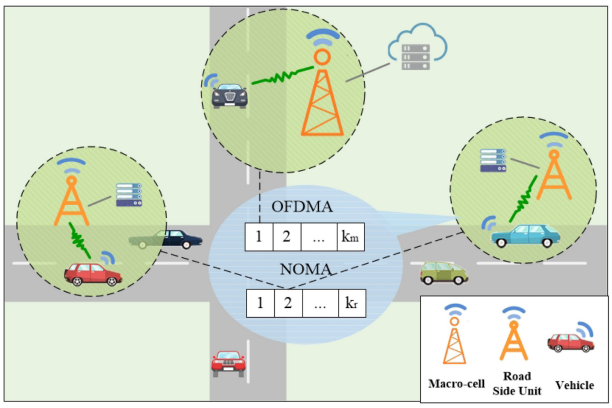


Fig. 1. An illustration of 5G-enabled vehicular networks.

- Partial offloading allows vehicles to decide the offloading ratio by paying for offloading services. However, the offloading ratio is related to many factors, including the characteristics of offloading tasks, communication channels, payoff for offloading and peer decisions. It is difficult to obtain the optimal offloading ratio directly.
- Vehicular users are rational and selfish in the real world. They always decide the offloading ratio by measuring their profits and costs, which may violate the scheduling policy made by the network operator. Therefore, it is essential to design an appropriate payoff strategy to stimulate vehicular users to obey the rules made by the network operator.

In this paper, we construct a partial computation offloading framework for 5G-enabled vehicular networks, where vehicles can process tasks locally (local computing) or offload their tasks to RSUs that are equipped with MEC servers through Non-Orthogonal Multiple Access (NOMA). An illustrative network architecture is presented in Fig. 1. Task scheduling includes three parts, i.e., channel resources allocation, offloading ratio decision and offloading payoff policy for vehicles. Jointly considering the profit of both vehicles and the network operator, the optimization problem is formulated to maximize the system-wide profit, including the profits of both vehicles and the network operator that provides offloading services. Due to the coupling of decision variables, the original optimization problem is divided into three subproblems. First, to allocate channel resources, a two-sided matching algorithm is developed by utilizing the defined preference mapping and list. The objective of the first subproblem is equivalent to minimizing the transmission delay cost. Then, the second subproblem is relaxed to a concave optimization problem. The offloading ratio can be derived through KKT conditions. Finally, considering the incentive compatibility and individual rationality of vehicles, a non-cooperative game is formulated to obtain the offloading payoff policy adaptively, and we prove that the Nash Equilibrium (NE) between the profits of vehicles and those of the network operator can be reached. The main contributions of this paper are summarized as follows:

- We construct a partial offloading and adaptive task scheduling framework for 5G-enabled vehicular networks. By considering the incentive compatibility and individual rationality of vehicles, the optimization problem is formulated to maximize the system-wide profit (including the profits of both vehicles and the network operator). The profit of vehicles can also be guaranteed through payoff policy.
- To solve the formulated optimization problem, it is divided into three subproblems by decoupling the decision variables. With the objective of minimizing the average delay cost, a two-sided matching algorithm is first developed to derive the optimal channel resources allocation strategy. The optimal offloading ratio of the second subproblem is obtained through convex optimization.
- To prevent vehicles from deviating the task scheduling, a non-cooperative game is constructed to decide the payoff under the constraints of incentive compatibility and individual rationality of vehicles. Integrated with game theory and convex optimization, the payoff policy is designed to reach the equilibrium between vehicles and the network operator, based on which the offloading ratio at the NE of the non-cooperative game is also proved to be the optimal offloading ratio of the second subproblem.
- Performance evaluations based on the real-world traces of taxies in Hangzhou (China) demonstrate the effectiveness of our solution from the aspects of system-wide profits and the profits of vehicles.

The rest of paper is organized as follows. Related work is reviewed in Section 2. Section 3 illustrates the system model and formulates the optimization problem. Section 4 proposes a partial computation offloading and adaptive task scheduling algorithm. Performance evaluations are illustrated in Section 5, followed by the conclusion in Section 6.

2 Related Work

Recent efforts have been made to study the MEC-enabled computation offloading. An offloading architecture is constructed in [6] to minimize the energy cost of computation with the assistance of MEC. A computation offloading framework is proposed in [7], where a single user can offload tasks to multiple servers. The optimization problem is formulated to minimize the total cost, including both the energy consumption of users and the total execution latency of tasks. The dynamic resource and task allocation problem is studied in [8], which regulates the local CPU and cloud resources to minimize network total energy cost. The authors in [9] jointly consider computation offloading, resource allocation and content caching in heterogeneous wireless networks, and propose a dynamic method to maximize the total network revenue. An adaptive online scheduling algorithm is presented in [10] to improve the energy efficiency of mobile devices. Its objective is to minimize the total energy cost of multiple mobile applications while satisfying users' performance expectation. The authors in [11] integrate cloud radio network and MEC to investigate dynamic resource scheduling. In order to maximize the profit of service provider, a unified system is designed to make a trade-off between power consumption and performance. The stochastic resource scheduling problem is solved by an improved Lyapunov algorithm. Resource scheduling with partial information is investigated in [12]. A perturbed Lyapunov technique is developed to maximize the network utility by relaxing the formulated knapsack problem to adapt out-of-date network knowledge.

In order to maximize network throughput, a two-sided matching algorithm is proposed to optimize channel assignment and power allocation [13]. The authors in [14] focus on the cooperation of MEC and cloud computing with NOMA. The resource allocation problem is formulated to minimize the total energy consumption and the delay of users. In addition, a few researches have leveraged game theory in computation offloading. A decentralized offloading game is formulated in [15], where users can decide whether to fully offload tasks to the cloud server or not, with the objective of minimizing task execution overhead. The game admits NEs for both homogenous and heterogenous wireless access networks. The authors in [16] formulate the multi-user computation offloading problem as a game. They derive the upper bound of the convergence time of the game theoretic strategy. Centralized computation offloading and transmission scheduling are investigated in [17] for delay-sensitive applications. A queueing game model is formulated to manage system dynamically. The authors in [18] study decentralized offloading in a dense wireless network. Mobile users are considered to be selfish and independent. With the objective of minimizing the computation cost, a game theory based algorithm is developed to compute the equilibrium. The upper bound of the price of anarchy caused by decentralization is theoretically derived. Offloading in heterogeneous cellular networks is researched in [19]. Markov approximation approach and log-linear learning are utilized to mitigate interference, associate users and allocate resources. The designed method can converge to the near optimal solution without complete global information. Based on game theory, the log-linear learning can reach a mixed-strategy Nash equilibrium.

Different from above researches, our work jointly considers channel resource allocation, offloading ratio decision and payoff policy determination. A two-sided matching algorithm is proposed to allocate channels with NOMA. The convex optimization is leveraged to obtain the offloading ratio. A non-cooperative game is formulated to design the payoff policy to maximize system-wide profits while considering the incentive compatibility and individual rationality of vehicles.

3 System Model

In this section, 5G-enabled the vehicular network model is first described. Then, offloading strategies and profits of both vehicles and the network operator are illustrated. Finally, the optimization problem is formulated.

3.1 Vehicular Network Model

We consider a 5G-enabled vehicular network as shown in Fig. 1, where a macro-cell station and a set of RSUs $\mathcal{M} = \{1, 2, \ldots, M\}$ equipped with MEC servers are deployed to perform task scheduling and provide offloading services for a set of vehicles $\mathcal{N} = \{1, 2, \ldots, N\}$, respectively. Vehicles access RSUs through K homogeneous unlicensed channels by utilizing NOMA technology. In order to capture the mobilities of vehicles, offloading service is operated in a discrete time slot structure, denoted by $t \in \mathcal{T} = \{1, 2, \ldots, T\}$. In each time slot, with the objective of maximizing their own profits, vehicles decide offloading ratios based on the designed payoff policy. All task offloading requests are sent to the macro-cell station, which allocates channel resources to vehicles. Since the data size of the offloading request signal is much less than that of the task, task scheduling delay can be ignored. Let λ_i represent the average generate rate of task flow, which is considered to follow a generally distributed random process similar to [17]. For vehicle i, offloading task l to be fulfilled is defined as $\tau_{i,l} = \left\{ s_{i,l}, c_{i,l}, T_{i,l}^{max}, i \in \mathcal{N}, 1 \leq l \leq \lambda_i \right\}$, where $s_{i,l}$ denotes task size, $c_{i,l}$ is the required number of CPU cycles to accomplish task l, and $T_{i,l}^{max}$ represents the tolerable latency for task $\tau_{i,l}$. For each MEC server, let us denote the available channels by $\mathcal{K} = \{1, 2, \ldots, K\}$, where K subchannels are identical.

3.2 Offloading Strategy

Upon generating a computation task, the vehicle decides about either processing it by local computing or offloading part of it to RSUs. In order to maximize the system-wide profit while guaranteeing the profit of all individuals, the macro-cell determines a long-term system-wide profit maximization scheme for partial computation offloading and adaptive task scheduling.

Denote $\Xi = \{\xi_1, \xi_2, \ldots, \xi_N\}$ as the offloading strategy of all vehicles, where $\xi_i \in [0, 1]$ indicates offloading ratio. Then, the local computing and MEC flows for vehicle i can be presented by $\lambda^{\text{local}} = (1 - \xi_i)\lambda_i$ and $\lambda^{\text{MEC}} = \xi_i \lambda_i$, respectively. We mainly focus on the uplink transmission delay and ignore other potential latency (e.g., packet pre-processing and queueing delay). The channel condition is assumed to be stable during the uplink transmission of the offloading task. The channel gain is modeled by the free space propagation path-loss model with Rayleigh fading [20]. Denote p_0 and p as the recieved signal power at d_0 and d away from the transmitter, respectively. Variable α is the exponent of the path loss, and complex Guassian variable h_j^0 represents the Rayleigh fading. Then, signal power $p = p_0(d/d_0)^{-\alpha}|h_j^0|^2$ holds. For simplicity, we set $d_0 = 1$. Channel gain $h_{i,j}$ from vehicle i to server j ($j \in \{0\} \cup \mathcal{M}$, where $j = 0$ denotes local computing) can be expressed as [21]:

$$|h_{i,j}|^2 = G|d_{i,j} + v_{i,j}t_w|^{-\alpha}|h_j^0|^2, \tag{1}$$

where G is the fixed power gain influenced by amplifier and antenna; $d_{i,j}$ denotes the distance between vehicle i and server j; $v_{i,j}$ illustrates the relative velocity of vehicle i to server j; and t_w represents the queuing latency. Let $p_{i,j}$ and $x_{i,j}$ denote the transmission power and original signal of vehicle i, respectively. Since multiple vehicles share one cellular channel, they suffer from interferences. The received signal from vehicle i to server j can be computed by:

$$y_{i,j} = \sqrt{p_{i,j}} h_{i,j} x_{i,j} + \sum_{n \neq i, n \in \mathcal{N}} \sqrt{p_{n,j}} h_{n,j} x_{n,j} + \sigma, \tag{2}$$

where σ is the additive white Gaussian noise. Let us denote the transmission schedule by a binary set $\Theta = \{\theta_{i,j} | i \in \mathcal{N}, j \in \mathcal{M}\}$, where $\theta_{i,j} = 1$ indicates that vehicle i accesses to MEC server j. We utilize Signal to Interference plus Noise Ratio (SINR) to measure the interferences among vehicles, which can be calculated by:

$$\Gamma_{i,j} = \frac{p_{i,j} h_{i,j}^2 \theta_{i,j}}{\sum_{n \in N, n \neq i} \theta_{n,j} p_{n,j} h_{n,j}^2 + \sigma^2}. \tag{3}$$

