



<b>Title</b>	Accelerated Projected Gradient Method for the Optimization of Cell-Free Massive MIMO Downlink
<b>Authors(s)</b>	Farooq, Muhammad, Ngo, Hien Quoc, Tran, Le-Nam
<b>Publication date</b>	2020-09-03
<b>Publication information</b>	Farooq, Muhammad, Hien Quoc Ngo, and Le-Nam Tran. "Accelerated Projected Gradient Method for the Optimization of Cell-Free Massive MIMO Downlink." IEEE, September 3, 2020. <a href="https://doi.org/10.1109/pimrc48278.2020.9217298">https://doi.org/10.1109/pimrc48278.2020.9217298</a> .
<b>Conference details</b>	The 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, Virtual Conference, 31st August - 3 September 2020
<b>Publisher</b>	IEEE
<b>Item record/more information</b>	<a href="http://hdl.handle.net/10197/11906">http://hdl.handle.net/10197/11906</a>
<b>Publisher's statement</b>	© 2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.
<b>Publisher's version (DOI)</b>	10.1109/pimrc48278.2020.9217298

Downloaded 2026-05-02 00:27:58

The UCD community has made this article openly available. Please share how this access benefits you. Your story matters! (@ucd\_oa)



© Some rights reserved. For more information

# Accelerated Projected Gradient Method for the Optimization of Cell-Free Massive MIMO Downlink

Muhammad Farooq\*, Hien Quoc Ngo<sup>†</sup>, and Le Nam Tran\*

\*School of Electrical and Electronic Engineering, University College Dublin, Ireland

Email: muhammad.farooq@ucdconnect.ie; nam.tran@ucd.ie

<sup>†</sup>Institute of Electronics, Communications and Information Technology, Queen's University Belfast, Belfast BT3 9DT, U.K.

Email: hien.ngo@qub.ac.uk

**Abstract**—We consider the downlink of a cell-free massive multiple-input multiple-output (MIMO) system where large number of access points (APs) simultaneously serve a group of users. Two fundamental problems are of interest, namely (i) to maximize the total spectral efficiency (SE), and (ii) to maximize the minimum SE of all users. As the considered problems are non-convex, existing solutions rely on successive convex approximation to find a sub-optimal solution. The known methods use off-the-shelf convex solvers, which basically implement an interior-point algorithm, to solve the derived convex problems. The main issue of such methods is that their complexity does not scale favorably with the problem size, limiting previous studies to cell-free massive MIMO of moderate scales. Thus the potential of cell-free massive MIMO has not been fully understood. To address this issue, we propose an accelerated projected gradient method to solve the considered problems. Particularly, the proposed solution is found in closed-form expressions and only requires the first order information of the objective, rather than the Hessian matrix as in known solutions, and thus is much more memory efficient. Numerical results demonstrate that our proposed solution achieves far less run-time, compared to other second-order methods.

**Index Terms**—Cell-free massive MIMO, sum-rate, power-control, gradient

## I. INTRODUCTION

Cell-free massive multiple-input multiple-output (MIMO) was introduced in [1] to overcome the inter-cell interference which is the main inherent limitation of cellular-based networks. In cell-free massive MIMO, many access points (APs) distributed over the whole network serve many users in the same time-frequency resource. There are no cells, and hence, no boundary effects. Unlike colocated massive MIMO, each AP is equipped with just a few antennas. But when the number of APs is very large, cell-free massive MIMO is still able to exploit the favorable propagation and channel hardening properties, like the colocated massive MIMO. In particular, in the downlink, each AP uses its local channel estimates acquired during the uplink training phase to perform simple beamforming techniques. With this way, there is no need for exchanging the instantaneous channel state information (CSI) among the APs or the central processing unit (CPU) [2].

The research on cell-free massive MIMO is still in its infancy and thus deserves more extensive studies. We discuss here some of the noticeable and related studies in the literature. In [1], Ngo *et al.* considered the problem of minimum rate maximization

to provide uniformly good services to all users. The problem was then solved using a bisection search and a sequence of linear feasibility problems. In [3], Nguyen *et al.* adopted zero-forcing precoding and studied the energy efficiency maximization (EEmax) problem. In this work, an iterative method based on successive convex approximation (SCA) was derived. In [4], both the max-min fairness and sum-rate maximization problems were considered and solved by SCA. The SCA-based method was also used in [2] and [5] to solve the EEmax and max-min fairness power controls with different cell-free massive MIMO setups, respectively.

A common feature of all the above mentioned pioneer studies on cell-free massive MIMO is the use of a second-order interior-point method which requires the computation of the Hessian matrix of the objective, and thus their computational complexity and memory requirement makes them infeasible in practical power control of large-scale cell-free massive MIMO. It only allows us to characterize the performance of cell-free massive MIMO for a relatively small area. For example, the work of [1] was able to consider an area of 1 km × 1 km, consisting of 100 APs serving 40 users. Numbers with the same order of magnitude were also observed in the above mentioned papers. The performance of these scenarios fractionally represents the full potential of cell-free massive MIMO.

