Lu *et al.*
EURASIP Journal on Advances in Signal Processing (2022) 2022:44
https://doi.org/10.1186/s13634-022-00867-2

EURASIP Journal on Advances in Signal Processing

RESEARCH

Open Access

Analytical offloading design for mobile edge computing-based smart internet of vehicle

Jinrong Lu[1], Lunyuan Chen[1], Junjuan Xia[1*], Fusheng Zhu[2*], Maobin Tang[1*], Chengyuan Fan[3] and Jiangtao Ou[3]

*Correspondence: xiajunjuan@gzhu.edu.cn; zhufusheng@gdcni.cn; tmb178@gzhu.edu.cn

[1] School of Computer Science, Guangzhou University, Guangzhou, China
[2] Guangdong New Generation Communication and Network Innovative Institute (GDCNi), Guangzhou, China
Full list of author information is available at the end of the article

Abstract

In this paper, we investigate how to analytically design an analytical offloading strategy for a multiuser mobile edge computing (MEC)-based smart internet of vehicle (IoV), where there are multiple computational access points (CAPs) which can help compute tasks from the vehicular users. As it is difficult to derive an analytical offloading ratio for a general MEC-based IoV network, we turn to provide an analytical offloading scheme for some special MEC networks including one-to-one, one-to-two and two-to-one cases. For each case, we study the system performance by using the linear combination of latency and energy consumption, and derive the analytical offloading ratio through minimizing the system cost. Simulation results are finally presented to verify the proposed studies. In particular, the proposed analytical offloading scheme can achieve the optimal performance of the brute force (BF) scheme. The analytical results in this paper can serve as an important reference for the analytical offloading design for a general MEC-based IoV.

Keywords: Internet of vehicle, Mobile edge computing, Offloading strategy, Latency, Energy consumption

1 Introduction

With the development and deployment of the fifth-generation (5G) wireless communication system [1–3], IoV has emerged as one of the most important components in smart cities, which has attracted much attention from both the academy and industry. In the IoV systems, various sensors at the vehicle can collect travelling state data and make real-time decision through analyzing the data, which can effectively prevent the traffic collision and congestion [4, 5]. From this process, one can clearly find that the development of IoV systems involves massive communication among the sensors, and massive computing to perform the analysis on the massive data [6–8]. Hence, some new communication and computing techniques should be employed to develop the IoV systems.

To support the development of IoV systems, some new communication techniques should be developed. One of the promising communication techniques for the next-generation communication is the multiple-input multiple-output (MIMO) [9, 10], where multiple antennas are equipped at the transmitter and receiver to provide plenty of spatial resources in order to increase the transmission data rate significantly [11–13].

Another promising technique is the intelligent reflecting surface (IRS), which installs a lot of reflection units on the transmitter and receiver [3, 14]. In the IRS systems, the transmission phase, amplitude and frequency can be adjusted to enhance the system performance significantly [15–17]. Moreover, the caching can be integrated into the wireless transmission system, which can help enlarge the dimension of the system resources [18]. In addition, resource allocation based on novel technologies can improve the system performance [19, 20]. In further, some intelligent algorithms, such as the channel state information (CSI) feedback [21], federated learning [22], and reinforcement learning [23] can be applied to enhance the communication quality.

Besides the communication techniques, some new computing techniques should be developed to support the development of IoV systems. Cloud computing was proposed to help compute the intensive computational tasks through offloading the tasks to the cloud server, which however, causes a heavy overload and a severe issue of information leakage. Accordingly, mobile edge computing (MEC) has been proposed to set the computational access points (CAPs) nearby the users, which can effectively reduce the system latency and energy consumption [24, 25]. In this direction, the authors in [26] investigated the impact of outdated CSI on the performance of MEC networks, through analyzing the system outage probability by taking into account the latency and energy consumption. Moreover, the performance of MEC networks in eavesdropping environments was studied in [27], where the deep Q-network (DQN)-based intelligent algorithms was proposed to devise the offloading strategy. In further, for the training networks with a large number of parameters, the network parameters can be split to be computed on different nodes in the network [28]. However, these approaches require high overhead of space and time to obtain offloading strategy. Different from reinforcement learning scheme, the analytical solutions have no exploration overhead, and can directly obtain the offloading strategy with the complexity of O(1). Moreover, to the best of our knowledge, there has been little work on the study of analytical methods to solve the offloading design for the MEC networks. In further, analytical method can provide theoretical basis for other offloading optimization schemes. It can help better understand the offloading mechanism and the influence of each parameter on the system performance. Therefore, the above views motivate the work in this paper. However, there are some difficulties in providing analytical solutions to offloading strategy. Specifically, when there are multiple CAPs and multiple users, it is difficult for us to devise an offloading strategy, taking into account the complexity from the increasing number of users and CAPs.

In this paper, we investigate how to analytically design an analytical offloading strategy for a multiuser MEC-based smart IoV, where there are multiple CAPs which can help compute tasks from the vehicular users. As it is difficult to derive an analytical offloading ratio for a general MEC-based IoV network, we turn to provide an analytical offloading scheme for some special MEC networks including one-to-one, one-to-two and two-to-one cases.

For each case, we study the system performance by using the linear combination of latency and energy consumption, and derive the analytical offloading ratio through minimizing the system cost. Simulation results are finally presented to verify the proposed studies. In particular, the proposed analytical offloading scheme can achieve the optimal performance of the BF scheme. The analytical results in this paper can serve as an important reference for the analytical offloading design for a general MEC-based IoV. The main contributions of this paper are summarized below.

- We investigate how to optimize the linear combination of latency and energy consumption by making offloading decision in vehicular MEC network. We theoretically analyze the system cost for one-to-one, one-to-two and two-to-one vehicular MEC system, which can serve as an important reference for later studies for a general vehicular MEC network.
- According to the analytical results for two-to-one and one-to-two cases, we propose an optimization based on the linear programming method, which can directly achieve the optimal offloading ratio at computational complexity $\mathcal{O}(1)$. Moreover, compared to the learning-based methods, the proposed strategy can avoid the incentive computational cost.
- We also conduct simulations to verify the effectiveness of the proposed strategy. Specifically, both the simulated and theoretical results are provided and show the superiority of the proposed strategy over the learning-based solutions.

The rest of this paper is organized as follows. After the introduction, we will discuss the system model of MEC-based IoV network in Sec. II. Then, we give the optimization problem in Sec. III, and provide the analytical solution for the offloading design in order to minimize the system latency and energy consumption. We further give some simulation results and discussions in Sec. IV, and finally conclude the work of this paper in Sec. V.

2 System model

Figure 1 shows the system model of the vehicular MEC network, where there are M users denoted by $\mathcal{M} = \{1, 2, \ldots, M\}$ and N CAPs denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. Users need to execute some computation-intensive and latency-sensitive tasks, which can be partitioned into several subtasks by some reasonable task partition method and then offloading to the CAPs for computing. Specifically, in our model, at the beginning of each time slot, different users generate computational tasks, which can be partitioned into several subtasks. After that, the users offload some subtasks to the same CAP in parallel, where the users access the CAP through the some orthogonal spectrum resources. When the tasks of all users are completed, one time slot is used up. For user $m \in \mathcal{M}$, it needs to offload its subtasks through wireless links characterized by the offloading decision $\boldsymbol{\alpha}_m$, which can be written as

$$\boldsymbol{\alpha}_m = [\alpha_{m,0}, \alpha_{m,1}, \alpha_{m,2}, \ldots, \alpha_{m,n}], \tag{1}$$

where $\alpha_{m,n}$ $(n > 0)$ is offloading ratio from user m to CAP n, while $\alpha_{m,0}$ is local computational ratio for user m.