The bandwidth of subchannels is represented by B. Variable σ^2 is the additive white Gaussian noise. The uplink transmission rate of vehicle i to server j can be expressed as:

$$r_{i,j} = \theta_{i,j} B \log_2 \left(1 + \Gamma_{i,j}\right). \tag{4}$$

3.3 Profit Function

Our objective is to maximize system-wide profit, including the profits of the network operator and vehicles. Generally, vehicles are rational and potentially selfish, which may violate the optimal offloading strategies regulated by the network operator [22]. Therefore, the profits of both the offloading system and all individuals need to be jointly taken into consideration. Given the task flow and strategies of offloading ratio as well as channel resource allocation, the system-wide profit is defined as:

$$\mathcal{U} = \sum_{i=1}^{N} U_i + \sum_{j=1}^{M} U_j^{\mathrm{MEC}}, \tag{5}$$

where $\sum_{i=1}^{N} U_i$ and $\sum_{j=1}^{M} U_j^{\mathrm{MEC}}$ denote the profits gained by vehicles and the network operator, respectively. The profit of vehicle i is defined as:

$$U_i = \sum_{l=1}^{\lambda_i} w_{i,l} - C_i(\lambda_i, \xi_i, \Theta), \tag{6}$$

where $w_{i,l}$ denotes the reward gained by vehicle i after completing offloading task l. Variable $C_i(\lambda_i, \xi_i, \Theta)$ is the cost of executing all offloading tasks, consisting of uplink transmission cost C_i^{tran}, processing energy cost by local computing C_i^{local}, and that by MEC C_i^{MEC}. Following the similar definitions in [9] and [7], we elaborate these cost functions in detail as follows.

For each vehicle $i, \forall i \in \mathcal{N}$, the generated computation task can be executed by the cooperation of local computing and MEC. The energy cost of local computing for task computation can be computed by:

$$C_i^{\mathrm{local}} = (1 - \xi_i) \sum_{l=1}^{\lambda_i} c_{i,l} \varphi_i, \tag{7}$$

where φ_i denotes the energy consumption per CPU cycle for local computing, and it can be obtained by the existing approaches in [23].

In contrast, there is no energy cost of task processing for vehicles when it is offloaded to RSUs. However, vehicles are charged by the network operator for offloading services. The average payoff for vehicle i through MEC can be calculated by:

$$C_i^{\mathrm{pro}} = \xi_i \pi_i^{\mathrm{pro}} \sum_{l=1}^{\lambda_i} c_{i,l}, \tag{8}$$

where π_i^{pro} denotes the average payoff per CPU cycle.

When vehicles offload tasks to RSUs, the transmission cost can be evaluated by energy consumption and channel resources occupation, denoted by:

$$C_i^{\mathrm{tran}} = C_i^{\mathrm{e}} + C_i^{\mathrm{c}}, \tag{9}$$

where transmission energy consumption C_i^{e} depends on the transmission power and delay. Let $s_i / r_{i,j}$ be the transmission time of vehicle i, and C_i^{e} can be calculated by:

$$C_i^{\mathrm{e}} = \xi_i p_{i,j} \sum_{l=1}^{\lambda_i} \frac{s_{i,l}}{r_{i,j}}. \tag{10}$$

Denote the payoff per bit for task transmission by π^{tran}, the channel occupation charge C_i^{c} can be computed by:

$$C_i^{\mathrm{c}} = \xi_i \pi_i^{\mathrm{tran}} \sum_{l=1}^{\lambda_i} s_{i,l}. \tag{11}$$

By substituting equations (7)-(11) to (6), the profit of vehicle i can be expressed by:

$$\begin{aligned} U_i = & \sum_{l=1}^{\lambda_i} w_{i,l} - (1 - \xi_i) \sum_{l=1}^{\lambda_i} c_{i,l} \varphi_i \\ & - \xi_i \sum_{l=1}^{\lambda_i} \left(c_{i,l} \pi_i^{\mathrm{pro}} + p_{i,j} \frac{s_{i,l}}{r_{i,j}} + s_{i,l} \pi_i^{\mathrm{tran}} \right), \end{aligned} \tag{12}$$

The profit function of the network operator that provides offloading services by deploying MEC servers can be specified as:

$$U_j^{\mathrm{MEC}} = \sum_{i=1}^{N} \sum_{l=1}^{\lambda_i} \theta_{i,j} \xi_i (c_{i,l} \pi^{\mathrm{pro}} + s_{i,l} \pi_i^{\mathrm{tran}}) - \mathcal{C}_j^{\mathrm{MEC}}(\xi_i \lambda_i), \tag{13}$$

where $\mathcal{C}_j^{\mathrm{MEC}}(\xi_i \lambda_i)$ denotes the network operating cost (i.e., computation energy consumption, equipment maintenance and wireless spectrum management cost) of MEC servers. Following the general assumption, such as in [24], [25], [26], we consider that $\mathcal{C}(\Xi, \lambda)$ is a monotonically increasing convex function in terms of task flow λ.

Denote the computation capabilities of vehicle $i, i \in \mathcal{N}$ and MEC servers by f_i^h and f^e, respectively. MEC servers are assumed to be homogeneous and their computation capabilities are the same. The task execution latency of vehicle i can be represented by:

$$T_i = \max \left\{ (1 - \xi_i) \sum_{l=1}^{\lambda_i} \frac{c_{i,l}}{f_i^h}, \xi_i \sum_{l=1}^{\lambda_i} \left(\frac{s_{i,l}}{r_{i,j}} + \frac{c_{i,l}}{f^e} \right) \right\}. \tag{14}$$

In our system model, due to the complexity of the transmission scheduling problem, we utilize the expected task execution time to approximate the actual latency. Specifically, for task $\tau_{i,l}$, the execution time cannot exceed threshold $T_{i,l}^{max}$, i.e.,

$$\begin{aligned}
&\left(1-\sum_{j=1}^{M}\theta_{i,j}\right)\frac{c_{i,1}}{f_i^h}+\sum_{j=1}^{M}\theta_{i,j}\left(\frac{s_{i,1}}{r_{i,j}}+\frac{c_{i,1}}{f^e}\right)\leq T_{i,1}^{max},\\
&\left(1-\sum_{j=1}^{M}\theta_{i,j}\right)\frac{c_{i,2}}{f_i^h}+\sum_{j=1}^{M}\theta_{i,j}\left(\frac{s_{i,2}}{r_{i,j}}+\frac{c_{i,2}}{f^e}\right)\leq T_{i,2}^{max},\\
&\cdots\\
&\left(1-\sum_{j=1}^{M}\theta_{i,j}\right)\frac{c_{i,\lambda_i}}{f_i^h}+\sum_{j=1}^{M}\theta_{i,j}\left(\frac{s_{i,\lambda_i}}{r_{i,j}}+\frac{c_{i,\lambda_i}}{f^e}\right)\leq T_{i,\lambda_i}^{max}.
\end{aligned}\tag{15}$$

For channel schedule $\Theta=\{\theta_{i,j}|i\in\mathcal{N},j\in\mathcal{M}\}$, there are $N\cdot M$ transmission scheduling variables. However, there are $\lambda_i\cdot N$ constraints as shown in equation (15) for all vehicles. If $\lambda_i>M$, i.e., the number of constraints exceeds that of variables, the transmission scheduling problem becomes unsolvable. Thus, we utilize the expected values of task accomplish reward, data size and required CPU cycles to estimate the actual task execution latency and profit. Let us denote the expected reward of vehicle i by $\mathbb{E}[w_i]$. Variables $\mathbb{E}[s_i]$ and $\mathbb{E}[c_i]$ denote the finite mean of the random data size and the required number of CPU cycles, respectively. The above expected values can be calculated by:

$$\begin{aligned}
\mathbb{E}[w_i]&=\sum_{l=1}^{\lambda_i}\frac{w_{i,l}}{\lambda_i},\\
\mathbb{E}[s_i]&=\sum_{l=1}^{\lambda_i}\frac{s_{i,l}}{\lambda_i},\\
\mathbb{E}[c_i]&=\sum_{l=1}^{\lambda_i}\frac{c_{i,l}}{\lambda_i}.
\end{aligned}\tag{16}$$

Then, the expected task execution latency for vehicle i can be calculated by:

$$T_i=\max\left\{\lambda_i(1-\xi_i)\frac{\mathbb{E}[c_i]}{f_i^h},\xi_i\lambda_i\left(\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right]+\frac{\mathbb{E}[c_i]}{f^e}\right)\right\},\tag{17}$$

where the latency of backhaul transmission is ignored since the data size of the output data is much smaller than that of the upload input data. Similar assumption has been widely made in many existing researches, such as [16], [18], [27]. The utilization of the expected value leads to estimation deviation. Performance evaluations in Section 5 demonstrate the optimal approximation of our proposed methods.

3.4 Problem Formulation

The system-wide profit relies on three parameters: offloading ratio Ξ, channel allocation Θ, and the payoff for offloading service, i.e., $\boldsymbol{\pi}=\{\pi^{\text{tran}},\pi^{\text{pro}}\}$. The channel allocation and payoff policy are determined by the network operator, while vehicles can decide the offloading ratio, denoted by $\widetilde{\Xi}=\{\widetilde{\xi}_1,\widetilde{\xi}_2,\cdots,\widetilde{\xi}_N\}$. Given the offloading request information, the partial computation offloading and adaptive task scheduling problem is formulated as follows:

$$\arg\max_{\Xi,\Theta,\boldsymbol{\pi}}\mathcal{U}=\sum_{i=1}^{N}U_i+\sum_{j=1}^{M}U_j^{\text{MEC}},\tag{18}$$

s.t.

$$0\leqslant\xi_i\leqslant 1,\forall i\in\mathcal{N},\tag{18a}$$

$$\pi_i^{\text{tran}},\pi_i^{\text{pro}}\geqslant 0,\forall i\in\mathcal{N},\tag{18b}$$

$$T_i\leqslant T_i^{max},\forall i\in\mathcal{N},\tag{18c}$$

$$U_i(\xi_i)\geqslant U_i\left(\widetilde{\xi}_i\right),\forall\widetilde{\xi}_i\in[0,1],i\in\mathcal{N},\tag{18d}$$

$$\mathbb{E}[U_i(\xi_i)]\geqslant 0,\forall i\in\mathcal{N},\tag{18e}$$

$$\sum_{j=1}^{M}\theta_{i,j}=1,\forall i\in\mathcal{N},\tag{18f}$$

$$\theta_{i,j}\in\{0,1\},\forall i\in\mathcal{N},j\in\mathcal{M},\tag{18g}$$

where constraints (**18**a) (**18**b) and (**18**g) indicate the ranges of the offloading ratio, payoff policy and channel allocation for offloading services, respectively. Constraint (**18**c) denotes the delay tolerance. Constraint (**18**d) indicates the incentive compatibility of vehicles, where the personal profit of each vehicle needs to be maximized. Constraint (**18**e) indicates the individual rationality of vehicles that their profits need to be non-negative under the task schedule of the network operator. Constraint (**18**f) shows that each vehicle can merely occupy one channel.

