Joint Pilot and Data Transmission Power Control and Computing Resource Allocation Algorithm for Massive MIMO-MEC Networks

WENJIANG FENG, JIALI ZHENG, AND WEIHENG JIANG, (Member, IEEE)
College of Microelectronics and Communication Engineering, Chongqing University, Chongqing 400044, China
Corresponding author: Jiali Zheng (781459521@qq.com)

This work was supported in part by the Chongqing Basic Science and Frontier Technology Research Project under Grant cstc2017jcyjBX0047.

ABSTRACT This paper designs a single-cell multi-user massive MIMO-MEC network. In order to ensure the fairness of users, a joint pilot transmission, data transmission and resource allocation model during the computation execution process with the goal of minimizing the maximum offload computing delay for all users is constructed. The resulted problem is non-convex and non-linear optimization, thus difficult to be solved optimally. To tackle this challenge, an improved fruit fly optimization algorithm (FOA) based on the external penalty function steepest descent algorithm (IFOA-PFSA) is proposed. The point obtained by the steepest descent algorithm based on the external penalty function has been employed as the initial point of the fruit fly optimization algorithm, which can greatly reduce the population size and the maximum number of iterations in the random search process of the traditional fruit fly optimization algorithm, reducing the algorithm complexity. Simulation results show that the proposed algorithm IFOA-PFSA has a smaller delay than the traditional FOA (TFOA) algorithm. The complexity of the proposed algorithm is also lower than the TFOA algorithm.

INDEX TERMS Mobile edge computing, massive MIMO, resource allocation, power control.
task scheduling policy was proposed aimed to maximize the profit of mobile network by jointly optimizing service rate, transmit power, and subcarrier allocation with satisfying power and delay constraints. In [10], an efficient task scheduling and radio resource allocation algorithm was developed to minimize the total execution delay of users in a single-cell power-domain non-orthogonal multiple access (PD-NOMA) based MEC system with multiple users and single MEC server. In [11], a power allocation algorithm was proposed for joint pilot and data transmission for the uplink of a single-cell massive MIMO system. It took spectrum efficiency as the performance index, coherent interval total energy budget and power control as decision variables to optimize total user spectral efficiency. The authors of [12], [13] studied the resource allocation of MEC networks based on game theory. In [14], two-stage decision optimization was studied, including user's offload computing decisions and cloud resource allocation based on computing needs. Task-based random arrival was considered in [13], analyzed the resource competition and equilibrium decision in the double-layer cloud network, proved the existence of equilibrium, and proposed an algorithm that can converge to the equilibrium solution. In [14], the joint computing and communication resource allocation of mobile edge computing networks based on millimeter wave multi-antennas was considered. With the constraint of delay and the goal of minimizing the total transmission power of the terminal, joint resource allocation is studied. In [15], the joint task assignment, transmission, and computing resource allocation in multilayer mobile edge computing systems was studied. For MEC network, unmanned aerial vehicle (UAV) integrates computing resources, while macro base stations integrate more powerful central cloud servers. The authors of [16] designed a segmented fog cloud computing architecture based on multiple unmanned aerial vehicles. The authors of [17] discussed the uplink resource allocation of the DM-MIMO system, aimed at minimizing user energy consumption. Using distributed fog computing, a signal processing and power control mechanism was proposed to reduce system complexity, overhead, and hardware cost.

For joint channel estimation and data transmission resource allocation, [18] aims at optimizing the system energy efficiency. A fractional planning and parameter planning optimization model was established with the pilot length, pilot and data transmission power as variables and the constraints on transmission power and spectral efficiency. A three-layer iterative energy efficiency resource allocation optimization algorithm was proposed. The authors of [19] studied joint scheduling and offload decision optimization based on integer programming. In [20], joint resource allocation optimization under MIMO cellular networks was studied. Jointly optimize the computing and communication resources under user delay constraints to minimize the energy consumption of multiple users. Because the established optimization model is non-convex, a continuous convex relaxation technique was proposed to obtain a stable solution. The authors of [21] provided a spectral and energy efficiency evaluation framework for massive MIMO heterogeneous networks with wireless backhaul. They developed a novel alternating optimization algorithm to maximize the weighted sum of spectral efficiency and energy efficiency. The authors of [22] investigated the energy-efficient resource allocation in two-tier massive MIMO heterogeneous networks with wireless backhaul.

A. RESEARCH CONTRIBUTION

The main contributions of this paper are summarized as follows:

• This paper designs a single-cell multi-user massive MIMO-MEC network. Combining MIMO with MEC, the scenario is novel.

• For a single-cell scenario, taking into account the pilot assisted channel estimation and offload computing, we jointly optimize pilot transmission power, data transmission power, and server computing resource allocation under the given the constraints of user energy consumption and computing resource constraints. We have established an optimization model. To ensure fairness, our goal is to minimize the maximum offload computing delay among all users. The models we build are more complex and more realistic than previous work.

• The constructed model is a non-convex non-linear optimization model. We first use a low-complexity suboptimal algorithm based on the external penalty function to transform the constraint problem into an unconstrained optimization problem.

• Then, the steepest descent algorithm based on the external penalty function is proposed to obtain some initial optimal points, which can reduce the complexity of the algorithm and speed up the convergence speed of the algorithm.

• Finally, the initial points are further optimized by using improved fruit fly optimization algorithm (IFOA) to obtain the local optimal solution of the optimization problem. By tracking the initial points, the fruit flies can conduct a local random search centered on the current optimal solution and search in a better direction. Then multiple local optimal solutions are obtained, and the local optimal solutions are further compared to obtain the global sub-optimal solution. The initial point of the fruit fly optimization algorithm, which can greatly reduce the population size and the maximum number of iterations in the random search process of the traditional fruit fly optimization algorithm. It can be seen from the simulation results that the proposed IFOA-PFSA algorithm has a better delay effect than TFOA.

B. ORGANIZATION

The remainder of this paper is organized as follows. Section II presents the system model of the MIMO-MEC and formulates problem of minimizing the maximum offload computing delay for all users. Section III proposes an improved
fruit fly optimization algorithm (FOA) [23] based on the steepest descent algorithm of the external penalty function (IFOA-PFSA) to solve the problem. Section IV provides simulation results to validate the effectiveness of the proposed algorithm. Section V concludes this paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

This section introduces the massive MIMO-MEC network system model and the joint resource allocation and power control optimization model in offload computation. The system model includes the network model, task model, communication model, and computation model.

