The Synergy of Edge and Central Cloud Computing with Wireless MIMO Backhaul

Xiaoyan Hu[†], Lifeng Wang[†], Kai-Kit Wong[†], Meixia Tao[‡], Yangyang Zhang[*], and Zhongbin Zheng[§]
[†]Department of Electronic and Electrical Engineering, University College London, London, UK
[‡]Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, P. R. China
[*]Kuang-Chi Institute of Advanced Technology, Shenzhen, P. R. China
[§]East China Institute of Telecommunications, Shanghai, P. R. China
Email:[†]{xiaoyan.hu.16, lifeng.wang, kai-kit.wong}@ucl.ac.uk, [‡]mxtao@sjtu.edu.cn,
[*]yangyang.zhang@kuang-chi.org, [§]ben@ecit.org.cn

***Abstract*—In this paper, the synergy of combining the edge and central cloud computing is studied in heterogeneous cellular networks (HetNets). Multi-antenna small base stations (SBSs) equipped with edge cloud servers offer computing services for user equipment (UEs) proximally, whereas a macro base station (MBS) provides central cloud computing services for UEs via wireless multiple-input multiple-output (MIMO) backhaul allocated to their associated SBSs. With task processing latency constraints for UEs, the network energy consumption is minimized through jointly optimizing the cloud selection, the UEs' transmit powers, the SBSs' receive beamformers, and the SBSs' transmit covariance matrices. A mixed-integer and non-convex optimization problem is formulated, and a decomposition algorithm is proposed to obtain a tractable solution iteratively. The simulation results confirm that great performance improvement can be achieved compared with the traditional scheme with central cloud computing only.**

***Index Terms*—Edge computing, central cloud computing, HetNets, MIMO, wireless backhaul.**

I. Introduction

The explosively increasing computing demand for resource-limited user equipment (UEs) has driven the emergence and development of edge computing, which has been regarded a promising technology to achieve high energy efficiency and low latency. The rationale behind edge computing is that cloud services can be provided at the edge of wireless networks, to liberate the UEs from heavy computation workload and prolong their battery lifetime [1, 2]. Recently, mobile edge computing (MEC) has been widely used in cellular networks, aiming at improving the energy efficiency or reducing the latency, e.g., [3–7]. The tradeoff between energy consumption and latency in information transmission and computation is studied in [3]. In [4], game-theoretical solutions are proposed to maximize the cell load and minimize the cost in terms of time and energy. Later in [5], time and frequency allocation problems are solved for improving energy efficiency. The work of [6] examines a single-cloudlet scenario, and a successive convex optimization approach is developed. A UAV-assisted MEC architecture is proposed in [7] to leverage the flexibility of UAV. The implementation of energy harvesting in MEC networks can further improve the system performance by providing sustainable energy supply for users and prolonging their lifetime, such as the works in [8–11].

However, the aforementioned works mainly focus on the small-scale edge computing networks. Actually, edge computing cannot entirely replace the traditional central cloud for the reason that edge computing provides limited processing and storage at the proximity of UEs but may be incapable of handling massive data processing. The latest white paper published by ETSI has further illustrated that edge computing and central cloud computing are highly complementary and significant benefits can be achieved when utilizing both [12].

Therefore, this paper studies the synergy of combining the edge and central cloud computing in a two-tier heterogeneous cellular network (HetNet), where UEs can offload their computing tasks to the small base stations (SBSs) with limited edge computing capabilities, or to the macro BS (MBS) providing central cloud computing services via wireless multiple-input multiple-output (MIMO) backhaul allocated to their associated SBSs [13]. Our aim is to minimize the network's total energy consumption under UEs' task processing latency constraints through jointly optimizing the cloud selection decisions, the UEs' transmit powers, the SBSs' receive beamforming vectors and transmit covariance matrices. The formulated problem is mixed-integer and non-convex, and an iterative algorithm is proposed to solve such a combinatorial problem properly. It is confirmed that the integrated edge and central cloud computing scheme proposed in this work can achieve better performance than the traditional central cloud computing scheme.

Notations—In this paper, the notations $(\cdot)^H$ and $(\cdot)^\dagger$ are conjugate transpose and conjugate operators, respectively. In addition, $[x]^+ = \max\{x, 0\}$. $\mathrm{eig}\{\mathbf{X}\}$ denotes the set of all the eigenvalues for matrix $\mathbf{X}$ and $\mathrm{eigvec}\{\cdot\}$ gives the eigenvector for a given eigenvalue of $\mathbf{X}$. $\langle \mathbf{X}_1, \mathbf{X}_2 \rangle \triangleq \Re\{\mathrm{tr}(\mathbf{X}_1^H \mathbf{X}_2)\}$, where $\Re\{\cdot\}$ is the real-value operator.

II. System Model and Problem Formulation

A two-tier HetNet is considered, in which an M-antenna MBS is fiber-optic connected to the central cloud with super computing capability and provides high-speed wireless backhaul to N SBSs with edge clouds offering limited computing capabilities. In each small cell, a SBS equipped with L antennas serves a single-antenna UE, and each UE has an atomic computation-intensive task which cannot be divided and has to be offloaded as a whole for computing. The case for serving multiple UEs in each small cell can be dealt with by using existing orthogonal multiple access techniques such as time-division. Let $\mathcal{N}=\{1,\ldots,N\}$ denote the set of the SBSs and UEs, and B^{a} and B^{b} denote the bandwidths allocated to the access and backhaul links, respectively.

Since the computing tasks offloaded by the UEs could be executed either at the edge cloud or central cloud, cloud selection needs to be appropriately determined before evaluating the computation latency and energy consumption. Let the binary indicator c_n denote the computing decision, where $c_n=1$ indicates edge computing, and $c_n=0$ indicates central cloud computing for each UE $n\in\mathcal{N}$. In the sequel, we will study the latency and energy consumption of the network, and then formulate the optimization problem for minimizing the network's total energy consumption under each UE's task processing latency constraint.

