

## PAPER

# Joint Optimization of Task Offloading and Resource Allocation for UAV-Assisted Edge Computing: A Stackelberg Bilayer Game Approach

Peng WANG<sup>†a)</sup>, Student Member, Guifen CHEN<sup>†b)</sup>, and Zhiyao SUN<sup>†c)</sup>, Nonmembers

**SUMMARY** Unmanned Aerial Vehicle (UAV)-assisted Mobile Edge Computing (MEC) can provide mobile users (MU) with additional computing services and a wide range of connectivity. This paper investigates the joint optimization strategy of task offloading and resource allocation for UAV-assisted MEC systems in complex scenarios with the goal of reducing the total system cost, consisting of task execution latency and energy consumption. We adopt a game theoretic approach to model the interaction process between the MEC server and the MU Stackelberg bilayer game model. Then, the original problem with complex multi-constraints is transformed into a duality problem using the Lagrangian duality method. Furthermore, we prove that the modeled Stackelberg bilayer game has a unique Nash equilibrium solution. In order to obtain an approximate optimal solution to the proposed problem, we propose a two-stage alternating iteration (TASR) algorithm based on the subgradient method and the marginal revenue optimization method. We evaluate the effective performance of the proposed algorithm through detailed simulation experiments. The simulation results show that the proposed algorithm is superior and robust compared to other benchmark methods and can effectively reduce the task execution latency and total system cost in different scenarios.

**key words:** edge computing, Stackelberg game, Nash equilibrium, task offloading, resource allocation

## 1. Introduction

The increasing number of mobile devices brings about a large increase in data traffic, which poses a great challenge to the performance of traditional networks [1]. MEC technology can well alleviate the computational pressure on the center server by deploying the server at the edge of the network. As it is closer to the mobile devices, it can better fulfill the demand of low latency. Most of the existing research work relies on fixing MEC servers in terrestrial base stations (BS) [2], [3], however, terrestrial BSs are expensive to deploy, fixed in location, and have limited coverage, which prevents them from providing reliable communication services in remote areas as well as emergency scenarios such as large-scale concerts. With low deployment cost, easy maneuvering, high mobility and flexibility, UAV can be rapidly deployed as a service node in the network to guarantee communication services in emergency scenarios [4].

UAV-assisted MEC can reduce the connection latency of the edge network, and improve the flexibility and robustness of the edge network [5]. UAV can be used as both a relay node in the edge network to help IoT devices that cannot access the edge server directly to forward computations. The UAV can either act as a relay node to help IoT devices forward computational tasks [6] or carry MEC servers to provide computational services for mobile MU terminals with limited computational capabilities [7].

The use of game theory to incentivize parties to participate in computation offloading has been widely studied. Wang M et al. [8] considered a scenario in which a terrestrial BS is damaged and unable to provide computation services, and by overlaying UAVs into the network of the damaged BSs, they constructed a Stackelberg game model to solve the problem of minimising energy consumption. Zhou H et al. [9] modeled the interaction between a mobile MU and an edge interaction between mobile MUs and service providers modeled as a Stackelberg game model and proposed a gradient-based iterative search algorithm to obtain an approximate optimal solution with the objective of maximizing their utility. Although the combined air-ground edge computing approach has received a lot of attention, the limitation of computational resources and energy supply for UAVs is a challenge for existing work [10], [11]. In addition, the research on multi-UAV resource allocation in complex environments in existing work needs to be further deepened and improved.

We consider a layered system for UAV-assisted edge computing in multi-MU, multi-UAV, and multi-BS scenarios and model it as a multi-leader, multi-follower Stackelberg bilayer game model, where the UAVs act as leader and follower, respectively, in the bilayer game. An alternating iteration algorithm is then proposed to solve the problem of minimising the total cost of the system. The main contributions of this paper are as follows: (1) We study the joint optimisation problem of computational offloading and resource allocation for UAV-assisted MEC in a complex scenario with multiple MUs, multiple UAVs, and BSs hosting MEC servers. (2) We develop a bilayer model of the multi-leader, multi-follower Stackelberg game with the optimisation objective of minimising the total system cost of task execution. We utilise the Lagrange duality method to transform the complex, constrained problem. (3) We prove that the established game model has a unique Nash equilibrium point. In order

Manuscript received December 8, 2023.

Manuscript revised March 25, 2024.

Manuscript publicized May 21, 2024.

<sup>†</sup>College of Electronic Information Engineering, Changchun University of Science and Technology, China.

a) E-mail: wangpeng990226@163.com

b) E-mail: 2022100736@mails.cust.edu.cn

c) E-mail: sunzhiyao@mails.cust.edu.cn

DOI: 10.1587/transinf.2023EDP7261

to obtain an approximate optimal solution to the proposed problem, a TAsR algorithm is proposed. (4) Simulation results show that the TAsR algorithm can make the bilayer game converge to a Nash equilibrium with some robustness. In addition, compared with the benchmark method, TAsR has certain superiority under different numbers of MUs.

## 2. System Model

We consider a UAV-assisted MEC task offloading system. As shown in Fig. 1, the system consists of MUs and BSs located on the ground and UAVs located in the air. The UAV and BS are equipped with MEC servers from different service providers with independent internal communication and computation modules. Considering the situation that the user and the BS cannot communicate effectively with each other due to factors such as obstacle blockage, the UAV can be used as a relay node and computation server for forwarding tasks. When the MU has a demand for task computation, a partial offloading mode is used to offload a part of the task to the UAV and the other part is computed locally by the MU. The UAV subdivides the received task into two parts: one part is offloaded to the BS for computation and paid to the BS for computation of rewards; the other part is computed locally by the UAV and receives rewards paid by the MU.

### 2.1 Computational Model

The UAV-assisted MEC system contains  $B$  BS ( $B = \{1, 2, \dots, B\}$ ),  $K$  UAVs ( $K = \{1, 2, \dots, K\}$ ), and  $N$  MUs ( $N = \{1, 2, \dots, N\}$ ). Denote the task size of the  $MU_i$  by  $D_i$  ( $i \in \{1, 2, \dots, N\}$ ), and  $T_i^{\max}$  denotes the maximum tolerance time of the task, assuming that there is one and only one task demanded by an MU during the time of  $T_i^{\max}$ .  $C$  denotes the number of CPU cycles required by the device to compute the 1-bit data.