A. NETWORK MODEL

The single-cell massive MIMO-MEC network model is shown in Fig. 1. It is a distributed topology and includes \( K \) single-antenna users and a wireless MEC server. The wireless MEC server is equipped with \( M \) antennas as an access point (AP), that is, the wireless MEC server has both communication and computation capabilities. Considering the local computing capabilities, energy supply constraint and other factors, the users’ computing tasks can be partially or wholly offloaded to the wireless MEC server for execution based on the complexity of the task. We assume that all computing tasks are offloaded to achieve optimal latency and energy consumption.

\[ G = \{g_1, \ldots, g_K\} \in C^{M \times K}, \quad g_k \sim CN(0, \beta_k \mathbf{I}), \quad k = 1, 2, \ldots, K \]

where the variance \( \beta_k > 0 \) represents the large-scale fading of the \( k \)-th user, including path loss and shadows, and the large-scale fading is normalized by the noise variance at the wireless MEC server. It is further assumed that large-scale fading is known at the wireless MEC server. According to [26], the large-scale fading can be denoted as:

\[ \beta_k = PL_k \cdot 10^{-\frac{\sigma_{sh,k}}{10}} \]

where \( PL_k \) represents the path loss (dB) between the \( k \)-th user and the wireless MEC server, \( 10^{0.1z_k/10} \) is the shadow fading with standard deviation \( \sigma_{sh,k} \), and \( z_k \sim CN(0, 1) \) is the shadowing coefficient describing the correlation of shadow fading between the \( k \)-th user and the wireless MEC server. Denote the distance between the \( k \)-th user and the

B. TASK MODEL

There are two commonly used computing task models for MEC networks, namely binary offload computing tasks [24] and partial offload computing tasks [25]. This paper assumes that the tasks required offload computing are binary offload computing tasks. The binary offload computing task is an indivisible computing task, which is either performed locally at the user equipment or offloaded to a wireless MEC server for execution as a whole. For a binary offload computing task \( A(L_{in}, \tau_d, X, L_{out}) \), \( L_{in} \) represents the number of data bits of the offloading task, \( L_{out} \) represents the number of data bits of the computed result, \( \tau_d \) denotes the deadline for the task, and \( X \) is the density/complexity of the task computing (i.e., the number of CPU cycles/bit).

\[ L_{in}X \]

The required number of CPU cycles is \( L_{in}X \), and then the \( L_{out} \) bits of the computed result generated by the task execution is fed back to the user equipment. The task has a deadline \( \tau_d \), and it is necessary to ensure that the task is completed before the deadline \( \tau_d \).

C. COMMUNICATION MODEL

Assuming that the channel fading remains unchanged at time interval \( T = \max\{\tau_k, d[k = 1, \ldots, K]\} \), \( \tau_k, d \) indicates the deadline of the task of the \( k \)-th user. The flat fading channel matrix from the wireless MEC server to all users is defined as \( G = \{g_1, \ldots, g_K\} \in C^{M \times K} \), and the \( k \)-th column \( g_k \) denotes the channel vector of the \( k \)-th user, which follows the following distribution:

\[ g_k \sim CN(0, \beta_k \mathbf{I}), \quad k = 1, 2, \ldots, K \]

where the variance \( \beta_k > 0 \) represents the large-scale fading of the \( k \)-th user, including path loss and shadows, and the large-scale fading is normalized by the noise variance at the wireless MEC server. It is further assumed that large-scale fading is known at the wireless MEC server. According to [26], the large-scale fading can be denoted as:

\[ \beta_k = PL_k \cdot 10^{-\frac{\sigma_{sh,k}}{10}} \]

where \( PL_k \) represents the path loss (dB) between the \( k \)-th user and the wireless MEC server, \( 10^{0.1z_k/10} \) is the shadow fading with standard deviation \( \sigma_{sh,k} \), and \( z_k \sim CN(0, 1) \) is the shadowing coefficient describing the correlation of shadow fading between the \( k \)-th user and the wireless MEC server. Denote the distance between the \( k \)-th user and the
wireless MEC server as \(d_k\), and its path loss is:

\[
PL_k = \begin{cases} 
-L - 35 \log_{10}(d_k) & \text{if } d_k > d_1 \\
-L - 15 \log_{10}(d_1) - 20 \log_{10}(d_k) & \text{if } d_0 < d_k \leq d_1 \\
-L - 15 \log_{10}(d_1) - 20 \log_{10}(d_0) & \text{if } d_k \leq d_0 
\end{cases}
\]

(3)

where

\[
L = 46.3 + 33.9 \log_{10}(f) - 13.82 \log_{10}(h_b) - (1.1 \log_{10}(f) - 0.7)h_u + (1.56 \log_{10}(f) - 0.8)
\]

(4)

and where \(f\) is the carrier frequency (in MHz), \(h_b, h_u\) is the antenna height (in m) of the wireless MEC server and the user, respectively.

Fig. 2 defines the pilot transmission time length as \(t_p\) (in time-frequency samples), where \(\phi_k\) represents the pilot sequence sent by the \(k\)-th user, \(||\phi_k||^2 = 1\), and the received signal of the wireless MEC server is represented by an \(M \times t_p\) order matrix:

\[
Y_p = \sum_k \sqrt{p_k} g_k \phi_k^H + N_p
\]

(5)

where \(p_k\) represents the pilot transmit power of the \(k\)-th user, \(N_p = [n_1^k, \ldots, n_K^k]\) represents the additive noise matrix at the wireless MEC server, and its elements are independent and uniformly distributed (i.i.d.) random variables obeying the distribution of \(CN(0, \sigma_n^2)\). The wireless MEC server estimates the channel vector \(g_k\) based on the received signal \(Y_p\) and the prior knowledge of the pilot sequence. If the pilot transmit power \(p_k^p\) and large-scale fading coefficient \(\beta_k\) are known, the channel estimation based on Linear Minimum Mean Square Error (LMMSE) is:

\[
\hat{g}_k = \frac{\sqrt{t_p p_k^p \beta_k}}{1 + t_p p_k^p \beta_k} \left( \sqrt{t_p} p_k^p g_k + n_k^p \right)
\]

(6)

In the data transmission stage, the received signals of the wireless MEC server can be denoted as:

\[
y = \sum_{k=1}^K g_k s_k \sqrt{p_d^k} + n
\]