A. Transmission and Computing Latency

1) Access Transmission Latency: The uplink transmission rate for offloading the computation task of UE n to its serving SBS is given as

$$R_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)=B^{\mathrm{a}}\log_2\left(1+\gamma_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)\right) \tag{1}$$

with the signal-to-interference-plus-noise ratio (SINR)

$$\gamma_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)=\frac{p_n^{\mathrm{u}}|\mathbf{w}_n^H\mathbf{h}_{n,n}^{\mathrm{a}}|^2}{\sum_{i=1,i\neq n}^N p_i^{\mathrm{u}}|\mathbf{w}_n^H\mathbf{h}_{i,n}^{\mathrm{a}}|^2+|\mathbf{w}_n^H\mathbf{n}_n|^2}, \tag{2}$$

where $\mathbf{w}_n$ is the receive beamforming vector of the n-th SBS, $\mathbf{h}_{i,n}^{\mathrm{a}}\in\mathbb{C}^{L\times 1}$ is the channel vector between UE i and SBS n, $\mathbf{n}_n$ is a vector of additive white Gaussian noise with zero mean and variance σ_n^2, and $\mathbf{p}^{\mathrm{u}}\triangleq[p_1^{\mathrm{u}},\ldots,p_N^{\mathrm{u}}]^T\in\mathbb{R}^{N\times 1}$ denotes the transmit power vector of the UEs. Therefore, given an arbitrary offloaded computation task size of the n-th UE's, denoted as I_n (bits), its uplink transmission latency for task offloading to the SBS n can be calculated as

$$T_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)=\frac{I_n}{R_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)}. \tag{3}$$

2) Edge Computing Latency ($c_n=1$): Let f_n and ϑ_n denote the SBS n's CPU frequency and the number of CPU cycles per bit data required for computing UE n's task. Then the computation latency at the n-th SBS can be described as

$$T_n^{\mathrm{edge}}=\vartheta_n I_n/f_n. \tag{4}$$

3) Central Cloud Processing Latency ($c_n=0$): The central cloud processing latency results from backhaul transmission and task execution at the central cloud. Due to the central cloud's super computing capability, its computing time is much lower than edge computing, thus we assume that the central cloud computing time is negligible. Hence, the central cloud processing latency, i.e., the backhaul transmission latency for the n-th UE is calculated as

$$T_n^{\mathrm{central}}(\mathbf{Q})=\frac{I_n}{R_n^{\mathrm{b}}(\mathbf{Q})}, \tag{5}$$

where $R_n^{\mathrm{b}}(\mathbf{Q})$ is the backhaul transmission rate given by

$$R_n^{\mathrm{b}}(\mathbf{Q})=B^{\mathrm{b}}\log_2\det\left(\mathbf{I}+\Psi(\mathbf{Q}_{-n})^{-1}\mathbf{H}_n^{\mathrm{b}}\mathbf{Q}_n\left(\mathbf{H}_n^{\mathrm{b}}\right)^H\right), \tag{6}$$

with the noise-plus-interference covariance matrix denoted as $\Psi(\mathbf{Q}_{-n})=\sigma^2\mathbf{I}+\sum_{i=1,i\neq n}^N\mathbf{H}_i^{\mathrm{b}}\mathbf{Q}_i\left(\mathbf{H}_i^{\mathrm{b}}\right)^H$. In (6), $\mathbf{Q}_n$ is the transmit covariance matrix of SBS n, $\mathbf{Q}=\{\mathbf{Q}_n\}_{n=1}^N$ and $\mathbf{Q}_{-n}=\{\mathbf{Q}_i\}_{i=1,i\neq n}^N$ are the compact transmit covariance matrices and the compact transmit covariance matrices except $\mathbf{Q}_n$, respectively, and $\mathbf{H}_n^{\mathrm{b}}\in\mathbb{C}^{M\times L}$ is the backhaul channel matrix from SBS n to the MBS. Hence, the total latency for completing UE n's computation task can be described as

$$T_n^{\mathrm{total}}=T_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n)+c_nT_n^{\mathrm{edge}}+(1-c_n)\,T_n^{\mathrm{central}}(\mathbf{Q}). \tag{7}$$

In addition, it is assumed that the size of computing outputs (usually a few command bits) is small and the downlink overhead such as time and energy consumption for delivering them to the UEs is negligible and therefore ignored.

B. Energy Consumption

Energy consumption mainly comes from task offloading energy and task execution energy. Based on Section II-A, the amount of energy consumption for UE n to offload its computing tasks to its serving SBS is

$$E_n^{\mathrm{a}}=p_n^{\mathrm{u}}T_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}},\mathbf{w}_n). \tag{8}$$

If the task is executed by the edge cloud at the SBS n, the energy consumption is given by [8]

$$E_n^{\mathrm{edge}}=\varrho_n\vartheta_nI_nf_n^2, \tag{9}$$

where ϱ_n is the effective switched capacitance of the edge server n associated with the SBS n. Else, if the task is executed by the central cloud, we then have

$$E_n^{\mathrm{central}}=\mathrm{tr}\left(\mathbf{Q}_n\right)T_n^{\mathrm{central}}(\mathbf{Q})+\zeta_nE_n^{\mathrm{edge}}, \tag{10}$$

where ζ_n is the ratio of central cloud's energy consumption to that of the edge cloud for computing the same task, and it is related to the CPU frequency. We assume that $\zeta_n>1$ since the CPU frequency of the central cloud server should be much higher than edge servers. Thus, the total energy consumption for offloading and computation can be calculated as

$$E_{\mathrm{total}}=\sum_{n=1}^N\left(E_n^{\mathrm{a}}+c_nE_n^{\mathrm{edge}}+(1-c_n)\,E_n^{\mathrm{central}}\right). \tag{11}$$