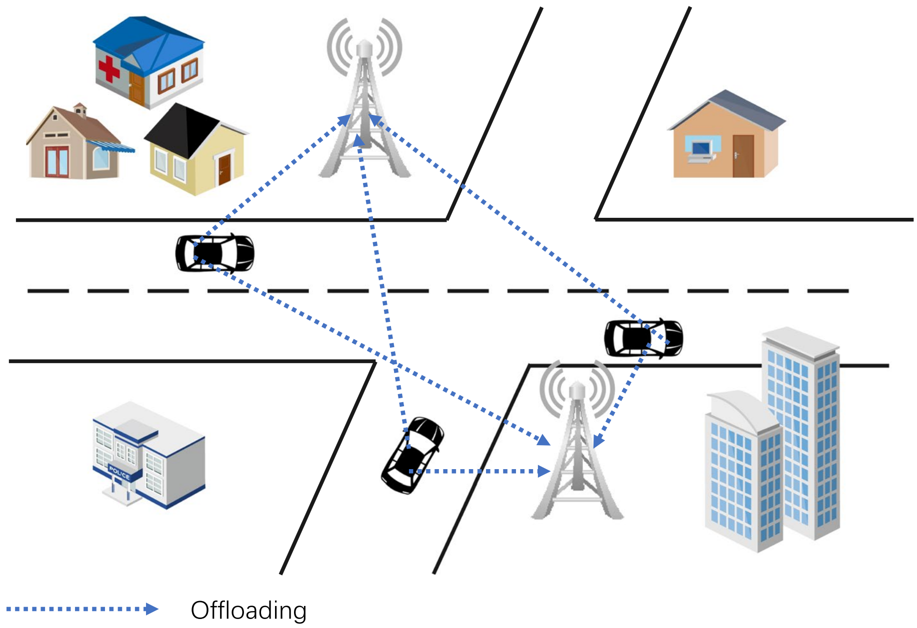


Fig. 1 Task offloading in the vehicular MEC network

In the following subsections, we will firstly present the local computational model, the communication model and the offloading computational model, respectively. Then, we will discuss the optimization problem according to the proposed model.

2.1 Local computational model

In the local computational model, let f_m denote CPU cycles per second at the user m. Then, the local computational latency can be expressed by

$$T_m^{local} = \frac{\alpha_{m,0} l_m \omega}{f_m}, \tag{2}$$

where ω denotes the number of cycles required for the CPU to compute per bit of tasks and l_m is the task size of user m. Thus, the local computational energy consumption for user m can be given by

$$E_m^{local} = T_m^{local} P_m^{local}, \tag{3}$$

where P_m^{local} is the local computational power for user m. In summary, the local computational energy consumption of M users can be summarized as

$$E^{local} = \sum_{m=1}^{M} E_m^{local}. \tag{4}$$

2.2 Communication model

In the communication model, according to the Shannon theory, we can obtain the transmission rate from user m to CAP n as [29–31]

$$R_{m,n} = W_{m,n} \log_2 \left(1 + \frac{|h_{m,n}|^2 P_m^{trans}}{\sigma^2} \right), \tag{5}$$

where $h_{m,n} \sim \mathcal{CN}(0, \beta)$ is the channel parameter of the link between user m and CAP n, $W_{m,n}$ is the communication bandwidth of user m allocated by CAP n, P_m^{trans} is transmission power at user m, and σ^2 is the variance of the additive white Gaussian noise (AWGN) at CAP n, where the noise effect on the wireless transmission is detailed in the literature [32, 33]. Then, the latency of user m offloading some subtasks to CAP n can be given by

$$T_{m,n}^{trans} = \frac{\alpha_{m,n} l_m}{R_{m,n}}. \tag{6}$$

The associated energy consumption can be given by

$$E_{m,n}^{trans} = T_{m,n}^{trans} P_m^{trans}. \tag{7}$$

In summary, the total transmission energy consumption of M users can be given by

$$E^{trans} = \sum_{m=1}^{M} \sum_{n=1}^{N} E_{m,n}^{trans}. \tag{8}$$

2.3 Offloading computational model

When the subtasks have been transmitted to the CAPs, they can be computed by the CAPs with a more powerful computational capability. The computational latency at CAP n can be given by

$$T_{m,n}^{CAP} = \frac{\alpha_{m,n} l_m \omega}{F_{m,n}}, \tag{9}$$

where $F_{m,n}$ is the computational capability assigned by CAP n to user m.

Since the users perform the local computing and offloading in parallel, the system total latency is the maximum value among the latency executed at local and CAPs. Then, we can obtain the total latency for user m as,

$$T_m = \max\{T_m^{local}, T_{m,1}^{trans} + T_{m,1}^{CAP}, \ldots, T_{m,N}^{trans} + T_{m,N}^{CAP}\}. \tag{10}$$

In addition, each user uses its individual computational resource to calculate the tasks. And the offloading is implemented in parallel among M users, as each user employs an orthogonal spectrum resource for the wireless transmission. Hence, the total system latency can be given by,

$$T^{total} = \max\{T_1, T_2, T_3, \ldots, T_M\}. \tag{11}$$

Similarly, we can obtain the total system energy consumption as,

$$E^{total} = E^{local} + E^{trans}. \tag{12}$$

Considering that both the latency and energy consumption have a significant impact on the system, we should optimize both the energy consumption and latency for the system. Therefore, we use a linear combination of T^{total} and E^{total} to measure the system cost, given by

$$\Phi = \lambda T^{total} + (1-\lambda)E^{total}, \tag{13}$$

where $\lambda \in [0,1]$ is a weight factor between the latency and energy consumption. We can get a flexible form of the system cost for the MEC system, by adjusting the value of λ, according to the practical requirements.

2.4 Problem formulation

As the offloading strategy can directly affect the system latency and energy consumption, we should take into account the offloading parameter $\{\alpha_{m,n} | 1 \leq m \leq M, 0 \leq n \leq N\}$, and devise the optimization objective to minimize the system cost, given by

$$\begin{aligned} \min_{\{\alpha_{m,n}\}} \quad & \Phi = \lambda T^{total} + (1-\lambda)E^{total} \\ s.t. \quad & C_1 : \alpha_{m,n} \in [0,1], \forall m \in M, \forall n \in N, \\ & C_2 : \sum_{n=0}^{N} \alpha_{m,n} = 1, \end{aligned} \tag{14}$$

where constraint C_1 represents that each user can partition its tasks and partially offload to the CAPs, and constraint C_2 denotes that the sum of the offloading ratio of each user should be unity.

3 Optimization of offloading strategy

For the offloading strategy in the MEC networks, many researchers have employed to use some intelligent algorithms, such as reinforcement learning, to obtain a feasible offloading decision. However, these approaches require a large number of iterations in the training and cause a heavy overhead. Therefore, in this paper, we intend to provide an offloading strategy that can directly obtain the offloading ratio by using some analytical method, which can significantly reduce the computational complexity. However, it is generally difficult to obtain an analytical offloading strategy for a large scale MEC network with M users and N CAPs. Hence, we turn to study some special cases to help deal with the tricky problem, which includes one user to one CAP case (called one-to-one), two users to one CAP case (called two-to-one), and one user to two CAPs case (called one-to-two). Specifically, for the one-to-one case, we will get the optimal offloading ratio by deriving the objective function. For the two-to-one and one-to-two cases, based on the characteristics of the max function in the latency formula, we will partition the analytical domain into several regions, and then use the linear programming (LP) method to find the local optimal offloading solution in each feasible region. The three special cases are detailed one by one in the following.