(7)

where \(s_k\) is the transmitted signal of the \(k\)-th user subject to the Gaussian distribution with zero mean and unit variance, and \(n \sim CN(0, I)\) is the additive noise in the data transmission stage. Subsequently, the channel estimation results are used by the wireless MEC server to perform signal detection based on the maximum ratio combining (MRC), that is, to multiply the received signal \(y\) by \(G^H = [\hat{g}_1, \ldots, \hat{g}_K]^H\) to achieve decoding \(s_1, \ldots, s_K\). The achievable ergodic rate for the \(k\)-th user can be expressed as:

\[
R_k = \log_2 (1 + SINR_k)
\]

(8)

where

\[
SINR_k = \frac{M_p \gamma_k}{1 + \sum_{j=1}^K \beta_j \gamma_j}, \quad \gamma_k = \frac{t_p \beta_k^2}{1 + t_p p_k^p \beta_k^2}.
\]

It should be noted that this work assumes that the wireless MEC server can obtain perfect channel state information (CSI) for each user. However, in reality, there is an estimation error in the channel estimation algorithm due to bit errors in the transmission of pilot symbols. Future work needs to quantify the impact of imperfect channel state information on performance. On the other hand, it is also necessary to evaluate the impact of pilot (signaling) overhead on resource efficiency caused by improving the accuracy of channel estimation. In this work, we are considering a massive MIMO scenario and the coverage area of the base station in each cell is small. Therefore, in principle, the SINR of each user in the cell can meet the constraint conditions of the minimum SINR.

D. CALCULATION MODEL

The wireless MEC server allocates VMs to perform computing services for different users in parallel, so the VM queue time is \(t_{vmQ} \to 0\). The computation performance depends on the CPU frequency allocated by the server. Assume that the maximum CPU frequency of the server is \(f_s\), and define \(f_{k,s}\), \(\tau_{k,s}\) as the CPU frequency allocated by the server to the \(k\)-th user. Therefore, for a given offloading computing task \(A(L_{k,in}, v_k, d, X_k, L_{k.out})\) by the \(k\)-th user, the delay of the wireless MEC server computing the task is given by:

\[
t_{vm}^k = L_{k,in} X_k / f_{k,s}
\]

(9)

Let \(R_{k,u}\) and \(R_{k,d}\) be the uplink and downlink transmission rates obtained by the \(k\)-th user, respectively, ignoring the uplink and downlink queuing time, i.e. \(t_{uQ}, t_{dQ} \to 0\). Assume that the task calculation result data is small, ignoring the downlink transmission time, i.e. \(t_d \to 0\). Therefore, the total computing offload delay \(t_o\) of the user is composed of three parts: pilot transmission delay \(t_p\), data transmission delay \(t_d\) and server computation delay \(t_{vm}\). For the \(A(L_{k,in}, v_k, d, X_k, L_{k.out})\) of \(k\)-th user, the total computing offload delay can be expressed as:

\[
t_{k,o} = t_p + \frac{L_{k,in} + L_{k,in} X_k}{R_{k,u}} / f_{k,s}
\]

(10)

\[
= t_p + t_d^k + t_{vm}^k
\]

where \(t_d^k = L_{k,in} / R_{k,u}\), \(t_{vm}^k = L_{k,in} X_k / f_{k,s}^k\).

For user equipment (UE), the energy consumption of offload computing originates from three parts: uplink transmission of computing task offload, downlink reception of computed result feedback, and circuit power consumption. We assume that the circuit power consumption is a fixed value \(P_c^k\). The uplink transmission energy consumption includes pilot transmission and data transmission, which depends on the transmission power and transmission time. Given the maximum energy consumption \(E_k\) of the user equipment. This paper assumes that the computed result has a small amount of data, and therefore ignores the downlink receiving power consumption of the user equipment. The energy consumption of task offload computation is:

\[
E_{k,o} = \frac{L_{k,in} (p_d^k + p_c^k)}{R_{k,u}} + t_p (p_k^p + p_c^k)
\]

(11)

where \(E_{k,o} \leq E_k\).
E. PROBLEM FORMULATION

The massive MIMO-MEC network optimization model constructed in this paper aims to minimize the maximum computing offload delay for all users. The decision variables include pilot transmit power, data transmit power, and the CPU frequency of the wireless MEC server. Our optimization problem can be formulated as:

$$\mathcal{P}_1 \min_{p, s, f} \max_k T_k = t_p + \frac{L_{k, in}}{R_{k, s}} + \frac{L_{k, in}X_k}{f_{k,s}} + t^*_m$$

s.t. $C1: t_p(p^k_p + p^k_d) + t^*_k(q^k_d + p^k_c) \leq E_k$, $k = 1, \ldots, K$

$C2: \sum_{k=1}^{K} f_{k,s} \leq f_S$

$C3: 0 \leq p^k_p \leq p^k_{max}, 0 \leq p^k_d \leq p^k_{max}$

$k = 1, \ldots, K$ (12)

The problem in (12) is a minimum-maximum fairness problem, where $C1$ is the energy consumption constraint of the $k$-th user and $C2$ is the computing resource constraint of the wireless MEC server, that is, problem $\mathcal{P}_1$ is to jointly optimize pilot transmission power, data transmission power and server computing resource allocation under the given constraints of user energy consumption and computing resource constraints. The goal of the problem $\mathcal{P}_1$ is to minimize the maximum offload computing delay among all users. For problem $\mathcal{P}_1$, the decision variables in constraint $C1$ are coupled with each other to form a non-convex set. Therefore, the optimization problem is non-convex and cannot be directly solved by convex optimization techniques. The paper intends to simplify the original problem by analyzing the objective function of $\mathcal{P}_1$ and the structural characteristics of the constraint conditions, and then design a solution algorithm.

III. IMPROVED FRUIT FLY OPTIMIZATION ALGORITHM BASED ON THE EXTERNAL PENALTY FUNCTION STEEPEST DESCENT ALGORITHM

In order to solve the optimization problem $\mathcal{P}_1$, the following lemmas are proposed as the basis for the subsequent algorithm design, and they are used as the basis for judging the optimal solution of this optimization problem.