To fully understand the performance limits of cell-free massive MIMO, we need to devise more scalable resource allocation methods. To this end, we propose in this paper a first order method to maximize the total spectral efficiency and the minimum rate of the downlink where the conjugate beamforming is adopted. A similar method has been used in [6] to solve the EEmax problem. In proposed method, we customize the accelerated proximal gradient (APG) method presented in [7], which mainly requires the computation of the gradient of the objective. The proposed method is provably convergent and is numerically shown to achieve much lower run-time compared to an existing second-order method given in [2].

**Notations:** Bold lower and upper case letters represent vectors and matrices.  $\mathcal{CN}(0, a)$  denotes a complex Gaussian random variable with zero mean and variance  $a$ .  $\mathbf{X}^T$  and  $\mathbf{X}^\dagger$  stand for the transpose and Hermitian of  $\mathbf{X}$ , respectively.  $x_i$  is the  $i$ -th entry of vector  $\mathbf{x}$ ;  $[\mathbf{X}]_{i,j}$  is the  $(i, j)$ -th entry of  $\mathbf{X}$ .  $\nabla f(\mathbf{x})$  represents the gradient of  $f(\mathbf{x})$  and  $\frac{\partial}{\partial x_i} f(\mathbf{x})$  is the partial

gradient with respect to  $\mathbf{x}_i$ .  $[\mathbf{x}]_+$  denotes the projector onto the positive orthant.  $\|\cdot\|$  represents the Euclidean norm;  $|\cdot|$  is the absolute value of the argument.

## II. SYSTEM MODEL AND PROBLEM FORMULATION

### A. System Model

We consider the downlink of a cell-free massive MIMO system model as in [2]. In particular, there are  $M$  APs serving  $K$  single-antenna users in time division duplex (TDD) mode. Each AP is equipped with  $N$  antennas. All the APs and the users are assumed to be distributed in a large area. As TDD operation is adopted, APs first estimate the channels using pilot sequences from the uplink (commonly known as uplink training) and then apply a beamforming technique to transmit signals to all users in the downlink, or use a matched filter technique to detect signals in the uplink. Since this work focuses on the downlink transmission, we neglect the uplink phase. Let us denote by  $T_c$  and  $T_p$  the length of the coherence time and the uplink training phase in samples. The uplink training and downlink payload transmission phases are summarized as follows. The details can be found in [2].

1) *Uplink Training and Downlink Payload Transmission*: We assume the channel is reciprocal, i.e., the channel gains on the uplink and on the downlink are the same. Consequently, APs can estimate the downlink channel based on the pilot sequences sent by all users on the uplink. Let  $\sqrt{T_p}\boldsymbol{\psi}_k \in \mathbb{C}^{T_p \times 1}$ , where  $\|\boldsymbol{\psi}_k\|^2 = 1$ , be the pilot sequence transmitted from the  $k$ -th user,  $k = 1, \dots, K$ . As in [2], we model the channel between the  $m$ -th AP and the  $k$ -th user as

$$\mathbf{g}_{mk} = \beta_{mk}^{1/2} \mathbf{h}_{mk},$$

where  $\beta_{mk}$  represents the large-scale fading and  $\mathbf{h}_{mk} \in \mathbb{C}^{N \times 1}$  comprises of small-scale fading coefficients between the  $N$  antennas of the  $m$ -th AP and the  $k$ -th user. We further assume that the entries of  $\mathbf{h}_{mk}$  follow i.i.d.  $\mathcal{CN}(0, 1)$ .

Let  $\mathbf{r}_{mk} = \mathbf{R}_{\text{up},m} \boldsymbol{\psi}_k$ , where  $\mathbf{R}_{\text{up},m}$  is the received signal at the  $m$ -th AP. Given  $\mathbf{r}_{mk}$ , the minimum mean-square error (MMSE) estimate of  $\mathbf{g}_{mk}$  is calculated as [2]

$$\hat{\mathbf{g}}_{mk} = \frac{\sqrt{\zeta_p T_p} \beta_{mk}}{1 + \zeta_p T_p \sum_{i=1}^K \beta_{mi} \left| \boldsymbol{\psi}_i^\dagger \boldsymbol{\psi}_k \right|^2} \mathbf{r}_{mk}, \quad (1)$$

where  $\zeta_p$  is the normalized transmit signal-to-noise ratio (SNR) of each pilot symbol. The mean square of any element of  $\hat{\mathbf{g}}_{mk}$  is given by

$$\nu_{mk} = \mathbb{E}\{|\hat{\mathbf{g}}_{mk}|^2\} = \frac{\zeta_p T_p \beta_{mk}^2}{1 + \zeta_p T_p \sum_{i=1}^K \beta_{mi} \left| \boldsymbol{\psi}_i^\dagger \boldsymbol{\psi}_k \right|^2}. \quad (2)$$