The latency and energy consumption of a task computed locally are expressed as

$$t_i^{MU} = \frac{(1 - \alpha_i)CD_i}{f_i^{MU}} \quad (1)$$

$$e_i^{MU} = \gamma^{MU}(1 - \alpha_i)CD_i(f_i^{MU})^2 \quad (2)$$

Where  $f_i^{MU}$  denotes the local computational capacity,  $\alpha_i$  denotes the ratio of tasks offloading processing, and  $\gamma^{MU}$  denotes the hardware-related coefficient.  $\beta_{i,j}$  denotes the percentage of received tasks that the UAV offloads to the BS. The latency and energy consumption required by the UAV

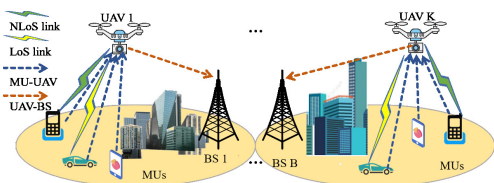


Fig. 1 System model

to compute the task are denoted by

$$t_{i,j}^{UAV} = \frac{\alpha_i(1 - \beta_{i,j})CD_i}{f_{i,j}^{UAV}} \quad (3)$$

$$e_{i,j}^{UAV} = \gamma^{UAV} \alpha_i(1 - \beta_{i,j})CD_i(f_{i,j}^{UAV})^2 \quad (4)$$

Where  $f_{i,j}^{UAV}$  denotes the computational resources allocated to the task by UAV. Let the computational resource assigned to the task by BS be  $f_{i,j,b}^{BS}$  ( $b \in \{1, 2, \dots, B\}$ ), then the latency and energy consumption of the task computed by the BS are

$$t_{i,j,b}^{BS} = \frac{\alpha_i\beta_{i,j}CD_i}{f_{i,j,b}^{BS}} \quad (5)$$

$$e_{i,j,b}^{BS} = \gamma^{BS} \alpha_i\beta_{i,j}CD_i(f_{i,j,b}^{BS})^2 \quad (6)$$

### 2.2 Transmission Model

There may be obstacle occlusion between the UAV and the MU or BS, resulting in additional path loss. According to the 3rd Generation Partnership Project (3GPP) technical specification [12], the probability that the UAV communicates with the user with a line-of-sight (LoS) link is

$$P_{i,j}^{LoS} = \begin{cases} 1, & d_{i,j}^{2D} \leq d_1 \\ \frac{d_1}{d_{i,j}^{2D}} + \exp\left(\frac{-d_{i,j}^{2D}}{p_1}\right) \left(1 - \frac{d_1}{d_{i,j}^{2D}}\right), & d_{i,j}^{2D} > d_1 \end{cases} \quad (7)$$

Where  $d_{i,j}^{2D}$  denotes the two-dimensional distance between the UAV and the user or the base station,  $p_1 = 233.98 \log_{10}(h_j^{uav}) - 0.95$  and  $d_1 = \max(294.05 \log_{10}(h_j^{uav}) - 432.94, 18)$  are the modelling parameters of the 3GPP specifications. The  $h_j^{uav}$  denotes the height of the UAV. Therefore, the transmission speed of the offloading task from the MU to the UAV can be obtained as

$$R_{i,j}^{M-U} = B_{i,j}^{M-U} \log_2 \left( 1 + \frac{P_{i,j}^{M-U} g_{i,j}^{M-U}}{N_0^2 + (1 - P_{i,j}^{LoS})N_{NL}} \right) \quad (8)$$

Where  $B_{i,j}^{M-U}$  denotes the channel bandwidth between the MU and the UAV,  $g_{i,j}^{M-U} = (d_{i,j}^{M-U})^{-\eta}$  denotes the channel gain, and  $d_{i,j}^{M-U}$  is the 3D distance between the MU and the UAV.  $\eta$  is the path loss exponent,  $N_{NL}$  denotes the additional loss due to the non-line-of-sight (NLoS) link, and  $N_0^2$  denotes the noise power. Similarly, the transmission rate of the UAV to the BS offloading task is denoted as

$$R_{j,b}^{U-B} = B_{j,b}^{U-B} \log_2 \left( 1 + \frac{P_{j,b}^{U-B} g_{j,b}^{U-B}}{N_0^2 + (1 - P_{j,b}^{LoS})N_{NL}} \right) \quad (9)$$

The latency and energy consumption incurred during task transmission can be calculated

$$t_{i,j}^{M-U} = \frac{\alpha_i D_i}{R_{i,j}^{M-U}} \quad (10)$$

$$e_{i,j}^{M-U} = p_{i,j}^{M-U} t_{i,j}^{M-U} \quad (11)$$

$$t_{i,j,b}^{U-B} = \frac{\alpha_i \beta_{i,j} D_i}{R_{j,b}^{U-B}} \quad (12)$$

$$e_{i,j,b}^{U-B} = p_{j,b}^{U-B} t_{i,j,b}^{U-B} \quad (13)$$

In general, the amount of data in the result download phase is much smaller than the amount of data unloaded by the task, so the latency and energy consumption incurred in the result download phase are ignored. The specific formulas for the task offloading time  $t_i^{off}$  and the total task completion time  $t_i^{all}$  are as follows:

$$t_i^{off} = t_{i,j}^{M-U} + \max \left\{ t_{i,j}^{UAV}, t_{i,j,b}^{U-B} + t_{i,j,b}^{BS} \right\} \quad (14)$$

$$t_i^{all} = \max \left\{ t_i^{MU}, t_i^{off} \right\} \quad (15)$$

### 3. Stackelberg Game Model

In the three-party game, the revenue of UAV is not only limited by the offloading decision of MU but also by the resource pricing of BS. In order to effectively utilize the computational resources of the UAV, the overall game problem is decoupled into the MU-UAV game problem and the UAV-BS game problem.

