Joint Uplink/Downlink and Offloading Optimization for Mobile Cloud Computing with Limited Backhaul

Ali Najdi Al-Shuwaili, Alireza Bagheri, and Osvaldo Simeone
Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark 07102, NJ, USA.
Email: ana24@njit.edu, ab745@njit.edu, and osvaldo.simeone@njit.edu

***Abstract*—Mobile cloud computing enables the offloading of computationally heavy applications, such as for gaming, object recognition or video processing, from mobile users (MUs) to a cloud server connected to wireless access points. The optimization of the operation of a mobile cloud computing system amounts to the problem of minimizing the energy required for offloading across all MUs under latency constraints at the application layer. In a scenario with multiple MUs transmitting over a shared wireless medium across multiple cells, this problem requires the management of interference for both the uplink, through which MUs offload the data needed for computation in the cloud, and for the downlink, through which the outcome of the cloud computation are fed back to the MUs, as well as the allocation of backhaul resources for communication between wireless edge and cloud and of computing resources at the cloud. In this paper, this problem is formulated for general multi-antenna, or MIMO, channels, and tackled by means of successive convex approximation methods. The numerical results illustrate the advantages of a joint allocation of computing and communication resources.**

***Index Terms*—Mobile cloud computing, 5G, successive convex approximation, application offloading, backhaul.**

I. Introduction

Mobile cloud computing enables the offloading of computationally heavy applications, such as for gaming, object recognition, video processing, or virtual reality, from mobile users (MUs) to a cloud server connected to wireless access points [1], [2]. Given the battery-limited nature of mobile devices, mobile cloud computing is deemed to be an important enabler for the provision of advanced services [3]. When studied purely at the application layer, the optimization of a mobile cloud computing system entails the fine-grained decision of which subtasks of the call graph of a given application should be offloaded as a function of the latency constraints of the application with the aim of minimizing energy expenditure (see, e.g., [4]). Given that offloading requires transmission and reception on the wireless interface, a more systematic approach involves the joint optimization of offloading decisions and communication parameters, such as power allocation [5], [6].

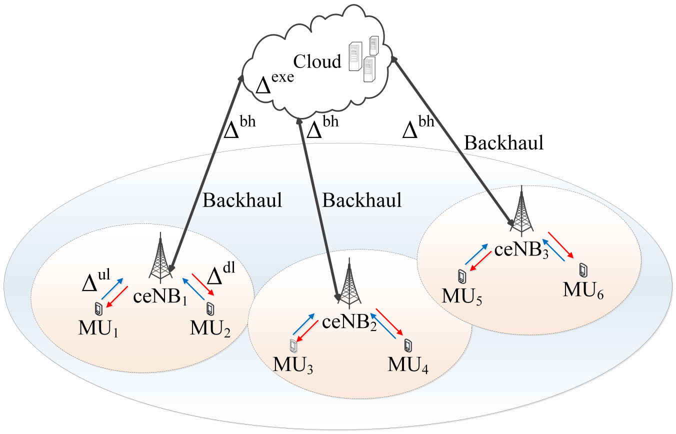

Fig. 1: Illustration of the system model.

While the mentioned problem formulations encompass the operation of a single MU, in a scenario with multiple MUs transmitting over a shared wireless medium across multiple cells, the design of a mobile cloud computing system requires: (*i*) the management of interference for the uplink, through which MUs offload the data needed for computation in the cloud; (*ii*) the management of interference for the downlink, through which the outcome of the cloud computations are fed back to the MUs; (*iii*) the allocation of backhaul resources for communication between wireless edge and cloud; and (*iv*) the allocation of computing resources at the cloud. In prior works [7], [8], a problem formulation that includes elements (*i*) and (*iv*) was studied, and the resulting problem tackled by means of the successive convex approximation (SCA) method [9].

In this paper, a problem formulation is considered that accounts for all aspects (*i*)-(*iv*) mentioned above for general multi-antenna, or MIMO, channels, in both uplink and downlink. This work is mostly motivated by the importance of taking into accout backhaul capacity limitations in the system design, since backhaul is well understood to be often the bottleneck in modern dense network deployments [10]. Moreover, unlike [9], we also explicitly model the optimization of the downlink for downloading the outcome of the optimization. The problem is addressed by adapting the SCA method to the set-up under study.

The rest of the paper is organized as follows. Section II, presents the system model and the problem formulation. The suggested iterative SCA scheme is explained in Section III. The simulation results are shown in Section IV and conclusions are finally provided in Section V.

II. System Model and Problem Formulation

In this section, we first describe the system model, and then the problem formulation.

A. System Model

We consider a network composed of N_c small cells which generalizes the system model in [7] by accounting also for limited backhaul resources and for downlink transmissions as shown in Fig. 1.

In each cell $n = 1, ..., N_c$, there is a small-cell base station, referred to as small-cell cloud-enhanced e-Node B (ceNB), which is connected to a common server, or pool of servers, named cloud, that provides computational resources. Each ceNB serves K Mobile Users (MUs) in orthogonal spectral resources, say in the frequency domain. We denote by i_n the MU in cell n that is scheduled on the ith spectral resource, and by $\mathcal{I} \triangleq \{i_n : i = 1, ..., K, n = 1, ..., N_c\}$ the set of all the users. Each MU i_n and ceNB n is equipped with $N_{T_{i_n}}$ transmit and N_{R_n} receive antenna, respectively. Note that MUs in different cells that are scheduled on the same spectral resources interfere with one other.