For downlink payload data transmission, the APs use the channel estimates (1) and conjugate beamforming to form separate radio beams to the  $K$  users. Denote the symbol to be sent to the  $k$ -th user by  $c_k$  and the power control coefficient between the  $m$ -th AP and the  $k$ -th user by  $\eta_{mk}$ . Then the transmitted signal at the  $m$ -th AP is

$$\mathbf{x}_m = \sqrt{\zeta_d} \sum_{k=1}^K \sqrt{\eta_{mk}} \hat{\mathbf{g}}_{mk}^* c_k, \quad (3)$$

where  $\zeta_d$  is the maximum downlink transmit power at each AP normalized to the noise power. Note that the total power at each AP is

$$\mathbb{E}\{\|\mathbf{x}_m\|^2\} = \zeta_d N \sum_{k=1}^K \eta_{mk} \nu_{mk}. \quad (4)$$

The received signal at the  $k$ -th user is written as

$$r_k = \sum_{m=1}^M \mathbf{g}_{mk}^\top \mathbf{x}_m + w_k, \quad (5)$$

where  $w_k \sim \mathcal{CN}(0, 1)$  is the white Gaussian noise.

2) *Spectral Efficiency*: By using the use-and-then-forget capacity bounding technique, we can obtain the following spectral efficiency of the  $k$ -th user [2]

$$\text{SE}_k(\bar{\boldsymbol{\eta}}) = \left(1 - \frac{T_p}{T_c}\right) \log_2(1 + \gamma_k(\bar{\boldsymbol{\eta}})) \quad (\text{bit/s/Hz}), \quad (6)$$

where

$$\gamma_k(\bar{\boldsymbol{\eta}}) = \frac{\zeta_d \left| \boldsymbol{\nu}_{kk}^\top \bar{\boldsymbol{\eta}}_k \right|^2}{\zeta_d \sum_{i \neq k}^K \left| \boldsymbol{\nu}_{ik}^\top \bar{\boldsymbol{\eta}}_i \right|^2 + \frac{\zeta_d}{N} \sum_{i=1}^K \|\mathbf{D}_{ik} \bar{\boldsymbol{\eta}}_i\|_2^2 + \frac{1}{N^2}} \quad (7)$$

and where  $\bar{\boldsymbol{\eta}}_k = [\sqrt{\eta_{1k}}; \dots; \sqrt{\eta_{Mk}}] \in \mathbb{R}_+^M$  consists of all power control coefficients associated with user  $k$ ,  $\bar{\boldsymbol{\eta}} = [\bar{\boldsymbol{\eta}}_1; \bar{\boldsymbol{\eta}}_2; \dots; \bar{\boldsymbol{\eta}}_K] \in \mathbb{R}_+^{MK}$ ,  $\mathbf{D}_{ik} \in \mathbb{R}_+^{M \times M}$  is a diagonal matrix with  $[\mathbf{D}_{ik}]_{m,m} = \sqrt{\nu_{mi} \beta_{mk}}$ , and  $\boldsymbol{\nu}_{ik} \triangleq \left| \boldsymbol{\psi}_i^\dagger \boldsymbol{\psi}_k \right| \left[ \nu_{1i} \frac{\beta_{1k}}{\beta_{1i}}; \nu_{2i} \frac{\beta_{2k}}{\beta_{2i}}; \dots; \nu_{Mi} \frac{\beta_{Mk}}{\beta_{Mi}} \right]$ .

### B. Problem Formulation

To formulate the considered problem and to facilitate the development of the proposed algorithm, we define  $\boldsymbol{\mu}_m \in \mathbb{R}_+^K$  to be the vector of all power control coefficients associated with the  $m$ -th AP as  $\boldsymbol{\mu}_m \triangleq [\mu_{m1}; \mu_{m2}; \dots; \mu_{mK}]$ , where  $\mu_{mk} = \sqrt{\eta_{mk} \nu_{mk}}$ ,  $m = 1, \dots, M$ ,  $k = 1, \dots, K$ . We also define  $\boldsymbol{\mu} \triangleq [\boldsymbol{\mu}_1; \boldsymbol{\mu}_2; \dots; \boldsymbol{\mu}_M] \in \mathbb{R}_+^{MK \times 1}$  to include the power control coefficients of all APs. To express the spectral efficiency in (6) as a function of  $\boldsymbol{\mu}$ , we denote by  $\bar{\boldsymbol{\mu}}_k = [\mu_{1k}; \mu_{2k}; \dots; \mu_{Mk}]$  the vector of power control coefficients associated with user  $k$ . Thus we can write  $\boldsymbol{\nu}_{ik}^\top \bar{\boldsymbol{\eta}}_i$  as  $\bar{\boldsymbol{\nu}}_{ik}^\top \bar{\boldsymbol{\mu}}_k$ , where  $\bar{\boldsymbol{\nu}}_{ik} \triangleq \left| \boldsymbol{\psi}_i^\dagger \boldsymbol{\psi}_k \right| \left[ \sqrt{\nu_{1i} \frac{\beta_{1k}}{\beta_{1i}}}; \sqrt{\nu_{2i} \frac{\beta_{2k}}{\beta_{2i}}}; \dots; \sqrt{\nu_{Mi} \frac{\beta_{Mk}}{\beta_{Mi}}} \right]$ . Similarly, we can write  $\mathbf{D}_{ik} \bar{\boldsymbol{\eta}}_i$  as  $\bar{\mathbf{D}}_i \bar{\boldsymbol{\mu}}_i$ , where  $\bar{\mathbf{D}}_i$  is a diagonal matrix with the  $m$ -th diagonal entry equal to  $\sqrt{\beta_{mi}}$ . Now the spectral efficiency of the  $k$ -th user (in nat/s/Hz) can be expressed as