#### 3.1 MU-UAV Game Model

In the MU-UAV game model, the UAV serves as the leader and provides resource allocation and pricing strategies to the MU, which functions as the follower and offers offloading strategies to the UAV. The MU is focused on minimising the latency and integrated cost of task execution, while the UAV aims to maximise revenue with its limited computing resources. The cost of MU is expressed as follows:

$$P1 : \min U_{mu}^{M-U} = M_i + \mu_1 t_i^{all} + \mu_2 (e_i^{MU} + e_{i,j}^{M-U}) \quad (16)$$

$$s.t. 0 \leq t_i^{all} \leq T_i^{\max} \quad (16a)$$

$$0 \leq \alpha_i \leq 1 \quad (16b)$$

Where  $\mu_1$  and  $\mu_2$  are weight parameters to eliminate the difference in the values of latency cost, energy consumption costs, and monetary cost.  $M_i$  denotes the monetary cost of the MU. Constraint (16a) indicates that the time to complete the task should satisfy the maximum time latency requirement of the task. Constraint (16b) denotes that the task offloading ratio should be between 0 and 1. The utility of UAV is expressed as follows:

$$P2 : \max U_{uav}^{M-U} = \sum_{i=1}^N (m_j f_{i,j}^{UAV} - \mu_2 e_{i,j}^{UAV}) \quad (17)$$

$$s.t. m_j f_{i,j}^{UAV} = M_i \quad (17a)$$

$$0 \leq f_{i,j}^{UAV} \leq F_{\max}^{UAV} \quad (17b)$$

$$m_j > 0 \quad (17c)$$

Where  $m_j$  denotes the uniform unit price established by UAV <sub>$j$</sub>  ( $j \in K$ ) for the computational resource. Constraint (17a) represents the monetary cost of the MU in relation to the computational resources and unit prices. Constraint (17b) indicates that the computational resources allocated by the UAV for a task should be less than the maximum computational resources of the UAV. Constraint (17c) indicates that resource pricing should be positive.

#### 3.2 UAV-BS Game Model

In the UAV-BS game model, BS as a leader provides resource allocation strategy and resource pricing strategy to UAV, and UAV as a follower provides offloading strategy to BS. The cost of a UAV is represented as follows:

$$P3 : \min U_{uav}^{U-B} = \sum_{i=1}^N (q_b f_{i,j,b}^{BS} + \mu_2 e_{i,j,b}^{U-B}) \quad (18)$$

$$s.t. 0 \leq \beta_{i,j} \leq 1 \quad (18a)$$

Where  $q_b$  denotes the uniform unit price established by BS <sub>$b$</sub>  ( $b \in B$ ) for the computational resources, and the constraint (18a) indicates that the task offloading ratio should be between 0 and 1. The BS needs a reasonable pricing strategy and resource allocation strategy to incentivize UAVs to offload tasks to it, and the BS utility is expressed as follows:

$$P4 : \max U_{bs}^{U-B} = \sum_{i=1}^N \sum_{j=1}^K (q_b f_{i,j,b}^{BS} - \mu_2 e_{i,j,b}^{BS}) \quad (19)$$

$$s.t. \mu_2 e_{i,j,b}^{U-B} + q_b f_{i,j,b}^{BS} \leq \mu_2 \gamma^{UAV} \alpha_i \beta_{i,j} D_i C (f_{i,j}^{UAV})^2 \quad (19a)$$

$$0 \leq f_{i,j,b}^{BS} \leq F_{\max}^{BS} \quad (19b)$$

$$q_b > 0 \quad (19c)$$

Where constraint (19a) ensures that the UAV's own interests are not jeopardised. The constraint (19b) indicates that the computational resources allocated by the BS for the task should not be greater than the maximum amount of computational resources. Constraint (19c) indicates that the unit price of resources should be positive.

Problem P4 is a standard convex optimisation problem when the offloading policy of the MU is known as well as the offloading policy and resource allocation policy of the UAV. However, in the actual solution, the limitations of constraint (19a) must be considered. In order to obtain a more efficient solution design, we transform problem P4 using the Lagrange duality method. Let  $\lambda_{i,j} \geq 0$  ( $i \in N, j \in K$ ) denote the duality variables corresponding to the constraint (19a), and the partial Lagrange expression for problem P4 is as follows:

$$\begin{aligned} L(\lambda_{i,j}, q_b, f_{i,j,b}^{BS}) \\ = \sum_{i=1}^N \sum_{j=1}^K [q_b f_{i,j,b}^{BS} - \mu_2 e_{i,j,b}^{BS} \end{aligned}$$

$$\begin{aligned}
& - \lambda_{i,j}(\mu_2 e_{i,j,b}^{U-B} + q_b f_{i,j,b}^{BS} \\
& - \mu_2 \gamma^{UAV} \alpha_i \beta_{i,j} D_i C (f_{i,j}^{UAV})^2] \quad (20)
\end{aligned}$$

Problem P4 can be rephrased as

$$\begin{aligned}
P4.1: \quad & \max_{q_b, f_{i,j,b}^{BS}} \min_{\lambda} L(\lambda_{i,j}, q_b, f_{i,j,b}^{BS}) \quad (21) \\
s.t. \quad & \lambda_{i,j} \geq 0, (19b), (19c)
\end{aligned}$$

The duality function of problem P4 is expressed as

$$g(\lambda_{i,j}, q_b, f_{i,j,b}^{BS}) = \max_{q_b, f_{i,j,b}^{BS}} L(\lambda_{i,j}, q_b, f_{i,j,b}^{BS}) \quad (22)$$

Further, the duality problem for P4 can be obtained

$$\begin{aligned}
P4.2: \quad & \min_{\lambda} g(\lambda_{i,j}, q_b, f_{i,j,b}^{BS}) \quad (23) \\
s.t. \quad & \lambda_{i,j} \geq 0, (19b), (19c)
\end{aligned}$$

As long as  $q_b$  and  $f_{i,j,b}^{BS}$  are set to sufficiently small positive values so that the constraints (19a) satisfy the strict inequality, there is always a feasible interior solution to Problem P4, and thus the Slater condition holds [13], i.e., the original Problem P4 has strong duality with the duality Problem P4.2, and solving the duality Problem P4.2 is equivalent to solving problem P4. Therefore, we first maximize the Lagrange function given the value of  $\lambda_{i,j}$  to obtain the expression of the duality function, and then obtain the optimal solution of the duality variable  $\lambda_{i,j}^*$  by minimizing the duality function. An approximate optimal solution to the original problem can be obtained based on the optimal duality variables.