3.1 One-to-one

In the one-to-one case, the single user accesses and offloads its task to the single CAP. Consequently, the offloading ratio is $\boldsymbol{\alpha_1} = [\alpha_{1,0}, \alpha_{1,1}]$, where we can see that $\alpha_{1,0} = 1 - \alpha_{1,1}$. Thus, we can rewrite the system cost for the one-to-one case with respect to $\alpha_{1,1}$ only, given by

$$\begin{aligned} \Phi_1(\alpha_{1,1}) &= \lambda T^{total}(\alpha_{1,1}) + (1-\lambda)E^{total}(\alpha_{1,1}) \\ &= \lambda \max\left\{ \frac{l_1\big(1-\alpha_{1,1}\big)\omega}{f_1}, \frac{l_1\alpha_{1,1}}{R_{1,1}} + \frac{l_1\alpha_{1,1}\omega}{F_{1,1}} \right\} \\ &\quad + (1-\lambda)\left[\frac{l_1\big(1-\alpha_{1,1}\big)\omega}{f_1} P_1^{local} + \frac{l_1\alpha_{1,1}}{R_{1,1}} P_{1,1}^{trans} \right]. \end{aligned} \tag{15}$$

In order to minimize the system cost Φ_1 and obtain the optimal offloading ratio for the one-to-one case, we consider three cases, which are $\lambda = 1$, $\lambda = 0$, and $0 < \lambda < 1$, for (15).

3.1.1 $\lambda = 1$

When $\lambda = 1$, the system cost of the linear combination of latency and energy consumption degenerates into latency only. In this case, minimizing the linear combination is equivalent to minimizing the latency. The system cost can be given by

$$\begin{aligned} \Phi_1(\alpha_{1,1}) &= T^{total}(\alpha_{1,1}) \\ &= \max\left\{ \frac{l_1\big(1-\alpha_{1,1}\big)\omega}{f_1}, \frac{l_1\alpha_{1,1}}{R_{1,1}} + \frac{l_1\alpha_{1,1}\omega}{F_{1,1}} \right\}. \end{aligned} \tag{16}$$

From (16), we can see that $\frac{l_1(1-\alpha_{1,1})\omega}{f_1}$ decreases with the increase of $\alpha_{1,1}$, while $\frac{l_1\alpha_{1,1}}{R_{1,1}} + \frac{l_1\alpha_{1,1}\omega}{F_{1,1}}$ increases with the increase of $\alpha_{1,1}$. In order to facilitate the analysis, we remove the max operation in (16) and convert $\Phi_1(\alpha_{1,1})$ into

$$\Phi_1(\alpha_{1,1}) = \begin{cases} \dfrac{l_1\big(1-\alpha_{1,1}\big)\omega}{f_1}, & \alpha_{1,1} \le \gamma_1, \\ \dfrac{l_1\alpha_{1,1}}{R_{1,1}} + \dfrac{l_1\alpha_{1,1}\omega}{F_{1,1}}, & \alpha_{1,1} > \gamma_1, \end{cases} \tag{17}$$

where $\gamma_1 = l_1 \dfrac{A}{A+B+C}$, $A = \dfrac{\omega}{f_1}$, $B = \dfrac{1}{R_{1,1}}$, and $C = \dfrac{\omega}{F_{1,1}}$. From (17), we can obtain the optimal offloading ratio of $a_{1,1}^*$ as,

$$\alpha_{1,1}^* = \gamma_1. \tag{18}$$

3.1.2 $\lambda = 0$

When $\lambda = 0$, the system cost of the linear combination of latency and energy consumption degenerates into energy consumption only. In this case, minimizing the linear combination is equivalent to minimizing the energy consumption. The system cost can be given by

$$\Phi_1(\alpha_{1,1}) = E^{total}(\alpha_{1,1}) = l_1\left[(1-\alpha_{1,1})\frac{P_1^{local}\omega}{f_1} + \alpha_{1,1}\frac{P_{1,1}^{trans}}{R_{1,1}}\right]. \tag{19}$$

We can see that (19) is a linear function with $\alpha_{1,1}$. So, we can get the optimal offloading ratio by comparing $\frac{P_1^{local}\omega}{f_1}$ and $\frac{P_{1,1}^{trans}}{R_{1,1}}$, given by

$$\alpha_{1,1}^* = \begin{cases} 1, & \frac{P_1^{local}\omega}{f_1} \geq \frac{P_{1,1}^{trans}}{R_{1,1}}, \\ 0, & \frac{P_1^{local}\omega}{f_1} < \frac{P_{1,1}^{trans}}{R_{1,1}}. \end{cases} \tag{20}$$

3.1.3 $0 < \lambda < 1$

When $0 < \lambda < 1$, the system cost is the linear combination of latency and energy consumption. In this case, we have to consider both energy consumption and latency and we can find the optimal offloading ratio by studying the monotonicity of the system cost. Specifically, the case $0 < \lambda < 1$ is the combination of $\lambda = 0$ and $\lambda = 1$, and from the above analysis, we can see that the system cost of the case $\lambda = 1$ is a piecewise function, while the system cost of the case $\lambda = 0$ is a monotonic function. Consequently, the system cost of the case $1 < \lambda < 1$ is also a piecewise function, which is given by

$$\Phi_1(\alpha_{1,1}) = \begin{cases} \lambda T_1^{local} + (1-\lambda)E^{total}, & \alpha_{1,1} \leq \gamma_1, \\ \lambda(T_{1,1}^{trans} + T_1^{CAP}) + (1-\lambda)E^{total}, & \alpha_{1,1} > \gamma_1. \end{cases} \tag{21}$$

Then, to study the monotonicity of the system cost, we take the derivative of (21) with respect to $a_{1,1}$ that is given by

$$\Phi_1{}'(\alpha_{1,1}) = \begin{cases} \underbrace{l_1\left[-\frac{\lambda\omega}{f_1} + (\lambda-1)\frac{P_1^{local}\omega}{f_1} + (1-\lambda)\frac{P_{1,1}^{trans}}{R_{1,1}}\right]}_{J_1}, & \alpha_{1,1} < \gamma_1, \\ \underbrace{l_1\left[\lambda\left(\frac{1}{R_{1,1}} + \frac{\omega}{F_{1,1}}\right) + (1-\lambda)\left(\frac{P_{1,1}^{trans}}{R_{1,1}} - \frac{P_1^{local}\omega}{f_1}\right)\right]}_{J_2}, & \alpha_{1,1} > \gamma_1, \end{cases} \tag{22}$$

where

$$\begin{aligned} J_1 &= l_1\left[-\frac{\lambda\omega}{f_1} + (\lambda-1)\frac{P_1^{local}\omega}{f_1} + (1-\lambda)\frac{P_{1,1}^{trans}}{R_{1,1}}\right], \\ J_2 &= l_1\left[\lambda\left(\frac{1}{R_{1,1}} + \frac{\omega}{F_{1,1}}\right) + (1-\lambda)\left(\frac{P_{1,1}^{trans}}{R_{1,1}} - \frac{P_1^{local}\omega}{f_1}\right)\right]. \end{aligned} \tag{23}$$

Hence, we can analyze the optimal offloading ratio as

$$\alpha_{1,1}^{*}=\begin{cases}1, & J_1\le 0, J_2\le 0,\\ \gamma_1, & J_1\le 0, J_2>0,\\ 0, & J_1>0, J_2>0.\end{cases} \tag{24}$$

3.2 Two-to-one

In the two-to-one case, two users can access and offload their tasks to the single CAP at each time slot. Consequently, offloading ratios of user 1 and user 2 are $\boldsymbol{\alpha_1}=[\alpha_{1,0},\alpha_{1,1}]$ and $\boldsymbol{\alpha_2}=[\alpha_{2,0},\alpha_{2,1}]$. We can simplify the system cost and rewrite it with respect to $\alpha_{1,1}\in[0,1]$ and $\alpha_{2,1}\in[0,1]$ by using $\alpha_{1,0}=1-\alpha_{1,1}$ and $\alpha_{2,0}=1-\alpha_{2,1}$, given by

$$\Phi_2\big(\alpha_{1,1},\alpha_{2,1}\big)=\lambda T^{total}\big(\alpha_{1,1},\alpha_{2,1}\big)+(1-\lambda)E^{total}\big(\alpha_{1,1},\alpha_{2,1}\big). \tag{25}$$