$$\text{SE}_k(\boldsymbol{\mu}) = \left(1 - \frac{T_p}{T_c}\right) \log(1 + \gamma_k(\boldsymbol{\mu})), \quad (8)$$

where  $\gamma_k(\boldsymbol{\mu})$  is the SINR of the  $k$ -th user given by

$$\gamma_k(\boldsymbol{\mu}) = \frac{\zeta_d (\bar{\boldsymbol{\nu}}_{kk}^\top \bar{\boldsymbol{\mu}}_k)^2}{\zeta_d \sum_{i \neq k}^K (\bar{\boldsymbol{\nu}}_{ik}^\top \bar{\boldsymbol{\mu}}_i)^2 + \frac{\zeta_d}{N} \sum_{i=1}^K \|\bar{\mathbf{D}}_i \bar{\boldsymbol{\mu}}_i\|_2^2 + \frac{1}{N^2}}. \quad (9)$$

The total spectral efficiency of the system is defined as

$$\text{SE}(\boldsymbol{\mu}) \triangleq \sum_{k=1}^K \text{SE}_k(\boldsymbol{\mu}). \quad (10)$$

In this paper, we consider a total power constraint at each AP which is given by  $\mathbb{E}\{\|\mathbf{x}_m\|^2\} \leq \zeta_d$  or  $\|\boldsymbol{\mu}_m\|^2 \leq \frac{1}{N}$ ,  $m = 1, 2, \dots, M$ , which follows from (4). For the problem formulation purpose, we define the following set  $\mathcal{S} = \{\boldsymbol{\mu} | \boldsymbol{\mu} \geq 0; \|\boldsymbol{\mu}_m\|^2 \leq 1/N, m = 1, 2, \dots, M\}$ . In this paper, we consider the following two common power control optimization problems, namely

- The problem of maximizing the total spectral efficiency (SEmax)

$$(\mathcal{P}_1) : \underset{\boldsymbol{\mu}}{\text{maximize}} \left\{ \sum_{k=1}^K \text{SE}_k(\boldsymbol{\mu}) \mid \boldsymbol{\mu} \in \mathcal{S} \right\} \quad (11)$$

- The problem of maximizing the minimum rate (MRmax) among all users (also known as max-min fairness maximization)

$$(\mathcal{P}_2) : \underset{\boldsymbol{\mu}}{\text{maximize}} \left\{ \min_{1 \leq k \leq K} \text{SE}_k(\boldsymbol{\mu}) \mid \boldsymbol{\mu} \in \mathcal{S} \right\} \quad (12)$$

For the above problems, SCA has gradually become a standard mathematical tool [1], [2]. In all known solutions for the considered problems or related ones, interior point methods (through the use of off-the-shelf convex solvers) are invoked to solve these convex problems [2], [4], [5], which do not scale with the problem size. In the next section, we propose methods that can tackle this scalability problem.

### III. PROPOSED SOLUTIONS

In this section, we present solutions to  $(\mathcal{P}_1)$  and  $(\mathcal{P}_2)$ , using the accelerated proximal gradient (APG) methods introduced in [7], which concerns the following problem

$$\min_{\boldsymbol{\mu} \in \mathbb{R}^n} \{F(\boldsymbol{\mu}) \equiv f(\boldsymbol{\mu}) + g(\boldsymbol{\mu})\}, \quad (13)$$

where  $f(\boldsymbol{\mu})$  is differentiable (but possibly *nonconvex*) and  $g(\boldsymbol{x})$  can be both nonconvex and *nonsmooth*. If we let  $g(\boldsymbol{\mu})$  be the indicator function of  $\mathcal{S}$ , defined as

$$\delta_{\mathcal{S}}(\boldsymbol{\mu}) = \begin{cases} 0 & \boldsymbol{\mu} \in \mathcal{S} \\ +\infty & \boldsymbol{\mu} \notin \mathcal{S}, \end{cases} \quad (14)$$

then (13) is actually equivalent to  $(\mathcal{P}_1)$ . Basically the objective of  $(\mathcal{P}_2)$  is nonsmooth, but we can still solve  $(\mathcal{P}_2)$  by applying a proper smoothing technique. We also note that when  $g(\boldsymbol{\mu})$  is the indicator function of  $\mathcal{S}$ , the proximal operator of  $g(\boldsymbol{\mu})$  becomes the Euclidean projection onto  $\mathcal{S}$ . In the following, we customize the APG methods to solve the considered problems.

