<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper presented at *IEEE WCNC 2017*.

Citation for the original published paper:

Masoudi, M., Khamidehi, B., Cavdar, C. (2017)
Green Cloud Computing for Multi Cell Networks.
In:

N.B. When citing this work, cite the original published paper.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-205202>

Green Cloud Computing for Multi Cell Networks

Meysam Masoudi[†], Behzad Khamidehi[‡], and Cicek Cavdar[†]

[†]Communication Systems Department, KTH Royal Institute Of Technology, Sweden

[‡]Department of Electrical and Computer Engineering, University of Tehran, Iran

Email: masoudi@kth.se, khamidehi@ut.ac.ir, cavdar@kth.se

Abstract—This paper investigates the power minimization problem for the user terminals by application offloading in multi-cell multi-user OFDMA mobile cloud computing networks where some practical constraints such as backhaul capacity limitation, interference level on each channel and maximum tolerable delay as user's quality of service is taken into account. Furthermore, the mixed integer nonlinear problem is converted into a convex form using D.C. approximation. Moreover, to solve the optimization problem, we have proposed joint power allocation and decision making (J-PAD) algorithm which can make offloading decision and allocate power at the same time. Simulation results illustrate that by utilizing the J-PAD algorithm, in comparison with baselines, considerable power saving could be achieved e.g. about 30% for delays more than 100 ms.

I. INTRODUCTION

Mobile applications with high computational resource requirements are developing swiftly. However, mobile terminals cannot always be considered as a platform for resource hungry applications due to their limited processing and power capacity [1]. Battery lifetime as one of the key concerns of mobile users [2] must be taken into account knowing that increasing the clock frequency of a CPU increases its power consumption [3]. Therefore, there is a tension between resource hungry applications and resource poor mobile devices which causes severe challenges for mobile platform development.

Unfortunately, battery technologies are still a major issue for the mobile terminals. Therefore, one can remove the processing load of the terminals by offloading the burden onto the cloud. Mobile cloud computing (MCC), which provides infrastructure, platform, and software as services, is a promising solution for supporting the resource-poor mobile users [4]. Utilizing the MCC can bridge the gap between the increasing computing demands and limited mobile resources [5] by offloading the data processing from users' side to the cloud data centers [6] to enhance the application's performance in terms of computing power, storage capacity and user's power consumption. Cloud computing can potentially save energy for mobile users [2] although offloading may not always result in the minimum energy consumption [7].

Delay sensitivity of mobile applications should be also taken into account which may require local computing. Therefore, a decision making procedure is necessary to determine whether executing locally or offloading to the cloud is efficient while satisfying the delay requirements of applications. To find the optimal policy for offloading, a Markov decision process based

model is adopted in [7]. For energy consumption and latency minimization problem, partial computation offloading considering both optimizing the computational speed of mobile terminal, their transmit power, and the ratio of locally executed bits to the total input data bits, is explored in [8]. In [9], the authors deal with the latency issue by means of cloudlet infrastructure, as a data center to bring the cloud closer to the users. The authors in [10], derive an offloading policy considering data rate and energy consumption under single stochastic wireless channel with only "good" or "bad" channel state. Interference and user's QoS is neglected in this study. In [11], it is shown that wireless access has an inevitable effect on the performance of MCC, using experimental measurements. The authors in [12], consider the problem of resource scheduling for multi service multi user MCC networks. Also, in [13] a heuristic approach is adopted to minimize the energy consumption of all users while making decision on offloading and resource allocation for each task. The study in [14], models the decision offloading in a multi radio interface to figure out the optimal solution of the conflicting objectives, namely, computation costs and the execution time of the application. In [15], a game theoretic approach is used to design an offloading mechanism for mobile devices. In their model, they have considered multi user case but the QoS of users and other users' effects are not addressed. In [16], a decentralized offloading game is proposed to make decision among mobile device users but they only consider single channel scenario.

There exists many questions to be answered in MCC networks. First, how users can make decision on offloading to save their energy consumption while satisfying the QoS requirements. Second, if a user decides to offload, how much resources should be allocated and what is the optimal procedure? To address the aforementioned questions, we aim at minimizing the power consumption of users while considering the QoS for users in terms of delay, maximum tolerable interference on each channel, and backhaul capacity limitation. We formulate the resource allocation and offloading problem and show that the problem is mixed integer nonlinear problem (MINLP), where the optimal solution is intractable. We convert the problem to the convex form and J-PAD algorithm is proposed to perform decision making and resource allocation in a polynomial time.