Lemma 1: If problem $\mathcal{P}_1$ has an optimal solution, then at the optimal solution, the offload computing delays of all users must be equal:

$$T_k = T^*, \quad \forall k = 1, \ldots, K$$

Proof: Prove by contradiction. First, suppose that at the optimal solution, the delay $T_j$ of one user is not equal to the delay $T^*$ of other users. Then, since it is at the optimal solution, there must be $T_j > T^*$, and $T_j$ is the minimum value of the maximum delay of all users under the constraint conditions. Then, according to (10), it can be seen that the computing delay is related to the allocated computing resources, i.e. $t_m = \frac{L_{k, in}X_k}{f_{k,s}}$, by reducing the resources allocated to other users, the computing resources allocated to the user can be increased to reduce the computing delay of the user. The delay in computing offload of the user will be reduced to $T_j'$, while satisfies the delay of the user is the largest among all users. However, $T_j' < T_j$, it conflicts with the assumption that the optimal solution is the minimum value of the maximum delay among all users under the constraint conditions. The original proposition holds, that is, at the optimal solution, all users have the same computing offload delay.

Lemma 2: If problem $\mathcal{P}_1$ has an optimal solution, then the optimal solution satisfies that the sum of the computing resources allocated by all users is the maximum computing resource that the wireless MEC server can provide:

$$\sum_{k=1}^{K} f_{k,s} = f_S$$

Proof: Prove by contradiction. First, from Lemma 1, when the optimal solution exists, there must exist that offload computing delay of all users equal to $T^*$. It can be known from $t_m = \frac{L_{k, in}X_k}{f_{k,s}}$ that under a certain task amount and task complexity, the computing delay is inversely proportional to the allocated resources. Assume that at the optimal solution, there exists:

$$\sum_{k=1}^{K} f'_{k,s} < f_S$$

Continuing to increase the computing resources allocated to each user can reduce the offload computing delay, which contradicts the assumption that the optimal solution is the lowest delay, so the assumption does not hold, and the original proposition is proved.

Lemma 3: If problem $\mathcal{P}_1$ has an optimal solution, then for all users, their energy consumption is their respective maximum energy consumption:

$$t_p(p^k_p + p^k_d) + t^*_k(q^k_d + p^k_c) = E_k, \quad k = 1, \ldots, K$$

Proof: Prove by contradiction. The SINR of the $k$-th user is monotonically increasing with respect to $p^k_p$ and is independent of the pilot power of other users. At the optimal solution, assuming that some users do not use the full energy in power allocation, they can increase the pilot transmission power to increase their own SINR without reducing the SINR of other users. Increasing the SINR will increase the user transmission rate and thus reduce the transmission delay. It is equivalent to creating a solution that is superior to the optimal solution. This is inconsistent with assumptions, so the original proposition is true.

Proposition 1: Given the CPU frequency of the virtual machine, if the data transmission delay of all users has been determined, the corresponding computing resource allocation method is also determined. The solution under this condition should satisfy Lemma 1 and Lemma 2.
According to Lemma 1, the allocation of computing resources should satisfy the following equations:

\[
\begin{aligned}
& t_p + t_d^1 + L_{1, in}X_1/t_{f,1}^s = t_p + t_d^2 + L_{2, in}X_2/f_{k,s} \\
& t_p + t_d^1 + L_{1, in}X_1/t_{f,1}^s = t_p + t_d^3 + L_{3, in}X_3/f_{k,s} \\
& t_p + t_d^1 + L_{1, in}X_1/t_{f,1}^s = t_p + t_d^K + L_{K, in}X_K/f_{k,s}
\end{aligned}
\]  

(17)

According to Lemma 2, in addition to satisfying the constraint that \( f_{k,s} > 0 \), the following constraints should also be satisfied:

\[
\sum_{k=1}^{K} f_{k,s} = f_S 
\]

(18)

According to the above equation and constraints, the following one-variable equation can be obtained:

\[
\sum_{k=1}^{K} L_{k, in}X_k / (t_{d,1}^i + L_{1, in}X_1/f_{f,1}^s) = f_S 
\]

(19)

In (19), except for \( f_{1,1} \), all variables are known, and in the interval \((0, +\infty)\), the left side of the (19) increases monotonically with \( f_{1,1} \), so it has and only has a unique positive solution. Then, the computing resource \( f_{k,s} \) of all users can be determined. In summary, the proposition is proved.

When the fruit fly optimization algorithm is used for subsequent optimization, the computing resource is no longer a decision variable, but a dependent variable determined by pilot transmission power and data transmission power.

Based on the above lemma, it is known that the optimal solution of the problem \( P_1 \) needs to meet the same offload computing delay of all users, the computing resources of the server are used 100\%, and the user’s energy is completely exhausted. However, because the objective function minimizes the maximum offload computing delay for all users is a non-convex non-linear function related to pilot transmission power, data transmission power, and computing resources, in order to solve this problem, the auxiliary variable \( \tau \) is introduced to convert the problem \( P_1 \) into the following equivalent problem:

\[
P_2 \quad \min_{(f_{k,s}, p_{d}, p_p, x)} \tau \quad s.t. \quad C0 : t_p + t_d^k + t_{vm} \leq \tau, \quad k = 1, \ldots, K \]

\[
C1 : t_p(p_{d,k} + p_{p,k}) + t_d^k(p_{d,k} + p_{p,k}) \leq E_k, \quad k = 1, \ldots, K \\
C2 : \sum_{k=1}^{K} f_{k,s} \leq f_S \\
C3 : 0 \leq p_{d,k} \leq p_{d,\text{max}}, 0 \leq p_{p,k} \leq p_{p,\text{max}}, \quad k = 1, \ldots, K
\]

(20)

where constraint \( C0 \) indicates that the time for each user to offload computing tasks should be less than a constant, that is, the original minimum-maximization optimization target is converted into a minimum optimization target. However, the problem is still non-convex.