The task scheduling strategy for optimization problem (18) can be obtained based on the assumption that all vehicles follow the centralized schedule. However, in realistic scenarios, vehicles intend to pursue personal profits by determining the offloading ratio on their own, which may deviate from the task schedule made by the network operator. Therefore, the optimal solution needs to guarantee the profits of vehicles and stimulate all vehicles to follow the scheduling made by the network operator.

The optimization problem in (18) is a mixed integer nonlinear programming problem, which is NP-hard. Note that the payoff item is excluded from the system-wide profit, and only included in constraints (**18**e) and (**18**f). The transmission latency in constraint (**18**d) mainly depends on channel allocation Θ. The above observations motivate us to divide the formulated problem into three subproblems to derive the optimal schedule.

4 Partial Computation Offloading and Adaptive Task Scheduling

In this section, an efficient Partial computation Offloading and adaptivE Task Scheduling algorithm (POETS) for 5G-enabled vehicular networks is proposed, by decoupling the optimization problem in (18) into three subproblems. We first develop an asymptotically optimal channel allocation discipline to minimize the task transmission delay of all vehicles with a given offloading ratio. Then, a convex optimization problem is formulated to derive the optimal offloading ratio of the second subproblem that can maximize the system-wide profit. Finally, the payoff policy for offloading services is determined to achieve the equilibrium of the constructed non-cooperative game.

4.1 Transmission Scheduling Discipline

After vehicles decide their offloading ratios, denoted by $\widetilde{\Xi}=\{\widetilde{\xi}_1,\widetilde{\xi}_2,\ldots,\widetilde{\xi}_N\}$, we can specify the system-wide profit as follows:

$$\begin{aligned}\mathcal{U} &= \sum_{i=1}^{N}(\lambda_i\mathbb{E}[w_i] - (1-\widetilde{\xi}_i)\lambda_i\mathbb{E}[c_i]\varphi_i) \\ &\quad - \sum_{j=1}^{M}\sum_{i=1}^{N}\widetilde{\xi}_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] - \sum_{j=1}^{M}\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i) \\ &= \sum_{i=1}^{N}\mathcal{V}(\widetilde{\xi}_i) - \sum_{j=1}^{M}\sum_{i=1}^{N}\widetilde{\xi}_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] - \sum_{j=1}^{M}\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i),\end{aligned} \tag{19}$$

where $\mathcal{V}(\widetilde{\xi}_i) = \lambda_i\mathbb{E}[w_i] - (1-\widetilde{\xi}_i)\lambda_i\mathbb{E}[c_i]\varphi_i$. It can be observed that $\mathcal{V}(\widetilde{\xi}_i)$ and $\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i)$ only depend on the offloading ratio. In addition, the payoff of vehicles and the earnings of the network operator cancel each other out. Thus, the payoff policy does not affect the system-wide profit. In this situation, the channel allocation discipline can be derived with a determined offloading ratio $\widetilde{\Xi}$. The first subproblem can be presented by:

$$\begin{aligned}\mathbf{\Theta} &= \arg\max \mathcal{U} \\ &= \arg\max \sum_{i=1}^{N}\mathcal{V}(\widetilde{\xi}_i) - \sum_{j=1}^{M}\sum_{i=1}^{N}\widetilde{\xi}_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] \\ &\quad - \sum_{j=1}^{M}\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i) \\ &\Leftrightarrow \arg\min \sum_{j=1}^{M}\sum_{i=1}^{N}\widetilde{\xi}_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right],\end{aligned} \tag{20}$$

s.t.

$$T_i \leqslant T_i^{max}, \forall i \in \mathcal{N}, \tag{20a}$$

$$\sum_{j=1}^{M}\theta_{i,j} = 1, \forall i \in \mathcal{N}, \tag{20b}$$

$$\theta_{i,j} \in \{0,1\}, \forall i \in \mathcal{N}, j \in \mathcal{M}. \tag{20c}$$

We illustrate the complication of the transmission scheduling from three aspects. First, vehicles that occupy the same channel suffer from interferences, influencing their own delay cost. It means the transmission scheduling of all vehicles is inter-dependent. Second, simply allocating the resources of MEC servers to the vehicles that send offloading requests may cause the waste of resources and unbalanced load. Thus, the resources of MEC servers need to be carefully utilized to maximize the system-wide profit. Third, the formulated transmission scheduling problem in (20) is a mixed-integer linear programing problem, which is NP-hard to solve.

Given the offloading ratio, the system-wide profit can be maximized by minimizing the expected transmission energy cost. Since constraint (**20**a) only depends on channel resource allocation, the problem in (20) can be relaxed by the Lagrangian multiplier method. Define $\mathcal{L}_i(T_i - T_i^{max})$ as the Lagrangian function with respect to the time difference between the actual task execution delay and the corresponding constraint. Intuitively, $\mathcal{L}_i(\cdot)$ can be interpreted as the dissatisfaction level of the experienced delay. Different from traditional homogeneous linear delay cost functions in [28], $\mathcal{L}_i(\cdot)$ is a nonlinear and heterogeneous function. By substituting $\mathcal{L}_i(\cdot)$ into (20), the total delay cost can be represented by:

$$\mathcal{U}^{\mathrm{delay}} = \sum_{j=1}^{M}\sum_{i=1}^{N}\widetilde{\xi}_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \sum_{i=1}^{N}\mathcal{L}_i(T_i - T_i^{max}). \tag{21}$$

Since vehicles cannot decide spectrum resource scheduling, we devise the channel allocation discipline by a two-sided many-to-many matching algorithm. The allocation discipline $\mathbf{\Theta}$ can be viewed as a mapping from the set of vehicles $\mathcal{N}$ to the set of channels $\mathcal{K}$.

Definition 1 (Preference mapping). *Given two different mappings $\mathbf{\Theta}$ and $\mathbf{\Theta}'$, we prefer $\mathbf{\Theta}'$ to $\mathbf{\Theta}$ (denoted as $\mathbf{\Theta}' \succ \mathbf{\Theta}$) if and only if the delay cost can be decreased, i.e., $\mathcal{U}^{\mathrm{delay}}(\mathbf{\Theta}') < \mathcal{U}^{\mathrm{delay}}(\mathbf{\Theta})$.*

The preference mapping consists of a series of preference lists $\varsigma_k, \forall k \in \mathcal{K}$, i.e.,

Definition 2 (Preference list). *The macro-cell station maintains a preference list for channel k to record the vehicles that have completed the matching process. The preference list needs to satisfy the following constraints:*

$$\begin{aligned}&\text{a)}\ \varsigma_k \in \mathcal{N}, \forall k \in \mathcal{K}, \quad \text{b)}\ |\varsigma_k| \leqslant Q, \forall k \in \mathcal{K}, \\ &\text{c)}\ \varsigma_k \cap \varsigma_{k'} = \varnothing, \forall k \neq k', \quad \text{d)}\ \sum_{k\in\mathcal{K}}\varsigma_k = \mathcal{N},\end{aligned} \tag{22}$$

where Q denotes the maximum number of vehicles that can occupy the same cellular channel. Constraint b) in (22) indicates the number of vehicles sharing the same channel cannot exceed the threshold. Constraint c) denotes that each vehicle can merely occupy one channel.

The developed matching model is more complicated than conventional two-sided matching models [29] from two main aspects. First, the delay cost of one vehicle is not only determined by itself. The interferences among vehicles influence the profits of the vehicles that occupy the same channel and there are a huge number of potential matching combinations, so that all participants need to cooperate to derive the optimal scheduling. Second, traditional methods are likely to be unstable, failing to obtain the optimal solution. Based on Gale-Shapley algorithm, the proposed Two-sided Matching algorithm based Transmission Scheduling discipline (TMTS) is illustrated in Algorithm 1. Delay cost queues are managed in an ascending order, i.e., the first channel in the queue is preferred. Offloading requests through channel assignment is scheduled by judging whether the assignment can lead to the decrease of the total delay cost (i.e., $\mathbf{\Theta}' \succ \mathbf{\Theta}$).

Theorem 1. *Our TMTS algorithm converges to an optimal transmission scheduling $\mathbf{\Theta}^*$ within a limited number of iterations. The upper bound of its time complexity is $O(K(N+K))$, where K is the total number of available channels and N is the total number of vehicles.*

Proof. In each iteration, every vehicle transmits the offloading request through its most-preferred channel recorded in the cost queue. No matter the request is accepted or not, the vehicle removes this channel from its delay cost queue, and will not send the request to it any more. Thus, the maximum number of requests to be processed for each vehicle is the total number of available channels, denoted by K. The matching algorithm will converge within K iterations.

The upper bound of the number of vehicles that transmitting requests is the total number of vehicles (N), and K channels receive requests at most in each iteration. In practice, several vehicles can be matched up with the corresponding channels in each iteration. Besides, it is possible that some channels do not receive the request in several iterations. Thus, the number of inner loops is less than the sum of N and K. The upper bound of the time complexity of the developed TMTS is $O(K(N+K))$. □

Algorithm 1. Pseudo-code of TMTS

Initialize the preference mapping.
Initialize the preference list $\varsigma_k = \varnothing$.
for vehicle i **do**
  Compute delay cost if it occupies channel k.
end
Construct delay cost queues q_i in an ascending order.
offloading-indicator = 1, offloading-request = 0.
while offloading-indicator or offloading-request **do**
  offloading-indicator = 0.
  for vehicle $i \in \mathcal{N}$ **do**
    Propose the offloading request to $q_i[1]$.
    Remove $q_i[1]$ from q_i.
    offloading-request = 1.
  end
  for channel k has receieved any requirement **do**
    if $\varsigma_k \leqslant Q$ **then**
      Accept vehicle i' with the highest cost priority.
      Reject other vehicles.
    end
    else
      if $\mathcal{U}^{delay}((\varsigma_k/\{i'\}) \cup \{i\}) < \mathcal{U}^{delay}(\varsigma_k)$ **then**
        Accept vehicle i.
        Reject other vehicles.
      end
      else
        Reject all requests.
      end
    end
  end
  Until no more vehicles send offloading requests.
end

4.2 Optimal Offloading Ratio

With the optimal channel allocation discipline, the offloading ratio can be determined by solving the following system-wide profit maximization problem:

$$\begin{aligned}\Xi = \arg\max & \sum_{i=1}^{N}\mathcal{V}(\xi_i) - \sum_{j=1}^{M}\sum_{i=1}^{N}\xi_i\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] \\ & - \sum_{i=1}^{N}\mathcal{L}_i^{\Theta}(T_i - T_i^{max}) - \sum_{j=1}^{M}\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i) \\ = & \sum_{i=1}^{N}\mathcal{V}(\xi_i) - \mathcal{U}_{\Xi}^{\mathrm{delay}}(\Theta) - \sum_{j=1}^{M}\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i),\end{aligned} \tag{23}$$

s.t.

$$0 \leqslant \xi_i \leqslant 1, \forall i \in \mathcal{N}, \tag{23a}$$

where $\mathcal{U}_{\Xi}^{\mathrm{delay}}(\Theta)$ denotes the minimum delay cost derived by the channel allocation discipline in Section 4.1, given any offloading ratio Ξ. In the following, we first illustrate the existence of a unique optimal solution for the above optimization problem. Then, KKT conditions are utilized to obtain the optimal solution.