## 4. Nash Equilibrium Analysis of Games

### 4.1 Analysis of the UAV-BS Game

**Theorem 1:** For each fixed  $q_b$ , the Nash equilibrium of the UAV-BS game always exists and is unique. The optimal strategy is denoted as

$$f_{i,j,b}^{BS*} = \frac{(1 + \lambda)q_b}{2\mu_2\gamma^{BS}\alpha_i\beta_{i,j}CD_i} \quad (24)$$

$$\beta_{i,j}^* = \frac{q_b}{\mu_2\alpha_i D_i} \sqrt{\frac{(1 + \lambda)R_{j,b}^{U-B}}{2\gamma^{BS}C\rho_{j,b}^{U-B}}} \quad (25)$$

**Proof:** Using the KKT condition, the optimal closed-form analytic form of the optimization variable  $f_{i,j,b}^{BS}$  in the duality problem P4.2 is obtained by making the first-order partial derivatives of Eq. (20) with respect to  $f_{i,j,b}^{BS}$  to be zero while fixing  $\lambda$ , i.e., Eq. (24). Bringing  $f_{i,j,b}^{BS*}$  into the utility function in Eq. (18) yields  $\frac{\partial^2 U}{\partial(\beta_{i,j})^2} > 0$ , which indicates that the utility function is a convex function on  $\beta_{i,j}$  as a convex function. Let  $\frac{\partial U}{\partial\beta_{i,j}} = 0$  to obtain the optimal offloading policy for UAV, i.e., Eq. (25).

Moreover, Eq. (20) is continuous with respect to  $q_b$  and has zero second-order partial derivatives. Since the set of players involved in the game is finite and the strategy space of the game belongs to a bounded, nonempty convex set on the Euclidean space. Therefore, for each fixed  $q_b$ , the UAV-BS game is a convex multi-player game [13] with a unique Nash equilibrium [14]. With fixed  $\lambda_{i,j}$ , the optimal set of strategies  $\{\beta_{i,j}^*, q_b^*, f_{i,j,b}^{BS*}\}$  brings the UAV-BS game to the Nash equilibrium.

### 4.2 Analysis of the MU-UAV Game

**Theorem 2:** For each fixed  $m_j$ , the Nash equilibrium of the MU-UAV game always exists and is unique. The optimal strategy is denoted as

$$f_{i,j}^{UAV*} = \frac{m_j}{2\mu_2\gamma^{UAV}\alpha_i(1 - \beta_{i,j})CD_i} \quad (26)$$

$$\alpha_i^* = \begin{cases} \alpha_i^{(1)} & t_{i,j}^{UAV} \geq t_{i,j}^{U-B} + t_{i,j}^{BS} \\ \max\{\alpha_i^{(2)}, \alpha_i^{(3)}\} & t_{i,j}^{UAV} < t_{i,j}^{U-B} + t_{i,j}^{BS} \end{cases} \quad (27)$$

**Proof:** Solving for the second-order partial derivative of  $U_{uav}^{M-U}$  with respect to  $f_{i,j}^{UAV}$  yields  $\frac{\partial^2 U}{\partial(f_{i,j}^{UAV})^2} < 0$ . It shows that there exists a unique  $f_{i,j}^{UAV*}$  that maximises  $U_{uav}^{M-U}$ , such that  $\frac{\partial U}{\partial f_{i,j}^{UAV}} = 0$ , and obtains the optimal resource allocation policy, i.e., Eq. (26).

According to Eq. (15), the total task execution time  $t_i^{all} = t_i^{MU}$  when  $t_i^{MU} \geq t_i^{off}$ . At this point, calculating the first-order partial derivative of  $U_{mu}^{M-U}$  with respect to  $\alpha_i$  yields

$$\begin{aligned}
\frac{\partial U}{\partial\alpha_i} &= \frac{\mu_2 p_{i,j}^{M-U} D_i}{R_{i,j}^{M-U}} - \frac{m_j^2}{2\alpha_i^2 \mu_2 \gamma^{UAV} (1 - \beta_{i,j}) CD_i} \quad (28) \\
& - \frac{\mu_1 CD_i}{f_i^{MU}} - \mu_2 \gamma^{MU} CD_i (f_i^{MU*})^2
\end{aligned}$$

Since the transmission energy consumption of a task with the same number of tasks is less than the computation energy consumption,  $\frac{\partial U}{\partial\alpha_i} < 0$ . It shows that  $U_{mu}^{M-U}$  is monotonically decreasing with respect to  $\alpha_i$ , so that  $U_{mu}^{M-U}$  obtains a minimum at  $t_i^{MU} = t_i^{off}$  when  $t_i^{MU} \geq t_i^{off}$ .

Let  $t_i^{MU} = t_i^{off} = t_{i,j}^{M-U} + t_{i,j}^{UAV}$  and solve the offloading strategy as follows:

$$\alpha_i^{(1)} = \frac{m_j}{2\mu_2 D} \left( \sqrt{\frac{4\mu_2}{m_j} \left( \frac{C}{f_i^{MU}} - \frac{K^2}{\mu_2 D m_j} \right) + H_1^2} + H_1 \right) \quad (29)$$

Where  $H_1 = \frac{2\Phi}{m_j} - \frac{C}{f_i^{MU}} - \frac{1}{R_{i,j}^{M-U}}$ ,  $\Phi = q_b \sqrt{\frac{(1+\lambda)R_{j,b}^{U-B}}{2\gamma^{BS}C\rho_{j,b}^{U-B}}}$ .

Let  $t_i^{MU} = t_i^{off} = t_{i,j}^{M-U} + t_{i,j}^{U-B} + t_{i,j}^{BS}$  and solve the offloading strategy as follows:

$$\alpha_i^{(2)} = \frac{f_i^{MU} R_{j,b}^{U-B}}{f_i^{MU} + CR_{j,b}^{U-B}} \left( H_2 - \frac{2q_b CR_{j,b}^{U-B}}{\mu_2 DP_{j,b}^{U-B}} \right) \quad (30)$$

Where  $H2 = \frac{C}{f_i^{MU}} - \frac{\Phi}{\mu_2 DR_{j,b}^{U-B}}$ .