### A. EXTERNAL PENALTY FUNCTION CONSTRUCTION

For the constraint non-convex problem \( P_2 \), this paper first uses a low-complexity suboptimal algorithm based on the external penalty function. The core idea of the external penalty function is to appropriately punish the constraint function in the problem, construct the auxiliary function with parameters, and transform the constraint optimization problem into a series of unconstrained optimization problems that solve the auxiliary function. In particular, for the problem \( P_2 \), the external penalty function algorithm is used to solve the problem. The basic idea is that the iteration point moves outside the feasible region, and as the number of iterations increases, the penalty intensity increases, so that the iteration point approaches the feasible region to get an approximate optimal solution in an approximate feasible region. Based on this, construct auxiliary functions for \( P_2 \):

\[
F(p_d, p_p, f, \tau, \mu) = \tau + \mu \\
\left( \sum_{i=1}^{K} \left[ \max \left(0, t_p + t_d^i + t_{vm} - \tau \right) \right]^2 \right) + \left( \sum_{i=1}^{K} \left[ \max \left(0, t_p(p_{d,i} + p_{p,i}) + t_d^i(p_{d,i} + p_{p,i}) - E_i \right) \right]^2 \right) + \left( \left[ \max \left(0, \sum_{i=1}^{K} f_{i,s} - f_S \right) \right]^2 \right)
\]

(21)

where \( p_d = [p_{d,1}, p_{d,2}, \ldots, p_{d,K}], p_p = [p_{p,1}, p_{p,2}, \ldots, p_{p,K}], f = [f_1, f_2, \ldots, f_K], \mu \) is a sufficiently large positive number, \( \mu \) is an external penalty function. In this way, the constrained optimization problem \( P_2 \) is transformed into an unconstrained optimization problem \( P_3 \):

\[
P_3 \quad \min_{(p_d, p_p, f, \tau, \mu)} F(p_d, p_p, f, \tau, \mu)
\]

(22)

When the set of decision variables in the problem \( P_3 \) is in the feasible region, \( F(p_d, p_p, f, \tau, \mu) = \tau \). When it is not in the feasible region, the penalty term coefficient in the auxiliary function is set to a positive sequence \( \{\mu_k\} \) that increases and tends to infinity. Its existence is a punishment for the points that deviate from the feasible region, and it is to force the iteration point to get close to the feasible region during the optimization process to obtain the approximate optimal solution of the constraint optimization problem.

### B. STEEPEST DESCENT ALGORITHM BASED ON EXTERNAL PENALTY FUNCTION

After completing the construction of the auxiliary function, the unconstrained optimization problem is further solved by the steepest descent algorithm. From the geometric meaning of the gradient, it can be known that the negative gradient direction of the function at the approximate point is the direction in which the function value decreases fastest. Specific to the problem \( P_3 \), first, the first-order partial derivatives of the
\[ \frac{\partial F}{\partial \tau} = 1 - 2\mu \sum_{i=1}^{K} \max \left(0, t_p + t'_d + t'_m - \tau \right) \]  
\[ \frac{\partial F}{\partial f_k} = 2\mu \left[ \max \left(0, \sum_{i=1}^{K} f_i - f_s \right) \right] - \left( L_k / f_k \right) \max \left(0, t_p + t'_d + t'_m - \tau \right) \]  
\[ \frac{\partial F}{\partial p'_k} = 2\mu \left[ \max \left(0, t_p + t'_d + t'_m - \tau \right) \right] \frac{\partial f'_k}{\partial p'_k} \]  
\[ + \left( t_p + (p'_k + p'_c) \right) \frac{\partial f'_k}{\partial p'_k} \]  
\[ \times \left( 1 + t_p t'_d t'_k \right) \]  

where

\[ \frac{\partial f'_k}{\partial p'_p} = -L_k M p'_p t_p \beta_k \]  
\[ \left[ (1 + \text{SINR}_k) \left( 1 + \sum_{j=1}^{K} \beta_j p'_d \right) \right] \left[ 1 + t_p t'_d \right] \ln 2 \].

\[ \frac{\partial f'_k}{\partial p'_p} = 2\mu \left[ \max \left(0, t_p + t'_d + t'_m - \tau \right) \right] \]  
\[ + \left( t_p + (p'_k + p'_c) \right) \frac{\partial f'_k}{\partial p'_k} \]  
\[ \times \left( 1 + t'_d t'_m \right) \ln 2 \].

The detailed description and implementation process of the steepest descent algorithm based on the external penalty function (SDA-EPF) is given by Algorithm 1. Through SDA-EPF, an approximate optimal solution of the problem can be obtained.

**Algorithm 1 The Steepest Descent Algorithm Based on External Penalty Function (SDA-EPF)**

Given the initial point \((p_d, p_p, f, \tau)_{\text{(initial)}}\), \(\mu_{\text{(initial)}}\), the error accuracy \(\delta > 0\), \(\eta > 1\), \(n = 0\), \(m = 0\);

1: \((p_d, p_p, f, \tau)_{\text{(m)}}\) = \((p_d, p_p, f, \tau)_{\text{(initial)}}\), \(\mu_{\text{(m)}}\)
2: Construct auxiliary functions \(F(p_d, p_p, f, \tau)_{\text{(m)}}\), \(\mu_{\text{(m)}}\) in (21);
3: Calculate \(\nabla F((p_d, p_p, f, \tau)_{\text{(m)}})\) according to the first-order partial derivative in (23)-(26).

If \(||\nabla F((p_d, p_p, f, \tau)_{\text{(m)}})_{\text{(initial)}}|| \geq \delta\), update:

\[ p_d^{(m+1)} = p_d^{(m)} - t_m \frac{\partial F}{\partial p_d} \]  
\[ p_p^{(m+1)} = p_p^{(m)} - t_m \frac{\partial F}{\partial p_p} \]  
\[ f^{(m+1)} = f^{(m)} - t_m \frac{\partial F}{\partial f} \]  
\[ \tau^{(m+1)} = \tau^{(m)} - t_m \frac{\partial F}{\partial \tau} \]  

where \(t_m\) is the optimal step size, which satisfies:

\[ F(p_d, p_p, f, \tau)_{\text{(m+1)}} = F(p_d, p_p, f, \tau)_{\text{(m)}} - t_m \nabla F(p_d, p_p, f, \tau)_{\text{(m)}} - \tau^{(m+1)} \]  
\n\[ = \min F(p_d, p_p, f, \tau)_{\text{(m)}} - t_m \nabla F(p_d, p_p, f, \tau)_{\text{(m)}} - \tau^{(m+1)} \]  
\nwhere \(m = m + 1\), return to step 3. Otherwise:

\(4:\) if \(||\nabla F((p_d, p_p, f, \tau)_{\text{(m)}})|| < \delta\) and \(n > 0\), stop calculation, output the optimal solution:

\(5:\) \((p_d, p_p, f, \tau)_{\text{(m)}} = (p_d, p_p, f, \tau)_{\text{(m)}}\), go to step 4.

Otherwise:

\(6:\) \((p_d, p_p, f, \tau)_{\text{(m)}} = (p_d, p_p, f, \tau)_{\text{(m)}}\), go to step 1.