Each MU i_n wishes to run an application within a given maximum latency T_{i_n}. The application to be executed is characterized by the number V_{i_n} of CPU cycles necessary to complete it, by the number $B^I_{i_n}$ of input bits, and by the number $B^O_{i_n}$ of output bits encoding the result of the computation. Each MU offload computations to the ceNB in the same cell, as long as the latency constraint is satisfied.

We next derive energy and latency resulting from an offloading decision at all MUs. The offloading latency consist of the time $\Delta^{ul}_{i_n}$ needed for the MU to transmit the input bits to its ceNB in the uplink; the time $\Delta^{exe}_{i_n}$ necessary for the cloud to execute the instructions; the round-trip time $\Delta^{bh}_{i_n}$ for exchanging information between ceBN and the cloud through the backhaul link; and the time $\Delta^{dl}_{i_n}$ to send the result back to the MU in the downlink. We can hence write the total offloading latency for MU i_n as

$$\Delta_{i_n} = \Delta^{ul}_{i_n} + \Delta^{exe}_{i_n} + \Delta^{bh}_{i_n} + \Delta^{dl}_{i_n}. \tag{1}$$

The energy E_{i_n} of each MU i_n instead depends only on the power used for transmission in the uplink. These latency and energy terms are computed as a function of the radio and computational resources in the following.

1) *Uplink transmission:* The optimization variables at the physical layer for the uplink are the users' transmit covariance matrices $\mathbf{Q}^{ul} \triangleq (\mathbf{Q}^{ul}_{i_n})_{i_n\in\mathcal{I}}$, where $\mathbf{Q}^{ul}_{i_n} = \mathbb{E}[\mathbf{x}^{ul}_{i_n}\mathbf{x}^{ul^H}_{i_n}]$ with $\mathbf{x}^{ul}_{i_n} \sim \mathcal{CN}(\mathbf{0}, \mathbf{Q}^{ul}_{i_n})$ being the signal transmitted by the user i_n. These matrices are subject to power budget constraints so that the set of feasible uplink covariance matrix is given by

$$\mathcal{Q}^{ul}_{i_n} \triangleq \left\{\mathbf{Q}^{ul}_{i_n} \in C^{N_{T_{in}}\times N_{T_{in}}} : \mathbf{Q}^{ul}_{i_n} \succeq 0, \mathrm{tr}(\mathbf{Q}^{ul}_{i_n}) \leq P^{ul}_{i_n}\right\} \tag{2}$$

where $P^{ul}_{i_n}$ is the maximum allowed transmit energy per symbol of MU i_n. For any given profile $\mathbf{Q}^{ul} \triangleq (\mathbf{Q}^{ul}_{i_n})_{i_n\in\mathcal{I}}$, the achievable transmission rate of MU i_n in bits per symbol can be written as

$$r^{ul}_{i_n}(\mathbf{Q}) = \log_2 \det\Big(\mathbf{I} + \mathbf{H}^H_{i_n}\mathbf{R}^{ul}_n(\mathbf{Q}_{-i_n})^{-1}\mathbf{H}_{i_n}\mathbf{Q}^{ul}_{i_n}\Big), \tag{3}$$

where

$$\mathbf{R}^{ul}_n(\mathbf{Q}_{-i_n}) \triangleq \sigma^2_w\mathbf{I} + \sum_{i_m\in\mathcal{I},m\neq n} \mathbf{H}_{i_m n}\mathbf{Q}^{ul}_{i_m}\mathbf{H}^H_{i_m n} \tag{4}$$

is the covariance matrix of the sum of the noise and of the inter-cell interference affecting reception at the ceNB in the ith spectral resources; $\mathbf{H}_{i_n}$ is the uplink channel matrix for MU i_n to the ceNB in the cell n, whereas $\mathbf{H}_{i_m n}$ is the cross channel matrix between the interfering MU i_m in the cell m and the ceNB in cell n. The time, in seconds, necessary for user i in cell n to transmit the input bits $B^I_{i_n}$ to its ceNB in the uplink is then

$$\Delta^{ul}_{i_n}\left(\mathbf{Q}^{ul}\right) = \frac{B^I_{i_n}}{W^{ul} r^{ul}_{i_n}\left(\mathbf{Q}^{ul}\right)}, \tag{5}$$

where W^{ul} is the uplink channel bandwidth allocated to each one of the orthogonal spectral resources. The corresponding energy consumption due to offloading is defined as

$$E_{i_n}\left(\mathbf{Q}^{ul}\right) = B^I_{i_n}\frac{\mathrm{tr}\left(\mathbf{Q}^{ul}_{i_n}\right)}{r^{ul}_{i_n}\left(\mathbf{Q}^{ul}\right)}. \tag{6}$$

2) *Downlink transmission:* The optimization variables for the downlink are ceNBs' transmit covariance matrices $(\mathbf{Q}^{dl}_{i_n})_{i=1,...,K}$, which are subject to per-ceNB power constraints and hence must belong to the set

$$\mathcal{Q}^{dl}_n \triangleq \left\{(\mathbf{Q}^{dl}_{i_n})_{i=1,...,K} \in C^{N_{T_{in}}\times N_{T_{in}}} : \sum_{i=1}^{K}\mathrm{tr}(\mathbf{Q}^{dl}_{i_n}) \leq P^{dl}_n\right\}. \tag{7}$$