For convenience, we use the following notation,

$$t_{1A}=\frac{l_1\omega}{f_1},\qquad t_{1B}=l_1\left(\frac{1}{R_{1,1}}+\frac{\omega}{F_{1,1}}\right), \tag{26}$$

$$t_{2A}=\frac{l_2\omega}{f_2},\qquad t_{2B}=l_2\left(\frac{1}{R_{2,1}}+\frac{\omega}{F_{2,1}}\right), \tag{27}$$

$$E_{1A}=\frac{l_1P_1^{local}\omega}{f_1},\qquad E_{1B}=\frac{P_{1,1}^{trans}l_1}{R_{1,1}}, \tag{28}$$

$$E_{2A}=\frac{l_2P_2^{local}\omega}{f_2},\qquad E_{2B}=\frac{P_{2,1}^{trans}l_2}{R_{2,1}}. \tag{29}$$

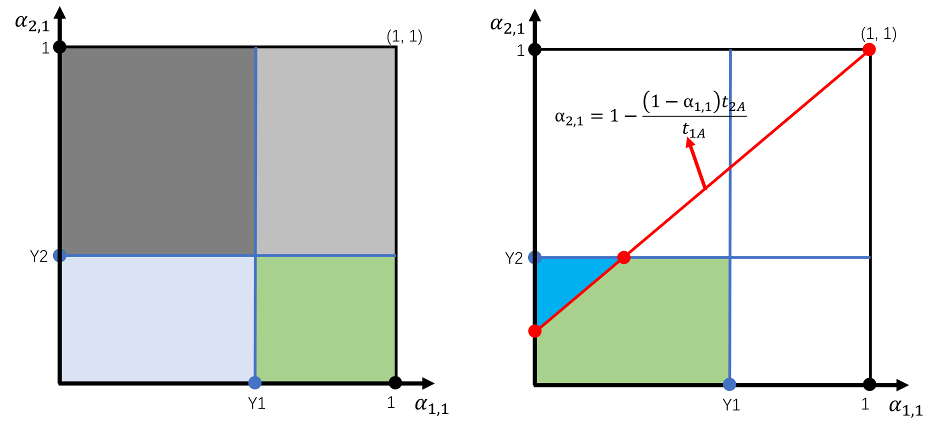
Then, the system cost can be rewritten into

$$\begin{aligned}\Phi_2(\alpha_{1,1}\alpha_{2,1})=&\ \lambda\max\big\{\max\big\{(1-\alpha_{1,1})t_{1A},\alpha_{1,1}t_{1B}\big\},\max\big\{(1-\alpha_{2,1})t_{2A},\alpha_{2,1}t_{2B}\big\}\big\}\\ &+\frac{1}{(1-\lambda)}[(1-\alpha_{1,1})E_{1A}+\alpha_{1,1}E_{1B}+(1-\alpha_{2,1})E_{2A}+\alpha_{2,1}E_{2B}].\end{aligned} \tag{30}$$

As (30) includes some max operations, it is hard to perform to analysis. To remove the max operation, we convert (30) into a piecewise function in

$$\Phi_2(\alpha_{1,1},\alpha_{2,1})=\begin{cases}\lambda\max\big\{(1-\alpha_{1,1})t_{1A},(1-\alpha_{2,1})t_{2A})\big\}+\frac{1}{(1-\lambda)}E^{total}, & \alpha_{1,1}<Y_1,\alpha_{2,1}<Y_2,\\ \lambda\max\big\{(1-\alpha_{1,1})t_{1A},\alpha_{2,1}t_{2B}\big\}+\frac{1}{(1-\lambda)}E^{total}, & \alpha_{1,1}<Y_1,\alpha_{2,1}\ge Y_2,\\ \lambda\max\big\{\alpha_{1,1}t_{1B},(1-\alpha_{2,1})t_{2A}\big\}+\frac{1}{(1-\lambda)}E^{total}, & \alpha_{1,1}\ge Y_1,\alpha_{2,1}<Y_2,\\ \lambda\max\big\{\alpha_{1,1}t_{1B},\alpha_{2,1}t_{2B}\big\}+\frac{1}{(1-\lambda)}E^{total}, & \alpha_{1,1}\ge Y_1,\alpha_{2,1}\ge Y_2,\end{cases} \tag{31}$$

where $Y_1=\dfrac{t_{1A}}{t_{1A}+t_{1B}}$ and $Y_2=\dfrac{t_{2A}}{t_{2A}+t_{2B}}$. Therefore, the domain of $\alpha_{1,1}$ and $\alpha_{2,1}$ can be partitioned into four regions, as shown in Fig. 2a.



(a) Partition according to (31). (b) Partition according to (33).

Fig. 2 Domain partition of the two-to-one case

Without lose of generality, we then analyze the case of $\alpha_{1,1} < Y_1$ and $\alpha_{2,1} < Y_2$. In this case, the system cost is given by

$$\Phi_2(\alpha_{1,1}, \alpha_{2,1}) = \lambda \max\left\{(1-\alpha_{1,1})t_{1A}, (1-\alpha_{2,1})t_{2A})\right\} + \frac{1}{(1-\lambda)}E^{total}. \tag{32}$$

To remove the max operation in (32), we convert it into a piecewise function in

$$\Phi_2(\alpha_{1,1}, \alpha_{2,1}) = \begin{cases} \lambda(1-\alpha_{1,1})t_{1A} + \frac{1}{(1-\lambda)}E^{total}, \ \alpha_{1,1} < Y_1, \alpha_{2,1} \in \left[1 - \frac{(1-\alpha_{1,1})t_{2A}}{t_{1A}}, Y_2\right), \\ \lambda(1-\alpha_{2,1})t_{2A} + \frac{1}{(1-\lambda)}E^{total}, \ \alpha_{1,1} < Y_1, \alpha_{2,1} < 1 - \frac{(1-\alpha_{1,1})t_{2A}}{t_{1A}}, \alpha_{2,1} < Y_2. \end{cases} \tag{33}$$

Then, we further partition the region into two parts as shown in Fig. 2b. For the part with $\alpha_{1,1} < Y_1, \alpha_{2,1} < 1 - \frac{(1-\alpha_{1,1})t_{2A}}{t_{1A}}$, and $\alpha_{2,1} < Y_2$, the system cost can be expressed as

$$\Phi_2(\alpha_{1,1}, \alpha_{1,2}) = \lambda(1-\alpha_{2,1})t_{2A} + (1-\lambda)E^{total}(\alpha_{1,1}, \alpha_{1,2}) \tag{34}$$

$$= \lambda t_{2A} + (1-\lambda)(E_{1A} + E_{2A}) + [(1-\lambda)(E_{2B} - E_{2A}) - \lambda t_{2A}]\alpha_{2,1} \tag{35}$$

$$+ (1-\lambda)(E_{1B} - E_{1A})\alpha_{1,1}. \tag{36}$$

Then, the corresponding problem optimization can be written as

$$\begin{aligned}
&\min_{\{\alpha_{1,1},\alpha_{2,1}\}} \quad \Phi_2(\alpha_{1,1},\alpha_{1,2}) - \lambda t_{2A} - (1-\lambda)(E_{1A}+E_{2A}) \\
&\quad = [(1-\lambda)(E_{2B}-E_{2A}) - \lambda t_{2A}]\alpha_{2,1} \\
&\quad + (1-\lambda)(E_{1B}-E_{1A})\alpha_{1,1} \\
&s.t. \quad C_1 : \alpha_{1,1} < Y_1, \\
&\qquad C_2 : \alpha_{2,1} < 1 - \frac{(1-\alpha_{1,1})t_{2A}}{t_{1A}}, \alpha_{2,1} < Y_2, \\
&\qquad C_3 : \alpha_{1,1} \geq 0, \alpha_{2,1} \geq 0.
\end{aligned} \tag{37}$$

We can observe that (37) is a linear programming problem and thus it can be solved by using some linear programming methods. Specifically, for non-integer linear programming, if the optimal solution exists, the optimal solution will be falling at the point of intersection. So, we only need to collect all points of intersection in this part and compare the associated costs to find the local optimal offloading decision. We can perform a similar operation for the other parts and get the global optimal offloading decision.