When  $t_i^{MU} \leq t_i^{off}$  and  $t_{i,j}^{UAV} \geq t_{i,j}^{U-B} + t_{i,j}^{BS}$ ,  $t_i^{all} = t_{i,j}^{M-U} + t_{i,j}^{UAV}$ . Bring the constraint (17a) to  $U_{mu}^{M-U}$  and then compute the second-order partial derivative with respect to  $M_i$  to obtain  $\frac{\partial^2 U}{\partial (M_i)^2} > 0$ . Making the first-order partial derivative equal to zero yields  $M_i^* = \sqrt{\mu_1 \alpha_i (1 - \beta_{i,j}) CD_i m_j}$ . Bringing  $M_i^*$  into the utility function and then computing the first-order partial derivative with respect to  $\alpha_i$  yields

$$\frac{\partial U}{\partial \alpha_i} = \sqrt{\frac{\mu_1 (1 - \beta_{i,j}) CD_i m_j}{\alpha_i}} + H3 \quad (31)$$

Where  $H3 = \frac{(\mu_1 + \mu_2 p_{i,j}^{M-U}) D_i}{R_{i,j}^{M-U}} - \mu_2 \gamma^{MU} CD_i (f_i^{MU})^2$ .

Then computing the second-order partial derivatives yields  $\frac{\partial^2 U}{\partial (\alpha_i)^2} < 0$ . It means that the first-order partial derivatives of  $U_{mu}^{M-U}$  are  $\frac{\partial U}{\partial \alpha_i}$  monotonically decreasing. When  $\alpha_i$  takes small enough positive values,  $\frac{\partial U}{\partial \alpha_i} > 0$ . When  $\alpha_i$  tends to a value of 1,  $\frac{\partial U}{\partial \alpha_i}$  tends to  $\sqrt{\mu_1 (1 - \beta_{i,j}) CD_i m_j} + H3$ . Where  $\sqrt{\mu_1 (1 - \beta_{i,j}) CD_i m_j}$  is equivalent to  $M_i^*$  when  $\alpha_i$  tends to 1 because the sum of the monetary cost and the transmission cost is greater than the computational cost with the same task size, so that  $\frac{\partial U}{\partial \alpha_i} > 0$ . Thus,  $U_{mu}^{M-U}$  is monotonically increasing with respect to  $\alpha_i$  and obtains a minimum at  $t_i^{MU} = t_{i,j}^{M-U} + t_{i,j}^{UAV}$ .

When  $t_i^{MU} \leq t_i^{off}$  and  $t_{i,j}^{UAV} < t_{i,j}^{U-B} + t_{i,j}^{BS}$ ,  $t_i^{all} = t_{i,j}^{M-U} + t_{i,j}^{U-B} + t_{i,j}^{BS}$ . Equation (24) and Eq. (25) are brought into  $U_{mu}^{M-U}$ , and then the first-order partial derivatives and second-order partial derivatives with respect to  $\alpha_i$  are calculated:

$$\frac{\partial U}{\partial \alpha_i} = H3 - \frac{\mu_2 D m_j^2}{2C \gamma^{UAV} (\mu_2 D \alpha_i - \Phi)^2} \quad (32)$$

$$\frac{\partial^2 U}{\partial (\alpha_i)^2} = \frac{(\mu_2 D m_j^2)^2}{C \gamma^{UAV} (\mu_2 D \alpha_i - \Phi)^3} \quad (33)$$

Based on constraint (18a) and Eq. (25), it can be concluded that  $\mu_2 D \alpha_i > \Phi$ , and therefore  $\frac{\partial^2 U}{\partial (\alpha_i)^2} > 0$ . So when  $t_i^{all} = t_{i,j}^{M-U} + t_{i,j}^{U-B} + t_{i,j}^{BS}$ ,  $U_{mu}^{M-U}$  is a convex function with respect to  $\alpha_i$ , and the first-order partial derivative is made equal to zero to obtain the offloading strategy

$$\alpha_i^{(3)} = \frac{1}{\mu_2 D} \left( \sqrt{\frac{\mu_2 D m_j^2}{2C \gamma^{UAV} H3}} + \Phi \right) \quad (34)$$

At this point, there are two cases: (i) When  $\alpha_i^{(2)} \geq \alpha_i^{(3)}$ ,  $U_{mu}^{M-U}$  is monotonically increasing in the range of  $\alpha_i \in [\alpha_i^{(2)}, 1]$ , the optimal offloading strategy is  $\alpha_i^* = \alpha_i^{(2)}$ . (ii) When  $\alpha_i^{(2)} < \alpha_i^{(3)}$ ,  $U_{mu}^{M-U}$  is a strictly convex function in the range of  $\alpha_i \in [\alpha_i^{(2)}, 1]$ , the optimal offloading strategy is  $\alpha_i^* = \alpha_i^{(3)}$ .

Moreover,  $U_{uav}^{M-U}$  is continuous with respect to  $m_j$  and has zero second order partial derivatives. Therefore, for each fixed  $m_j$ , there exists a unique Nash equilibrium point for the MU-UAV game [13], [14], i.e., the set of optimal strategies  $\{\alpha_i^*, m_j^*, f_{i,j}^{UAV*}\}$ .

Since UAV's set of strategies in the MU-UAV game and the UAV-BS game do not interfere with each other, there exists a unique Nash equilibrium for the Stackelberg bilayer game.

## 5. Algorithm Design

In order to obtain an approximate optimal solution to the proposed problem, as shown in Algorithm 1, we propose a multi-leader, multi-follower Stackelberg bilayer game TASR algorithm.

### Algorithm 1 TASR Algorithm

---

**Input:**  $B, K, N, C, D_i, P_{i,j}^{M-U}, P_{j,b}^{U-B}, P_{i,j}^{M-U}, B_{j,b}^{U-B}$   
Position coordinates of MUs, UAVs, and BSs

**Output:** Cost or utility for all game players

- 1: **Initialization:**  $\alpha_i, \beta_{i,j}, m_j, q_{i,j,b}, f_{i,j}^{UAV}, f_{i,j,b}^{BS}, \lambda_{i,j}$   
Server Selection Strategy
- 2: **while** Bilayer game reaches Nash equilibrium **do**
- 3:   **for each**  $MU_i, i \in N$  **do**
- 4:     **for each**  $UAV_j, j \in K$  **do**
- 5:       **if**  $UAV_j$  is the offloading destination for  $MU_i$  **then**
- 6:          Calculate the resource allocation strategy  $f_{i,j}^{UAV*}$   
and offloading strategy  $\alpha_i^*$ .
- 7:          Calculate the utility of this task for  $UAV_j$ .
- 8:       **for each**  $BS, b \in B$  **do**
- 9:          **if**  $BS_b$  is the offloading destination for  $UAV_j$  **then**
- 10:            Calculate the resource allocation strategy  $f_{i,j,b}^{BS*}$   
and the offloading strategy  $\beta_{i,j}^*$ .
- 11:            Calculate the utility of this task for  $BS_b$ .
- 12:          **end if**
- 13:       **end for**
- 14:        Calculate the offloading cost of  $UAV_j$ .
- 15:     **end if**
- 16:   **end for**
- 17:    Calculating the cost of task execution for  $MU_i$ .
- 18: **end for**
- 19: Update UAV pricing strategy  $m_j^*$ .
- 20: Update BS pricing strategy  $q_{i,j,b}^*$ .
- 21: Update the duality variable  $\lambda_{i,j}^*$ .
- 22: Update server selection strategy.
- 23: **end while**