Similar to the uplink, we can write the achievable rate in bits per symbol for each MU in the downlink and the corresponding required transmission time as

$$r^{dl}_{i_n}(\mathbf{Q}^{dl}) = \log_2 \det\Big(\mathbf{I} + \mathbf{G}^H_{i_n}\mathbf{R}^{dl}_n(\mathbf{Q}^{dl}_{-i_n})^{-1}\mathbf{G}_{i_n}\mathbf{Q}^{dl}_{i_n}\Big), \tag{8}$$

with

$$\mathbf{R}^{dl}_n(\mathbf{Q}^{dl}_{-i_n}) \triangleq \sigma^2_w\mathbf{I} + \sum_{i_m\in\mathcal{I},m\neq n} \mathbf{G}_{i_m n}\mathbf{Q}^{dl}_{i_m}\mathbf{G}^H_{i_m n}, \tag{9}$$

and

$$\Delta^{dl}_{i_n}\left(\mathbf{Q}^{dl}\right) = \frac{B^O_{i_n}}{W^{dl} r^{dl}_{i_n}\left(\mathbf{Q}^{dl}\right)}, \tag{10}$$

where $\mathbf{G}_{i_n}$ is the downlink channel matrix between the ceNB in cell n and the MU i_n and $\mathbf{G}_{i_m n}$ is the cross channel matrix between the interfering MU i_m in the cell m and the ceNB in cell n; and W^{dl} is the downlink channel bandwidth. Note that (7)-(8) implicitly assume that the downlink spectral resources are allocated to the MUs in the same way as for the uplink, so that MUs i_n for $n = 1, ..., N_c$ are mutually interfering in both uplink and downlink. This assumption can be easily alleviated at the cost of introducing additional notation.

3) *Cloud processing:* Let the capacity in terms of number of CPU cycles per second of the cloud be F_c. Moreover, let $f_{i_n} \geq 0$ be the fraction of the processing power F_c assigned to user i_n, so that $\sum_{i_n \in \mathcal{I}} f_{i_n} \leq 1$. The time needed to run V_{i_n} CPU cycles for user i_n remotely is then

$$\Delta_{i_n}^{exe}(f_{i_n}) = \frac{V_{i_n}}{f_{i_n} F_c}. \tag{11}$$

4) *Backhaul transmission:* We denote as C_n^{ul} the capacity in bits per second of the backhaul connecting the ceNB in cell n with the cloud, and as C_n^{dl} the capacity in bits per second of the backhaul connecting the cloud with the ceNB in cell n. Let $c_{i_n}^{ul}, c_{i_n}^{dl} \geq 0$ be the fraction of the backhaul capacities C_n^{ul} and C_n^{dl}, respectively, allocated to the ith MU in cell n. We then have the constraint $\sum_i c_{i_n}^{ul} \leq 1$ and $\sum_i c_{i_n}^{dl} \leq 1$. Moreover, the time delay due to the backhaul transfer between ceNB n and the femtocloud in both directions is given as

$$\Delta_{i_n}^{bh}(c_{i_n}^{ul}, c_{i_n}^{dl}) = \frac{B_{i_n}^I}{c_{i_n}^{ul} C_n^{ul}} + \frac{B_{i_n}^O}{c_{i_n}^{dl} C_n^{dl}}. \tag{12}$$

B. Problem Formulation

The optimal offloading problem can be stated as the minimization of the sum of the energies spent by all MUs to run their applications remotely, subject to individual latency and power constraint. Mathematically, this problem can be written as

$$\begin{array}{lll}
\underset{\mathbf{Q}^{ul},\mathbf{Q}^{dl},\mathbf{f},\mathbf{c}^{ul},\mathbf{c}^{dl}}{\text{minimize}} & E\left(\mathbf{Q}^{ul}\right) & = \sum_{i_n \in \mathcal{I}} E_{i_n}\left(\mathbf{Q}_{i_n}^{ul}, \mathbf{Q}_{-n}^{ul}\right) \\
 & & = \sum_{i_n \in \mathcal{I}} B_{i_n}^I \frac{\text{tr}\left(\mathbf{Q}_{i_n}^{ul}\right)}{r_{i_n}^{ul}(\mathbf{Q}^{ul})} \\
s.t. \quad \mathbf{C.1} & \multicolumn{2}{l}{\frac{B_{i_n}^I}{W^{ul} r_{i_n}^{ul}(\mathbf{Q}^{ul})} + \frac{B_{i_n}^I}{c_{i_n}^{ul} C_n^{ul}} + \frac{V_{i_n}}{f_{i_n} F_c}} \\
 & \multicolumn{2}{l}{+ \frac{B_{i_n}^O}{W^{dl} r_{i_n}^{dl}(\mathbf{Q}^{dl})} + \frac{B_{i_n}^O}{c_{i_n}^{dl} C_n^{dl}} \leq T_{i_n}} \\
\quad\quad \mathbf{C.2} & \multicolumn{2}{l}{f_{i_n} \geq 0, \sum_{i_n \in \mathcal{I}} f_{i_n} = 1} \\
\quad\quad \mathbf{C.3} & \multicolumn{2}{l}{c_{i_n}^{ul}, c_{i_n}^{dl} \geq 0, \sum_i c_{i_n}^{ul} = 1, \sum_i c_{i_n}^{dl} = 1} \\
\quad\quad \mathbf{C.4} & \multicolumn{2}{l}{\mathbf{Q}_{i_n}^{ul} \in \mathcal{Q}_{i_n}^{ul}, \mathbf{Q}_{i_n}^{dl} \in \mathcal{Q}_n^{dl}, \forall i_n \in \mathcal{I}}
\end{array} \tag{P.1}$$