Theorem 2. *There always exists a unique optimal offloading ratio for problem (23) to maximize the system-wide profit.*

Proof. The system-wide profit in (23) consists of three parts: i) $\mathcal{V}(\xi_i)$ is a monotonically increasing linear function with respect to $\xi_i, \forall i \in \mathcal{N}$, as defined in equation (19); ii) $\mathcal{U}_{\Xi}^{\mathrm{delay}}(\Theta)$ is defined as in (21), where $\xi_i\lambda_i p_{i,j}\mathbb{E}[s_i/r_{i,j}]$ and $\mathcal{L}_i(T_i - T_i^{max})$ are increasing linear and convex functions with respect to ξ_i; iii) $\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i)$ is also convex with respect to $\xi_i, \forall i \in \mathcal{N}$. In summary, the system-wide profit maximization problem in (23) is a concave maximization problem, where the feasible set is non-empty and compact. Therefore, the unique optimal offloading ratio for the subproblem exists. □

Since the second subproblem formulated in (23) is a convex optimization problem, KKT conditions can be utilized to derive its optimal offloading ratio.

Theorem 3. *Denote $\Xi^* = \{\xi_1^*, \xi_2^*, \ldots, \xi_N^*\}$ ($\xi_i^* \in [0,1]$) as the optimal solution for maximizing the system-wide profit, then Ξ^* should satisfy:*

$$\begin{aligned}\frac{\partial\mathcal{V}_i(\xi_i)}{\partial\xi_i} = & \sum_{j=1}^{M}\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] \\ & + \sum_{j=1}^{M}\sum_{z=1}^{N}\left(\xi_z^*\lambda_z p_{z,j}\frac{\partial\mathbb{E}\left[\frac{s_z}{r_{z,j,k}}\right]}{\partial\xi_i} + \frac{\partial\mathcal{L}_z^{\Theta}}{\partial\xi_i}\right) \\ & + \sum_{j=1}^{M}\frac{\partial\mathcal{C}_j^{\mathrm{MEC}}(\xi_i\lambda_i)}{\partial\xi_i} + \omega_i - \nu_i, \forall i \in \mathcal{N},\end{aligned} \tag{24}$$

where coefficients ω_i and ν_i are non-negative Lagrange multipliers, satisfying $\omega_i\xi_i^* = 0$ and $\nu_i(\xi_i^* - 1) = 0$, respectively.

Proof. Theorem 2 proves that the subproblem formulated in (23) is a concave maximization problem. Therefore, KKT conditions can be utilized to obtain the optimal solution of the offloading ratio. The partial derivative of equation (23) with respect to the offloading ratio of vehicle i results in equation (24). □

In the presented KKT conditions, all variables are explicit except for Lagrangian function $\mathcal{L}_z^{\Theta}$, since its parameter is computed by a max function shown in equation (17). Thus, the actual task execution latency requires to be clarified explicitly to make KKT conditions solvable. Based on the equation of the task execution latency and the corresponding delay constraint, the actual latency can be represented by the following threshold of the offloading ratio.

Theorem 4. *For vehicle $i, i \in \mathcal{N}$, the task execution delay can be represented by the following equation:*

$$T_i = \begin{cases}\lambda_i(1-\xi_i)\frac{\mathbb{E}[c_i]}{f_i^l}, 0 \leqslant \xi_i < \dot{\xi}_i, \\ \xi_i\lambda_i\left(\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \frac{\mathbb{E}[c_i]}{f^e}\right), \dot{\xi}_i \leqslant \xi_i \leqslant 1,.\end{cases} \tag{25}$$

where the offloading ratio threshold $\dot{\xi}_i = \frac{\frac{\mathbb{E}[c_i]}{f_i^l}}{\frac{\mathbb{E}[c_i]}{f_i^l} + \mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \frac{\mathbb{E}[c_i]}{f^e}}$.

Proof. In practical, local computing and MEC are performed simultaneously. The task execution latency equals the one with larger delay, as presented in equation (17). For vehicle i, when the delay of local computing is larger than that of MEC, the task generation rate satisfies the following inequality:

$$\begin{aligned}&\lambda_i(1-\xi_i)\frac{\mathbb{E}[c_i]}{f_i^l} > \xi_i\lambda_i\left(\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right]+\frac{\mathbb{E}[c_i]}{f^e}\right),\\ &0\leqslant\xi_i < \frac{\frac{\mathbb{E}[c_i]}{f_i^l}}{\frac{\mathbb{E}[c_i]}{f_i^l}+\mathbb{E}[\frac{s_i}{r_{i,j}}]+\frac{\mathbb{E}[c_i]}{f^e}} = \dot{\xi}_i.\end{aligned} \tag{26}$$

Similarly, when $\dot{\xi}_i \leqslant \xi_i \leqslant 1$, the task execution latency equals the delay caused by offloading to MEC servers. □

The procedure of deriving the global optimal offloading ratio that can maximize the system-wide profit is illustrated in Algorithm 2. First, vehicles initialize coefficients, i.e., the expected reward after completing one task and the energy consumption per CPU cycle for local computing. They are uploaded with other offloading information, including the generation rate of the task flow, the capability of local computing as well as the data size, required CPU cycles and execution latency constraint of the task.

Algorithm 2. Procedure of offloading ratio derivation

For each vehicle:
Initialize coefficients $\mathbb{E}[w_i]$ and φ_i.
Upload individual offloading information, including $\mathbb{E}[w_i]$, φ_i, ξ_i, f_i^l and $\tau_i = \{s_i, c_i, T_i^{max}\}$.
For the macro-cell station:
Gather global offloading information.
Allocate channel resources based on **Algorithm 1**.
Compute the task execution latency of each vehicle according to **Theorem 4**.
Derive the offloading ratio based on KKT conditions illustrated in **Theorem 3**.

4.3 Payoff Policy

After deriving the channel resources allocation and offloading ratio by TMTS algorithm and convex optimization, payoff policy $\boldsymbol{\pi}$ should be decided to guarantee the incentive compatibility and individual rationality of vehicles, as illustrated in constraints (**18**e) and (**18**f). In practical, vehicles can determine their offloading ratio strategically and independently, i.e., the strategy may deviate from the optimal solution derived by the network operator. In order to induce all vehicles follow the optimal task scheduling, the profits of vehicles need to be maximized by designing the payoff policy. Considering the task execution latency constraint, the profit of vehicle i in equation (12) is modified by introducing Lagrangian function $\mathcal{L}_i^{\Theta}(T_i - T_i^{max})$. Given strategies of all the other users $\xi_{-i} = \Xi/\xi_i$, function $U_i(\xi_i, \xi_{-i})$ is the profit of vehicle i with respect to strategy ξ_i.

$$\begin{aligned}U_i(\xi_i,\xi_{-i}) &= \mathcal{V}(\xi_i) - \mathcal{L}_i^{\Theta}(T_i - T_i^{max})\\ &\quad - \xi_i\lambda_i(\mathbb{E}[c_i]\pi_i^{\text{pro}} + \sum_{j=1}^{M} p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \mathbb{E}[s_i]\pi_i^{\text{tran}}).\end{aligned} \tag{27}$$

Vehicles compete with each other to maximize individual profit by adjusting their offloading ratios, which can be viewed as a non-cooperative game with a default channel allocation discipline derived in Section 4.1. The game can be formulated as:

$$\mathcal{G} \triangleq \{\mathcal{N}, \Xi, \{U_i(\xi_i, \xi_{-i})\}_{i\in\mathcal{N}}\}, \tag{28}$$

where vehicles in $\mathcal{N}$ act as participants. The strategy set of vehicles is denoted by $\Xi = \{\xi_1, \xi_2, \ldots, \xi_N\}$. The NE of game $\mathcal{G}$ is defined as follows:

Definition 3 (NE). *A strategy profile $\Xi^e = \{\xi_1^e, \xi_2^e, \ldots, \xi_N^e\}$ is an NE of game $\mathcal{G}$, if and only if it satisfies:*

$$U_i(\xi_i^e, \xi_{-i}^e) \geqslant U_i(\xi_i, \xi_{-i}^e), \forall \xi_i \in [0,1], i \in \mathcal{N}. \tag{29}$$

Given the channel allocation discipline, the NE of game $\mathcal{G}$, denoted by ξ^e, mainly depends on payoff $\boldsymbol{\pi} = \{\pi^{\text{pro}}, \pi^{\text{tran}}\}$. Since each vehicle intends to maximize its own profit and may behave selfishly, the optimal solution derived in Section 4.2 can be accepted by all participants only if the optimal offloading ratio of the second subproblem can jointly maximize the system-wide profit and the profits of vehicles, i.e., the payoff policy needs to guarantee that the optimal solution derived by the network operator is also an NE of game $\mathcal{G}$, expressed by:

$$\xi_i^* = \xi_i^e, \forall i \in \mathcal{N}. \tag{30}$$

In the following, we first present the designed optimal payoff policy, with which the NE of game $\mathcal{G}$ is proved to exist. Then, the optimal offloading ratio of the second subproblem is also proved to be an NE of game $\mathcal{G}$, which satisfies the incentive compatibility and individual rationality of vehicles.

Proposition 1 (Optimal payoff policy $\boldsymbol{\pi}^*$). *Given the optimal offloading ratio Ξ^*, the optimal payoff $\boldsymbol{\pi}^* = \{\pi_1^*, \pi_2^*, \ldots, \pi_N^*\}$ can be obtained by:*

$$\begin{aligned}\pi_i^{\text{tran}} &= \frac{1}{\lambda_i\mathbb{E}[s_i]}\cdot\sum_{z=1,z\neq i}^{N}\left(\frac{\partial\mathcal{L}_z^{\Theta}}{\partial\xi_i} + \sum_{j=1}^{M}\xi_z^*\lambda_z p_{z,j}\frac{\partial\mathbb{E}[\frac{s_z}{r_{z,j,k}}]}{\partial\xi_i}\right),\\ \pi_i^{\text{pro}} &= \frac{1}{\lambda_i\mathbb{E}[c_i]}\cdot\sum_{j=1}^{M}\frac{\partial\mathcal{C}^{\text{MEC}}(\xi_i\lambda_i)}{\partial\xi_i},\end{aligned} \tag{31}$$

where π_i^{tran} and π_i^{pro} are independent of ξ_i and can be determined by λ_i and Ξ^*.

Next, we demonstrate that the NE of game $\mathcal{G}$ exists. Then, optimal offloading ratio Ξ^* is proved to satisfy the NE condition.