C. Problem Formulation

We aim at minimizing the total energy consumption for offloading and computation under each UE's task processing latency constraint through jointly optimizing the cloud selection decisions ($\mathbf{c}=\{c_n\}_{n=1}^N$), UEs' transmit power vector ($\mathbf{p}^{\mathrm{u}}$), SBSs' receive beamformers ($\mathbf{w}=\{\mathbf{w}_n\}_{n=1}^N$), and the SBSs' transmit covariance matrices in ($\mathbf{Q}$). Hence, we can formulate the problem as follows:

$$\begin{aligned}\min_{\mathbf{c},\mathbf{p}^{\mathrm{u}},\mathbf{w},\mathbf{Q}}\quad & E_{\mathrm{total}} \qquad\qquad (12)\\ \text{s.t.}\quad & \mathrm{C1}: c_n\in\{0,1\},\ \ \forall n\in\mathcal{N},\\ & \mathrm{C2}: 0\leq p_n^{\mathrm{u}}\leq P_{\max}^{\mathrm{u}},\ \forall n\in\mathcal{N},\\ & \mathrm{C3}: \mathbf{Q}_n\succeq\mathbf{0},\ \forall n\in\mathcal{N},\\ & \mathrm{C4}: T_n^{\mathrm{total}}(\mathbf{c},\mathbf{p}^{\mathrm{u}},\mathbf{w},\mathbf{Q})\leq T_{\mathrm{th}},\ \forall n\in\mathcal{N},\end{aligned}$$

where C1 is the cloud selection constrains; C2 and C3 are constraints guaranteeing the non-negativeness of the parameters; C4 shows the UEs' delay constraints for completing their computation tasks, and $T_{\rm th}$ is a predefined threshold. According to the expression of $T_n^{\rm total}(\mathbf{c},\mathbf{p}^{\rm u},\mathbf{w},\mathbf{Q})$ in (7) and the definition of c_n in C1, the constraint C4 can be equivalently divided into the following two constraints

$$\mathrm{C4.1}: T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)+c_nT_n^{\rm edge}\le T_{\rm th},\forall n\in\mathcal{N}, \qquad (13)$$

$$\mathrm{C4.2}: T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)+\left(1-c_n\right)T_n^{\rm central}(\mathbf{Q})\le T_{\rm th},\forall n\in\mathcal{N}. \qquad (14)$$

In the following sections, we will mainly focus on the equivalent version of the above problem (12) with the processing latency constraints C4.1 and C4.2.

III. Algorithm Design

In fact, problem (12) is a mixed-integer optimization problem with coupled variables $\mathbf{c}$ and $\{\mathbf{p}^{\rm u},\mathbf{w},\mathbf{Q}\}$, which is non-convex and a NP-hard in general. To address this issue, a decomposition approach is adopted to solve (12) in an iterative manner. We first make the cloud selection decisions with given $\{\mathbf{p}^{\rm u},\mathbf{w},\mathbf{Q}\}$, and then optimize the transmit power ($\mathbf{p}^{\rm u}$), the beamformers ($\mathbf{w}$), and the covariance matrices ($\mathbf{Q}$) with a given cloud selection. A tractable solution can be finally obtained when the algorithm converges.

A. Edge or Central Cloud Decision

By relaxing $c_n\in\{0,1\}$ to $\widehat{c}_n\in[0,1]$, we find that given $\{\mathbf{p}^{\rm u},\mathbf{w},\mathbf{Q}\}$, problem (12) can be decomposed into

$$\begin{aligned}\min_{\widehat{\mathbf{c}}}\ & \sum_{n=1}^{N}\left(\widehat{c}_nE_n^{\rm edge}+\left(1-\widehat{c}_n\right)E_n^{\rm central}\right) \qquad (15)\\ \text{s.t.}\ & \widehat{\mathrm{C}}1:\widehat{c}_n\in\left[0,1\right],n\in\mathcal{N},\ \mathrm{C4.1},\ \mathrm{C4.2}.\end{aligned}$$

where $\widehat{\mathbf{c}}=\{\widehat{c}_n\}_{n\in\mathcal{N}}$. It is easy to note that problem (15) is a one-dimensional linear programming, and its solution is given in the following lemma.

Lemma 1. *The optimal cloud decision variable $\widehat{c}_n$ of problem* (15) *can be given in two cases:*

- *Case 1: If $E_n^{\rm edge}\le E_n^{\rm central}$, the objective function in problem* (15) *is a decreasing function of $\widehat{c}_n$, and thus the optimal $\widehat{c}_n^*$ is the maximum value that satisfies the constraints in* (15)*, i.e.,*

$$\widehat{c}_n^*=\left[\min\left\{\frac{T_{\rm th}-T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)}{T_n^{\rm edge}},1\right\}\right]^+. \qquad (16)$$

- *Case 2: If $E_n^{\rm edge}>E_n^{\rm central}$, the objective function in* (15) *is an increasing function of $\widehat{c}_n$, and the optimal $\widehat{c}_n^*$ is the minimum value that satisfies the constraints in* (15)*, i.e.,*

$$c_n^*=\left[1-\frac{T_{\rm th}-T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)}{T_n^{\rm central}(\mathbf{Q})}\right]^+. \qquad (17)$$

From Lemma 1, we can see that the edge/central cloud computing decision $\widehat{\mathbf{c}}^*$ is reliant on the optimal $\{\mathbf{p}^{\rm u},\mathbf{w},\mathbf{Q}\}$ of problem (12). Hence, we will respectively focus on obtaining the optimal $\{\mathbf{p}^{\rm u*},\mathbf{w}^*\}$ and $\mathbf{Q}^*$ based on a given cloud selection decision $\widehat{\mathbf{c}}$ in the following two subsections. The final cloud decision c^* is obtained by rounding $\widehat{\mathbf{c}}^*$.