where constraint C.1 enforces that the latency for any MU i_n be less than or equal to the maximum tolerable delay of T_{i_n} seconds; C.2 imposes the mentioned limit on the cloud computational resources; C.3 enforces the limited backhaul capacities in uplink and downlink; and C.4 guarantee that power budget constraint on the radio resources of both uplink and downlink is satisfied. Note that problem (P.1) depends only on the ratios $B_{i_n}^I/W^{ul}$, $B_{i_n}^I/C_n^{ul}$, V_{i_n}/F_c, $B_{i_n}^O/W^{dl}$ and $B_{i_n}^O/C_n^{dl}$. Problem (P.1) is not convex due to the non-convexity of the objective function and the constraint C.1. Therefore, in the next section, we explore an efficient algorithm based on SCA that aim at obtaining an effective suboptimal solution.

III. Successive Convex Approximation Optimization

Problem (P.1) is non-convex due to the non-convexity of the objective function and constraint C.1. To address this issue, we apply here the successive convex approximation (SCA) method proposed in [9] and used in [7] to tackle problem (P.1) only over the uplink precoding variables $\mathbf{Q}^{ul}$ and computing resource vector $\mathbf{f}$. To this end, we identify: (i) a family of strongly convex approximations $\tilde{E}\left(\mathbf{Q}^{ul}; \mathbf{Q}^{ul}(v)\right)$ of $E(\mathbf{Q}^{ul})$, which are parameterized by a current iterate $\mathbf{Q}^{ul}(v)$ with the key property that $\nabla_{\mathbf{Q}^{ul*}} \tilde{E}\left(\mathbf{Q}^{ul}; \mathbf{Q}^{ul}\right) = \nabla_{\mathbf{Q}^{ul*}} E\left(\mathbf{Q}^{ul}\right)$ for any feasible uplink precoding profile $\mathbf{Q}^{ul}$, where $\nabla_{\mathbf{Q}^*} f(\mathbf{Q})$ represents the conjugate gradient of function $f(\mathbf{Q})$; (ii) a family of convex upper bounds $\tilde{g}_{i_n}\left(\mathbf{Q}^{ul}, \mathbf{Q}^{dl}, f_{i_n}, \mathbf{c}_{i_n}^{ul}, \mathbf{c}_{i_n}^{dl}; \mathbf{Q}^{ul}(v), \mathbf{Q}^{dl}(v)\right) \geq g_{i_n}\left(\mathbf{Q}^{ul}, \mathbf{Q}^{dl}, f_{i_n}, \mathbf{c}_{i_n}^{ul}, \mathbf{c}_{i_n}^{dl}\right)$, parameterized by the current iterate $\mathbf{Q}^{ul}(v)$ and $\mathbf{Q}^{dl}(v)$ of the constraint C.1.

A. Convexification of the Objective Function

Following the same procedure in [7], a strongly convex approximation with the desired properties can be obtained as

$$\tilde{E}\left(\mathbf{Q}^{ul}; \mathbf{Q}^{ul}(v)\right) = \sum_{i_n \in \mathcal{I}} \tilde{E}_{i_n}\left(\mathbf{Q}^{ul}; \mathbf{Q}^{ul}(v)\right) \tag{13}$$

where

$$\begin{aligned}
\tilde{E}_{i_n}\left(\mathbf{Q}^{ul}; \mathbf{Q}^{ul}(v)\right) &= \text{tr}\left(\mathbf{Q}_{i_n}^{ul}(v)\right) \frac{B_{i_n}^I}{r_{i_n}^{ul}\left(\mathbf{Q}_{i_n}^{ul}, \mathbf{Q}_{-n}^{ul}(v)\right)} \\
&+ \text{tr}\left(\mathbf{Q}_{i_n}^{ul}\right) \frac{B_{i_n}^I}{r_{i_n}^{ul}\left(\mathbf{Q}_{i_n}^{ul}(v), \mathbf{Q}_{-n}^{ul}(v)\right)} \\
&+ \sum_{i_m \in \mathcal{I}, m \neq n} \left\langle \nabla_{\mathbf{Q}_{i_n}^{ul*}} E_{i_m}\left(\mathbf{Q}^{ul}(v)\right), \mathbf{Q}_{i_n}^{ul} - \mathbf{Q}_{i_n}^{ul}(v) \right\rangle \\
&+ \frac{c_{q_{i_n}}}{2} \left\|\mathbf{Q}_{i_n}^{ul} - \mathbf{Q}_{i_n}^{ul}(v)\right\|^2,
\end{aligned} \tag{14}$$