Theorem 5. *The game $\mathcal{G} \triangleq \{\mathcal{N}, \Xi, \{U_i(\xi_i, \xi_{-i})\}_{i\in\mathcal{N}}\}$ has at least one NE with the designed payoff $\boldsymbol{\pi}^*$.*

Proof. All participants can determine the strategy (i.e., the offloading ratio) by themselves. The strategy set of all vehicles can be denoted as the Cartesian product of each strategy, denoted as $\Xi = \prod_{i=1}^{N}[0,1] \subset \mathbb{R}^N$. It is obviously a non-empty, convex and compact set.

Based on equation (27), we calculate the first-order derivative of the profit of vehicle i as follows:

$$\begin{aligned}\frac{\partial U_i}{\partial\xi_i} &= \frac{\partial\mathcal{V}_i(\xi_i)}{\partial\xi_i} - \lambda_i(\mathbb{E}[c_i]\pi^{\text{pro}} + \mathbb{E}[s_i]\pi^{\text{tran}}) - \frac{\partial\mathcal{L}_i^{\Theta}}{\partial\xi_i}\\ &\quad - \sum_{j=1}^{M}\left(\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \xi_i\lambda_i p_{i,j}\frac{\partial\mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial\xi_i}\right).\end{aligned} \tag{32}$$

As illustrated in Definition 1, the payoff is independent of strategy ξ_i. Function $\mathcal{V}(\xi_i)$ is linear to ξ_i. Thus, the second-order derivative of U_i can be illustrated as:

$$\frac{\partial^2 U_i}{\partial \xi_i^2} = -\sum_{j=1}^{M}\left(\lambda_i p_{i,j}\frac{\partial \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial \xi_i} + \xi_i\lambda_i p_{i,j}\frac{\partial^2 \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial \xi_i^2}\right) - \frac{\partial^2 \mathcal{L}_i^{\Theta}}{\partial \xi_i^2}. \tag{33}$$

Recall that both $\mathbb{E}[s_i/r_{i,j}]$ and $\mathcal{L}_i^{\Theta}$ are monotonically increasing convex functions with respect to ξ_i. Thus, we have:

$$\frac{\partial \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial \xi_i} \geqslant 0, \frac{\partial^2 \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial^2 \xi_i} \geqslant 0, \frac{\partial^2 \mathcal{L}_i^{\Theta}}{\partial^2 \xi_i} \geqslant 0. \tag{34}$$

By substituting (34) into (33), it can be yielded that $\frac{\partial^2 U_i}{\partial \xi_i^2} \leqslant 0$, i.e., U_i is a concave function with respect to ξ_i.

In summary, the available strategy set is non-empty, convex and compact, and the utility function is continuous and concave. Therefore, game $\mathcal{G}$ has at least one NE Ξ^e. □

Theorem 6. *With the designed payoff in (31), the optimal offloading ratio is also an NE of game $\mathcal{G}$, i.e.,*

$$\Xi^* = \Xi^e. \tag{35}$$

Proof. The NE of game $\mathcal{G}$ needs to maximize the individual utilities of all participants, i.e.,

$$\text{s.t.} \quad \xi_i^e = \arg\max U_i, \tag{36}$$

$$0 \leqslant \xi_i^e \leqslant 1, \forall i \in \mathcal{N}. \tag{36a}$$

By substituting the designed payoff in (31) into the individual profit in (27), the KKT conditions that ξ_i^e satisfies can be calculated as follows:

$$\begin{aligned}
\frac{\partial \mathcal{V}_i(\xi_i)}{\partial \xi_i} &= \sum_{j=1}^{M}\left(\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \xi_i^e\lambda_i p_{i,j}\frac{\partial \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial \xi_i}\right) \\
&\quad + \lambda_i(\mathbb{E}[c_i]\pi^{\text{pro}} + \mathbb{E}[s_i]\pi^{\text{tran}}) + \frac{\partial \mathcal{L}_i^{\Theta}}{\partial \xi_i} + \omega_i - \nu_i \\
&= \sum_{j=1}^{M}\left(\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] + \xi_i^e\lambda_i p_{i,j}\frac{\partial \mathbb{E}[\frac{s_i}{r_{i,j}}]}{\partial \xi_i}\right) + \frac{\partial \mathcal{L}_i^{\Theta}}{\partial \xi_i} \\
&\quad + \lambda_i\mathbb{E}[s_i]\frac{1}{\lambda_i\mathbb{E}[s_i]}\cdot\sum_{z=1,z\neq i}^{N}\left(\frac{\partial \mathcal{L}_z^{\Theta}}{\partial \xi_i} + \right. \\
&\quad \left.\sum_{j=1}^{M}\xi_z^e\lambda_z p_{z,j}\frac{\partial \mathbb{E}[\frac{s_z}{r_{z,j,k}}]}{\partial \xi_i}\right) \\
&\quad + \lambda_i\mathbb{E}[c_i]\frac{1}{\lambda_i\mathbb{E}[c_i]}\cdot\sum_{j=1}^{M}\frac{\partial \mathcal{C}^{\text{MEC}}(\xi_i\lambda_i)}{\partial \xi_i} + \omega_i - \nu_i \\
&= \sum_{j=1}^{M}\lambda_i p_{i,j}\mathbb{E}\left[\frac{s_i}{r_{i,j}}\right] \\
&\quad + \sum_{j=1}^{M}\sum_{z=1}^{N}\left(\xi_z^e\lambda_z p_{z,j}\frac{\partial \mathbb{E}[\frac{s_z}{r_{z,j,k}}]}{\partial \xi_i} + \frac{\partial \mathcal{L}_z^{\Theta}}{\partial \xi_i}\right) \\
&\quad + \sum_{j=1}^{M}\frac{\partial \mathcal{C}_j^{\text{MEC}}(\xi_i\lambda_i)}{\partial \xi_i} + \omega_i - \nu_i, \forall i \in \mathcal{N},
\end{aligned} \tag{37}$$

where ω_i and ν_i are lagrange multipliers, holding that $\omega_i\xi_i^e = 0$ and $\nu_i(1-\xi_i^e) = 0$.

It can be observed that the KKT conditions (NE) for vehicle $i, \forall i \in \mathcal{N}$ are precisely equivalent to the KKT conditions obtained in (24), which indicates the optimal solution derived in Section 4.2 is also an NE of game $\mathcal{G}$, i.e., $\Xi^* = \Xi^e$. □

Theorems 5 and 6 illustrate that the NE of game $\mathcal{G}$ exists, and it can be derived by solving the sub-optimization problem in (23). By implementing the proposed POETS algorithm as illustrated in Algorithm 3, vehicles can maximize their profits and satisfy the incentive compatibility. In practice, after receiving offloading requests, the network operator can derive the optimal channel allocation and offloading ratio for all vehicles in advance. Then, the payoff policy can be determined at the same time without extra transmission overheads. Since incentive compatibility and individual rationality can be satisfied, all vehicles can be encouraged to follow the schedule made by the network operator. In the following, we prove that individual rationality can also be satisfied.

Algorithm 3. Procedure of POETS

Initialize task generation rates of all vehicles.
Initialize offloading transmission and computation resources.
Step 1 Derive the offloading ratio of vehicles based on **Algorithm 2**.
Step 2 Determine the payoff policy of offloading according to **Proposition 1**.
Step 3 Vehicles can verify whether the offloading strategy satisfies incentive compatibility and individual rationality according to **Theorem 6**.

Theorem 7 (Individual rationality). *With the implementation of POETS algorithm, the maximum profit of each vehicle $i, \forall i \in \mathcal{N}$, is non-negative, i.e.,*

$$\begin{aligned}
U_i(\xi_i^*) &= \mathcal{V}(\xi_i^*) - \mathcal{L}_i^{\Theta}(T_i - T_i^{max}) \\
&\quad - \xi_i^*\lambda_i(\mathbb{E}[c_i]\pi_i^{pro} + \sum_{j=1}^{M}p_{i,j}\mathbb{E}[\frac{s_i}{r_{i,j}}] + \mathbb{E}[s_i]\pi_i^{tran}) \geqslant 0.
\end{aligned} \tag{38}$$

Proof. Vehicles can obtain profit through processing computation tasks without offloading, i.e.,

$$U_i(\xi_i) \geqslant 0, \xi_i = 0, i \in \forall\mathcal{N}. \tag{39}$$

In addition, it has been proved in Theorem 5 that the utility function of vehicles is concave with respect to ξ_i, and ξ_i^* satisfies the KKT condition, i.e.,

$$U_i(\xi_i^*) = \max_{\xi_i\in[0,1]} U_i(\xi_i). \tag{40}$$

Thus, we have:

$$U_i(\xi_i^*) \geqslant U_i(0) \geqslant 0. \tag{41}$$

□

Since both incentive compatibility and individual rationality can be satisfied, vehicles can accept the task scheduling without deviation. After that, the macro-cell station derives the optimal task scheduling, including channel allocation, offloading ratio and payoff policy. In particular, the

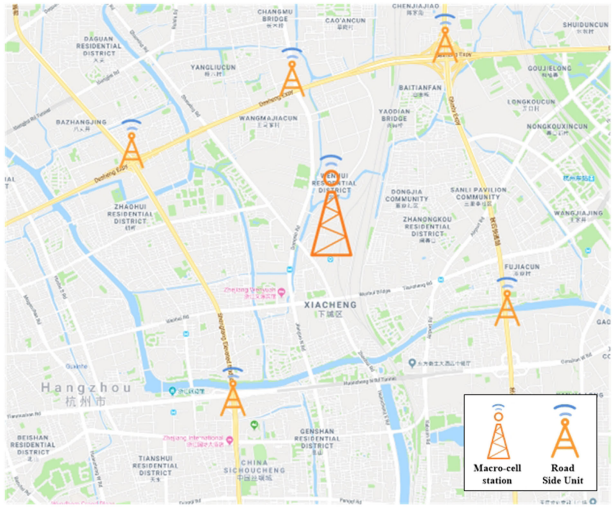


Fig. 2. The map of Xiacheng district in Hangzhou (China).

payoff policy is determined adaptively, where the charge on each vehicle depends on the characteristics of both its own and others' tasks. Traditional fixed charge mechanism cannot guarantee the profits of all vehicles. If the network operator intends to earn profits by falsely raising the payoff of offloading services, vehicles can simply detect the charging gap by verifying whether the payoff policy is the NE of game $\mathcal{G}$ in equation (28). In addition, vehicles do not need to execute iterative algorithms to reach the NE, and the corresponding convergence time can be saved.

5 Performance Evaluation

In this section, we first illustrate the simulation setup. Then, the proposed POETS algorithm is evaluated based on the real-world traces of taxies in Xiacheng District, Hangzhou (China), as shown in Fig. 2.