B. UEs' Transmit Powers and SBSs' Receive Beamformers

With a given cloud decision $\widehat{\mathbf{c}}$ and the SBSs' transmit covariance matrix $\mathbf{Q}$, the optimal $\{\mathbf{p}^{\rm u*},\mathbf{w}^*\}$ can be obtained by solving the following subproblem of (12):

$$\begin{aligned}\min_{\mathbf{p}^{\rm u},\mathbf{w}}\ & \sum_{n=1}^{N}p_n^{\rm u}T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n) \qquad (18)\\ \text{s.t.}\ & \mathrm{C2},\ \ \widehat{\mathrm{C}}4:T_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)\le T_n^{(1)},\end{aligned}$$

where $T_n^{(1)}=[T_{\rm th}-\max\{\widehat{c}_nT_n^{\rm edge},(1-\widehat{c}_n)T_n^{\rm central}(\mathbf{Q})\}]^+$. The objective function of the subproblem (18) is weighted sum-of-ratios related to $\mathbf{p}^{\rm u}$ and $\mathbf{w}$, and thus this problem is non-convex and challenging to solve. We first decouple the interplay between $\mathbf{p}^{\rm u}$ and $\mathbf{w}$, and the following Lemma 2 presents the optimal $\mathbf{w}^*$ of problem (18) when $\mathbf{p}^{\rm u}$ is fixed.

Lemma 2. *With a given $\mathbf{p}^{\rm u}$, problem* (18) *with respect to (w.r.t.) $\mathbf{w}_n$ can be equivalently transformed into a generalized eigenvector problem, thus the optimal $\mathbf{w}_n^*$ is given by*

$$\mathbf{w}_n^*=\mathrm{eigvec}\left\{\max\left\{\mathrm{eig}\{\left(\mathbf{\Phi}_{-n}\right)^{-1}\mathbf{\Phi}_n\}\right\}\right\}, \qquad (19)$$

where $\mathbf{\Phi}_{-n}=\sum_{i=1,i\neq n}^{N}p_i^{\rm u}\mathbf{h}_{i,n}^{\rm a}(\mathbf{h}_{i,n}^{\rm a})^H+\sigma_n^2\mathbf{I}_L$ and $\mathbf{\Phi}_n=p_n^{\rm u}\mathbf{h}_{n,n}^{\rm a}(\mathbf{h}_{n,n}^{\rm a})^H$.

With the help of auxiliary variables $\mathbf{t}=\{t_n\}_{n=1}^N$, it can be verified that problem (18) over the UEs' transmit power vector $\mathbf{p}^{\rm u}$ for fixed $\mathbf{w}$ can be equivalently transformed as

$$\begin{aligned}\min_{\mathbf{p}^{\rm u},\mathbf{t}}\ & \sum_{n=1}^{N}I_nt_n \qquad (20)\\ \text{s.t.}\ & \mathrm{C2},\ \ \widehat{\mathrm{C}}4,\ \ \mathrm{C5}:p_n^{\rm u}-t_nR_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)\le 0,\ \forall n\in\mathcal{N}.\end{aligned}$$

Lemma 3. *The optimal solution $(\mathbf{p}^{\rm u*},\mathbf{t}^*)$ of problem* (20) *satisfies the Karush-Kuhn-Tucker (KKT) conditions of the following N $(n\in\mathcal{N})$ subproblems*

$$\begin{aligned}\min_{p_n^{\rm u}}\ & \left(\lambda_n+M_n\right)p_n^{\rm u}-\lambda_nt_nR_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n) \qquad (21)\\ \text{s.t.}\ & \widetilde{\mathrm{C}}2:0\le p_n^{\rm u}\le P_{\rm max}^{\rm u},\\ & \widetilde{\mathrm{C}}4:I_n/T_n^{(1)}-R_n^{\rm a}(\mathbf{p}^{\rm u},\mathbf{w}_n)\le 0,\end{aligned}$$

with

$$\begin{aligned}M_n=&\sum_{j=1,j\neq n}^{N}\lambda_jt_j\frac{B_{\rm a}}{\ln 2}\frac{\left(\gamma_j^{\rm a}\right)^2|\mathbf{w}_j^H\mathbf{h}_{n,j}^{\rm a}|^2}{p_j^{\rm u}|\mathbf{w}_j^H\mathbf{h}_{j,j}^{\rm a}|^2\left(1+\gamma_j^{\rm a}\right)}+ \qquad (22)\\ &\sum_{j=1,j\neq n}^{N}\mu_j\frac{\left(\gamma_j^{\rm a}\right)^2|\mathbf{w}_j^H\mathbf{h}_{n,j}^{\rm a}|^2}{p_j^{\rm u}|\mathbf{w}_j^H\mathbf{h}_{j,j}^{\rm a}|^2},\end{aligned}$$

where $\{\mu_n\}_{n=1}^N$ and $\{\lambda_n\}_{n=1}^N$ are the Lagrange multipliers associated with the constraints $\widehat{\mathrm{C}}4$ and C5 of problem (20)*, and $M_n=-\sum_{j\neq n}^{N}\lambda_jt_j\frac{\partial R_j^{\rm a}}{\partial p_n^{\rm u}}-\sum_{j\neq n}^{N}\mu_j\frac{\partial\gamma_j^{\rm a}}{\partial p_n^{\rm u}}$. For optimal $(\mathbf{p}^{\rm u*},\mathbf{w}^*)$, λ_n and t_n are respectively calculated as*

$$\lambda_n=\frac{I_n}{R_n^{\rm a}\left(\mathbf{p}^{\rm u*},\mathbf{w}_n^*\right)},\ t_n=\frac{p_n^{\rm u*}}{R_n^{\rm a}\left(\mathbf{p}^{\rm u*},\mathbf{w}_n^*\right)}. \qquad (23)$$

Given λ_n and t_n, subproblem (21) is convex w.r.t. p_n^{u}. Therefore, we have the following theorem.