where $\langle \mathbf{A}, \mathbf{B} \rangle \triangleq \text{Re}\left\{\text{tr}\left(\mathbf{A}^H \mathbf{B}\right)\right\}$ and the conjugate gradient $\nabla_{\mathbf{Q}_{i_n}^{ul*}} E_{i_m}\left(\mathbf{Q}^{ul}(v)\right)$ is given by [7, eq. (17)]

$$\begin{aligned}
\nabla_{\mathbf{Q}_{i_n}^{ul*}} E_{j_m}\left(\mathbf{Q}^{ul}(v)\right) &= \frac{\text{tr}\left(\mathbf{Q}_{j_m}^{ul}(v)\right) \Delta_{j_m}^{ul}\left(\mathbf{Q}^{ul}(v)\right)}{\log(2)\, r_{j_m}\left(\mathbf{Q}^{ul}(v)\right)} \\
.[\mathbf{H}_{i_n m}^H (\mathbf{R}_m^{ul}\left(\mathbf{Q}_{-j_m}^{ul}(v)\right)^{-1} &- (\mathbf{R}_m^{ul}\left(\mathbf{Q}_{-j_m}^{ul}(v)\right) \\
&+ \mathbf{H}_{j_m} \mathbf{Q}_{j_m}^{ul}(v) \mathbf{H}_{j_m}^H)^{-1}) \mathbf{H}_{i_n m}],
\end{aligned} \tag{15}$$

and the last term is a quadratic regularization term added to make $\tilde{E}_{i_n}$ uniformly strongly convex, with $c_{q_{i_n}}$ being an arbitrary positive constant.

B. Inner Convexification of the Constraints

Finally, we need to calculate an upper bound on the right-hand side of C.1. Let us define the non-convex part of latency expression as

$$g_{i_n}\left(\mathbf{Q}^{ul}, \mathbf{Q}^{dl}\right) \triangleq \frac{B_{i_n}^I}{W^{ul} r_{i_n}^{ul}\left(\mathbf{Q}^{ul}\right)} + \frac{B_{i_n}^O}{W^{dl} r_{i_n}^{dl}\left(\mathbf{Q}^{dl}\right)}. \tag{16}$$

To build the desired bound on g_{i_n}, we exploit first the concave-convex structure of the rate functions $r_{i_n}^{ul}\left(\mathbf{Q}^{ul}\right)$

$$\begin{aligned} r_{i_n}^{ul}\left(\mathbf{Q}^{ul}\right) &= \log_2 \det\left(\mathbf{I}+\mathbf{H}_{i_n}^{H}\mathbf{R}_n^{ul}\left(\mathbf{Q}_{-i_n}^{ul}\right)^{-1}\mathbf{H}_{i_n}\mathbf{Q}_{i_n}^{ul}\right) \\ &= \underbrace{\log_2 \det\left(\mathbf{R}_n^{ul}\left(\mathbf{Q}_{-i_n}^{ul}\right)+\mathbf{H}_{i_n}^{H}\mathbf{H}_{i_n}\mathbf{Q}_{i_n}^{ul}\right)}_{r_{i_n}^{ul\,+}\left(\mathbf{Q}^{ul}\right)} \\ &\quad - \underbrace{\log_2 \det\left(\mathbf{R}_n^{ul}\left(\mathbf{Q}_{-i_n}^{ul}\right)\right)}_{r_{i_n}^{ul\,-}\left(\mathbf{Q}_{-n}^{ul}\right)} \end{aligned} \tag{17}$$

where $r_{i_n}^{ul\,+}\left(\mathbf{Q}^{ul}\right)$ is a concave function, and $-r_{i_n}^{ul\,-}\left(\mathbf{Q}_{-n}^{ul}\right)$ is a convex function. The same applies to $r_{i_n}^{dl}(\mathbf{Q}^{\mathbf{dl}})$. The desired inner convex approximation $\tilde{g}_{i_n}$ on constraint functions of the difference convex (DC) − type can be obtained from g_{i_n} by retaining the convex parts in (16) and linearizing the non-convex parts, resulting in:

$$\begin{aligned} \tilde{g}_{i_n}\left(\mathbf{Q}^{ul},\mathbf{Q}^{dl};\mathbf{Q}^{ul}(v),\mathbf{Q}^{dl}(v)\right) \triangleq & \frac{B_{i_n}^{I}}{W^{ul}\tilde{r}_{i_n}^{ul}\left(\mathbf{Q}^{ul};\mathbf{Q}^{ul}(v)\right)} \\ & + \frac{B_{i_n}^{O}}{W^{dl}\tilde{r}_{i_n}^{dl}\left(\mathbf{Q}^{dl};\mathbf{Q}^{dl}(v)\right)}, \end{aligned} \tag{18}$$

where

$$\begin{aligned} \tilde{r}_{i_n}^{ul}\left(\mathbf{Q}^{ul};\mathbf{Q}^{ul}(v)\right) &= r_{i_n}^{ul+}\left(\mathbf{Q}^{ul}\right)-r_{i_n}^{ul-}\left(\mathbf{Q}_{-n}^{ul}(v)\right) \\ &- \sum_{n\neq m=1}^{N_c}\sum_{j=1}^{K}\left\langle \nabla_{\mathbf{Q}_{i_n}^{ul}{}^{*}} r_{i_m}^{ul\,-}\left(\mathbf{Q}_{-n}^{ul}(v)\right),\mathbf{Q}_{i_m}^{ul}-\mathbf{Q}_{i_m}^{ul}(v)\right\rangle, \end{aligned} \tag{19}$$