3.3 One-to-two

In the one-to-two case, the single user can access and offload its task to the two CAPs at each time slot. Consequently, the user offloading ratio is $\boldsymbol{\alpha_1} = [\alpha_{1,0}, \alpha_{1,1}, \alpha_{1,2}]$. We can simplify the system cost and rewrite it with respect to $\alpha_{1,1} + \alpha_{1,2} \leq 1$ by using $\alpha_{1,0} = 1 - \alpha_{1,1} - \alpha_{1,2}$, given by

$$\Phi_3\left(\alpha_{1,1},\alpha_{1,2}\right) = \lambda T^{total}\left(\alpha_{1,1},\alpha_{1,2}\right) + (1-\lambda)E^{total}\left(\alpha_{1,1},\alpha_{1,2}\right). \tag{38}$$

For convenience, we use the following notations,

$$t_{0A} = \frac{l_1\omega}{f_1}, \qquad E_{0A} = \frac{l_1 P_1^{local}\omega}{f_1}, \tag{39}$$

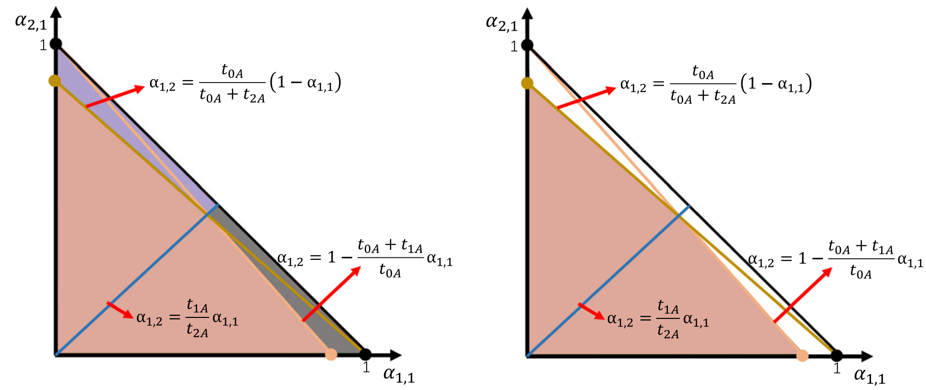
$$t_{1A} = l_1\left(\frac{1}{R_{1,1}} + \frac{\omega}{F_{1,1}}\right), \qquad E_1A = \frac{l_1 P_{1,1}^{trans}}{R_{1,1}}, \tag{40}$$

$$t_{2A} = l_1\left(\frac{1}{R_{1,2}} + \frac{\omega}{F_{1,2}}\right), \qquad E_2A = \frac{l_1 P_{1,2}^{trans}}{R_{1,1}}. \tag{41}$$

Then, the system cost can be rewritten as

$$\begin{aligned}
\Phi_3(\alpha_{1,1},\alpha_{1,2}) = \lambda &\max\left\{t_{0A}(1-\alpha_{1,1}-\alpha_{1,2}), t_{1A}\alpha_{1,1}, t_{2A}\alpha_{1,2}\right\} \\
&+ (1-\lambda)E_{0A}(1-\alpha_{1,1}-\alpha_{1,2}) \\
&+ (1-\lambda)E_{1A}\alpha_{1,1} + (1-\lambda)E_{2A}\alpha_{1,2}.
\end{aligned} \tag{42}$$

As (42) includes max operation, it is hard to perform the analysis. To remove the max operation, we convert (42) into a piecewise function in



(a) Partition according to (43). (b) Partition according to (44).

Fig. 3 Domain partition of the one-to-two case

$$\Phi_3(\alpha_{1,1},\alpha_{1,2})=\begin{cases}\lambda t_{0A}(1-\alpha_{1,1}-\alpha_{1,2})+(1-\lambda)E^{total}, & \alpha_{1,2}\le 1-\dfrac{(t_{0A}+t_{1A})\alpha_{1,1}}{t_{0A}},\alpha_{1,2}\le\dfrac{t_{0A}(1-\alpha_{1,1})}{t_{0A}+t_{2A}},\\ \lambda t_{1A}\alpha_{1,1}+(1-\lambda)E^{total}, & \alpha_{1,2}\le\dfrac{t_{1A}}{t_{2A}}\alpha_{1,1},\alpha_{1,2}>1-\dfrac{t_{0A}+t_{1A}}{t_{0A}}\alpha_{1,1},\\ \lambda t_{2A}\alpha_{1,2}+(1-\lambda)E^{total}, & \alpha_{1,2}>\dfrac{t_{1A}}{t_{2A}}\alpha_{1,1},\alpha_{1,2}>\dfrac{t_{0A}(1-\alpha_{1,1})}{t_{0A}+t_{2A}}.\end{cases} \tag{43}$$

Consequently, the domain can be partitioned into three regions, as shown in Fig. 3a.

Without loss of generality, we then analyze the case of $\alpha_{1,2}<1-\dfrac{t_{0A}+t_{1A}}{t_{0A}}\alpha_{1,1}$ and $\alpha_{1,2}<\dfrac{t_{0A}}{t_{0A}+t_{2A}}(1-\alpha_{1,1})$. In this case, the system cost becomes,

$$\begin{aligned}\Phi_3(\alpha_{1,1},\alpha_{1,2})&=\lambda t_{0A}(1-\alpha_{1,1}-\alpha_{1,2})+(1-\lambda)E^{total}\\&=\lambda t_{0A}+(1-\lambda)E_{0A}+[(1-\lambda)(E_{1A}-E_{0A})-\lambda t_{0A}]\alpha_{1,1}\\&\quad+[(1-\lambda)(E_{2A}-E_{0A})-\lambda t_{0A}]\alpha_{1,2}.\end{aligned} \tag{44}$$

The corresponding part is shown in Fig. 3b. Then, the corresponding problem optimization can be written as

$$\begin{aligned}\min_{\{\alpha_{1,1},\alpha_{1,2}\}}\quad&\Phi_3(\alpha_{1,1},\alpha_{1,2})-\lambda t_{0A}-(1-\lambda)E_{0A}\\&=[(1-\lambda)(E_{1A}-E_{0A})-\lambda t_{0A}]\alpha_{1,1}+[(1-\lambda)(E_{2A}-E_{0A})-\lambda t_{0A}]\alpha_{1,2}\\ s.t.\quad&C_1:\alpha_{1,2}<1-\frac{t_{0A}+t_{1A}}{t_{0A}}\alpha_{1,1},\\&C_2:\alpha_{1,2}<\frac{t_{0A}}{t_{0A}+t_{2A}}(1-\alpha_{1,1}),\\&C_3:\alpha_{1,1}+\alpha_{1,2}\le 1,\\&C_4:\alpha_{1,1}\ge 0,\alpha_{1,2}\ge 0.\end{aligned} \tag{45}$$

We can observe that (45) is also a linear programming problem and thus it can be effectively solved by using some linear programming methods. Specifically, for the non-integer linear programming, if the optimal solution exists, the optimal solution will be falling at the point of intersection. Consequently, we only need to collect all points of intersection in this region and compare the associated system costs to find the local optimal offloading decision. We can perform a similar operation for the other regions and finally get the global optimal offloading decision.

4 Simulation

In this section, we will demonstrate some simulations to verify the effectiveness of the proposed analytical method for the three special cases including one-to-one, two-to-one, and one-to-two cases. In the following, we will firstly introduce the environment setup of these simulations. Then, we will give some related discussions, based on the simulations.

4.1 Environment setup

The local computational capabilities of the two users are set to $f_1 = 3 \times 10^8$ cyc/s and $f_2 = 2.2 \times 10^8$ cyc/s, respectively, and the computational capabilities at the two CAPs are set to $F_1 = 16 \times 10^8$ cyc/s and $F_2 = 13 \times 10^8$ cyc/s, respectively. In addition, the task sizes of the two users are set to $l_1 = 60$Mb and $l_2 = 120$Mb, respectively, and the variance of the AWGN is set to 0.01. Moreover, the wireless bandwidth for the two CAPs is set to 5 MHz, and the local computational powers of the two users are set to $P_1 = 2$ W and $P_2 = 3$ W, respectively. In further, we simulate the experiments on the Python platform.