Theorem 1. *The solution of subproblem* (21) *is given by*

$$p_n^{\mathrm{u}*} = \begin{cases} \dfrac{\tau}{\Lambda_n}, & \text{if } \Omega_n < \dfrac{\tau}{\Lambda_n}, \\ \Omega_n, & \text{if } \dfrac{\tau}{\Lambda_n} \le \Omega_n \le P_{\max}^{\mathrm{u}}, \\ P_{\max}^{\mathrm{u}}, & \text{if } \Omega_n > P_{\max}^{\mathrm{u}}, \end{cases} \tag{24}$$

$$\mu_n^* = \begin{cases} \dfrac{\lambda_n + M_n}{\Lambda_n} - \dfrac{B_{\mathrm{a}}}{\ln 2}\dfrac{\lambda_n t_n}{\tau+1}, & \text{if } \Omega_n < \dfrac{\tau}{\Lambda_n}, \\ 0, & \text{otherwise}, \end{cases} \tag{25}$$

$$\nu_n^* = \begin{cases} 0, & \text{if } \Omega_n \le P_{\max}^{\mathrm{u}}, \\ \dfrac{B_{\mathrm{a}}}{\ln 2}\dfrac{\lambda_n t_n}{P_{\max}^{\mathrm{u}} + 1/\Lambda_n} - \lambda_n - M_n, & \text{otherwise}, \end{cases} \tag{26}$$

where $\tau = 2^{\frac{I_n}{B^{\mathrm{a}} T_n^{(1)}}} - 1$, $\Lambda_n \triangleq \frac{|\mathbf{w}_n^H \mathbf{h}_{n,n}^{\mathrm{a}}|^2}{\sum_{i=1, i\neq n}^N p_i^{\mathrm{u}} |\mathbf{w}_n^H \mathbf{h}_{i,n}^{\mathrm{a}}|^2 + |\mathbf{w}_n^H \mathbf{n}_n|^2}$, $\Omega_n \triangleq \frac{B_{\mathrm{a}}}{\ln 2}\frac{\lambda_n t_n}{\lambda_n + M_n} - \frac{1}{\Lambda_n}$, *and* ν_n^* *and* μ_n^* *are the optimal Lagrange multipliers associated with the constraints* $\widetilde{\mathrm{C}}2$ *and* $\widetilde{\mathrm{C}}4$ *of problem* (21)[1], *respectively.*

Proof. See Appendix A. □

C. SBSs' Transmit Covariance Matrixes

For fixed cloud selection $\widehat{\mathbf{c}}$ and $\{\mathbf{p}^{\mathrm{u}}, \mathbf{w}\}$ obtained in the above subsection, the optimal $\mathbf{Q}^*$ can be obtained by solving the following subproblem:

$$\begin{aligned} \min_{\mathbf{Q}} \ & y(\mathbf{Q}) = \sum_{n=1}^N (1-\widehat{c}_n) \operatorname{tr}(\mathbf{Q}_n) T_n^{\mathrm{central}}(\mathbf{Q}) \\ \text{s.t.} \ & \mathrm{C3}, \ \widehat{\mathrm{C}}4.2: R_n^{\mathrm{b}}(\mathbf{Q}) \ge (1-\widehat{c}_n) I_n / T_n^{(2)}, \ \forall n \in \mathcal{N}, \end{aligned} \tag{27}$$

where $T_n^{(2)} = T_{\mathrm{th}} - T_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}}, \mathbf{w}_n)$. Problem (27) is non-convex because of the non-convex objective function and constraints in $\widehat{\mathrm{C}}4.2$. To solve it, a successive pseudoconvex approach is leveraged, which is noted for its fast convergence and parallel computation capability [14].

First, let $\mathbf{Q}^l$ denote the $\mathbf{Q}$ value in the l-th iteration. Hence, the non-convex terms $\operatorname{tr}(\mathbf{Q}_n) T_n^{\mathrm{central}}(\mathbf{Q})$ for $n \in \mathcal{N}$ in the objective function can be approximated as a pseudoconvex function at $\mathbf{Q}^l$, which is written as

$$\widehat{y}_n(\mathbf{Q}_n; \mathbf{Q}^l) \triangleq \frac{I_n \operatorname{tr}(\mathbf{Q}_n)}{R_n^{\mathrm{b}}(\mathbf{Q}_n; \mathbf{Q}^l)} + z_n(\mathbf{Q}_n), \ n \in \mathcal{N} \tag{28}$$

where $z_n(\mathbf{Q}_n) = \sum_{j\neq n} I_j \operatorname{tr}(\mathbf{Q}_j^l) \Big\langle (\mathbf{Q}_n - \mathbf{Q}_n^l), \nabla_{\mathbf{Q}_n^\dagger} \frac{1-c_j}{R_j^{\mathrm{b}}(\mathbf{Q}^l)} \Big\rangle$ is a function obtained by linearizing the non-convex function $\sum_{j\neq n}^N \operatorname{tr}(\mathbf{Q}_j) T_j^{\mathrm{central}}(\mathbf{Q})$ in $\mathbf{Q}_n$ at the point $\mathbf{Q}^l$ and $\nabla_{\mathbf{Q}_j^\dagger} \frac{1-c_j}{R_j^{\mathrm{b}}(\mathbf{Q}^l)}$ is the Jacobian matrix of $\frac{1-c_j}{R_j^{\mathrm{b}}(\mathbf{Q}^l)}$ w.r.t. $\mathbf{Q}_j^\dagger$. Based on (28), the objective function $y(\mathbf{Q})$ of problem (27) at $\mathbf{Q}^l$ can be approximated as

$$\widetilde{y}(\mathbf{Q}; \mathbf{Q}^l) = \sum_{n=1}^N (1-c_n) \widehat{y}_n(\mathbf{Q}_n; \mathbf{Q}^l). \tag{29}$$

It can be verified that $\widetilde{y}(\mathbf{Q}; \mathbf{Q}^l)$ is a pseudoconvex function and has the same gradient with $y(\mathbf{Q})$ at $\mathbf{Q} = \mathbf{Q}^l$.