**C. IFOA-PFSA ALGORITHM**

Based on this, this paper adopts the improved local search scheme based on a novel improved fruit fly optimization algorithm to obtain the
local optimal solution of the optimization problem \( P_2 \). Here, we first introduce a new swarm intelligence optimization algorithm, namely the fruit fly optimization algorithm (FOA), which is based on the foraging behavior of the fruit fly. The fruit fly first uses its excellent sense of smell to search odors to find food sources. After flying near the food, use a keen vision to find food and companions, and fly to the gathering place of the companions. The implementation process of the traditional fruit fly optimization algorithm (TFOA) is shown in Algorithm 2.

**Algorithm 2 Traditional Fruit Fly Optimization Algorithm (TFOA)**

**Phase 1: Initialization**
1. Set the population size (popsize), the maximum number of iterations (maxgen), the position range of the fruit fly population (LR), and the single flight range (FR) of the fruit fly;
2. The position information of each individual in the fruit fly population is given by its corresponding \((X, Y)\) two-dimensional coordinates, and its initial position is:
   
   \[ X_{axis} = \text{rand}(LR), Y_{axis} = \text{rand}(LR); \]

**Phase 2: The olfactory search process**
3. When each fruit fly in the group uses its olfactory search, it is given a random flying direction and distance. The new position of the \( i \)-th fruit fly is:
   
   \[ X_i = X_{axis} + \text{rand}(FR), Y_i = Y_{axis} + \text{rand}(FR); \]
4. Since the location of the food odor source is unknown, the distance between the fruit fly and the origin is calculated by using
   
   \[ \text{Dist}_i = \sqrt{X_i^2 + Y_i^2}; \]
5. Use \( S_i = 1/\text{Dist}_i \) to calculate the judgment value of odor concentration;
6. Use \( \text{Smell}_i = \text{fitness} (S_i) \) to calculate the odor concentration of each fruit fly in the current population, where \( \text{fitness} \) represents the odor concentration judgment function, which is usually set as the objective function;
7. Select the fruit fly with the best odor concentration value in the current population, record its odor concentration value and the corresponding position as:

   \[ \text{[bestSmell, bestIndex]} = \min (\text{Smell}); \]

**Phase 3: The visual search process**
8. Maintain the optimal odor concentration value and the position information of the corresponding fruit fly. Other fruit flies in the group use vision to fly to this position, namely:

   \[ \text{Smell}_{\text{best}} = \text{bestSmell}, X_{axis} = X (\text{bestIndex}), Y_{axis} = Y (\text{bestIndex}); \]

**Phase 4: Repeat phases 2 and 3 until the number of iterations of the algorithm reaches maximum number of iterations (maxgen).**

Specific to the problem solved in this paper, the approximate optimal solution obtained by the steepest descent algorithm of the external penalty function is used as the initial point of the local search scheme based on the fruit fly optimization algorithm. By tracking the information of the current optimal solution to guide the next search of the fruit fly population, the group can conduct a local random search centered on the current optimal solution and search in a better direction. The point obtained by the steepest descent algorithm based on the external penalty function has been determined as the initial point of the fruit fly optimization algorithm, which can greatly reduce the population size and the maximum number of iterations in the random search process of the traditional fruit fly optimization algorithm, reducing the algorithm complexity. When selecting the initial point in Algorithm 1, due to the low implementation complexity of the algorithm, multiple random initial points can be selected and applied to the steepest descent algorithm based on the external penalty function, so that multiple good initial points can be obtained. Then multiple local optimal solutions are obtained, and the local optimal solutions are further compared to obtain the global sub-optimal solution. The global sub-optimal solution is used as the initial point for the fruit fly optimization algorithm. The implementation process of the proposed improved FOA based on external penalty function steepest descent algorithm (IFOA-PFSA) for solving problem \( P_1 \) is shown in Algorithm 3.

**Algorithm 3 Improved FOA Based on External Penalty Function Steepest Descent Algorithm (IFOA-PFSA)**

1. Initialize \( N \) initial points randomly, including the pilot transmission power, data transmission power, frequency allocation, and target values \( \tau \) of all users;
2. Construct the auxiliary function in problem \( P_3 \). Based on the current initial point, use Algorithm 1 to obtain \( N \) initial points;
3. Among the \( N \) initial points, select a point that satisfies the energy constraint and has the smallest objective function value;
4. Extract the pilot transmission power and data transmission power of all users from the point obtained in step 3 to form the initial point of the fruit fly optimization algorithm and start the fruit fly optimization algorithm;
5. Perform the fruit fly optimization algorithm using the objective function in problem \( P_1 \) as the fitness function;
6. Output the optimal allocation scheme \((p_d, p_v, f)\).

**IV. SIMULATION RESULTS**

This section analyzes and evaluates the performance of the proposed improved FOA based on external penalty function steepest descent algorithm (IFOA-PFSA). The traditional FOA algorithm (TFOA) is used as a comparison algorithm, as shown in Algorithm 2. The simulation parameters are shown in Table 1. The simulation analysis mainly focuses on indicators such as the number of users, the amount of user offload tasks, number of antennas, and algorithm complexity to evaluate the performance of different algorithms. The results are shown in Fig. 3-7. The data in the figure is...
TABLE 1. Simulation parameters.

| Parameter                                | Value        |
|------------------------------------------|--------------|
| Pilot transmission time                  | 10ms         |
| Maximum transmit power of each user      | 0.3W         |
| Power iteration step size                | 0.01W        |
| Circuit power consumption of each user   | 0.1W         |
| User and base station spacing            | uniform distribution: [10, 300] m |
| Base station antenna height              | 15m          |
| User antenna height                      | 1.65m        |
| Carrier frequency                        | 1.9GHz       |
| Transmission bandwidth                   | 20MHz        |
| User number change range                 | [20, 30)     |
| Energy constraint ceiling                | 0.2J         |
| Number of base station antennas          | 128          |
| User offload computing data              | 5000 bits    |
| $\sigma_0, d_1, d_2$                     | 8dB, 50m, 10m |

the average of the simulation results of 10,000 times. The number of fruit fly generations in IFOA-PFSA and TFOA in Fig. 3-6 is 10 and 100, respectively.