5.1 Simulation Setup

We utilize Anaconda 4.3 to conduct performance evaluations in a 64bit Windows 10 operating system computer, which has a 16.0 GB RAM and an Intel(R) Core(TM) i7-8700 CPU with 3.20 GHz frequency. Based on the characteristics of cellular network communications [30], the transmission power of each vehicle is set to 100 mW. In addition, the generation rate of offloading tasks for each vehicle is chosen within [2, 5] per minute. Considering the heterogeneity of both users and tasks, the data size is generated randomly from [40, 60] Mb, and the required CPU cycles are chosen from [800, 1200] Megacycles. Inspired by [17], Lagrangian function $\mathcal{L}(\Delta t)$ is defined as $\mathcal{L}(\Delta t) = \epsilon \Delta t^2$, where Δt is the time difference between the actual delay and delay constraint. Coefficient ϵ is the marginal delay sensitivity, which is selected within [1, 6]. Simulation parameters are summarized in Table 1. Performance indicators are as follows:

- Average delay cost: The channel resource allocation discipline is developed to maximize the system-wide profit, which is equivalent to minimize the average delay cost.
- System-wide profit: The objective of the MEC-enabled partial offloading system is to maximize the system-wide profit, including the profit of task execution for vehicles and the profit of providing offloading services for the macro-cell station.
- Number of vehicles benefiting from MEC: Vehicles that select MEC servers to process tasks are regarded as benefiting from MEC. On the premise of maximizing the system-wide profit, MEC servers are expected to provide services for users as many as possible.

TABLE 1
Simulation Parameters Setting

Notations	Description	Value
M	The number of MEC servers	5
$p_{i,j}$	Transmission power of vehicle i	100 mW
f_i^h	CPU frequency of vehicle i	5-10 GHz
f^e	CPU frequency of MEC servers	20 GHz
λ_i	Average task arrival rate of vehicle i	2-5
$s_{i,l}$	Data size of single task	40-60 Mb
$c_{i,l}$	Required CPU cycles of single task	800-1200 M
B	The bandwidth of subchannels	2 M

To demonstrate the effectiveness of our proposed TMTS algorithm, the following methods are leveraged for comparison:

- DTMS [31]: It is a first-come-first-serve based scheduling algorithm, which assigns the same priority to all users without considering their heterogeneous delay requirements and channel conditions.
- IHRA [32]: A branch-and-bound based heuristic method, which intends to minimize the offloading delay under the cooperation of edge computing and cloud computing.

In addition, the proposed POETS algorithm is compared with the following schemes to illustrate the effectiveness with respect to the system-wide profit and the number of vehicles benefiting from MEC:

- MECO [33]: It optimizes resource allocation based on OFDMA by considering the priorities, energy consumption and channel conditions of vehicular users.
- POETS without NOMA access (POETS w.o. NOMA): It is utilized as a benchmark, which merely depends on OFDMA technology.
- Local computing: All users execute their tasks through the macro-cell station.

Finally, we conduct a simplified case study to illustrate that our results can approximate the optimal solution. Since the original problem is NP-hard, we keep the parameters in Table ?? unchanged for the simplified case. The number of vehicles and channels is reduced to 5, i.e., $N = K = 5$. The average task arrival rate varies from 2 to 7.

5.2 Simulation Results

Figs. 3 and 4 illustrate the performance comparison of different transmission scheduling schemes. Performance of the average delay cost with different number of vehicles is shown in Fig. 3 ($K = 10, \epsilon = 2$). We can observe that the average delay cost increases when the number of vehicles becomes large. This is because that a large number of vehicles lead to intense competitions on channels, increasing the interference among vehicles. Thus, low transmission rate results in high delay. Our proposed TMTS discipline performs 20 and

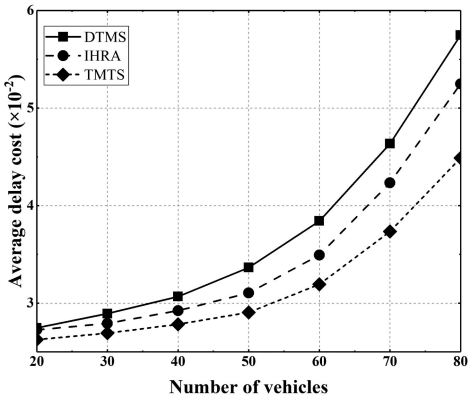


Fig. 3. Average delay cost with different number of vehicles.

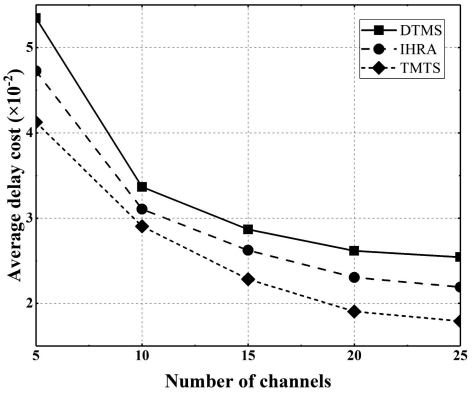


Fig. 4. Average delay cost with different number of channels.

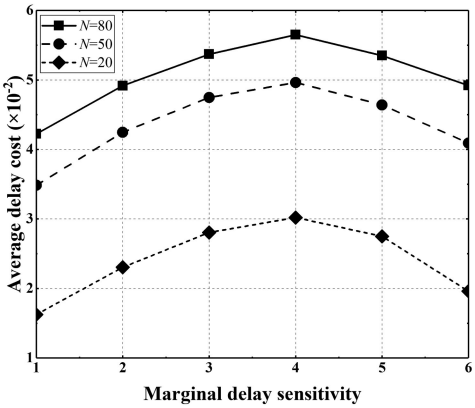


Fig. 5. Average delay cost with different marginal delay sensitivity.

15 percent better than those of DTMS and IHRA, respectively. This is because that the former ignores the potential priority of vehicles, and the latter aims to minimize the total transmission delay, while profits cannot be guaranteed. Our method takes the heterogeneity of vehicles into account, and makes a trade-off between minimizing the total transmission delay and guaranteeing the profits of vehicles.

Performance of average delay cost with different number of channels is shown in Fig. 4 ($N = 50, \epsilon = 2$). It can be observed that the average delay cost first deceases rapidly (e.g., $5 \leqslant K \leqslant 10$), and then decreases slowly ($20 \leqslant K \leqslant 25$) with the increasing number of channels. When channel resources are limited, vehicles suffer from severe interferences, and delay cost is large. In this case, the increasing number of channels can decrease the number of vehicles sharing one channel

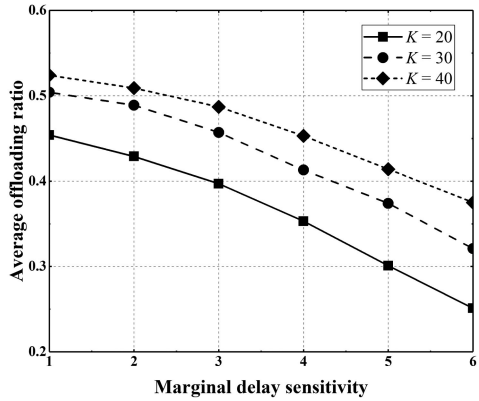


Fig. 6. Average offloading ratio with different marginal delay sensitivity.

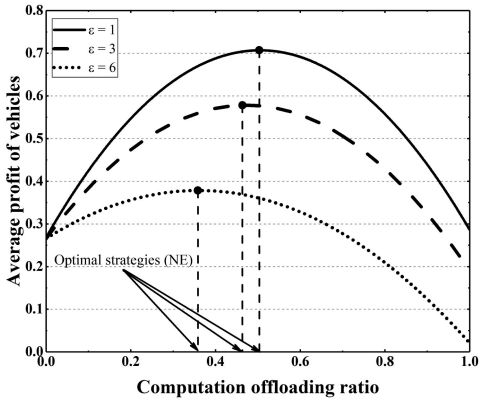


Fig. 7. Optimal offloading strategies of vehicles.

due to incentive compatibility. Correspondingly, the delay cost can be significantly reduced. However, the reduction rate of the average delay cost becomes slow when channel resources are sufficient. This is because most vehicles can upload tasks to MEC servers with relatively small interferences. Our proposed TMTS algorithm can reduce the average delay cost by 16 and 27 percent compared with IHRA and DTMS algorithms, respectively.

The impacts of marginal delay sensitivity on average delay cost and offloading ratio are illustrated in Fig. 5 ($K = 10$) and Fig. 6 ($N = 50$), respectively. We can observe that the average delay cost of vehicles is similar to a quadratic function, which increases first and then decreases with the increasing of marginal delay sensitivity. When marginal delay sensitivity is relatively small, the delay tolerance of vehicles is high, and the corresponding delay cost is low. Vehicles prefer MEC to local computing, and the offloading ratio is large. However, when marginal delay sensitivity becomes large, vehicles are highly sensitive to transmission delay, resulting in the growth of delay cost. In order to guarantee profits, vehicles decrease the offloading ratio, which is illustrated in Fig. 6. In addition, with the rising of marginal delay sensitivity, the profit of invoking MEC cannot compensate for the increasing delay cost. Vehicles tend to choose local computing to decrease delay cost. Thus, the delay cost reduces with the decline of offloading ratio. Furthermore, since vehicles compete with each other to occupy communication channels, less number of channels (e.g., $K = 20$) results in lower offloading ratios.

In Fig. 7, the incentive compatibility of vehicles is examined by the average profit of vehicles varying from different

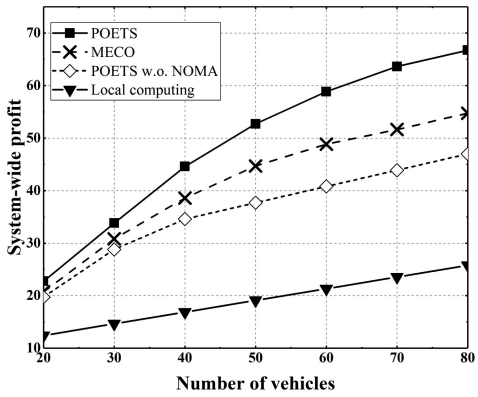


Fig. 8. System-wide profit with different number of vehicles.

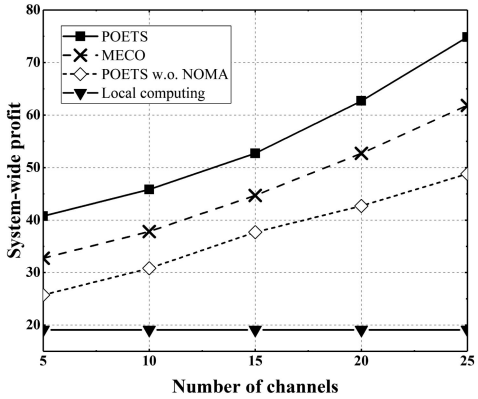


Fig. 9. System-wide profit with different number of channels.

offloading ratios ($N = 50, K = 10$). First, the profits of vehicles increase with the rise of offloading ratio. This is because vehicles can obtain profits by decreasing the consumption of local computing and exploiting the benefit from MEC. When the offloading ratio reaches and exceeds a certain point (i.e., the optimal strategy indicated in Fig. 7), the profits decrease with the growth of offloading ratio. This is because the offloading consumption (including task transmission and computation consumption) becomes dominant, and the benefit gained by vehicles is less than offloading payoff. Obviously, the optimal strategy (NE) is the point that can maximize the profits of vehicles, as well as guarantee incentive compatibility. It is also the optimal solution derived by the macro-cell station as proved in Theorem 6.