Then, by rewriting the non-concave function $R_n^{\mathrm{b}}(\mathbf{Q})$ in $\widehat{\mathrm{C}}4.2$ as a difference of two concave functions equivalently as in (30a) and leveraging the first-order Taylor expansion at $\mathbf{Q}^l$ for the second function, $R_n^{\mathrm{b}}(\mathbf{Q})$ can be approximated as

$$R_n^{\mathrm{b}}(\mathbf{Q}) = B^{\mathrm{b}} \log_2 \det\left(\sigma^2 \mathbf{I} + \boldsymbol{\Xi}(\mathbf{Q})\right) - R_n^{\mathrm{b2}}(\mathbf{Q}) \tag{30a}$$

$$\begin{aligned} &\geqslant B^{\mathrm{b}} \log_2 \det\left(\sigma^2 \mathbf{I} + \boldsymbol{\Xi}(\mathbf{Q})\right) - R_n^{\mathrm{b2}}(\mathbf{Q}^l) - \\ &\sum_{j\neq n}^N \Big\langle (\mathbf{Q}_j - \mathbf{Q}_j^l), \nabla_{\mathbf{Q}_j^\dagger} R_n^{\mathrm{b2}}(\mathbf{Q}^l) \Big\rangle \triangleq \widetilde{R}_n^{\mathrm{b}}(\mathbf{Q}), \end{aligned} \tag{30b}$$

where $\boldsymbol{\Xi}(\mathbf{Q}) = \sum_{i=1}^N \mathbf{H}_i^{\mathrm{b}} \mathbf{Q}_i \left(\mathbf{H}_i^{\mathrm{b}}\right)^H$, $R_n^{\mathrm{b2}}(\mathbf{Q}) = B^{\mathrm{b}} \log_2 \det \left(\sum_{i\neq n}^N \mathbf{H}_i^{\mathrm{b}} \mathbf{Q}_i \left(\mathbf{H}_i^{\mathrm{b}}\right)^H + \sigma^2 \mathbf{I}\right)$, and $\nabla_{\mathbf{Q}_j^\dagger} R_n^{\mathrm{b2}}(\mathbf{Q}^l)$ is the Jacobian matrix of $R_n^{\mathrm{b2}}(\mathbf{Q}^l)$ w.r.t. $\mathbf{Q}_j^\dagger$. Here, $\widetilde{R}_n^{\mathrm{b}}(\mathbf{Q})$ expressed in (30b) is a concave function over $\mathbf{Q}$.

Therefore, at $\mathbf{Q}^l$, the original problem (27) can be approximately transformed as

$$\begin{aligned} \min_{\mathbf{Q}} \ & \widetilde{y}(\mathbf{Q}; \mathbf{Q}^l) \\ \text{s.t.} \ & \mathrm{C3}, \ \widetilde{\mathrm{C}}4.2: \widetilde{R}_n^{\mathrm{b}}(\mathbf{Q}) \ge (1-c_n) I_n / T_n^{(2)}, \ \forall n \in \mathcal{N}. \end{aligned} \tag{31}$$

The objective function in problem (31) is a sum of N pseudoconvex functions each containing a fractional function and a linear function. In addition, all the constraints in problem (31) are convex. Hence, by introducing a set of auxiliary variables for the N fractional functions in the objective function and leveraging the Dinkelbach-like algorithm [15], problem (31) can be transformed into a solvable convex optimization problem, which can be effectively solved by CVX [16] and owns provable convergence [14]. Let $\mathbf{Q}^{l*}$ represent the solution of problem (31) at the l-th iteration, and thus the value of $\mathbf{Q}$ in the next $(l+1)$-th iteration can be updated as

$$\mathbf{Q}^{l+1} = \mathbf{Q}^l + \varepsilon(l)(\mathbf{Q}^{l*} - \mathbf{Q}^l), \tag{32}$$

where $\mathbf{Q}^{l*} - \mathbf{Q}^l$ is the descent direction of $y(\mathbf{Q})$ and $\varepsilon(l)$ is the step size at the l-th iteration that can be obtained through the successive line search. Therefore, the solution of problem (27) can be iteratively obtained.

IV. Simulation Results

In this section, simulation results are presented to evaluate the performance of the proposed solution and show the effects of the key parameters including the uniform task size ($I = I_n$), the UEs' processing latency threshold (T_{th}), and the uniform SBSs' CPU frequency ($f = f_n$), in combination with the uniform ratio of energy consumption between central and edge cloud computing ($\zeta = \zeta_n$). The performance of traditional "Central-cloud-only" computing scheme is also given as a benchmark. Note that the total energy consumption shown in the following figures are averaged over 500 independent channel realizations. All the small-scale fading channel coefficients follow independent and identically complex Gaussian distribution with zero mean and unit variance. The pathloss between SBS and UE and between MBS and SBS are respectively set as $140.7 + 36.7 \log_{10} d(\mathrm{km})$ and $100.7 + 23.5 \log_{10} d(\mathrm{km})$ according to 3GPP TR 36.814 [17], where d is the distance

[1] $\widetilde{\mathrm{C}}4$ can be equivalently expressed as $\tau - \gamma_n^{\mathrm{a}}(\mathbf{p}^{\mathrm{u}}, \mathbf{w}_n) \le 0$, which is used in the derivation of Theorem 1.

TABLE I
SIMULATION PARAMETERS

Parameter	Symbol	Value
Bandwidth for an access or backhaul link	B^{a}, B^{b}	10 MHz
Noise power spectral density for an access or backhaul link	$\sigma_n^2, n \in \mathcal{N}$, σ^2	-174 dBm/Hz
Radius of the small cell, macro cell	r^{a}, r^{b}	50 m, 500 m
Number of of antennas for the MBS	M	16
Number of SBSs/UEs	N	6
Number of antennas for each SBS	L	2
UEs' maximum transmit power	$P_{\max}^{\mathrm{u}}$	23 dBm
Required CPU cycles per bit	$\vartheta_n, n \in \mathcal{N}$	300 cycles/bit
the effective switched capacitance of the SBS processor	$\varrho_n, n \in \mathcal{N}$	10^{-28}

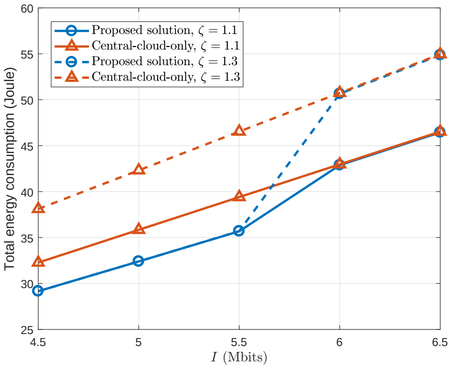

Fig. 1. The total energy consumption of the system versus UEs' uniform task size I: $T_{\mathrm{th}} = 0.3$ s and $f = 6$ GHz.