#### A. Proposed Solution to $(\mathcal{P}_1)$

Since  $f(\boldsymbol{\mu})$  for  $(\mathcal{P}_1)$  is differentiable, the proposed algorithm for solving  $(\mathcal{P}_1)$  follows closely the monotone APG method in [7], which is outlined in Algorithm 1. In Algorithm 1,  $\alpha > 0$  is called the step size which should be sufficiently small to guarantee its convergence. Also, the notation  $P_{\mathcal{S}}(\mathbf{u})$  denotes the projection onto  $\mathcal{S}$ , i.e.,  $P_{\mathcal{S}}(\mathbf{u}) = \arg \min \{ \|\mathbf{x} - \mathbf{u}\| \mid \mathbf{x} \in \mathcal{S} \}$ . From a given operating point, we move along the direction of the gradient of  $f(\boldsymbol{\mu})$  with the step size  $\alpha$ , and then project the resulting point onto the feasible set. In particular,  $\mathbf{y}^n$  in Step 4 is an extrapolated point which is used for convergence acceleration. However,  $\mathbf{y}^n$  can be a bad extrapolation unlike APG methods for the convex case, and thus Step 7 is there to fix this issue. We now give the details for the two main operations of Algorithm 1, namely: the projection onto the feasible set  $\mathcal{S}$  and the gradient of  $f(\boldsymbol{\mu})$ .

---

#### Algorithm 1 General Description of Proposed Algorithm for Solving $(\mathcal{P}_1)$ and $(\mathcal{P}_2)$

---

```

1: Input:  $\boldsymbol{\mu}^0 \succcurlyeq 0, t_0 = t_1 = 1, \alpha > 0$ 
2:  $\boldsymbol{\mu}^1 = \mathbf{z}^1 = \boldsymbol{\mu}^0$ 
3: for  $n = 1, 2, \dots$  do
4:    $\mathbf{y}^n = \boldsymbol{\mu}^n + \frac{t_{n-1}}{t_n}(\mathbf{z}^n - \boldsymbol{\mu}^n) + \frac{t_{n-1}-1}{t_n}(\boldsymbol{\mu}^n - \boldsymbol{\mu}^{n-1})$ 
5:    $\mathbf{z}^{n+1} = P_{\mathcal{S}}(\mathbf{y}^n + \alpha \nabla f(\mathbf{y}^n))$ 
6:    $\mathbf{v}^{n+1} = P_{\mathcal{S}}(\boldsymbol{\mu}^n + \alpha \nabla f(\boldsymbol{\mu}^n))$ 
7:    $\boldsymbol{\mu}^{n+1} = \begin{cases} \mathbf{z}^{n+1} & f(\mathbf{z}^{n+1}) \geq f(\mathbf{v}^{n+1}) \\ \mathbf{v}^{n+1} & \text{otherwise} \end{cases}$ 
8:    $t_{n+1} = 0.5(\sqrt{4t_n^2 + 1} + 1)$ 
9: end for
10: Output:  $\boldsymbol{\mu}^*$ 

```

---

1) *Projection onto  $\mathcal{S}$* : We show that the projection in Steps 5 and 6 in Algorithm 1 can be done *in parallel* and by *closed-form expressions*. Recall that for a given  $\mathbf{x} \in \mathbb{R}^{MK \times 1}$ ,  $P_{\mathcal{S}}(\mathbf{x})$  is the solution to the following problem

$$\underset{\boldsymbol{\mu} \in \mathbb{R}^{MK \times 1}}{\text{minimize}} \left\{ \|\boldsymbol{\mu} - \mathbf{x}\|^2 \mid \boldsymbol{\mu} \geq 0; \|\boldsymbol{\mu}_m\|^2 \leq \frac{1}{N}, \right. \\ \left. m = 1, 2, \dots, M \right\}. \quad (15a)$$

It is easy to see that the above problem can be decomposed into sub-problems at each AP  $m$  as

$$\underset{\boldsymbol{\mu}_m \in \mathbb{R}^{K \times 1}}{\text{minimize}} \left\{ \|\boldsymbol{\mu}_m - \mathbf{x}_m\|^2 \mid \boldsymbol{\mu}_m \geq 0; \|\boldsymbol{\mu}_m\|^2 \leq \frac{1}{N} \right\}. \quad (16a)$$

The above problem is in fact the projection onto the intersection of a ball and the positive orthant. Interestingly, the analytical solution to this problem can be found by applying [8, Theorem 7.1], which produces

$$\boldsymbol{\mu}_m = \frac{\sqrt{1/N}}{\max\left(\sqrt{1/N}, \|\mathbf{x}_m\|_+\right)} [\mathbf{x}_m]_+. \quad (17)$$