The rest of the paper is organized as follows. The system model and problem formulation is explained in section II. The solution methodology and algorithm design is discussed in

This work was supported in part by EU Celtic Plus Project SooGREEN in KTH.

section III. To evaluate the proposed algorithm and to illustrate the enhancements, simulation results are given in section IV. Finally concluding remarks are summarized in section V.

II. SYSTEM MODEL

A. System Description

We consider a cellular network with N_c cells where macro-cell users (MUs) are uniformly distributed within a cell range. Each cell serves up to F_i active users. We assume that the bandwidth is B and it is divided into N subchannels. Also, we consider OFDMA as an access method hence users in the same cell cannot share same subchannel with each other; however, each user might experience an interference from neighboring cells. In our model, j th user in cell i has a bit stream of size $L_{i,j}$ to be executed whether in cloud or in the mobile device locally. Single user cannot use both scheme, e.g. sending a portion of its data to the cloud and process the remaining data locally. Also, each user has an application deadline $T_{i,j}$, that is its data should be processed within this time.

B. Power Model

Power Model for Local Processing:

When users decide to execute the data locally, the CPU power consumption is dominant. It is composed of dynamic power, short circuit power, and leakage power [8]. The authors in [10], showed that under the optimal value for CPU frequency, the minimum power consumption of CPU is proportional to the T^{-3} and also proportional to the L^3 , where T is application delay deadline (delay threshold) and L is user's bit stream size. Consequently, we use the following model for local execution power consumption:

$$p_{i,j}^{Local} = M \frac{L_{i,j}^3}{T_{i,j}^3}, \quad (1)$$

where $p_{i,j}^{Local}$ denotes the power consumption of user j in cell i when a user is supposed to process its data locally and M is a constant value depending on the CPU and application parameters [10].

Power Model for Offloading:

The transmission power, for sending data to the cloud is,

$$p_{i,j}^{Tx} = \sum_{n=1}^N a_{i,j,n} p_{i,j,n}, \quad (2)$$

where $p_{i,j}^{Tx}$ denotes the power consumption of user i in cell j . Also, $a_{i,j,n}$ is a binary variable representing whether the corresponding subchannel is assigned to the user or not. It is worth mentioning that in transmission power model, a constant power representing the circuit power can be considered. Here without loosing any generality, we do not consider the constant power because this power is the same for both offloading and local execution.

Power Model for the Network:

Total power consumption of the users in the network is composed of local processing power and transmission power which is given by:

$$\mathbf{P}^{Total} = \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} p_{i,j} \quad (3)$$

$$\text{where } p_{i,j} = s_{i,j} p_{i,j}^{Tx} + (1 - s_{i,j}) p_{i,j}^{Local} \quad (4)$$

$$= s_{i,j} \sum_{n=1}^N a_{i,j,n} p_{i,j,n} + (1 - s_{i,j}) \frac{ML_{i,j}^3}{T_{i,j}^3}. \quad (5)$$

The integer variable $s_{i,j}$ takes value of 0 if user j in the i th cell uses its own processor, and is 1 if this user sends its data to the cloud. Therefore, the total power consumption can be written as:

$$\mathbf{P}^{Total} = \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} \sum_{n=1}^N s_{i,j} a_{i,j,n} p_{i,j,n} + \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} (1 - s_{i,j}) \frac{ML_{i,j}^3}{T_{i,j}^3}. \quad (6)$$

Moreover, the signal to noise plus interference ratio at the base station in cell i is given by:

$$\gamma_{i,j,n} = \frac{p_{i,j,n} h_{i,j,n}}{\sigma^2 + I_i^{(n)}}, \quad (7)$$

where the channel gain from j th MU of i th cell is denoted by $h_{i,j,n}$. Also, the channel gain from user m , in cell k to the cell i is denoted by $h_{k,m,n}^i$. The first term in the denominator of (7) is noise power and the second term is an interference from other cells on channel n in cell i which can be calculated as:

$$I_i^{(n)} = \sum_{k=1}^{N_c} \sum_{m=1}^{F_i} a_{k,m,n} s_{k,m} p_{k,m,n} h_{k,m,n}^i. \quad (8)$$

In our assumption, the users must utilize the whole duration so power minimization is in line with energy minimization. Therefore, because T is assumed to be fixed for each user, the more power is used, the less energy is consumed.