---

Since the duality variable  $\lambda_{i,j}$  introduced in Problem P4.2 is a variable outside the set of game strategies,  $\lambda_{i,j}$  and the set of game strategies need to be iterated alternately in order to obtain the near-optimal solution. In each iteration, the game strategy is first updated with  $\lambda_{i,j}$  fixed, and then  $\lambda_{i,j}$  is updated with the game strategy fixed. The optimal duality solution and the optimal set of strategies are obtained when the iterative algorithm converges. Algorithm 1 describes the detailed algorithmic procedure.

For the duality variable  $\lambda_{i,j}$ , the subgradient method is used to update the variable values at each iteration, and the update expression is denoted as



$$\lambda_{i,j}^{\tau+1} = [\lambda_{i,j}^{\tau} + \theta^{\tau} \Delta \lambda_{i,j}^{\tau}]^+ \quad (35)$$

$$\begin{aligned} \Delta \lambda_{i,j}^{\tau} = & \mu_2 * e^{U-B} + q_b f_{i,j,b}^{BS} \\ & - \mu_2 \gamma^{UAV} \alpha_i \beta_{i,j} D_i C (f_{i,j}^{UAV})^2 \end{aligned} \quad (36)$$

Where  $\Delta \lambda_{i,j}^{\tau}$  denotes the corresponding subgradient, and  $\theta^{\tau}$  denotes the step size of the first  $\tau$  iteration.

Although increasing the amount of computing resources allocated increases the monetary gain of the server, it is accompanied by a quadratic increase in energy cost, which is consistent with the principle of diminishing marginal returns in economics [15], [16]. Using the marginal revenue formula, the server price update expression is expressed as

$$m_j^{\tau+1} = m_j^{\tau} + \delta MR_{UAV}^{\tau} \quad (37)$$

$$q_b^{\tau+1} = q_b^{\tau} + \delta MR_{BS}^{\tau} \quad (38)$$

Where  $\delta$  is the iterative step adjustment factor, and  $MR_{UAV}^{\tau}$  and  $MR_{BS}^{\tau}$  denote the marginal returns of UAV and BS, respectively. Introducing a very small price change factor  $\varepsilon$ ,  $MR_{UAV}^{\tau}$  and  $MR_{BS}^{\tau}$  can be specified as

$$MR_{UAV}^{\tau} = \frac{U_{uav}^{M-U}(m_j^{\tau} + \varepsilon) - U_{uav}^{M-U}(m_j^{\tau} - \varepsilon)}{\sum_{i=1}^N [f_{i,j}^{UAV*}(m_j^{\tau} + \varepsilon) - f_{i,j}^{UAV*}(m_j^{\tau} - \varepsilon)]} \quad (39)$$

$$MR_{BS}^{\tau} = \frac{L(q_b^{\tau} + \varepsilon) - L(q_b^{\tau} - \varepsilon)}{\sum_{i=1}^N \sum_{j=1}^K [f_{i,j,b}^{BS*}(q_b^{\tau} + \varepsilon) - f_{i,j,b}^{BS*}(q_b^{\tau} - \varepsilon)]} \quad (40)$$

In addition, a server selection algorithm based on the principle of least cost is designed in order to ensure competitiveness among multi-leaders. As shown in Algorithm 2, the MUs and UAVs select the service that minimises their offloading cost as the new task offloading destination after each iteration.

## 6. Simulation Results

In this section, a detailed simulation experiment of the proposed algorithm is carried out using the MATLAB tool, and the simulation results are compared with the benchmark scenario. The parameters used in the simulation are based on the experimental simulation of the real scenario, as shown in Table 1 [6], [17]. The 2D coordinates of the BS, UAV, and MU are randomly generated in a range of  $1000\text{m} \times 1000\text{m}$ , where the heights of the UAV, BS, and MU are fixed at 100m, 2m, and 0m, respectively.

Figure 2 shows the iterative process of task offloading ratio. In the initial stage of the game, the task offloading proportion in the MU-UAV game shows an increasing trend. In contrast, the proportion of task offloading in the UAV-BS game has a decreasing trend. After about 90 iterations, the offloading strategies in the bilayer game begin to fluctuate in a small range until they finally converge. In Fig. 2, the offloading ratio of task 3 showed a wide range of changes

### Algorithm 2 Server Selection Algorithm

---

**Input:**  $B, K, N, m_j^*, q_{i,j,b}^*, \lambda_{i,j}^*, \alpha_i^*$   
Position coordinates of MUs, UAVs, and BSs

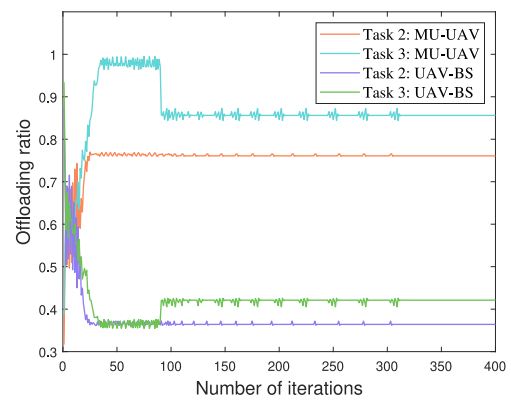
**Output:** Server Selection Strategy

- 1: **for** each  $MU_i, i \in N$  **do**
- 2:   **for** each  $UAV_j, j \in K$  **do**
- 3:     Calculate the resource allocation strategy  $f_{i,j}^{UAV*}$ .
- 4:     Calculate the task execution cost of  $MU_i$  for selecting  $UAV_j$  as the offloading destination.
- 5:     **for** each  $BS, b \in B$  **do**
- 6:       Calculate the offloading strategy  $\beta_{i,j}^*$  and the resource allocation strategy  $f_{i,j,b}^{BS*}$ .
- 7:       Calculate the task execution cost of  $UAV_j$  for selecting  $BS_b$  as the offloading destination.
- 8:     **end for**
- 9:     The  $UAV_j$  selects the BS that minimises its offloading cost as the offloading destination.
- 10:   **end for**
- 11:   The  $MU_i$  selects the UAV that minimises its offloading cost as the offloading destination.
- 12: **end for**