4.2 Simulation results and discussions

Figure 4 demonstrates the impact of the weight factor λ on the system cost of the proposed analytical method for the one-to-one case, where the wireless bandwidth is set to 5MHz, the task size of user$_1$ is set to 60Mb, the computational capability of CAP$_1$ is set to 16×10^8 cyc/s, the computational capability of user$_1$ is set to 3×10^8 cyc/s, and the weight factor λ varies from 0.0 to 1.0. For comparison, we also provide the system cost of the BF approach in solving the offloading strategy. From this figure, we can find that for

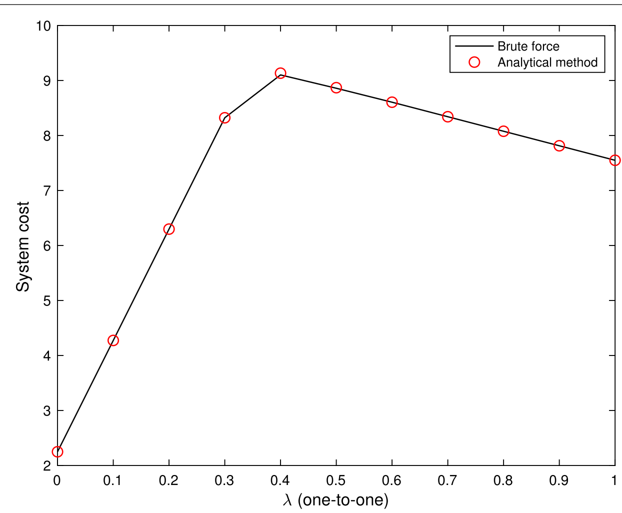


Fig. 4 System cost of the analytical method and BF versus the weight factor λ for the one-to-one case

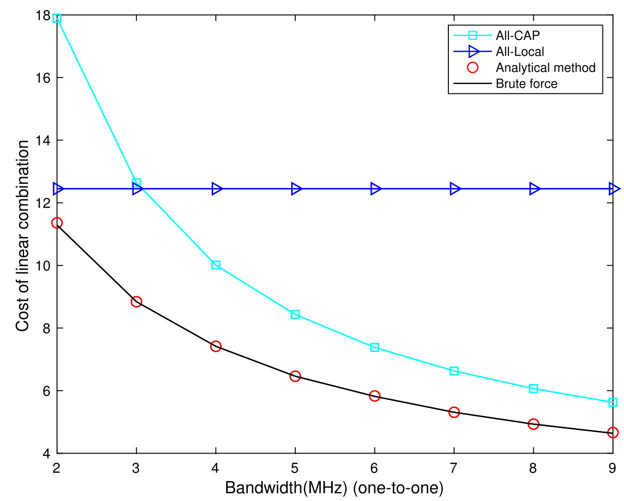


Fig. 5 System cost versus the wireless bandwidth with $\lambda = 0.5$ for the one-to-one case

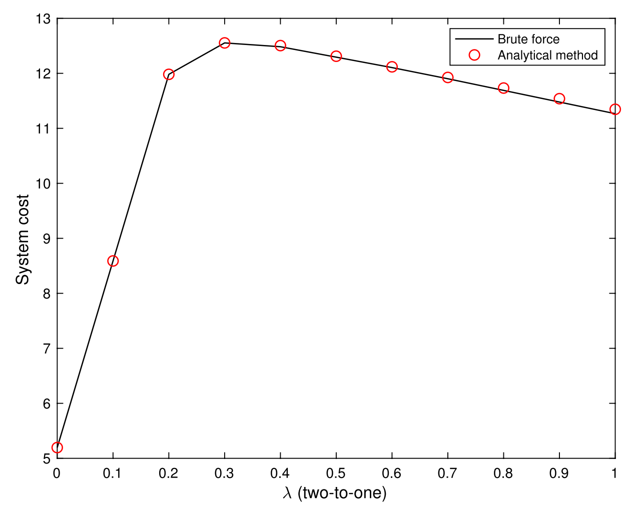


Fig. 6 System cost of the analytical method and BF versus weight factor λ for the two-to-one case

various values of the weight factor λ, the proposed analytical method matches well with the BF approach. As the BF approach gives the optimal offloading ratio for each λ, the proposed analytical method can calculate the optimal offloading ratio for a wide range of λ. Moreover, for the special cases that the system only considers the energy consumption or the latency, i.e., $\lambda = 0$ and $\lambda = 1$, both the proposed analytical method and BF approach can effectively reduce the system cost to about 2.2 and 7.5.

Figure 5 illustrates the impact of the wireless bandwidth on the system cost of different offloading schemes for the one-to-one case, where the weight factor λ is 0.5, the task size of user$_1$ is 60Mb, the computational capability of CAP$_1$ is 16×10^8 cyc/s, the computational capability of user$_1$ is 3×10^8 cyc/s and the wireless bandwidth varies from 2 MHz to 9MHz. Besides the proposed analytical method and BF approach, we also study the system cost of the All-CAP offloading scheme and the All-Local offloading scheme, where the All-CAP offloading scheme and the All-Local offloading scheme indicate that all the tasks are computed at the CAP and local, respectively. We can find from Fig. 5 that the analytical method and the BF approach outperform the All-CAP offloading scheme and the All-Local offloading scheme, indicating that the proposed analytical method can effectively exploit the bandwidth resources to obtain an offloading strategy similar to the BF approach. Moreover, we can observe that for various values of the wireless bandwidth, the system costs of the analytical method and BF are almost the same, which further validates the effectiveness of the proposed analytical method. In further, the system cost of the All-CAP offloading scheme, BF approach and our analytical method is reduced when the wireless bandwidth increases, while the system cost of the All-Local offloading scheme remains unchanged for various wireless bandwidth settings. This is because that the transmission rate increases with a larger bandwidth, which can result in a reduction in both transmission latency and transmission energy consumption, while the All-Local offloading scheme does not involve any data transmission.

Figure 6 shows the system cost of the proposed analytical method versus the weight factor λ varying from 0.0 to 1.0 for the two-to-one case, where the task size of the two users is set to $l_1 = 60$ Mb and $l_2 = 120$ Mb, the wireless bandwidth is set to $W_{1,1} = 5$ MHz and $W_{2,1} = 5$ MHz, and the computational capability of the CAP is set to $F_{1,1} = 7 \times 10^8$cyc/s and $F_{2,1} = 9 \times 10^8$ cyc/s, and the computational capabilities of the two users are set to $f_1 = 3 \times 10^8$ cyc/s and $f_2 = 2.2 \times 10^8$ cyc/s. To demonstrate the correctness of our proposed analytical method, we use the BF approach for comparison. As shown in Fig. 6, we can find that for different values of the weight factor λ, the performance of our proposed analytical method is very close to that of the BF approach, which validates the correctness of the proposed analytical method. Moreover, we can see that when $\lambda \in [0, 0.3]$, the system cost of the BF method and the proposed analytical method increases with a larger λ, since the latency plays a more important role than the energy consumption. In contrast, when $\lambda \in (0.3, 1]$, the

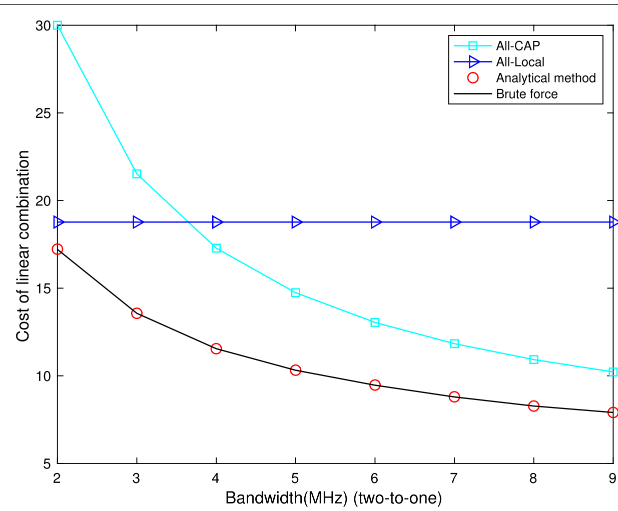


Fig. 7 System cost versus the wireless bandwidth with $\lambda = 0.5$ for the two-to-one case

system cost of BF method and the proposed analytical method decreases with a larger λ, since the energy consumption plays a more important role than the latency.