C. Problem Formulation

We first define the investigated optimization problem and then the solution methodology will be discussed. In the resource allocation and offloading problem defined in (9), the objective is to minimize the total power consumption of all active MUs in the network.

$$\min_{\{\mathbf{a}, \mathbf{p}, \mathbf{s}\}} \mathbf{P}^{Total} \quad (9)$$

subject to

$$\text{C1: } 0 \leq p_{i,j}^{Tx} \leq p_{max}, \quad \forall i, j,$$

$$\text{C2: } \sum_{k=1}^{N_c} \sum_{j=1}^{F_i} s_{k,j} a_{k,j,n} p_{k,j,n} h_{k,j,n}^i \leq I_{th}^{(n)}, \quad \forall i, n,$$

$$\text{C3: } T_{i,j} \leq T_{max}, \quad \forall i, j,$$

$$\text{C4: } \sum_{j=1}^{F_i} \sum_{n=1}^N s_{i,j} a_{i,j,n} \log_2(1 + \gamma_{i,j,n}) \leq R_{max}^{Backhaul}, \quad \forall i,$$

$$\text{C5: } \sum_{n=1}^N a_{i,j,n} \leq 1,$$

$$\text{C6: } a_{i,j,n} \in \{0, 1\}, \quad \forall i, j, n,$$

$$\text{C7: } s_{i,j} \in \{0, 1\}, \quad \forall i, j.$$

The constraint C1 indicates that the transmit power of each user is limited to p_{max} . The constraint C2 states that for each

base station i , the interference arising from MUs of other cells on each subchannel is restricted to be within a threshold. The constraint C3 enforces the maximum tolerable delay for user j in the i th cell to T_{max} if the aforementioned user sends its data to the cloud. If a user executes its data locally, then the CPU is responsible for satisfying this constraint. In our analysis we assume that CPU uses the entire available time to reduce the power consumption. The constraint C4 addresses the backhaul capacity limitation. The constraint C5 guarantees that each subchannel is assigned to at most one user in each cell. The constraints C6 and C7 indicate that the subchannel and data offloading indices are binary variables. It is worth mentioning that the constraint C3 can be written in an equivalent form. Using C3 we will have

$$\frac{L_{i,j}}{T_{i,j}} \geq \frac{L_{i,j}}{T_{max}}. \quad (10)$$

Defining $R_{min} \triangleq \frac{L_{i,j}}{T_{max}}$ and noting that the left side of (10) is the total data rate of the j th user in the i th cell, we obtain the following equivalent constraint for C3

$$s_{i,j} \sum_{n=1}^N a_{i,j,n} \log_2(1 + \gamma_{i,j,n}) \geq s_{i,j} R_{min}, \forall i, j. \quad (11)$$

In the rest of this paper, we consider the constraint C3 in the form presented in (11).

III. SOLUTION METHODOLOGY AND ALGORITHM DESIGN

In this section, we convert the MINLP defined in section II-C to a convex form. The problem is MINLP because of two types of binary variables, subchannel assignment and data offloading indices and also because of the interference term in the denominator of SINR and rate constraints C3 and C4. Due to these difficulties, finding the optimal solution is NP hard and therefore we cannot solve the joint optimization problem thus an iterative method should be employed. To solve the optimization problem of (9), we separate the subchannel assignment part from power allocation and data offloading:

$$\begin{aligned} & \text{Initialization} \quad \text{Iteration } t-1 \\ & \mathbf{a}[0] \rightarrow (\mathbf{p}[0], \mathbf{s}[0]) \rightarrow \dots \rightarrow \mathbf{a}[t-1] \rightarrow (\mathbf{p}[t-1], \mathbf{s}[t-1]) \\ & \quad \quad \quad \text{Iteration } t \quad \quad \quad \text{Optimal Solution} \\ & \rightarrow \mathbf{a}[t] \rightarrow (\mathbf{p}[t], \mathbf{s}[t]) \rightarrow \dots \rightarrow \mathbf{a}^* \rightarrow (\mathbf{p}^*, \mathbf{s}^*). \end{aligned} \quad (12)$$

The problem of power allocation and data offloading are solved jointly thus it has closer results to the optimal solution. Moreover, it is worth mentioning that this problem should be solved at the base stations that will be explained later.