2) *Gradient of  $f(\boldsymbol{\mu})$  for  $(\mathcal{P}_1)$* : To implement Algorithm 1, we also need to compute  $\nabla_{\boldsymbol{\mu}} f(\boldsymbol{\mu})$ , which is found as

$$\nabla f(\boldsymbol{\mu}) = \left[ \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_1} f(\boldsymbol{\mu}); \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_2} f(\boldsymbol{\mu}), \dots, \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_K} f(\boldsymbol{\mu}) \right], \quad (18)$$

where  $\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} f(\boldsymbol{\mu}) = \sum_{k=1}^K \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} \text{SE}_k(\boldsymbol{\mu})$ . Thus it basically boils down to finding  $\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} \text{SE}_k(\boldsymbol{\mu})$ . To this end, let us define  $b_k(\boldsymbol{\mu}) = \zeta_d (\bar{\mathbf{V}}_{kk}^T \bar{\boldsymbol{\mu}}_k)^2$  and  $c_k(\boldsymbol{\mu}) = \zeta_d \left( \sum_{i \neq k}^K (\bar{\mathbf{V}}_{ik}^T \bar{\boldsymbol{\mu}}_i)^2 + \frac{1}{N} \sum_{i=1}^K \|\bar{\mathbf{D}}_i \bar{\boldsymbol{\mu}}_i\|^2 \right) + \frac{1}{N^2}$ . Then we can rewrite  $\text{SE}_k(\boldsymbol{\mu})$  (6) as

$$\text{SE}_k(\boldsymbol{\mu}) = \log(b_k(\boldsymbol{\mu}) + c_k(\boldsymbol{\mu})) - \log c_k(\boldsymbol{\mu}). \quad (19)$$

The gradient of  $\text{SE}_k(\boldsymbol{\mu})$  with respect to  $\bar{\boldsymbol{\mu}}_i$ ,  $i = 1, 2, \dots, K$ , is found as

$$\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} \text{SE}_k(\boldsymbol{\mu}) = \frac{\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} (b_k(\boldsymbol{\mu}) + c_k(\boldsymbol{\mu}))}{b_k(\boldsymbol{\mu}) + c_k(\boldsymbol{\mu})} - \frac{\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} c_k(\boldsymbol{\mu})}{c_k(\boldsymbol{\mu})}. \quad (20)$$

Now we recall the following equality  $\nabla \|\mathbf{A}\mathbf{x}\|^2 = 2\mathbf{A}^T \mathbf{A}\mathbf{x}$  for any symmetric matrix  $\mathbf{A}$ , and thus  $\nabla_{\bar{\boldsymbol{\mu}}_i} b_k(\boldsymbol{\mu})$  and  $\nabla_{\bar{\boldsymbol{\mu}}_i} c_k(\boldsymbol{\mu})$

are respectively given by

$$\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} b_k(\boldsymbol{\mu}) = \begin{cases} 2\zeta_d \bar{\mathbf{v}}_{kk} \bar{\mathbf{v}}_{kk}^T \bar{\boldsymbol{\mu}}_k, & i = k \\ 0, & i \neq k \end{cases} \quad (21)$$

$$\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} c_k(\boldsymbol{\mu}) = \begin{cases} 2(\zeta_d/N) \bar{\mathbf{D}}_k^2 \bar{\boldsymbol{\mu}}_k, & i = k \\ 2\zeta_d \bar{\mathbf{v}}_{ik} \bar{\mathbf{v}}_{ik}^T \bar{\boldsymbol{\mu}}_i + \frac{2\zeta_d}{N} \bar{\mathbf{D}}_i^2 \bar{\boldsymbol{\mu}}_i, & i \neq k \end{cases}. \quad (22)$$

### B. Proposed Solution to $(\mathcal{P}_2)$

We recall that for  $(\mathcal{P}_2)$  the objective function is

$$f(\boldsymbol{\mu}) = \min_{1 \leq k \leq K} \text{SE}_k(\boldsymbol{\mu}), \quad (23)$$

which is *non-differentiable*. Thus a straightforward application of the APG method is impossible. To overcome this issue, we adopt a smoothing technique. In particular,  $f(\boldsymbol{\mu})$  is approximated by the following log-sum-exp function given by [9]

$$f_\tau(\boldsymbol{\mu}) = -\frac{1}{\tau} \log\left(\frac{1}{K} \sum_{k=1}^K \exp(-\tau \text{SE}_k(\boldsymbol{\mu}))\right), \quad (24)$$