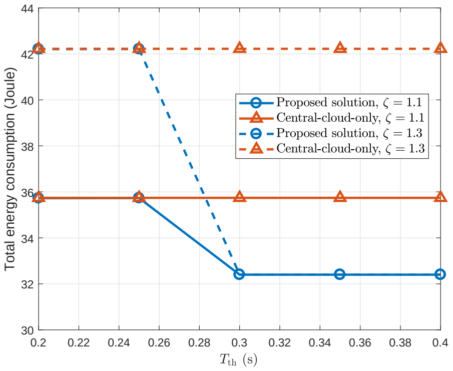

Fig. 2. The total energy consumption of the system versus the latency threshold of completing UEs' task T_{th}: $I = 5$ Mbits and $f = 6$ GHz.

between two nodes. The other basic simulation parameters are listed in Table I.

In Fig. 1, the total energy consumption of the system versus the uniform task sizes I for the cases of $\zeta = 1.1$ and $\zeta = 1.3$ are presented. It is easy to understand that computing more input data consumes more energy, and thus the energy cost of each scheme increases with I. We can see that the proposed solution is superior to the "Central-cloud-only" scheme in both cases. Besides, the gap between the proposed solution and the "Central-cloud-only" scheme becomes larger when ζ increases due to the fact that more energy will be consumed for central cloud computing with a larger ζ. We also observe that in both two cases, the results of using the proposed solution gradually approach to those of the Central-cloud-only scheme when I increases, which means that more UEs will select the central cloud for computing tasks with large sizes. The reason is that when the task is large, T_n^{edge} will be large accordingly, and thus the use of edge computing may no longer satisfy the latency constraint C4.1 of problem (12) due to the limited edge computing capability, and central cloud has to be chosen for computing so as to guarantee the UEs' computing tasks being completed within the given duration.

Fig. 2 depicts the total energy consumption of the system varying with the latency threshold of completing UEs' tasks (T_{th}) for the cases of $\zeta = 1.1$ and $\zeta = 1.3$. It is seen that the proposed solution is a non-increasing function of T_{th} and outperforms the baseline scheme with central cloud only. Similar to Fig. 1, the gaps between the proposed solution and the "Central-cloud-only" scheme become larger when ζ increases. We can see that the Central-cloud-only scheme is insensitive to T_{th}, and its performance is almost invariant thanks to its super computing capability for low computing latency. Note that in both two cases, the two schemes consume almost same amount of energy when T_{th} is small, e.g., $T_{\mathrm{th}} = 0.2$ s. The reason is that the latency constraint C4.1 cannot be met for the reason that $T_n^{\mathrm{edge}} > T_{\mathrm{th}}$ and only central cloud computing can be employed. With the increasing of T_{th}, the edge cloud computing becomes more feasible, and thus the consumed energy of the proposed solution decreases with T_{th} and its performance improvement becomes more obvious since more UEs are allowed to choose the energy-efficient edge cloud computing for large T_{th}.

Fig. 3 shows the total energy consumption of the system versus the SBSs' uniform CPU frequency f for the cases of $I = 3$ Mbits and $I = 5$ Mbits. According to the curves of these two cases, we see that the effect of f is heavily reliant on the computing task size I. When I is not large, i.e., $I = 3$ Mbits, the energy consumption of both two schemes may increase with f for the reason that the energy consumption of both the edge cloud computing and central cloud computing increase with f. However, when I becomes large, the network's energy consumption of the proposed solution may decrease with f in certain scenario, where there is an obvious decrease as $f \in [5, 6] \times 10^9$ Hz in the case of $I = 5$ Mbits. This is

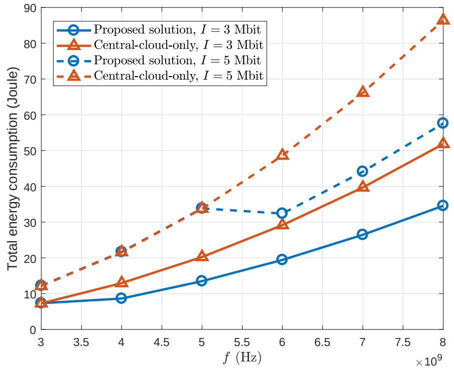

Fig. 3. The total energy consumption of the system versus the SBSs' uniform CPU frequency f: $I = 5$ Mbits and $\zeta = 1.5$.

because when f is small, e.g., less than 4×10^9 Hz, the edge cloud computing cannot satisfy the latency constraint C4.1, i.e., $T_n^{\text{edge}} > T_{\text{th}}$ and the central cloud computing becomes the only option; as f increases, the energy-efficient edge cloud computing becomes feasible for more UEs to save energy, thus the total energy cost will first decrease then increase with f.

V. Conclusion

In this paper, we studied the joint design of computing services when edge cloud computing and central cloud computing coexist in a two-tier HetNet with wireless MIMO backhaul. By optimizing the cloud selection decisions, the UEs' transmit powers, the SBSs' receive beamforming vectors and the transmit covariance matrices, the network's energy consumption can be minimized while meeting the latency constraints of completing UEs' computation tasks. An iterative algorithm was proposed, which can achieve better performance than the traditional scheme with central cloud computing only. The simulation results have further confirmed the performance enhancement of leveraging the proposed solution.

Acknowledgment

This work is supported in part by the U.K. EPSRC under Grants EP/K015893/1 and EP/L026031/1.