A. Optimal Sub-channel Assignment for a Fixed Power Allocation and Data Offloading

Proposition 1. *To minimize the power consumption, each subchannel in each cell should be assigned to the MU with the highest effective interference on that subchannel.*

Proof. Because the problem is power minimization and also minimum data rate requirement of user should be satisfied, the

minimum power is consumed when the inequality of minimum required rate becomes the equality. Now let's assume that all users are given the best possible channel to reach their data rate with minimum power consumption. Also, let a user have a channel with effective interference value lower than a highest value and the user has data rate r_{min} on that channel. Thus, the consumed power on that channel is

$$\log_2(1 + \gamma_{i,j,n}) = r_{min} \quad (13)$$

$$p_{i,j,n} = \frac{C}{\frac{h_{i,j,n}}{\sigma^2 + I^n}}, \quad (14)$$

where C here is a constant value. Also from our assumption we know that, the effective interference in a denominator of (14) e.g. $\frac{h_{i,j,n}}{\sigma^2 + I^n}$, is not the highest possible value. Hence, if we assign the highest effective interference value to this user, the total power consumption will be lower and this is in contrast with the assumption of minimum power consumption. Therefore, minimum power is consumed when maximum effective interference is a criterion for channel allocation. In other words, user with higher effective interference, uses less power to reach its required rate. $\square$

Let $EI_{i,j,n}$ denotes the effective interference vector of a user on the channel n . Therefore, the decision for channel allocation will be made based on the following criteria:

$$\tilde{a}_{i,j,\tilde{n}} = 1 \Big|_{\tilde{n}=\max EI_{i,j,n}} \quad \forall i, j. \quad (15)$$

Thus, a channel allocation matrix $\mathbf{a}[t]$ at time t , can be formed with the elements obtained from equation (15).

B. Joint Power Allocation and Decision Making (J-PAD) Problem:

Given a subchannel assignment, the problem of joint power allocation and data offloading can be specified as follows

$$\min_{\{\mathbf{p}, \mathbf{s}\}} \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} \sum_{n=1}^N s_{i,j} p_{i,j,n} + \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} (1 - s_{i,j}) \frac{ML_{i,j}^3}{T_{i,j}^3} \quad (16)$$

subject to

$$\text{C1: } 0 \leq s_{i,j} p_{i,j}^{Tx} \leq p_{max}, \quad \forall i, j,$$

$$\text{C2: } \sum_{k=1}^{N_c} \sum_{j=1}^{F_i} s_{k,j} p_{k,j,n} h_{k,j,n}^i \leq I_{th}^{(n)}, \quad \forall i, n,$$

$$\text{C3: } s_{i,j} \sum_{n=1}^N \log_2(1 + \gamma_{i,j,n}) \geq s_{i,j} R_{min}, \quad \forall i, j,$$

$$\text{C4: } \sum_{j=1}^{F_i} \sum_{n=1}^N s_{i,j} \log_2(1 + \gamma_{i,j,n}) \leq R_{max}^{Backhaul}, \quad \forall i,$$

$$\text{C7: } s_{i,j} \in \{0, 1\}, \quad \forall i, j.$$

To solve (16), we first reformulate it to a more mathematically tractable form. Since $s_{i,j}$ is a binary variable, we can write $s_{i,j} \log_2(1 + \gamma_{i,j,n}) = \log_2(1 + s_{i,j} \gamma_{i,j,n})$. Moreover, the

problem consists of the product terms of $s_{i,j}p_{i,j,n}$. We use the following change of variable

$$\tilde{p}_{i,j,n} = s_{i,j}p_{i,j,n}, \quad (17)$$

to recast the optimization problem. Also, the optimization problem includes integer variables hence to convert $s_{i,j}$'s to continuous variables, we can express the constraint C7 as the intersection of the following regions:

$$\begin{aligned} \mathcal{R}_1 : 0 \leq s_{i,j} \leq 1, \forall i, j, \\ \mathcal{R}_2 : \sum_j \sum_i (s_{i,j} - s_{i,j}^2) \leq 0. \end{aligned} \quad (18)$$

Hence, we can write the optimization problem of (16) as follows

$$\begin{aligned} \min_{\tilde{\mathbf{p}}, \mathbf{s}} \mathbf{P}^{Total} \\ \text{s.t. C1-C4, } \mathcal{R}_1, \mathcal{R}_2. \end{aligned} \quad (19)$$

The problem of (19) is a continuous optimization problem with respect to all variables. However, we aim to find integer solutions for $s_{i,j}$'s. To attain this goal, we add a penalty to the objective function of (19) to penalize it if the values of $s_{i,j}$'s are not integer. Thus, the problem can be modified to

$$\begin{aligned} \min_{\tilde{\mathbf{p}}, \mathbf{s}} \mathcal{L}(\tilde{\mathbf{p}}, \mathbf{s}, \lambda) \\ \text{s.t. C1-C4, } \mathcal{R}_1. \end{aligned} \quad (20)$$