---

**Table 1** Simulation parameters.

Parameters	Values
B	2
K	4
N	8
$\eta$	3.76
C	1000 cycles/bit
$N_{NL}$	20 dB
$N_0^2$	-174 dBm
$D_i$	[6,10] Mbit
$f_i^{MU}$	[0.5,1] GHz
$P_{i,j}^{M-U}, P_{i,j,b}^{U-B}$	[0.2,0.3] W
$B_{i,j}^{M-U}, B_{i,j,b}^{U-B}$	[1,2] MHz
$\gamma^{MU}, \gamma^{UAV}, \gamma^{BS}$	$[1, 5] \times 10^{-27}$



**Fig. 2** Variation in offloading ratio

after about 90 iterations. It indicates that  $MU_3$  changed its task offloading destination after this iteration, i.e., another UAV was selected as the server, which also implies that the BS serving Task 3 also changed.

Figure 3 shows how the MU cost changes as the number of iterations increases. At the beginning of the iteration, the cost of each MU shows an increasing trend. As the number

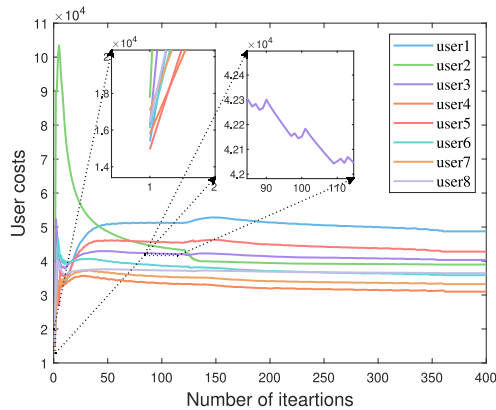


Fig. 3 Variations in the cost of MUs

of iterations increases, the cost of MUs starts to show a decreasing potential and reaches a converged state at the end. When the costs of all MUs are in the converged state, the system reaches Nash equilibrium. Jointly observing task 3 in Fig. 2 and MU<sub>3</sub> in Fig. 3, the cost of MU<sub>3</sub> shows a small fluctuation near the 90th iteration and then continues to show a decreasing trend, indicating that the change of a single node does not affect the performance of the proposed algorithm, which confirms that the TASR algorithm has a certain degree of robustness.

Next, to verify the superior performance of TASR, we introduce three benchmark schemes for comparison. (1) Full offloading of distribution scheme (FODS): MU offloads all the tasks to the UAV for processing [18], and a single-layer game is played between the UAV and the BS to determine the offloading policy. (2) Uniform offloading distribution scheme (UODS): The same proportion of data is computed by BS, UAV, and MU for each task. The other strategies are solved in the same way as TASR. (3) Randomised offloading ratio scheme (RORS): The task offloading ratio is randomly generated between [0, 1].

Figure 4 compares the impact of different offloading schemes on the total system cost. TASR achieves the joint optimization of offloading strategies and computational resources by solving the Nash equilibrium of the bilayer game, which reduces the total system cost. UODS is unable to take into account the differences in the resources of different nodes, which leads to the imbalance in the utilization of the resources. The decision-making of RODS is stochastic and the optimization is inefficient. FODS in the case of only a single-layer game lacks dynamic analysis of user requirements and is completely dependent on server resources. Therefore, under the same conditions TASR can obtain the minimum system cost and always has better performance. This proves the superiority of TASR. Moreover, TASR converges after about 360 iterations, which proves the existence of Nash equilibrium from an experimental point of view.

In addition, we conduct simulation experiments for different numbers of MUs. Figure 5 and Fig. 6 compare the effects of the number of MUs on the average latency and the total system cost of executing tasks under different of-

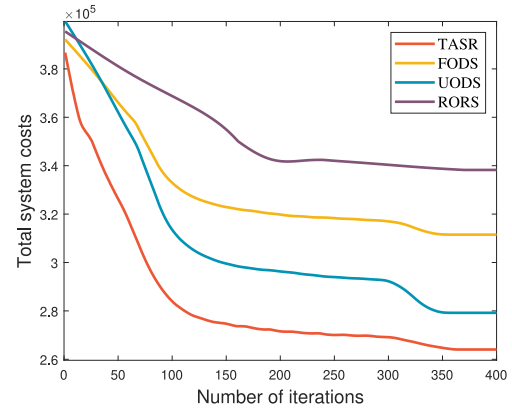


Fig. 4 Total system cost for different offloading schemes

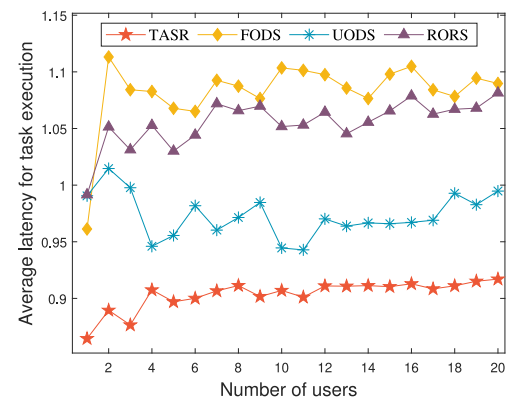


Fig. 5 Average latency with different number of MUs

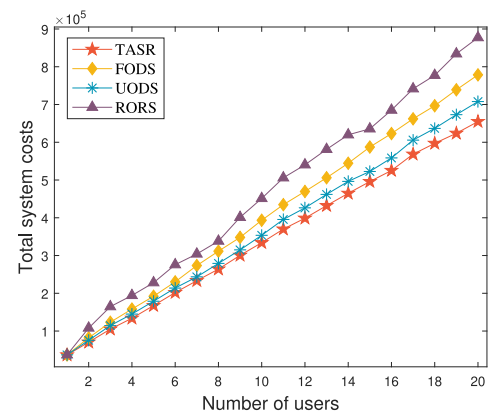


Fig. 6 Total energy consumption with different number of MUs

floading scenarios, respectively. Figure 5 shows that TASR obtains the lowest average latency, and the increase in the number of MU does not cause significant fluctuations in the average latency. Figure 6 indicates that TASR obtains the lowest total system cost in the different number of MUs in an environments, TASR obtains the lowest total system cost. When the number of users is high, TASR reduces the energy consumption by about 7% compared to UODS and about 16% compared to FODS. The results validate the robustness and superiority of TASR.