Fig. 3 is the curve of the user maximum offload computing delay change with the number of users of the two algorithms. The maximum CPU frequency of the server is 100 GHz. It can be seen from Fig. 3 that the maximum user offload computing delays of IFOA-PFSA and TFOA increase with the number of users. The reason is that on one hand, in a massive MIMO system with a given number of antennas, as the number of user increases, the interference between users during the transmission process increases, which leads to a decrease in the SINR of the achievable rate, leading to an increase in offload computing delay. On the other hand, due to the limitation of the total computing resources, the computing resources that each user can be allocated from the wireless MEC server will decrease as the number of users increases, resulting in an increase in the maximum offload computing time for users. Simulation results confirm that the performance of IFOA-PFSA is better than TFOA, and it becomes more and more significant as the number of users increases. It is shown that using the low-complexity-based steepest descent algorithm based on the external penalty function to find multiple initial points is beneficial to reduce the maximum offload calculation delay of the user.

Fig. 4 is the curve of the user maximum offload computing delay change with the user’s offload computing data of the two algorithms. The number of users is 25, and the maximum CPU frequency of the server is 100 GHz. As can be seen from Fig. 4, the user maximum offload computing delays of IFOA-PFSA and TFOA increase with the increase of the amount of user offload computing data. The reason is that on one hand, the increase in the amount of offload computing data will increase the transmission delay at the same achievable rate. On the other hand, an increase in the amount of data that needs to be processed by the same server computing resources will also increase computing latency. We can find that the performance of IFOA-PFSA is better than TFOA algorithm with the increase of the amount of offload computing data, which verifies the advantage of IFOA-PFSA in processing the original problem.

Fig. 5 is the curve of the user maximum offload computing delay change with the computing resources (namely the maximum CPU frequency of the server) of the two algorithms. The number of users is 25. As the frequency changes, the delay of the IFOA-PFSA algorithm is smaller than that of TFOA, even if the difference is small. It can be seen from Fig. 5 that the maximum offload computing delays of users of IFOA-PFSA and TFOA decrease with the increase of computing resources. The reason is that the more CPU frequencies allocated to users, the lower the computing delay. The tendency of the CPU frequency to change in $10^5 - 10^9$ is larger because the offload computing delay of each user can be reduced by increasing the computing resources, while when the CPU frequency is changed in $10^5 - 10^9$, the user offload computing delay declines slowly, the reason is that at this time the computing resources are large enough and the computing delay that can be optimized is no longer.
significant. Offload transmission has become the major factor in user offload computing delays.

Fig. 6 is the curve of the user maximum offload computing delay change with the number of antennas of the two algorithms. Set the number of users to 25, and the maximum CPU frequency of the server to 100 GHz. As is shown in Fig. 6, the maximum offload computing delays of users of IFOA-PFSA and TFOA decrease with the increase in the number of antennas, mainly because the increase in the number of antennas can bring greater spatial diversity gain and increase for each user SINR also improves the achievable rate of each user, thereby reducing transmission delay. When the number of antennas varies from 256 to 512, the delay decreased relatively small with the increase in the number of antennas. A reasonable number of antennas can be set according to the number of users and the amount of offload computing data to achieve a reasonable resource allocation.

Fig. 7 is the curve of the user maximum offload computing delay change with the number of generations of FOA. Fig. 7 is the curve of the user maximum offload computing delay change with the number of fruit fly generations of the two algorithms, reflecting the complexity of the algorithms and their convergence performance. The number of users is 25, and the maximum CPU frequency of the server is 100 GHz. As can be seen from Fig. 7, in IFOA-PFSA, the maximum offload computing delay of the user only needs to execute about 20 generations of the fruit flies to converge to the optimal value, while the TFOA algorithm needs to execute about 80 generations to converge. TFOA has a high search complexity due to too many generations, while IFOA-PFSA can reduce the number of fruit fly generations by using the initial point obtained by the steepest descent algorithm based on the external penalty function. It is verified that the proposed algorithm IFOA-PFSA is a low complexity algorithm with better performance. The computational complexity of the proposed algorithm is $O(K^3 + NT)$, where $N$ is the number of fruit flies, $T$ is the number of generations.

V. CONCLUSION

This paper combines mobile edge computing with massive MIMO and designs a single-cell multi-user massive MIMO-MEC system based on the study of joint resource allocation. In order to ensure user fairness, a non-convex non-linear optimization model of joint pilot allocation, data transmission process resource allocation, and resource allocation during the computing execution process with the goal of minimizing the maximum offload computing delay for all users is constructed. Since this problem cannot be solved directly, an improved fruit fly optimization algorithm based on the steepest descent algorithm of external penalty function (IFOA-PFSA) is proposed. Simulation analysis shows that when the number of users increases, the algorithm IFOA-PFSA has a more significant delay effect than TFOA when optimizing maximum user delay, and it can better reflect the superior performance of IFOA-PFSA when dealing with a large number of users. With the number of offload bits increases, and the performance of IFOA-PFSA is better than TFOA when processing a large amount of data.
The reason is that the IFOA-PFSA algorithm can find better secondary optimal points faster. When the number of antennas is less than 256, IFOA-PFSA has a significant delay effect in optimizing the user’s maximum delay, because IFOA-PFSA uses space gain more efficiently. In addition, when analyzing the complexity of the algorithm, IFOA-PFSA only needs to execute the fruit fly algorithm 20 times to converge immediately, while TFOA needs to execute 80 times, which verifies the low complexity of the proposed algorithm. This work mainly focuses on resource allocation within a single cell. In future studies, we will consider resource allocation in multiple cells and consider inter-cell interference.