In (20), $\mathcal{L}(\tilde{\mathbf{p}}, \mathbf{s}, \lambda)$ is the Lagrangian of (19), and is defined as

$$\mathcal{L}(\tilde{\mathbf{p}}, \mathbf{s}, \lambda) \triangleq \mathbf{P}^{Total} + \lambda \sum_j \sum_i (s_{i,j} - s_{i,j}^2), \quad (21)$$

where λ is the penalty factor which should be $\lambda \gg 1$. It can be shown that, for sufficiently large values of λ , the optimization problem of (20) is equivalent to (19) and attains the same optimal value [17]. Now, the optimization problem can be converted to the following problem

$$\begin{aligned} \min_{\{\tilde{\mathbf{p}}, \mathbf{s}\}} \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} \sum_{n=1}^N \tilde{p}_{i,j,n} + \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} (1 - s_{i,j}) \frac{ML_{i,j}^3}{T_{i,j}^3} \\ + \lambda \left(\sum_i \sum_j (s_{i,j} - s_{i,j}^2) \right) \end{aligned} \quad (22)$$

subject to

$$\begin{aligned} \text{C1: } 0 \leq \sum_{n=1}^N \tilde{p}_{i,j,n} \leq s_{i,j} p_{max}, \quad \forall i, j, \\ \text{C2: } \sum_{k=1}^{N_c} \sum_{j=1}^{F_i} \tilde{p}_{i,j,n} h_{k,j,n}^i \leq I_{th}^{(n)}, \quad \forall i, n, \\ \text{C3: } \sum_{n=1}^N \log_2 \left(1 + \frac{\tilde{p}_{i,j,n} h_{i,j,n}}{\sigma^2 + \tilde{I}^{(n)}} \right) \geq s_{i,j} R_{min}, \quad \forall i, j, \\ \text{C4: } \sum_{j=1}^{F_i} \sum_{n=1}^N \log_2 \left(1 + \frac{\tilde{p}_{i,j,n} h_{i,j,n}}{\sigma^2 + \tilde{I}^{(n)}} \right) \leq R_{max}^{Backhaul}, \quad \forall i, \\ \text{C7: } s_{i,j} \in [0, 1], \quad \forall i, j, \end{aligned}$$

where $\tilde{I}_i^{(n)} \triangleq \sum_{k=1}^{N_c} \sum_{m=1}^{F_i} \tilde{p}_{k,m,n} h_{k,m,n}^i$. We can write the objec-

tive function in (22) as $f_1(\tilde{\mathbf{p}}, \mathbf{s}) - f_2(\tilde{\mathbf{p}}, \mathbf{s})$, where $f_1(\tilde{\mathbf{p}}, \mathbf{s}) \triangleq \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} \sum_{n=1}^N \tilde{p}_{i,j,n} + \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} ((1 - s_{i,j}) \frac{ML_{i,j}^3}{T_{i,j}^3} + \lambda s_{i,j})$, and

$f_2(\tilde{\mathbf{p}}, \mathbf{s}) \triangleq \lambda \sum_{i=1}^{N_c} \sum_{j=1}^{F_i} s_{i,j}^2$ are two convex functions. In a similar way, for $\forall i, j$, we define $z_{i,j,n}(\tilde{\mathbf{p}})$ and $q_{i,j,n}(\tilde{\mathbf{p}})$ as

$$z_{i,j,n}(\tilde{\mathbf{p}}) \triangleq \log_2 \left(\tilde{p}_{i,j,n} h_{i,j,n} + \sum_{k=1}^{N_c} \sum_{m=1}^{F_i} \tilde{p}_{k,m,n} h_{k,m,n}^i + \sigma^2 \right), \quad (23)$$

$$q_{i,j,n}(\tilde{\mathbf{p}}) \triangleq \log_2 \left(\sum_{k=1}^{N_c} \sum_{m=1}^{F_i} \tilde{p}_{k,m,n} h_{k,m,n}^i + \sigma^2 \right), \quad (24)$$

then, we can write constraints C3 and C4 as follows

$$\begin{aligned} \text{C3: } Z_{i,j}(\tilde{\mathbf{p}}) - Q_{i,j}(\tilde{\mathbf{p}}) &\geq s_{i,j} R_{min}, \forall i, j, \\ \text{C4: } Q_i(\tilde{\mathbf{p}}) - Z_i(\tilde{\mathbf{p}}) &\geq -R_{max}^{Backhaul}, \forall i, \end{aligned} \quad (25)$$

where $Z_{i,j}(\tilde{\mathbf{p}}) \triangleq \sum_{n=1}^N z_{i,j,n}(\tilde{\mathbf{p}})$, $Q_{i,j}(\tilde{\mathbf{p}}) \triangleq \sum_{n=1}^N q_{i,j,n}(\tilde{\mathbf{p}})$,