Fig. 8 demonstrates the effectiveness of the proposed POETS algorithm in terms of system-wide profit ($K = 10$, $\epsilon = 2$). Since more vehicles can obtain more profits by task processing (including both local computing and MEC), the system-wide profit increases with the rising number of vehicles. Without the assist of MEC, vehicles suffer from excessive energy consumption. Thus, local computing obtains the lowest system-wide profit. Since the channel occupation cost can be reduced by employing NOMA technique, POETS performs better than those of MECO and POETS w.o. NOMA. Specially, when the number of vehicles becomes large, the growth rate of the system-wide profit slows down. This is because channel resources are limited. A large number of vehicles result in the intense competitions on channels, hindering the increase of the system-wide profit. On average, POETS algorithm can increase the system-wide profit by 16

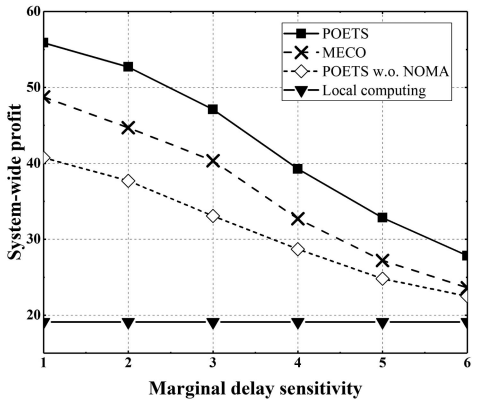


Fig. 10. System-wide profit with different marginal delay sensitivity.

and 37 percent compared with MECO and POETS w.o. NOMA, respectively.

The system-wide profit with different number of channels is illustrated in Fig. 9 ($N = 50, \epsilon = 2$). When channel resources are deficient (e.g., $K = 5$), MEC servers are available to a few vehicles, and most vehicles are forced to choose local computing. Thus, the system-wide profit is low. With the increasing number of channels, the number of vehicles sharing one channel decreases. Correspondingly, the task transmission rate raises, promoting more vehicles to invoke MEC rather than local computing. The system-wide profit grows steadily. Without channel multiplexing, the spectrum efficiency of MECO algorithm is low. In such cases, our proposed POETS algorithm performs 25 percent better than MECO algorithm in terms of the system-wide profit.

Fig. 10 investigates the system-wide profit with different marginal sensitivity of vehicles ($N = 50, K = 10$). It can be observed that the system-wide profit decreases with the increase of marginal delay sensitivity. It is because when marginal delay sensitivity increases, the tolerance of vehicles for interference is becoming lower and lower. A relatively small interference leads to large delay cost, and the system-wide profit decreases. Note that the decline rate of profit is non-linear, i.e., the decline rate becomes faster when marginal delay sensitivity raises from 1 to 4, and then slower when marginal delay sensitivity raises from 4 to 6. The reason is that the number of vehicles selecting MEC to process tasks is large when marginal delay sensitivity is relatively small (i.e., 1-4). With the increase of marginal delay sensitivity, a few vehicles choose local computing to avoid large delay cost. However, local computing consumes large energy, incurring the rapid decline of profit. When marginal delay sensitivity is relatively large (i.e., 4-6), although the growth of marginal delay sensitivity leads to the increase of delay cost, the number of vehicles invoking MEC is small. The decline rate of the system-wide profit slows down gradually.

Figs. 11 and 12 demonstrate the effectiveness of our proposed POETS algorithm in terms of the number of vehicles benefiting from MEC ($N = 50, \epsilon = 2$). For the proposed POETS algorithm, the number of vehicles benefiting from MEC increases with the rise of the number of channels. This is because sufficient channel resources enable a large number of vehicles transmit tasks to MEC servers with satisfactory interference. For MECO and POETS w.o. NOMA methods, the number of vehicles benefiting from MEC

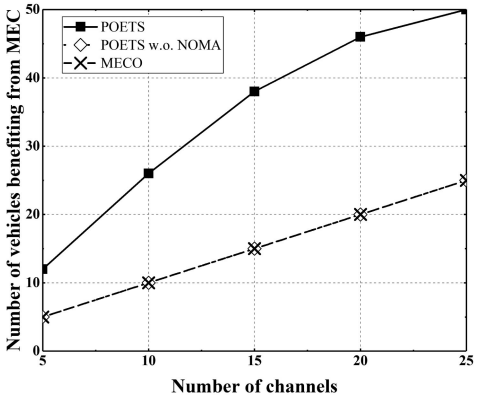


Fig. 11. Number of vehicles benefiting from MEC with different number of channels.

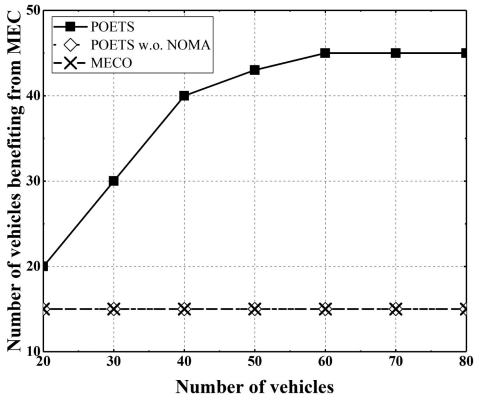


Fig. 12. Number of vehicles benefiting from MEC with different number of vehicles.

increases linearly with the rise of the number of channels. Specifically, the number of vehicles benefiting from MEC equals to the number of channels. It is because MECO and POETS w.o. NOMA methods utilize OFDMA technique to transmit tasks. Each channel can be merely occupied by one vehicle. Hence, the spectrum efficiency is low, and their performances is worse than that of POETS algorithm in terms of the number of vehicles benefiting from MEC.

The trend of the number of vehicles benefiting from MEC based on different number of vehicles is illustrated in Fig. 12. For POETS algorithm, when channel resources are sufficient (i.e., $N \leqslant 40$), it can be observed that the number of vehicles benefiting from MEC increases linearly with the increasing number of vehicles. When vehicles are excessive, the interference incurred by channel multiplexing becomes severe and the delay cost increases rapidly. Thus, these vehicles choose local computing to process tasks. The number of vehicles benefiting from MEC does not increase with the rise of the number of vehicles any more. For MECO and POETS w.o. NOMA methods, since the number of channels is finite and constant, the number of vehicles benefiting from MEC remains the same with the increasing number of vehicles. In summary, our proposed POETS can improve spectrum efficiency, and provide MEC services for more vehicles than that of the compared methods.

Figs. 13 and 14 compare the performance of our proposed TMTS and POETS algorithms with the optimal solution obtained by the brute force searching method, respectively.

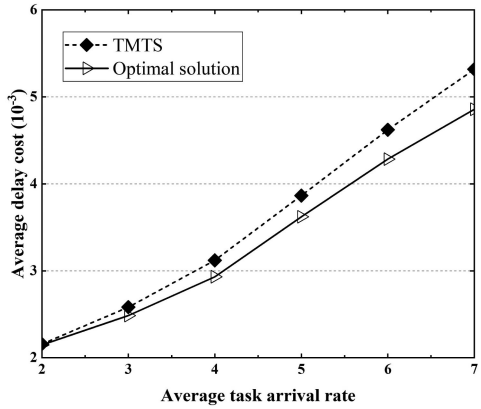


Fig. 13. Average delay cost with different task arrival rate.

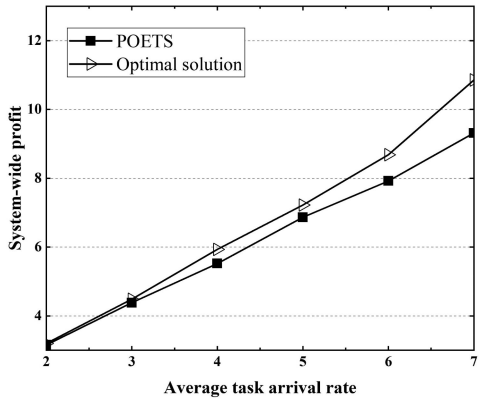


Fig. 14. System-wide profit with different task arrival rate.

Since we utilize the expected task execution latency to approximate the actual latency, the delay cost that computed by TMTS algorithm is 6 percent higher than that obtained by the optimal solution, and the performance gap becomes large with the increase of the average task arrival rate. This is because the deviation of the expected value estimation becomes large. Correspondingly, the performance of the system-wide profit shows a similar trend, and the optimal solution performs 8 percent better than our proposed POETS algorithm. In summary, the results obtained by our methods can approximate the optimal solution effectively with low computational complexity.

6 Conclusion

In this paper, we investigate the issues of transmission scheduling, offloading ratio and payoff decisions for 5G-enabled vehicular networks. To characterize the complexity of the formulated problem, constraints are considered including latency sensitivity, incentive compatibility and individual rationality. A partial computation offloading and adaptive task scheduling algorithm is proposed to maximize the system-wide profit, namely POETS. Theoretical analysis proves that both the incentive compatibility and individual rationality of vehicles can be satisfied. Performance evaluations based on real-world traces of taxies in Hangzhou (China) demonstrate the effectiveness of our solution from the aspects of system-wide profit and the number of vehicles benefiting from MEC. Specifically, our proposed transmission scheduling discipline can reduce the average delay cost by 20 percent

compared with benchmarks. The POETS algorithm can increase the system-wide profit by 25 percent on average. By improving the spectrum efficiency, our proposed algorithms provide MEC services for more vehicles than that of the compared methods.

Acknowledgments

This work was supported by the National Key R&D Program of China under Grant 2018YFE0206800, in part by the National Natural Science Foundation of China under Grants 61971084, 62001073 and 61771120, and in part by the National Natural Science Foundation of Chongqing under Grant cstc2019jcyjmsxmX0208.

References

[1] X. Wang, Z. Ning, and S. Guo, "Multi-agent imitation learning for pervasive edge computing: A decentralized computation offloading algorithm," *IEEE Trans. Parallel Distrib. Syst.*, vol. 32, no. 2, pp. 411–425, Feb. 2020.

[2] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L.-C. Wang, "Deep reinforcement learning for mobile 5G and beyond: Fundamentals, applications, and challenges," *IEEE Veh. Technol. Mag.*, vol. 14, no. 2, pp. 44–52, Jun. 2019.

[3] Z. Ning *et al.*, "Mobile edge computing enabled 5G health monitoring for internet of medical things: A decentralized game theoretic approach," 2020. [Online]. Available: https://www.researchgate.net/publication/339874488_Mobile_Edge_Computing_Enabled_5G_Health_Monitoring_for_Internet_of_Medical_Things_A_Decentralized_Game_Theoretic_Approach_IEEE_Journal_on_Selected_Areas_in_Communications_2020

[4] X. Wang, Z. Ning, S. Guo, and L. Wang, "Imitation learning enabled task scheduling for online vehicular edge computing," *IEEE Trans. Mobile Comput.*, early access, Jul. 28, 2020, doi: 10.1109/TMC.2020.3012509.

[5] Z. Ning *et al.*, "Joint computing and caching in 5G-envisioned internet of vehicles: A deep reinforcement learning-based traffic control system," *IEEE Trans. Intell. Transp. Syst.*, early access, Feb. 05, 2020, doi: 10.1109/TITS.2020.2970276.

[6] M. V. Barbera, S. Kosta, A. Mei, and J. Stefa, "To offload or not to offload? The bandwidth and energy costs of mobile cloud computing," in *Proc. IEEE INFOCOM*, 2013, pp. 1285–1293.