and

$$\begin{aligned} \tilde{r}_{i_n}^{dl}\left(\mathbf{Q}^{dl};\mathbf{Q}^{dl}(v)\right) &= r_{i_n}^{dl+}\left(\mathbf{Q}^{dl}\right)-r_{i_n}^{dl-}\left(\mathbf{Q}_{-n}^{dl}(v)\right) \\ &- \sum_{n\neq m=1}^{N_c}\sum_{j=1}^{K}\left\langle \nabla_{\mathbf{Q}_{i_n}^{dl}{}^{*}} r_{i_m}^{dl\,-}\left(\mathbf{Q}_{-n}^{dl}(v)\right),\mathbf{Q}_{i_m}^{dl}-\mathbf{Q}_{i_m}^{dl}(v)\right\rangle, \end{aligned} \tag{20}$$

with [7]

$$\nabla_{\mathbf{Q}_{j_m}^{ul}{}^{*}} r_{i_n}^{ul-}\left(\mathbf{Q}_{-n}^{ul}(v)\right)=\mathbf{H}_{j_m n}^{H}\mathbf{R}_n^{ul}\left(\mathbf{Q}_{-n}^{ul}(v)\right)^{-1}\mathbf{H}_{j_m n}, \tag{21}$$

$$\nabla_{\mathbf{Q}_{j_m}^{dl}{}^{*}} r_{i_n}^{dl-}\left(\mathbf{Q}_{-n}^{dl}(v)\right)=\mathbf{H}_{j_m n}^{H}\mathbf{R}_n^{dl}\left(\mathbf{Q}_{-n}^{dl}(v)\right)^{-1}\mathbf{H}_{j_m n}. \tag{22}$$

C. SCA Algorithm

The SCA algorithm operates by iteratively solving the following problem around the current iterate $\mathbf{Z}(v) \triangleq \left(\mathbf{Q}^{ul}(v),\mathbf{Q}^{dl}(v)\right)$,

$$\begin{aligned} &\hat{\mathbf{Z}}(\mathbf{Z}(v)) \triangleq \underset{\mathbf{Q}^{ul},\mathbf{Q}^{dl},\mathbf{f},\mathbf{c}^{ul},\mathbf{c}^{dl}}{\operatorname{argmin}} \tilde{E}\left(\mathbf{Q}^{ul};\mathbf{Q}^{ul}(v)\right)+\mathrm{E}(v) \\ &\text{subject to} \\ &\textbf{C.1}\quad \tilde{g}_{i_n}\left(\mathbf{Q}^{ul},\mathbf{Q}^{dl};\mathbf{Q}^{ul}(v),\mathbf{Q}^{dl}(v)\right) \\ &\qquad\qquad + \frac{B_{i_n}^{I}}{c_{i_n}^{ul}C_n^{ul}}+\frac{B_{i_n}^{O}}{c_{i_n}^{dl}C_n^{dl}}+\frac{V_{i_n}}{f_{i_n}F_c}-T_{i_n}\leq 0 \\ &\textbf{C.2}\quad f_{i_n}\geq 0,\ \sum_{i_n\in\mathcal{I}} f_{i_n}=1 \qquad\qquad (23) \\ &\textbf{C.3}\quad c_{i_n}^{ul},c_{i_n}^{dl}\geq 0,\sum_{i} c_{i_n}^{ul}=1,\sum_{i} c_{i_n}^{dl}=1 \\ &\textbf{C.4}\quad \mathbf{Q}_{i_n}^{ul}\in\mathcal{Q}_{i_n}^{ul},\ \mathbf{Q}_{i_n}^{dl}\in\mathcal{Q}_{n}^{dl},\ \forall\, i_n\in\mathcal{I} \end{aligned} \tag{P.2}$$

where $\hat{\mathbf{Z}}(\mathbf{Z}(v)) \triangleq (\hat{\mathbf{Q}}^{ul}(\mathbf{Z}(v)),\hat{\mathbf{Q}}^{dl}(\mathbf{Z}(v)),\hat{\mathbf{f}}(\mathbf{Z}(v)),\hat{\mathbf{c}}^{ul}$ $(\mathbf{Z}(v)),\hat{\mathbf{c}}^{dl}(\mathbf{Z}(v)))$ denotes the unique solution of the strongly convex optimization problem; and $\mathrm{E}(v) = \frac{c_p}{2}\left\|\mathbf{Q}^{dl}-\mathbf{Q}^{dl}(v)\right\|^2 + \frac{c_f}{2}\|\mathbf{f}-\mathbf{f}(v)\|^2 + \frac{c_{c^{ul}}}{2}\left\|\mathbf{c}^{ul}-\mathbf{c}^{ul}(v)\right\|^2 + \frac{c_{c^{dl}}}{2}\left\|\mathbf{c}^{dl}-\mathbf{c}^{dl}(v)\right\|^2$ is added to the objective function to include the quadratic terms in the $\mathbf{Q}^{dl},\mathbf{f},\mathbf{c}^{ul}$ and $\mathbf{c}^{dl}$ variables in order to make the objective function strongly convex in $\mathbf{Q}^{dl},\mathbf{f},\mathbf{c}^{ul}$ and $\mathbf{c}^{dl}$ with $c_p,c_f,c_{c^{ul}}$ and $c_{c^{dl}}$ being arbitrary positive constants.