$Z_i(\tilde{\mathbf{p}}) \triangleq \sum_{j=1}^{F_i} \sum_{n=1}^N z_{i,j,n}(\tilde{\mathbf{p}})$, and $Q_i(\tilde{\mathbf{p}}) \triangleq \sum_{j=1}^{F_i} \sum_{n=1}^N q_{i,j,n}(\tilde{\mathbf{p}})$ are

concave functions. Therefore, the problem is in the form of the difference of two convex (concave) functions (D.C.) programming [18]. In D.C. programming, we start from a feasible initial point and iteratively solve the optimization problem. Let k denote the iteration number. At the k th iteration, to make the problem convex, using the first order Taylor approximation for $f_2(\tilde{\mathbf{p}}, \mathbf{s})$, $Q_{i,j}(\tilde{\mathbf{p}})$ and $Z_i(\tilde{\mathbf{p}})$ as follows

$$\begin{aligned} \tilde{f}_2(\tilde{\mathbf{p}}, \mathbf{s}) &\approx f_2(\tilde{\mathbf{p}}, \mathbf{s}^{k-1}) + \nabla_{\mathbf{s}} f_2^T(\tilde{\mathbf{p}}, \mathbf{s}^{k-1}) \cdot (\mathbf{s} - \mathbf{s}^{k-1}), \\ \tilde{Q}_{i,j}(\tilde{\mathbf{p}}) &\approx Q_{i,j}(\tilde{\mathbf{p}}^{k-1}) + \nabla_{\tilde{\mathbf{p}}} Q_{i,j}^T(\tilde{\mathbf{p}}^{k-1}) \cdot (\tilde{\mathbf{p}} - \tilde{\mathbf{p}}^{k-1}), \\ \tilde{Z}_i(\tilde{\mathbf{p}}) &\approx Z_i(\tilde{\mathbf{p}}^{k-1}) + \nabla_{\tilde{\mathbf{p}}} Z_i^T(\tilde{\mathbf{p}}^{k-1}) \cdot (\tilde{\mathbf{p}} - \tilde{\mathbf{p}}^{k-1}), \end{aligned} \quad (26)$$

where $\tilde{\mathbf{p}}^{k-1}$ and $\mathbf{s}^{k-1}$ are the solutions of the problem at $(k-1)$ th iteration and ∇_x denotes the gradient operation with respect to x . Thus, at the k th iteration, instead of dealing with the problem of (16), we solve the following convex problem

$$\begin{aligned} \min_{\{\tilde{\mathbf{p}}, \mathbf{s}\}} f_1(\tilde{\mathbf{p}}, \mathbf{s}) - \tilde{f}_2(\tilde{\mathbf{p}}, \mathbf{s}) \\ \text{subject to: C1, C2, C7,} \\ \text{C3: } Z_{i,j}(\tilde{\mathbf{p}}) - \tilde{Q}_{i,j}(\tilde{\mathbf{p}}) &\geq s_{i,j} R_{min}, \quad \forall i, j, \\ \text{C4: } Q_i(\tilde{\mathbf{p}}) - \tilde{Z}_i(\tilde{\mathbf{p}}) &\geq -R_{max}^{Backhaul}, \quad \forall i. \end{aligned} \quad (27)$$

In [17], it is shown that the D.C. programming results in a sequence of feasible solutions that iteratively improves the solutions until it converges.

C. J-PAD Algorithm

Algorithm 1 performs **Joint Power Allocation and Decision making** and is called **J-PAD**. J-PAD is designed to solve the convex optimization problem presented in (27). The key idea of our algorithm is to make decision and allocate power simultaneously. In the algorithm, first the channels will be allocated and then the rest of the problem will be solved. Algorithm 1 represents the procedure of solving the optimization problem using J-PAD algorithm.