[7] T. Q. Dinh, J. Tang, Q. D. La, and T. Q. Quek, "Offloading in mobile edge computing: Task allocation and computational frequency scaling," *IEEE Trans. Commun.*, vol. 65, no. 8, pp. 3571–3584, Aug. 2017.

[8] J. Kwak, Y. Kim, J. Lee, and S. Chong, "Dream: Dynamic resource and task allocation for energy minimization in mobile cloud systems," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2510–2523, Dec. 2015.

[9] C. Wang, C. Liang, F. R. Yu, Q. Chen, and L. Tang, "Computation offloading and resource allocation in wireless cellular networks with mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 8, pp. 4924–4938, Aug. 2017.

[10] Y. Cui *et al.*, "Performance-aware energy optimization on mobile devices in cellular network," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 1073–1089, Apr. 2016.

[11] X. Wang *et al.*, "Dynamic resource scheduling in mobile edge cloud with cloud radio access network," *IEEE Trans. Parallel Distrib. Syst.*, vol. 29, no. 11, pp. 2429–2445, Nov. 2018.

[12] X. Lyu *et al.*, "Optimal schedule of mobile edge computing for internet of things using partial information," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2606–2615, Nov. 2017.

[13] B. Di, L. Song, and Y. Li, "Sub-channel assignment, power allocation, and user scheduling for non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7686–7698, Nov. 2016.

[14] Y. Liu, F. R. Yu, X. Li, H. Ji, and V. C. Leung, "Hybrid computation offloading in fog and cloud networks with non-orthogonal multiple access," in *Proc. IEEE Conf. Comput. Commun. Workshops*, 2018, pp. 154–159.

[15] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 974–983, Apr. 2014.

[16] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2015.

[17] C. Yi, J. Cai, and Z. Su, "A multi-user mobile computation offloading and transmission scheduling mechanism for delay-sensitive applications," *IEEE Trans. Mobile Comput.*, vol. 19, no. 1, pp. 29–43, Jan. 2019.

[18] S. Jošilo and G. Dán, "Selfish decentralized computation offloading for mobile cloud computing in dense wireless networks," *IEEE Trans. Mobile Comput.*, vol. 18, no. 1, pp. 207–220, Jan. 2018.

[19] T. Z. Oo, N. H. Tran, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Offloading in hetnet: A coordination of interference mitigation, user association, and resource allocation," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2276–2291, Aug. 2016.

[20] L.-W. Chen and C.-C. Chang, "Cooperative traffic control with green wave coordination for multiple intersections based on the Internet of vehicles," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1321–1335, Jul. 2016.

[21] A. Goldsmith, *Wireless Communications*. Cambridge, UK: Cambridge university press, 2005.

[22] Z. Ning *et al.*, "Intelligent edge computing in internet of vehicles: A joint computation offloading and caching solution," *IEEE Trans. Intell. Transp. Syst.*, early access, Jun. 05, 2020, doi: 10.1109/TITS.2020.2997832.

[23] Y. Wen, W. Zhang, and H. Luo, "Energy-optimal mobile application execution: Taming resource-poor mobile devices with cloud clones," in *Proc. IEEE INFOCOM*, 2012, pp. 2716–2720.

[24] V. Kantere, D. Dash, G. Francois, S. Kyriakopoulou, and A. Ailamaki, "Optimal service pricing for a cloud cache," *IEEE Trans. Knowl. Data Eng.*, vol. 23, no. 9, pp. 1345–1358, Sep. 2011.

[25] H. Xu and B. Li, "Dynamic cloud pricing for revenue maximization," *IEEE Trans. Cloud Comput.*, vol. 1, no. 2, pp. 158–171, Jul./Dec. 2013.

[26] Z. Xiong, S. Feng, W. Wang, D. Niyato, P. Wang, and Z. Han, "Cloud/Fog computing resource management and pricing for blockchain networks," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4585–4600, Jun. 2018.

[27] Z. Xiong, S. Feng, D. Niyato, P. Wang, A. Leshem, and Z. Han, "Joint sponsored and edge caching content service market: A game-theoretic approach," *IEEE Trans. Wireless Commun.*, vol. 18, no. 2, pp. 1166–1181, Feb. 2019.

[28] Y. Masuda and S. Whang, "Dynamic pricing for network service: Equilibrium and stability," *Manag. Sci.*, vol. 45, no. 6, pp. 857–869, 1999.

[29] A. E. Roth and M. Sotomayor, "A study in game-theoretic modeling and analysis," *Econometric Society Monographs*, Cambridge, UK: Cambridge Univ. Press, vol. 18, 1990.

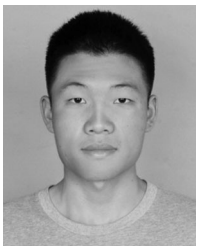
[30] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-Advanced for Mobile Broadband*. New York, NY, USA: Academic press, 2013.

[31] A. S. Alfa, *Queueing Theory for Telecommunications: Discrete Time Modelling of a Single Node System*. Berlin, Germany: Springer, 2010.

[32] Z. Ning, P. Dong, X. Kong, and F. Xia, "A cooperative partial computation offloading scheme for mobile edge computing enabled Internet of things," *IEEE Internet Things J.*, vol. 6, no. 3, pp. 4804–4814, Jun. 2018.

[33] 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. 2016.

Zhaolong Ning (Senior Member, IEEE) received the MS and PhD degrees from Northeastern University, Shenyang, China. He was a research fellow at Kyushu University, Japan. Currently, he is a distinguished professor at the Chongqing University of Posts and Telecommunications, China, and an associate professor with the Dalian University of Technology, China. His research interests include Internet of Things, mobile edge computing, and resource management. He has published more than 120 scientific papers in international journals and conferences. He is elected to be the Young Elite Scientists Sponsorship Program by CAST and Hong Kong Scholar.



Peiran Dong received BS degree from the Dalian University of Technology, Dalian, China, in 2018. He is currently working toward the MS degree in software engineering at the Dalian University of Technology. His research interests include mobile edge computing, network computation offloading, and resource management.

Xiaojie Wang received the MS degree from Northeastern University, China, in 2011, and the PhD degree from the Dalian University of Technology, Dalian, China, in 2019. From 2011 to 2015, she was a software engineer in NeuSoft Corporation, China. Currently, she is a postdoctor at the Hong Kong Polytechnic University. Her research interests include Internet of Things, mobile edge computing, and machine learning.

Xiping Hu received the PhD degree in electrical and computer engineering from the University of British Colombia, Vancouver, Canada. He is currently a professor with the Lanzhou University, China. He was the co-founder and CTO of Bravolol Limited, Hong Kong, a leading language learning mobile application company with more than 100 million users, and listed as top two language education platform globally. He has more than 90 articles published and presented in prestigious conferences and journals. His research interests include distributed intelligent systems, crowdsensing, social networks, and cloud computing.

Jiangchuan Liu (Fellow, IEEE) received the BEng (cum laude) degree in computer science from Tsinghua University, Beijing, China, in 1999, and the PhD degree in computer science from the Hong Kong University of Science and Technology, in 2003. He is currently a full professor (with University Professorship) at the School of Computing Science, Simon Fraser University, BC, Canada. He is a fellow of the Canadian Academy of Engineering and the NSERC E.W.R. Steacie Memorial fellow. He is a steering committee member of *IEEE Transactions on Mobile Computing*. He was a co-recipient of the Test of Time Paper Award of the IEEE INFOCOM, in 2015, the ACM TOMCCAP Nicolas D. Georganas Best Paper Award, in 2013, and the ACM Multimedia Best Paper Award, in 2012. He is an associate editor of the *IEEE/ACM Transactions on Networking*, the *IEEE Transactions on Big Data*, and the *IEEE Transactions on Multimedia*.

Lei Guo received the PhD degree from the University of Electronic Science and Technology of China, Chengdu, China, in 2006. He is currently a full professor at the Chongqing University of Posts and Telecommunications, Chongqing, China. He has authored or coauthored more than 200 technical papers in international journals and conferences. He is an editor for several international journals. His current research interests include communication networks, optical communications, and wireless communications.

Bin Hu (Senior Member, IEEE) is currently a professor at Lanzhou University, and a guest professor at ETH Zurich, Switzerland. He is an IET fellow, co-chairs of IEEE SMC TC on Cognitive Computing, and member at Large of ACM China, vice president of International Society for Social Neuroscience (China committee) etc. He has published more than 100 papers in peer reviewed journals, conferences, and book chapters including *Science*, *Journal of Alzheimer's Disease*, *PLoS Computational Biology*, *IEEE Trans.*, *IEEE Intelligent Systems*, AAAI, etc. He has served as chairs/co-chairs in many IEEE international conferences/workshops, and associate editors in *Peer Reviewed Journals on Cognitive Science and Pervasive Computing*, such as *IEEE Trans. Affective Computing*, *Brain Informatics*, *IET Communications*, etc.

Ricky Y. K. Kwok (Fellow, IEEE) received the BSc degree in computer engineering from the University of Hong Kong, in 1991, and the MPhil and PhD degrees, both in computer science, from the Hong Kong University of Science and Technology (HKUST), in 1994 and 1997, respectively. His research interests include designing efficient communication protocols and robust resources management algorithms toward enabling large scale distributed mobile computing. In these research areas, he has authored one textbook, co-authored another two textbooks, and published more than 200 technical papers in various leading journals, research books, and refereed international conference proceedings. He is a fellow of the HKIE, and the IET. From March 2006 to December 2011, Ricky served on the Editorial Board of the *Journal of Parallel and Distributed Computing* as a subject area editor in *Peer-to-Peer Computing*. He also served as an associate editor for the *IEEE Transactions on Parallel and Distributed Systems* from January 2013 to December 2016.

Victor C. M. Leung (Fellow, IEEE) is currently a distinguished professor of computer science and software engineering at Shenzhen University, China. He is also an Emeritus professor of electrical and computer engineering and the director of the Laboratory for Wireless Networks and Mobile Systems at the University of British Columbia (UBC), Canada. His research interests include wireless networks and mobile systems, and he has published widely in these areas. He is serving on the editorial boards of the *IEEE Transactions on Green Communications and Networking*, *IEEE Transactions on Cloud Computing*, *IEEE Access*, *IEEE Network*, and several other journals. He received the 1977 APEBC Gold Medal, 1977-1981 NSERC Postgraduate Scholarships, IEEE Vancouver Section Centennial Award, 2011 UBC Killam Research Prize, 2017 Canadian Award for Telecommunications Research, 2018 IEEE TCGCC Distinguished Technical Achievement Recognition Award, and 2018 ACM MSWiM Reginald Fessenden Award. He co-authored papers that won the 2017 IEEE ComSoc Fred W. Ellersick Prize, 2017 IEEE Systems Journal Best Paper Award, 2018 IEEE CSIM Best Journal Paper Award, and 2019 IEEE TCGCC Best Journal Paper Award. He is a Fellow of the Royal Society of Canada, Canadian Academy of Engineering, and Engineering Institute of Canada. He is named in the current Clarivate Analytics list of Highly Cited Researchers.