The SCA scheme summarized in Algorithm 1. It starts from a feasible point $\mathbf{Z}(0) \triangleq \left(\mathbf{Q}^{ul}(0);\mathbf{Q}^{dl}(0)\right)$. In step 2, the termination criterion is to stop when $\left|E\left(\mathbf{Q}^{ul}(v+1)\right)-E\left(\mathbf{Q}^{ul}(v)\right)\right| \leq \delta$, where $\delta > 0$ is the desired accuracy. In this algorithm, a memory in the update of the iterate $\hat{\mathbf{Z}}(\mathbf{Z}(v))$ is allowed in the form of a convex combination via $\mathbf{Z}(v)$. The step size rule is $\gamma(v) = \gamma(v-1)(1-\alpha\gamma(v-1))$ with $\gamma(0) \in (0,1]$ and $\alpha \in (0,1/\gamma(0))$.

Algorithm 1 Inner SCA Algorithm for (P.1)

```
1: Initialization: $\mathbf{Z}(0) \triangleq \left(\mathbf{Q}^{ul}(0);\mathbf{Q}^{dl}(0)\right) \in \mathcal{Z}$;
   $\{\gamma(v)\}_v \in (0,1]$; $c_p, c_f, c_{c^{ul}}, c_{c^{dl}} > 0$; $v = 0$.
2: If $\mathbf{Z}(v)$ satisfies the termination criterion, STOP.
3: Compute $\hat{\mathbf{Z}}(\mathbf{Z}(v))$;
4: Set $\mathbf{Z}(v+1) = \mathbf{Z}(v) + \gamma(v)\left(\hat{\mathbf{Z}}(\mathbf{Z}(v)) - \mathbf{Z}(v)\right)$;
5: $v \leftarrow v+1$, and return to step 2.
```

IV. Simulation Results

In this section, we present numerical results with the main aims of validating the performance of the proposed SCA schemes and of assessing the advantages of the considered joint optimization across uplink, downlink, backhaul and computing resources. To this end, we compare the discussed schemes in which joint optimization is performed, with more conventional solutions in which the the computing and backhaul resources are equally allocated to all MUs, that is $f_{i_n} = 1/(N_c K)$ and $c_{i_n}^{ul} = c_{i_n}^{dl} = 1/K$, while the covariance transmit

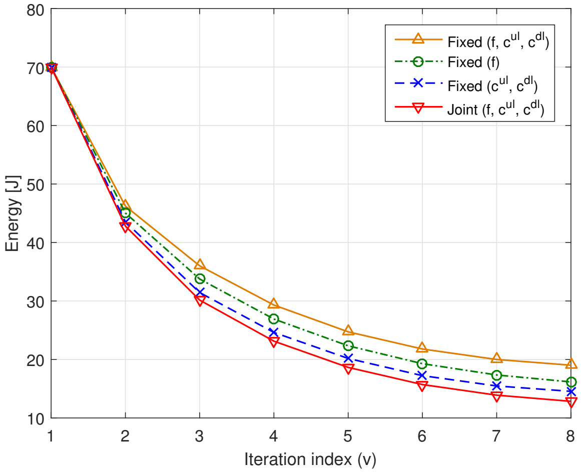

Fig. 2: Minimum average mobile energy consumption versus iteration index.

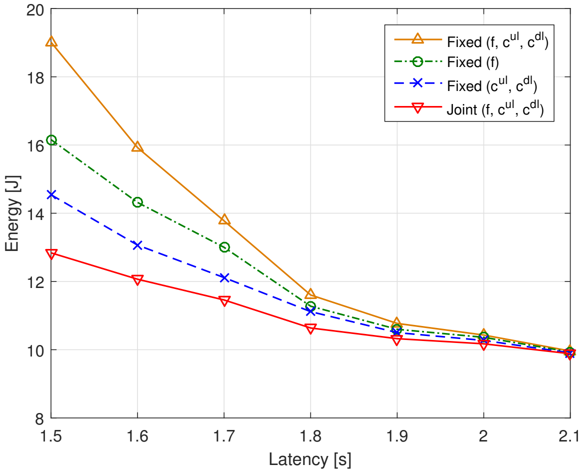

Fig. 3: Minimum average mobile energy consumption versus the latency T_{i_n}.

matrices at the physical layer are optimized using SCA. We label this scheme in the figures as "fixed (f, c^{ul}, c^{dl})". We also consider for reference scenarios in which only the computing resources or the backhaul resources are equally allocated, while the rest of the parameters are jointly optimized using SCA. These schemes are accordingly labeled by the variables that are kept fixed, i.e., equally allocated.