Algorithm 1 Joint Power Allocation and Decision Making (J-PAD) algorithm

- 1: Initialize power, $\mathbf{a}$, $\mathbf{s}$, I_{max} , λ , and $Counter = 0$
 - 2: **while** $Counter \leq I_{max}$ **do**
 - 3: **Channel Allocation**
 - 4: Calculate $EI_{i,j,n}$ based on (15) $\forall i, j, n$
 - 5: Form $\mathbf{a}[t]$ based on $EI_{i,j,n}$
 - 6: **Power Allocation and Offloading Decision**
 - 7: **for** $i=1$ to N_c **do**
 - a) Solve the problem (27) using interior point method
 - b) Update Power Vector based on the solution of (27)
 - c) Update $s_{k,u,n}$ according to the solution of (27)
 - 8: **end for**
 - 9: Update λ , $Counter = Counter + 1$
 - 10: Main Data Center updates the I based on (8) and sends this value back to the base stations.
 - 11: **end while**
-

J-PAD algorithm is composed of two main sections, channel allocation done based on the equation (15) and Power allocation and offloading decision. After performing the second part, the power vectors and offloading decisions are updated at each base station and will be sent to the main center. Then the main center updates the interference value on each channel and sends them back to each base station for next iteration. The problem is solved in each base station thus the base stations are the main coordinator of this algorithm. Also, choosing λ is an important issue. It is a penalty factor to punish the objective function for any value of s which is not equal to 0 or 1, therefore it should be large enough e.g. 10^5 , ($\lambda \gg 1$) [17]. One can fix this value to a predetermined high value but here at first we set the λ to a relatively low value ($\lambda > 1$). In this case, the value of s will be a real value in $[0,1]$. Then in next iterations we tighten the condition on s by choosing larger value for λ .

D. Complexity Analysis

In this section, we investigate the computational complexity of J-PAD algorithm. To assign each subchannel to the users, we have to find the user with highest effective interference. Let F denote the maximum number of users existing in a cell, i.e., $F = \max_{i=1, \dots, N_c} F_i$. Since finding the maximum of a set with K elements requires $O(K)$ operations, the subchannel assignment phase has the complexity in order of $O(NFN_c)$. For the data offloading and power allocation, we have totally

Fig. 1: Convergence performance of J-PAD algorithm and the effect of the algorithm on decision offloading value

$N_c F(N+1)$ decision variables and $N_c(3F+N+1)$ convex and linear constraints [17]. Therefore, the computational complexity of solving a joint data offloading and power allocation problem is given by

$$O((N_c F(N+1))^3 (N_c(3F+N+1))) \approx O(N_c^4 F^3 N^3 (3F+N))$$

IV. SIMULATION RESULTS

In this section we evaluate the performance of our proposed algorithm and we compare it with two baselines. The first baseline is equal power allocation where the channel allocation is based on (15) and power is assigned equally on each subchannel provided that all the constraints are satisfied. The second baseline is the random channel allocation. We assume that we have $N = 20$ subchannels. Up to $F = 10$ uniformly distributed active MUs share the same spectrum of W with a maximum transmit power of 23dbm with $(N_c - 1)$ neighboring cells. Where $N_c = 7$ is total number of cells. The power spectral density of noise and the interference threshold, I_n^{th} , is set to $N_0 = -174\text{dbm}/\text{Hz}$ and -101.2dbm , respectively. Path loss model is adopted from [19], and shadow fading is modeled as zero mean log normal distributions with variance of 8db . Also the cell radius is set to 500m . The numerical results are averaged over 100 simulations. According to the section III-C, first s can be a real value between $[0,1]$. Fig.1 represents the perfect behavior of our algorithm towards enforcing s to be zero or one. We have opted two users, one who decides to send its data to the cloud and one who prefer to execute its data locally. It can be seen that in the first iterations where s , offloading decision variable, has the freedom to get any positive value less than one while by increasing the λ the algorithm forces this value to be whether zero or one. Furthermore, the convergence performance of J-PAD algorithm is also depicted in Fig.1 where it confirms the rapid convergence of J-PAD Algorithm.

According to the Fig.2, it can be seen that increasing the tolerable delay threshold results in reduction in total power consumption of the network. Because by increasing the tolerable delay, lower data rate and also less power is needed to send data and likewise for local execution, mobile devices have more time to execute their data therefore they need less CPU frequency and consequently they consume less power. This result is in line with the power formula

Fig. 2: Maximum allowable delay effect on power consumption

Fig. 3: Maximum allowable delay effect on the locally executing users

Fig. 4: The Optimal region of typical user for offloading

defined in (1) as well. Also, according to the figure J-PAD algorithm consumes at least about 30% less power than the other algorithms. The reason for moderate slope for decreasing power in equal power algorithm shows the weakness of equal power algorithm where the delay is not considered. Here, there are some users who always send data to the main center and their sum power dominates the power consumption of other users. Fig.3 depicts the portion of locally executing users. The more tolerable delay is allowed, the more users prefer local execution. Because the transmission power costs more than local processing if heavy processing is not needed. In some cases that applications require very short latency and heavy processing, cloud execution is more preferable. Fig.4 depicts the offloading optimal region of a typical user. It can be seen that both bit stream size and acceptable delay have an effect on the offloading decision.