Figure 7 performs the comparison of four offloading schemes with $\lambda = 0.5$ for the two-to-one case, where the wireless bandwidth varies from 2 MHz to 9MHz, the task sizes of the two users are set to $l_1 = 60$ Mb and $l_2 = 120$ Mb, the wireless bandwidth is set to $W_{1,1} = 5$ MHz and $W_{2,1} = 5$ MHz, the computational capability of the CAP is set to $F_{1,1} = 7 \times 10^8$cyc/s and $F_{2,1} = 9 \times 10^8$ cyc/s, and the computational capabilities of the two users are set to $f_1 = 3 \times 10^8$ cyc/s and $f_2 = 2.2 \times 10^8$ cyc/s. Figure 7 shows that the system cost of All-CAP, analytical method and BF gradually decreases as the wireless bandwidth increases, as the transmission rate of the users become larger, resulting in a reduced system cost. In contrast, the system cost of the All-Local offloading scheme remains unchanged across different bandwidths. Moreover, the performance of the analytical method is better than that of All-CAP and All-local, which indicates the superiority of analytical method. In further, the analytical method and BF have almost the same cost for various values of the wireless bandwidth, indicating that the proposed analytical method can find the optimal offloading strategy.

Figure 8 illustrates the system cost of the proposed analytical method and BF versus the weight factor λ varying from 0.0 to 1.0 for the one-to-two case, where the task size of the user is set to $l_1 = 60$ Mb, the wireless bandwidth is set to $W_{1,1} = 5$ MHz, the computational capability of the CAP is set to $F_{1,1} = 7 \times 10^8$ cyc/s, and the local computational capability is set to $f_1 = 3 \times 10^8$ cyc/s. We can see from Fig. 8 that for various values of λ, the system cost of the analytical method and BF matches well. This also indicates that the performance of the analytical method is similar to that of the BF. In addition, we can find that when $\lambda = 0.2$, the system cost of the BF approach and the proposed analytical method is higher than that with other values of λ, indicating that the linear combination of latency and energy consumption at this point will dramatically affect the system performance.

Figure 9 depicts the system cost of four offloading strategies versus the wireless bandwidth with $\lambda = 0.5$ for the one-to-two case, where the bandwidth varies from 2

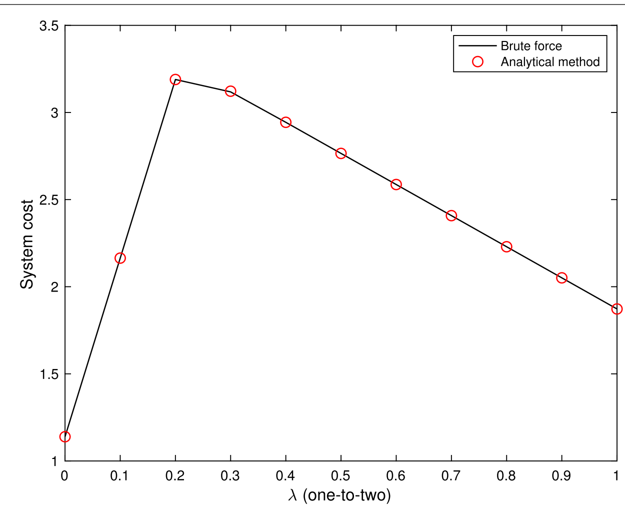


Fig. 8 System cost of the analytical method and BF versus weight factor λ for the one-to-two case

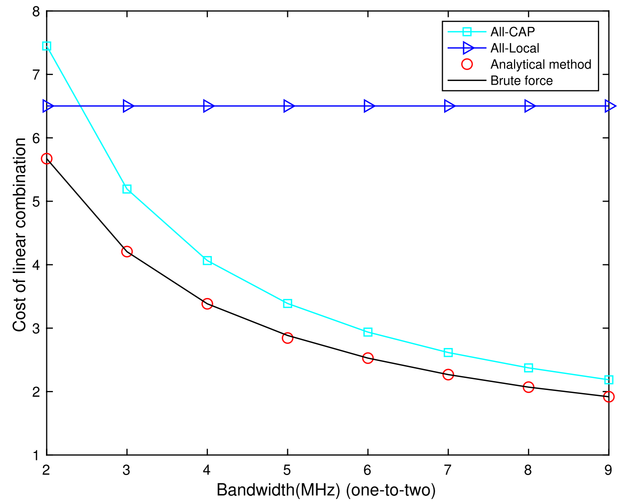


Fig. 9 System cost versus the wireless bandwidth with $\lambda = 0.5$ for the one-to-two case

MHz to 9MHz, the task size of the user is set to $l_1 = 60$ Mb, the computational capability of the CAP is set to $F_{1,1} = 7 \times 10^8$ cyc/s, and the local computational capability is set to $f_1 = 3 \times 10^8$ cyc/s. By observing Fig. 9, we can find that the system cost of the All-CAP offloading scheme, the analytical method, and the BF approach decreases with a larger bandwidth, while the system cost of the All-Local remains unchanged. This is because that the transmission rate of the All-CAP offloading scheme, the analytical method, and the BF approach is significantly affected by the wireless bandwidth, while All-Local performs the calculation locally, and it is not affected by the wireless resources. In addition, as the bandwidth increases, the downward trends of All-CAP, analytical method, and BF all decrease. This indicates that the improvement is gradually becoming saturated. Most importantly, it is evident from Fig. 9 that the analytical method and BF are consistently lowest among the four offloading strategies, which further illustrates the effectiveness of the proposed analytical method. Moreover, the system cost of proposed analytical method is similar to that of the BF approach, indicating that the performance of the proposed analytical method can achieve the optimal performance of the BF approach.

5 Conclusion

In this paper, we investigated how to analytically design an analytical offloading strategy for a multiuser vehicular MEC network, where there were multiple CAPs which could help compute tasks from the users. As it is difficult to derive an analytical offloading ratio for a general vehicular MEC network, we turned to provide an analytical offloading scheme for some special vehicular MEC networks including one-to-one, two-to-one and one-to-two cases. For each case, we studied the system performance by using the linear combination of latency and energy consumption, and derived the analytical offloading ratio through minimizing the system cost. Simulation results were finally presented to verify the proposed studies. In particular, the proposed analytical offloading scheme can achieve the optimal performance of the BF scheme.

The analytical results in this paper can serve as an important reference for the analytical offloading design for a general vehicular MEC network. Moreover, analysis of the more general MEC network can serve more application scenarios. Motivated by this, we will discussed the analytical method for a more general MEC network in the future works.

Abbreviations
MEC: Mobile edge computing; IoV: Internet of vehicle; CAP: Computational access point; BF: Brutee force; MIMO: Multiple-input multiple-output; IRS: Intelligent reflecting surface; CSI: Channel state information; DQN: Deep Q-network; AWGN: Additive white Gaussian noise; one-to-one: One user to one CAP; two-to-one: Two user to one CAP; one-to-two: One user to two CAP; LP: Linear programming; OFDM: Orthogonal frequency division multiplexing.

Acknowledgements
This work was supported by the Key-Area Research and Development Program of Guangdong Province (No. 2019B090904014), by the International Science and Technology Cooperation Projects of Guangdong Province (No. 2020A0505100060), by the Natural Science Foundation of Guangdong Province (No. 2021A1515011392), and in part by the research program of Guangzhou University (No. YJ2021003).