Throughout, we consider a network composed of two cells with two users in each cell, i.e. $N_c = 2$ and $K = 2$. All transceivers are equipped with $N_T = 2$ and $N_R = 2$ transmit and receive antennas, respectively. The channel matrices between a user and the ceNB in the same cell are generated to have zero mean complex Gaussian entries with power equal to 1, whereas the power of the channel coefficients between a user and ceNBs in different cells is set to 0.5. The power budget constraints for both uplink and downlink are set to $P^{ul}_{i_n} = P^{dl}_n = 0.8 \cdot 10^{-4}$. Other system parameters are set to: $W^{ul} = W^{dl} = 10$ MHz, $\sigma^2_w = 0.8 \cdot 10^{-5}$, $B^I_{i_n} = B^O_{i_n} = 10^6$ bits, $V_{i_n} = 10^5$ CPU cycles, $C^{ul}_n = C^{dl}_n = 100$ Mbits/s, $F_c = 10^7$ CPU cycles/s and $T_{i_n} = 1.5$ seconds unless stated otherwise.

Fig. 2 illustrates the average minimal sum-energy consumption $E\left(\mathbf{Q}^{ul}\right)$ versus iteration index v for the different schemes outlined above. We first observe the fast convergence of the optimization strategy. Furthermore, it can be seen that the proposed joint optimization method shows a considerable gain compared to the equal allocation of computational and backhaul resources.

Fig. 3 depicts the minimum average mobile energy as a function of the maximum latency constraints, assumed to be the same for all MUs. The figure confirms the advantages of joint optimization, particularly in the region where the latency constraint is more stringent.

Finally, in order to show the advantage of including the optimization over the backhaul capacity allocation $(c^{dl}_{i_n})$, Fig. 4 compares the energy performance of both joint optimization and of solutions that use an equal backhaul allocation versus the downlink backhaul capacity, assumed to be the same for both cells. The figure illustrates the gains of joint optimization, which are especially pronounced in the regime of small backhaul capacities in which the backhaul capacity should be properly allocated among the MUs in each cell.

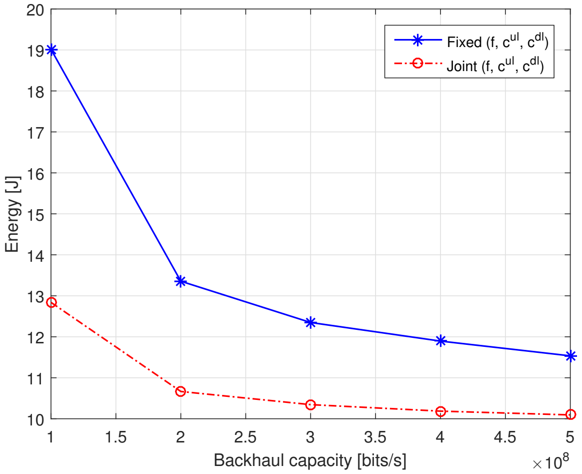

Fig. 4: Minimum average mobile energy consumption versus the backhaul capacity C^{dl}_n.

V. Conclusion

In this paper, we formulated the resource allocation problem in a multiple-cell multiple-users mobile cloud computing network as a joint optimization problem over radio, computational resources and backhaul resources and in both uplink and downkink directions. An iterative algorithm based on successive convex approximations is presented for solving the resulting nonconvex problem under latency and power constraints. Numerical results show the advantates of joint optimization as compared to more conventional solutions based on fixed allocations of computing and backhaul resources, particularly in the relevant regimes of small backhaul capacities and low latency requirements.

References

[1] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, “A survey of computation offloading for mobile systems,” *Mobile Networks and Applications*, vol. 18, no. 1, pp. 129–140, Feb. 2013.

[2] H. T. Dinh, C. Lee, D. Niyato, and P. Wang, “A survey of mobile cloud computing: architecture, applications, and approaches,” *Wireless Communications and Mobile Computing*, vol. 13, no. 18, pp. 1587–1611, Dec. 2013.

[3] S. Abolfazli, Z. Sanaei, E. Ahmed, A. Gani, and R. Buyya, “Cloud-based augmentation for mobile devices: motivation, taxonomies, and open challenges,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 337–368, Feb. 2014.

[4] Y.-H. Kao, B. Krishnamachari, M.-R. Ra, and F. Bai, “Hermes: Latency optimal task assignment for resource-constrained mobile computing,” in *Proc. INFOCOM*, pp. 1894-1902, Kowloon, Hong Kong, Apr. 2015.

[5] P. D. Lorenzo, S. Barbarossa, and S. Sardellitti, “Joint optimization of radio resources and code partitioning in mobile cloud computing,” *Submitted, arXiv:1307.3835*, 2015.

[6] S. Khalili and O. Simeone, “Inter-layer per-mobile optimization of cloud mobile computing: A message-passing approach,” *Submitted, arXiv:1207.0016*, 2015.

[7] S. Sardellitti, G. Scutari, and S. Barbarossa, “Joint optimization of radio and computational resources for multicell mobile-edge computing,” *Submitted, arXiv:1412.8416*, 2014.

[8] S. Barbarossa, S. Sardellitti, and P. D. Lorenzo, “Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks,” *IEEE Signal Proc. Magazine*, vol. 31, no.6, pp. 45–55, Nov. 2014.

[9] G. Scutari, F. Facchinei, L. Lampariello, and P. Song, “Distributed methods for constrained nonconvex multi-agent optimization-part I: Theory,” *Submitted, arXiv:1410.4754*, 2014.

[10] R. Taori and A. Sridharan, “Point-to-multipoint in-band mmwave backhaul for 5G networks,” *IEEE Communications Magazine*, vol. 53, no. 1, pp. 195–201, Jan. 2015.