where  $\tau > 0$  is the positive *smoothness* parameter. To obtain (24), we have used the fact that  $\min_{1 \leq k \leq K} \text{SE}_k(\boldsymbol{\mu}) = -\max_{1 \leq k \leq K} -\text{SE}_k(\boldsymbol{\mu})$ . Nesterov proved in [9] that  $f(\boldsymbol{\mu}) + \frac{\log K}{\tau} \geq f_\tau(\boldsymbol{\mu}) \geq f(\boldsymbol{\mu})$ . In other words,  $f_\tau(\boldsymbol{\mu})$  is a differentiable approximation of  $f(\boldsymbol{\mu})$  with a numerical accuracy of  $\frac{\log K}{\tau}$ . Thus, with a sufficiently high  $\tau$ , we can find an approximate solution to  $(\mathcal{P}_2)$  by running Algorithm 1 with  $f(\boldsymbol{\mu})$  being replaced by  $f_\tau(\boldsymbol{\mu})$  in (24). In this regard, the gradient of  $f_\tau(\boldsymbol{\mu})$  is easily found as  $\nabla_{\bar{\boldsymbol{\mu}}} f_\tau(\boldsymbol{\mu}) = [\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_1} f_\tau(\boldsymbol{\mu}), \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_2} f_\tau(\boldsymbol{\mu}), \dots, \frac{\partial}{\partial \bar{\boldsymbol{\mu}}_K} f_\tau(\boldsymbol{\mu})]$ , where  $\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} f_\tau(\boldsymbol{\mu})$  is given by

$$\frac{\partial}{\partial \bar{\boldsymbol{\mu}}_i} f_\tau(\boldsymbol{\mu}) = \frac{\sum_{k=1}^K \left( \exp(-\tau \text{SE}_k(\boldsymbol{\mu})) \nabla_{\bar{\boldsymbol{\mu}}_i} \text{SE}_k(\boldsymbol{\mu}) \right)}{\sum_{k=1}^K \exp(-\tau \text{SE}_k(\boldsymbol{\mu}))}. \quad (25)$$

## IV. COMPLEXITY AND CONVERGENCE ANALYSIS OF PROPOSED METHODS

### A. Complexity Analysis

We now provide the complexity analysis of the proposed algorithm for one iteration using the big-O notation. It is clear that the complexity of Algorithm 1 is dominated by the computation of three quantities: the objective, the gradient, and the projection. It is easy to see that  $KM$  multiplications are required to compute  $\text{SE}_k(\boldsymbol{\mu})$ . Therefore, the complexity of finding  $f(\boldsymbol{\mu}) = \sum_{k=1}^K \text{SE}_k(\boldsymbol{\mu})$  is  $\mathcal{O}(K^2M)$ . Similarly, we can find that the complexity of  $\nabla \text{SE}(\boldsymbol{\mu})$  which is  $\mathcal{O}(K^2M)$ . The projection of  $\boldsymbol{\mu}$  onto  $\mathcal{S}$  is given in (17), which requires the computation of the  $l_2$ -norm of  $K \times 1$  vector  $\mathbf{x}_m$  at each AP, and thus the complexity of the projection is  $\mathcal{O}(KM)$ . In summary, the per-iteration complexity of the proposed algorithm for solving  $(\mathcal{P}_1)$  is  $\mathcal{O}(K^2M)$ . We can also conclude that the per-iteration complexity for solving  $(\mathcal{P}_2)$  is also  $\mathcal{O}(K^2M)$ .

### B. Convergence Analysis

We now discuss the convergence result of Algorithm 1 for solving  $(\mathcal{P}_1)$  and  $(\mathcal{P}_2)$ . For  $(\mathcal{P}_1)$ , from (20), (21), (22), we can check that  $\nabla f(\boldsymbol{\mu})$  is Lipschitz continuous, or equivalently

$f(\boldsymbol{\mu})$  has Lipschitz continuous gradient. That is, there exists a constant  $L > 0$  such that

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L \|\mathbf{x} - \mathbf{y}\|. \quad (26)$$

For the considered problems, it is possible to find a Lipschitz constant for  $\nabla f(\boldsymbol{\mu})$  but we skip the details here for the sake of brevity. Let us simply denote by  $L$  the Lipschitz constant of  $\nabla f(\boldsymbol{\mu})$ . The convergence of Algorithm 1 is stated in the following lemma.

**Lemma 1.** *Let the step size  $\alpha$  in Algorithm 1 satisfy  $\alpha < \frac{1}{L}$ . Then the iterates  $\{\boldsymbol{\mu}^n\}$  generated by Algorithm 1 are bounded. Let  $\boldsymbol{\mu}^*$  be any accumulation point of  $\{\boldsymbol{\mu}^n\}$ , then  $\boldsymbol{\mu}^*$  is a critical point of  $(\mathcal{P}_1)$ .*

*Proof:* Please see Appendix A. ■

The same convergence result applies to Algorithm 1 for solving  $(\mathcal{P}_2)$  with the differential approximation in (24).