V. CONCLUSION

In this paper, mobile terminal power consumption as a critical aspect of mobile cloud computing networks is considered. Therefore, an optimization problem was defined which aimed at minimizing the aggregate power of all MUs. In order to consider the practical issues, the maximum allowable interference level on each subchannel, the maximum tolerable delay of users who send their data to the cloud, and limited backhaul capacity are taken into account. Knowing the inherent non-convexity of our primary problem, we applied the D.C. approximation and also a penalty factor to transform the non-convex problem to a convex one. The problem is then solved by a resource allocation and decision making algorithm called J-PAD. Finally, simulation results demonstrated that significant enhancement in terms of power saving could be achieved. Specifically, J-PAD algorithm outperformed the other algorithms by 30% while considering the maximum allowable delay.

REFERENCES

- [1] L. Lei, Z. Zhong, K. Zheng, J. Chen, and H. Meng, "Challenges on wireless heterogeneous networks for mobile cloud computing," *IEEE Wireless Communications*, vol. 20, no. 3, pp. 34–44, 2013.
- [2] K. Kumar and Y.-H. Lu, "Cloud computing for mobile users: Can offloading computation save energy?" *Computer*, vol. 43, no. 4, pp. 51–56, 2010.
- [3] J. Kwak, Y. Kim, J. Lee, and S. Chong, "Dream: dynamic resource and task allocation for energy minimization in mobile cloud systems," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2510–2523, 2015.
- [4] Y. Xu and S. Mao, "A survey of mobile cloud computing for rich media applications," *IEEE Wireless Commun.*, vol. 20, no. 3, pp. 1–0, 2013.
- [5] Y. Cai, F. R. Yu, and S. Bu, "Cloud radio access networks (c-ran) in mobile cloud computing systems," in *Computer Communications Workshops (INFOCOM WKSHPS), 2014 IEEE Conference on*. IEEE, 2014, pp. 369–374.
- [6] R. Kaewpuang, D. Niyato, P. Wang, and E. Hossain, "A framework for cooperative resource management in mobile cloud computing," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 12, pp. 2685–2700, 2013.
- [7] Y. Zhang, D. Niyato, and P. Wang, "Offloading in mobile cloudlet systems with intermittent connectivity," *IEEE Transactions on Mobile Computing*, vol. 14, no. 12, pp. 2516–2529, 2015.
- [8] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: Partial computation offloading using dynamic voltage scaling," 2016.
- [9] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for vm-based cloudlets in mobile computing," *IEEE pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.
- [10] W. Zhang, Y. Wen, K. Guan, H. Luo, and D. O. Wu, "Energy-optimal mobile cloud computing under stochastic wireless channel," *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, 2013.
- [11] M. V. Barbera, S. Kosta, A. Mei, and J. Stefa, "To offload or not to offload? the bandwidth and energy costs of mobile cloud computing," in *INFOCOM, 2013 Proceedings IEEE*. IEEE, 2013, pp. 1285–1293.
- [12] X. Liu, Y. Li, and H.-H. Chen, "Wireless resource scheduling based on backoff for multi-user multi-service mobile cloud computing," 2016.
- [13] M.-H. Chen, B. Liang, and M. Dong, "Joint offloading decision and resource allocation for multi-user multi-task mobile cloud," in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.
- [14] S. E. Mahmoodi, K. Subbalakshmi, and V. Sagar, "Cloud offloading for multi-radio enabled mobile devices," in *2015 IEEE International Conference on Communications (ICC)*. IEEE, 2015, pp. 5473–5478.
- [15] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient multi-user computation offloading for mobile-edge cloud computing," *IEEE/ACM Transactions on Networking*, vol. PP, no. 99, pp. 1–1, 2015.
- [16] X. Chen, "Decentralized computation offloading game for mobile cloud computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 4, pp. 974–983, 2015.
- [17] E. Che, H. D. Tuan, and H. H. Nguyen, "Joint optimization of cooperative beamforming and relay assignment in multi-user wireless relay networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 10, pp. 5481–5495, 2014.
- [18] H. H. Kha, H. D. Tuan, and H. H. Nguyen, "Fast Global Optimal Power Allocation in Wireless Networks by Local DC Programming," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 510–515, 2012.
- [19] E. U. T. R. Access, "Further advancements for e-utra physical layer aspects," *3GPP Technical Specification TR*, vol. 36, p. V2, 2010.