Author Contributions
J. Lu designed the proposed framework and performed the simulations, L. Chen helped improve the optimization method, J. Xia helped revise the manuscript in both the structure and grammar check, M. Tang helped conduct the simulations, F. Zhu helped enhance the design of the analytical method, C. Fan helped explained the simulation results in this paper, and J. Ou helped clarify the main contribution and novelty of this paper. J. Xia, F. Zhu and M. Tang are the corresponding authors of this paper. All authors read and approved the final manuscript.

Funding
This work was supported by the Key-Area Research and Development Program of Guangdong Province (No. 2019B090904014), by the International Science and Technology Cooperation Projects of Guangdong Province (No. 2020A0505100060), by the Natural Science Foundation of Guangdong Province (No. 2021A1515011392), and in part by the research program of Guangzhou University (No. YJ2021003).

Availability of data and materials
The authors state the data availability in this manuscript.

Declarations

Competing interests
The authors declare that they have no competing interests.

Ethics approval and consent to participate
Not applicable.

Author details
[1]School of Computer Science, Guangzhou University, Guangzhou, China. [2]Guangdong New Generation Communication and Network Innovative Institute (GDCNi), Guangzhou, China. [3]AI Sensing Technology, Chancheng District, Foshan, China.

Received: 13 January 2022 Accepted: 28 March 2022
Published online: 23 May 2022

References
1. B. Wang, S. Jin, H. Lin, G.Y. Li, Spatial- and frequency-wideband effects in millimeter-wave massive MIMO systems. IEEE Trans. Signal Process. **66**(13), 3393–3406 (2018)
2. X. Liu, X. Zhang, Noma-based resource allocation for cluster-based cognitive industrial internet of things. IEEE Trans. Ind. Inf. **16**(8), 5379–5388. https://doi.org/10.1109/TII.2019.2947435
3. X. Hu, C. Zhong, Y. Zhu, X. Chen, Z. Zhang, Programmable metasurface-based multicast systems: design and analysis. IEEE J. Sel. Areas Commun. **38**(8), 1763–1776 (2020)
4. Y. Xu, C. Shen, D. Cai, G. Zhu, Latency constrained non-orthogonal packets scheduling with finite blocklength codes. IEEE Trans. Veh. Technol. **69**(10), 12312–12316 (2020)
5. X. Liu, X.B. Zhai, W. Lu, C. Wu, Qos-guarantee resource allocation for multibeam satellite industrial internet of things with NOMA. IEEE Trans. Ind. Inf. **17**(3), 2052–2061 (2021). https://doi.org/10.1109/TII.2019.2951728
6. Z. Na, B. Li, X. Liu, J. Wan, M. Zhang, Y. Liu, B. Mao, UAV-based wide-area internet of things: an integrated deployment architecture. IEEE Netw. **35**(5), 122–128 (2021)
7. X. Liu, X. Zhang, Rate and energy efficiency improvements for 5G-based IoT with simultaneous transfer. IEEE Internet Things J. **6**(4), 5971–5980 (2019). https://doi.org/10.1109/JIOT.2018.2863267

8. Z. Na, J. Lv, F. Jiang, M. Xiong, N. Zhao, Joint subcarrier and subsymbol allocation-based simultaneous wireless information and power transfer for multiuser GFDM in IoT. IEEE Internet Things J. **6**(4), 5999–6006 (2019)
9. L. He, K. He, Towards optimally efficient search with deep learning for large-scale MIMO systems. IEEE Trans. Commun. (99), 1–12 (2022)
10. K. He, Y. Deng, Efficient memory-bounded optimal detection for GSM-MIMO systems. IEEE Trans. Commun. (99), 1–12 (2022)
11. X. Hu, J. Wang, Statistical CSI based design for intelligent reflecting surface assisted MISO systems. Sci. China Inf. Sci. **63**(12), 222303 (2020)
12. J. Zhang, Y. Zhang, Z. Zhang, Robust design for intelligent reflecting surfaces assisted MISO systems. IEEE Commun. Lett. **24**(10), 2353–2357 (2020)
13. H. Xie, F. Gao, S. Zhang, S. Jin, A unified transmission strategy for TDD/FDD massive MIMO systems with spatial basis expansion model. IEEE Trans. Veh. Technol. **66**(4), 3170–3184 (2017)
14. R. Zhao, M. Tang, Impact of direct links on intelligent reflect surface-aided MEC networks. Phys. Commun. (99), 1–10 (2022)
15. X. Hu, Y. Zhang, X. Chen, Z. Zhang, Location information aided multiple intelligent reflecting surface systems. IEEE Trans. Commun. **68**(12), 7948–7962 (2020)
16. Y. Wu, M. Tang, Intelligent task offloading for vehicular edge computing with imperfect CSI: a deep reinforcement approach. Phys. Commun. (99), 1–10 (2022)
17. Q. Tao, J. Wang, Performance analysis of intelligent reflecting surface aided communication systems. IEEE Commun. Lett. **24**(11), 2464–2468 (2020)
18. X. Lai, Secure mobile edge computing networks in the presence of multiple eavesdroppers. IEEE Trans. Commun. 1–12 (2021)
19. X. Liu, X. Zhang, M. Jia, L. Fan, W. Lu, X.B. Zhai, 5G-based green broadband communication system design with simultaneous wireless information and power transfer. Phys. Commun. **28**, 130–137 (2018). https://doi.org/10.1016/j.phycom.2018.03.015
20. Z. Na, Y. Liu, J. Shi, C. Liu, Z. Gao, UAV-supported clustered NOMA for 6G-enabled internet of things: trajectory planning and resource allocation. IEEE Internet Things J. **8**(20), 15041–15048 (2021)
21. S. Tang, Dilated convolution based CSI feedback compression for massive MIMO systems. IEEE Trans. Veh. Tech. PP(99), 1–5 (2021)
22. Z. Zhao, System optimization of federated learning networks with a constrained latency. IEEE Trans. Veh. Tech. PP(99), 1–5 (2021)
23. L. Chen, Intelligent ubiquitous computing for future UAV-enabled MEC network systems. Cluster Comput. **2021**(25), 1–10 (2021)
24. R. Zhao, M. Tang, Profit maximization in cache-aided intelligent computing networks. Phys. Commun. (99), 1–10 (2022)
25. L. Zhang, C. Gao, Deep reinforcement learning based IRS-asisted mobile edge computing under physical-layer security. Phys. Commun. (99), 1–10 (2022)
26. X. Lai, Outdated access point selection for mobile edge computing with cochannel interference. IEEE Trans. Veh. Tech. (99), 1–5 (2021)
27. L. Chen, Physical-layer security on mobile edge computing for emerging cyber physical systems. Comput. Commun. (99), 1–12 (2022)
28. S. Tang, L. Chen, Computational intelligence and deep learning for next-generation edge-enabled industrial IoT. IEEE Trans. Netw. Sci. Eng. (99), 1–12 (2022)
29. L. Fan, X. Lei, Learning-based MIMO detection with dynamic spatial modulation. Sci. China Inf. Sci. (99), 1–10 (2022)
30. J. Lu, M. Tang, Performance Analysis for IRS-Assisted MEC networks with Unit Selection. Phys. Commun. (99), 1–10 (2022)
31. Y. Guo, S. Lai, Distributed machine learning for multiuser mobile edge computing systems. IEEE J. Sel. Top. Signal Process. (99), 1–12 (2021)
32. L. Zhang, DQN based mobile edge computing for smart internet of vehicle. EURASIP J. Adv. Signal Process. (99), 1–10 (2022)
33. F. Li, K. Lam, X. Liu, J. Wang, K. Zhao, L. Wang, Joint pricing and power allocation for multibeam satellite systems with dynamic game model. IEEE Trans. Veh. Technol. **67**(3), 2398–2408 (2018). https://doi.org/10.1109/TVT.2017.2771770

Publisher's Note