REFERENCES

[1] M. Marjanovic, A. Antonic, and I. P. Zarko, “Edge computing architecture for mobile crowdsensing,” IEEE Access, vol. 6, pp. 10662–10674, 2018.
[2] X. Wu, W. Jiang, Y. Zhang, and W. Yu, “Online combinatorial based mechanism for MEC network resource allocation,” Int. J. Commun. Syst., vol. 32, no. 7, p. e3928, May 2019.
[3] N. Abbas, Y. Zhang, A. Taherkordi, and T. Skeie, “Mobile edge computing: A survey,” IEEE Internet Things J., vol. 5, no. 1, pp. 450–465, Feb. 2018.
[4] T. L. Marzetta, “Noncooperative cellular wireless with unlimited numbers of base station antennas,” IEEE Trans. Wireless Commun., vol. 9, no. 11, pp. 3590–3600, Nov. 2010.
[5] N. Zorba, A. I. Pérez-Neira, A. Foglar, and C. Verikoukis, “Cross layer QoS guarantees in multiuser WLAN systems,” Wireless Pers. Commun., vol. 51, no. 3, pp. 549–563, Nov. 2009.
[6] S. Sardellitti, G. Scutari, and S. Barbarossa, “Joint optimization of radio and computational resources for multicell mobile-edge computing,” IEEE Trans. Signal Inf. Process. Netw., vol. 1, no. 2, pp. 89–103, Jun. 2015.
[7] Z. Yang, Y. Du, C. Che, W. Wang, H. Mei, D. Zhou, and K. Yang, “Energy-efficient joint resource allocation algorithms for MEC-enabled emotional computing in urban communities,” IEEE Access, vol. 7, pp. 137410–137419, 2019.
[8] A. N. Al-Shawallih, A. Bagheri, and O. Simeone, “Joint uplink/downlink and offloading optimization for mobile cloud computing with limited backhaul,” in Proc. Annu. Conf. Inf. Sci. Syst. (CISS), Mar. 2016, pp. 424–429.
[9] P. Paymard and N. Mokari, “Resource allocation in PD-NOMA-based mobile edge computing system: Multiserver and multitask priority,” Trans. Emerg. Telecommun. Technol., vol. 1, p. e3631, May 2019.
[10] P. Paymard, S. Rezvani, and N. Mokari, “Joint task scheduling and uplink/downlink radio resource allocation in PD-NOMA based mobile edge computing networks,” Phys. Commun., vol. 32, pp. 160–171, Feb. 2019.
[11] H. V. Cheng, E. Bjoranson, and E. G. Larsson, “Optimal pilot and payload power control in single-cell massive MIMO systems,” IEEE Trans. Signal Process., vol. 65, no. 9, pp. 2363–2378, May 2017.
[12] Y. Wang, X. Lin, and M. Pedram, “A nested two stage game-based optimization framework in mobile cloud computing system,” in Proc. IEEE 7th Int. Symp. Service-Oriented Syst. Eng., Mar. 2013, pp. 494–502.
[13] V. Cardellini, V. De Nitto Personé, V. Di Valerio, F. Facchinei, V. Grassi, F. Lo Presti, and V. Piccialli, “A game-theoretic approach to computation offloading in mobile cloud computing,” Math. Program., vol. 157, no. 2, pp. 421–449, Jun. 2016.
[14] S. Barbarossa, E. Caci, M. Merluzzi, and E. Calvaneo-Sinistrari, “Enabling effective mobile edge computing using millimeterwave links,” in Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops), May 2017, pp. 367–372.
[15] P. Wang, C. Yao, Z. Zheng, G. Sun, and L. Song, “Joint task assignment, transmission, and computing resource allocation in multiplayer mobile edge computing systems,” IEEE Internet Things J., vol. 6, no. 2, pp. 2872–2884, Apr. 2019.
[16] K. Liang, L. Zhao, X. Zhao, Y. Wang, and S. Ou, “Joint resource allocation and coordinated computation offloading for fog radio access networks,” China Commun., vol. 13, no. 2, pp. 131–139, 2016.
[17] X. Chen, “Decentralized computation offloading game for mobile cloud computing,” IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 4, pp. 974–983, Apr. 2015.
[18] W. Yi, L. Yan, and L. I. Chunguo, “Energy-efficiency optimization of pilot duration and power allocation for downlink massive MIMO FDD systems,” J. Electron. Inf. Technol., vol. 39, no. 1, pp. 16–23, 2017.
[19] S. E. Mahmoudi, R. N. Uma, and K. P. Subbalakshmi, “Optimal joint scheduling and cloud offloading for mobile applications,” IEEE Trans. Cloud Comput., vol. 7, no. 2, pp. 301–313, Apr. 2019.
[20] A. Al-Shawallih, O. Simeone, A. Bagheri, and G. Scutari, “Joint uplink/downlink optimization for backhaul-limited mobile cloud computing with user scheduling,” IEEE Trans. Signal Inf. Process. Netw., vol. 3, no. 4, pp. 787–802, Dec. 2017.
[21] B. Huang and A. Guo, “Spectral and energy efficient resource allocation for massive MIMO HetNets with wireless backhaul,” IEEE Wireless Commun. Lett., vol. 8, no. 2, pp. 552–555, Apr. 2019.
[22] W. Hao, M. Zeng, Z. Chu, S. Yang, and G. Sun, “Energy-efficient resource allocation for mmWave massive MIMO HetNets with wireless backhaul,” IEEE Access, vol. 6, pp. 2457–2471, 2018.
[23] W.-T. Pan, “A new fruit fly optimization algorithm: Taking the financial distress model as an example,” Knowl.-Based Syst., vol. 26, pp. 69–74, Feb. 2012.
[24] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, “A survey of computation offloading for mobile systems,” Mobile Netw. Appl., vol. 18, no. 1, pp. 129–140, Feb. 2013.
[25] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, “Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks,” IEEE Signal Process. Mag., vol. 31, no. 6, pp. 45–55, Nov. 2014.
[26] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, “Cell-free massive MIMO versus small cells,” IEEE Trans. Wireless Commun., vol. 16, no. 3, pp. 1834–1850, Mar. 2017.

WENJIANG FENG received the Ph.D. degree in electrical engineering from Chongqing University, Chongqing, China, in 2000. He is currently a Professor with the College of Microelectronics and Communication Engineering, Chongqing University. His research interests include all aspects of MIMO communication, including limited feedback techniques, antenna design, interference management and full-duplex communication, cognitive radio, special mobile communication systems, and emergency communication. He is a Peer Review Expert of the Natural Science Foundation of China and is a Senior Member of the China Institute of Communications. He also serves as an Editorial Board Member of Data Communication, China.

JIALI ZHENG received the B.S. degree in communication engineering from Chongqing University, Chongqing, China, in 2018, where she is currently pursuing the master’s degree with the School of Microelectronics and Communication Engineering. Her current research interests include MIMO, MEC, and machine learning.

WEIHENG JIANG (Member, IEEE) received the Ph.D. degree in communication and information systems from Chongqing University, Chongqing, China, in 2015. He is currently a Tutor with the College of Microelectronics and Communication Engineering, Chongqing University. His main research interests include wireless communication and networking, mobile sensing and computing, and low-energy, and passive sensor networks. He serves as a Reviewer for international journals, such as International Journal of Distributed Sensor Network and TPC for international conferences such as PIMRC.