Appendix A: Proof of Theorem 1

The KKT conditions of subproblem (21) are expressed as

$$\lambda_n + M_n - \frac{B_{\text{a}}}{\ln 2} \frac{\lambda_n t_n \Lambda_n}{1 + \gamma_n^{\text{a}}} - \mu_n \Lambda_n + \nu_n = 0, \tag{A.1}$$

$$\nu_n \left(p_n^{\text{u}} - P_{\max}^{\text{u}}\right) = 0. \tag{A.2}$$

$$\mu_n \left(\tau - \gamma_n^{\text{a}}\right) = 0, \tag{A.3}$$

From (A.1), we see that the optimal $p_n^{\text{u}*}$ meets

$$p_n^{\text{u}*} = \frac{B_{\text{a}}}{\ln 2} \frac{\lambda_n t_n}{\lambda_n + M_n - \mu_n^* \Lambda_n + \nu_n^*} - \frac{1}{\Lambda_n}, \tag{A.4}$$

where ν_n^* and μ_n^* satisfy the KKT conditions (A.2) and (A.3), respectively. To explicitly obtain $\{p_n^{\text{u}*}, \mu_n^*, \nu_n^*\}$, we need to consider the following cases:

- Case 1: When $p_n^{\text{u}*} \in \left(\frac{\tau}{\Lambda_n}, P_{\max}^{\text{u}}\right)$, $\nu_n^* = \mu_n^* = 0$ according to (A.2) and (A.3). In this case, $p_n^{\text{u}*} = \Omega_n$ with $\Omega_n = \frac{B_{\text{a}}}{\ln 2} \frac{\lambda_n t_n}{\lambda_n + M_n} - \frac{1}{\Lambda_n}$ according to (A.4). Therefore, if $\Omega_n \in \left[\frac{\tau}{\Lambda_n}, P_{\max}^{\text{u}}\right]$, $p_n^{\text{u}*} = \Omega_n$ and $\nu_n^* = \mu_n^* = 0$.
- Case 2: If $\Omega_n < \frac{\tau}{\Lambda_n}$, it is seen from (A.4) that $\mu^* > 0$. In this case, $p_n^{\text{u}*} = \frac{\tau}{\Lambda_n}$ and $\nu_n^* = 0$ according to (A.2) and (A.3). Substituting $p_n^{\text{u}*} = \frac{\tau}{\Lambda_n}$ and $\nu_n^* = 0$ into (A.4), we obtain $\mu_n^* = \frac{\lambda_n + M_n}{\Lambda_n} - \frac{B_{\text{a}}}{\ln 2} \frac{\lambda_n t_n}{\tau + 1}$
- Case 3: If $\Omega_n > P_{\max}^{\text{u}}$, it is seen from (A.4) that $\nu_n^* > 0$. In this case, $p_n^{\text{u}*} = P_{\max}^{\text{u}}$ and $\mu_n^* = 0$ according to (A.2) and (A.3). Substituting $p_n^{\text{u}*} = P_{\max}^{\text{u}}$ and $\mu_n^* = 0$ into (A.4), we obtain $\nu_n^* = \frac{B_{\text{a}}}{\ln 2} \frac{\lambda_n t_n}{P_{\max}^{\text{u}} + 1/\Lambda_n} - \lambda_n - M_n$.

Thus, we get the optimal $\{p_n^{\text{u}*}, \mu^*, \nu_n^*\}$ shown in **Theorem** 1.

References

[1] S. Barbarossa, S. Sardellitti, and P. D. 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.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.

[3] O. Muñz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Tech.*, vol. 64, no. 10, pp. 4738–4755, Oct. 2015.

[4] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Trans. Netw.*, vol. 24, no. 5, pp. 2795–2808, Oct. 2016.

[5] C. You, K. Huang, H. Chae, and B. H. Kim, "Energy-efficient resource allocation for mobile-edge computation offloading," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1397–1411, Mar. 2017.

[6] A. Al-Shuwaili and O. Simeone, "Energy-efficient resource allocation for mobile edge computing-based augmented reality applications," *IEEE Wireless Commun. Lett.*, vol. 6, no. 3, pp. 398–401, June 2017.

[7] X. Hu, K.-K. Wong, K. Yang, and Z. Zheng, "UAV-assisted relaying and edge computing: Scheduling and trajectory optimization," *arXiv preprint arXiv:1812.02658*, 2018.

[8] C. You, K. Huang, and H. Chae, "Energy efficient mobile cloud computing powered by wireless energy transfer," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1757–1771, May 2016.

[9] X. Hu, K. K. Wong, and K. Yang, "Wireless powered cooperation-assisted mobile edge computing," *IEEE Trans. Wireless Commun.*, vol. 17, no. 4, pp. 2375–2388, Apr. 2018.

[10] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[11] S. Bi and Y. J. A. Zhang, "Computation rate maximization for wireless powered mobile-edge computing with binary computation offloading," *IEEE Trans. Wireless Commun.*, pp. 1–14, Early Access Articles 2018.

[12] ETSI White Paper No. 23: "Cloud RAN and MEC: A perfect pairing", Feb. 2018.

[13] X. Hu, L. Wang, K.-K. Wong, Y. Zhang, Z. Zheng, and M. Tao, "Edge and central cloud computing: A perfect pairing for high energy efficiency and low-latency," *arXiv preprint arXiv:1806.08943*, 2018.

[14] Y. Yang and M. Pesavento, "A unified successive pseudoconvex approximation framework," *IEEE Trans. Signal Process.*, vol. 65, no. 13, pp. 3313–3328, July 2017.

[15] A. Zappone and E. Jorswieck, *Energy Efficiency in Wireless Networks via Fractional Programming Theory*. Now Foundations and Trends, 2015, vol. 11, no. 3-4.

[16] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," 2008.

[17] 3GPP TR 36.814 v9.2.0, "3rd generation partnership project: Technical specification group radio access network: Evolved univerisal terrestrial radio access (E-UTRA): further advancements for E-UTRA physical layer aspects," Mar. 2017.