## 7. Conclusion

In this paper, we study MEC systems that contain multiple MEC server-equipped UAVs and BSs and multiple MUs. We model the interaction process between MEC servers and MUs as a multi-leader, multi-follower Stackelberg bi-layer game and prove the existence and uniqueness of Nash equilibrium. We propose the TASR algorithm to obtain an approximate optimal solution to the proposed problem and evaluate the effectiveness of the proposed algorithm through detailed simulation experiments. Results show that the TASR algorithm is significantly superior and robust compared to other benchmark schemes. Due to the variability of hotspot areas for task requirements, we will consider UAV location deployment and path planning in future research.

## References

- [1] X. Lyu, W. Ni, H. Tian, R.P. Liu, X. Wang, G.B. Giannakis, and A. Paulraj, "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.
- [2] G. Xie, Z. Wang, and Y. Liu, "Joint Task Partition and Resource Allocation for Multiuser Cooperative Mobile Edge Computing," *Wirel. Commun. Mob. Comput.*, vol.2022, pp.1–13, 2022.
- [3] Y. Mao, C. You, J. Zhang, K. Huang, and K.B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Commun. Surv. Tutor.*, vol.19, no.4, pp.2322–2358, 2017.
- [4] Z. Shah, U. Javed, M. Naeem, S. Zeadally, and W. Ejaz, "Mobile Edge Computing (MEC)-Enabled UAV Placement and Computation Efficiency Maximization in Disaster Scenario," *IEEE Trans. Veh. Technol.*, vol.72, no.10, pp.13406–13416, Oct. 2023.
- [5] G. Mitsis, E.E. Tsiropoulou, and S. Papavassiliou, "Data Offloading in UAV-Assisted Multi-Access Edge Computing Systems: A Resource-Based Pricing and User Risk-Awareness Approach," *Sensors*, vol.20, no.8, p.2434, Jan. 2020.
- [6] Z. Yu, Y. Gong, S. Gong, and Y. Guo, "Joint Task Offloading and Resource Allocation in UAV-Enabled Mobile Edge Computing," *IEEE Internet Things J.*, vol.7, no.4, pp.3147–3159, April 2020.
- [7] S. Jeong, O. Simeone, and J. Kang, "Mobile Edge Computing via a UAV-Mounted Cloudlet: Optimization of Bit Allocation and Path Planning," *IEEE Trans. Veh. Technol.*, vol.67, no.3, pp.2049–2063, March 2018.
- [8] M. Wang, L. Zhang, P. Gao, X. Yang, K. Wang, and K. Yang, "Stackelberg-Game-Based Intelligent Offloading Incentive Mechanism for a Multi-UAV-Assisted Mobile-Edge Computing System," *IEEE Internet Things J.*, vol.10, no.17, pp.15679–15689, 2023.
- [9] H. Zhou, Z. Wang, G. Min, and H. Zhang, "UAV-aided computation offloading in mobile-edge computing networks: A stackelberg game approach," *IEEE Internet Things J.*, vol.10, no.8, pp.6622–6633, April 2023.
- [10] H. Lu, X. He, M. Du, X. Ruan, Y. Sun, and K. Wang, "Edge QoE: Computation Offloading With Deep Reinforcement Learning for Internet of Things," *IEEE Internet Things J.*, vol.7, no.10, pp.9255–9265, Oct. 2020.
- [11] F. Costanzo, P.D. Lorenzo, and S. Barbarossa, "Dynamic Resource Optimization and Altitude Selection in Uav-Based Multi-Access Edge Computing," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.4985–4989, May 2020.
- [12] Technical Specification Group Radio Access Network, "Study on enhanced LTE support for aerial vehicles," document 3GPP TR 36.777 V15.0.0, Dec. 2017.
- [13] Y. Chen, Z. Li, B. Yang, K. Nai, and K. Li, "A Stackelberg game approach to multiple resources allocation and pricing in mobile edge computing," *Future Gener. Comput. Syst.*, vol.108, pp.273–287, July 2020.
- [14] J.B. Rosen, "Existence and Uniqueness of Equilibrium Points for Concave N-Person Games," *Econometrica*, vol.33, no.3, p.520, July 1965.
- [15] P.J. Coughlin, "Changes in Marginal Revenue and Related Elasticities," *South. Econ. J.*, vol.51, no.2, pp.568–573, 1984.
- [16] X. He, A. Prasad, and S.P. Sethi, "Cooperative advertising and pricing in a dynamic stochastic supply chain: Feedback Stackelberg strategies," *PICMET '08 - 2008 Portland International Conference on Management of Engineering & Technology*, pp.1634–1649, July 2008.
- [17] L. Zhang and N. Ansari, "Latency-Aware IoT Service Provisioning in UAV-Aided Mobile-Edge Computing Networks," *IEEE Internet Things J.*, vol.7, no.10, pp.10573–10580, Oct. 2020.
- [18] S. Bi and Y.J. Zhang, "Computation Rate Maximization for Wireless Powered Mobile-Edge Computing With Binary Computation Offloading," *IEEE Trans. Wirel. Commun.*, vol.17, no.6, pp.4177–4190, June 2018.



**Peng Wang** is pursuing a master's degree in information and communication engineering at Changchun University of Science and Technology. His main research interests are mobile edge computing, service caching, and the IoT.



**Guifen Chen** is a professor and doctoral supervisor in the field of information and communication engineering. Her main research interests are optical communication and communication technology. In recent years, Prof. Chen has completed more than ten national, provincial and horizontal projects.



**Zhiyao Sun** is pursuing her PhD in Information and Communication Engineering. She participated in and completed several national, provincial and horizontal projects during her PhD program. Her main research interests are mobile edge computing, heterogeneous networks, and optical communication technologies.