### C. Improved Convergence with Line Search

In practice, we do not need to compute a Lipschitz constant of  $\nabla f(\boldsymbol{\mu})$  for two reasons. First, the best Lipschitz constant of  $\nabla f(\boldsymbol{\mu})$  (i.e. the smallest  $L$  such that (26) holds) is hard to find. Second, the conditions  $\alpha < \frac{1}{L}$  is sufficient but not necessary for Algorithm 1 to converge. Thus, we can allow  $\alpha$  to take on larger values to speed up the convergence of Algorithm 1 by means of a linear search procedure. Specifically, let  $L_0 > 0$ ,  $\delta > 0$  and  $\rho \in (0, 1)$ . Then the step size  $\alpha$  in Step 5 of Algorithm 1 is set to  $\alpha = L_0 \rho^{m_n}$ , where  $m_n$  is the first nonnegative smallest integer such that

$$\mathbf{z}^{n+1} = P_{\mathcal{S}}(\mathbf{y}^n + L_0 \rho^{m_n} \nabla f(\mathbf{y}^n)), \quad (27a)$$

$$f(\mathbf{z}^{n+1}) \geq f(\mathbf{y}^n) + \delta \|\mathbf{z}^{n+1} - \mathbf{y}^n\|^2. \quad (27b)$$

The above line search follows the Armijo rule. The same line search procedure can be used for Step 6 of Algorithm 1. We refer to these modifications as Algorithm 1 with line search.

## V. NUMERICAL RESULTS

In this section, we evaluate the performance of the proposed method in terms of computational complexity and achieved spectral efficiency. The users and the APs are uniformly dropped over a  $D \times D$  km<sup>2</sup>. The large-scale fading coefficient between the  $m$ -th AP and the  $k$ -th user is generated as

$$\beta_{mk} = \text{PL}_{mk} \cdot z_{mk},$$

where  $\text{PL}_{mk}$  and  $z_{mk}$  represent the path loss and log-normal shadowing with mean zero and standard deviation  $\sigma_{\text{sh}}$ , respectively. In this paper we adopt the three-slope path loss model as in [2] in which  $\text{PL}_{mk}$  (in dB) is given by

$$\text{PL}_{mk} = \begin{cases} -L - 15 \log_{10}(d_1) - 20 \log_{10}(d_0) & d_{mk} < d_0, \\ -L - 35 \log_{10}(d_{mk}) & d_1 < d_{mk}, \\ -L - 15 \log_{10}(d_1) - 20 \log_{10}(d_{mk}) & \text{otherwise,} \end{cases}$$

where  $L$  is a constant dependent on carrier frequency,  $d_{mk}$  (in km) is the distance between the  $m$ -th AP and the  $k$ -th user, and  $d_0$  and  $d_1$  (both in km) are reference distances.

Similar to [2], we choose  $L = 140.7$  dB,  $d_0 = 0.01$  km and  $d_1 = 0.05$  km. We consider a system having a bandwidth

of  $B = 20$  MHz, the noise power density is  $N_0 = -174$  (dBm/Hz), and a noise figure of 9 dB. The length of the coherence time and the uplink training phase are set to  $T_p = 20$ ,  $T_c = 200$ , respectively. If not otherwise mentioned, we set the power transmit power for downlink data transmission and uplink training phase (before normalization) as  $\zeta_d = 1$  W and  $\zeta_p = 0.2$  W. Single antenna AP is considered in all numerical experiments. These parameters are taken from [2].

For comparison purpose, we modify the SCA-based method presented in [2], which is dedicated to the EEmax problem. However, it can be easily modified to deal with the problems considered in this paper. In this regard we use convex conic solver MOSEK [10] through the modeling tool YALMIP [11].

In the first numerical experiment, we compare the convergence rate of the proposed method with the SCA-based method. Figures 1(a) and 1(b) show the convergence of the proposed

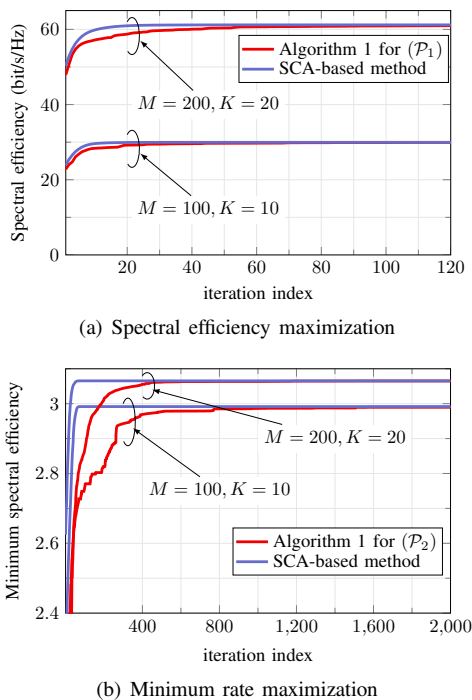


Fig. 1. Total spectral efficiency and the minimum spectral efficiency versus the number of iterations. The values of  $M$  and  $K$  are given explicitly the figure. Each AP is equipped with one antenna.

method and the SCA-based method for the total spectral efficiency and the min-rate maximization problem, respectively. We can see that the proposed and the SCA-based methods achieve the same performance but the SCA-based method requires fewer iterations. However, the main advantage of our proposed method over the SCA-based method is that each iteration of the proposed method is very memory efficient, and hence, performs very fast. As a result, the run-time of the proposed method is far less than that of the SCA-based method as shown in Table I. In Table I, we report the actual run-time of both methods to solve the SEmax problem. Here, we execute our codes on a 64-bit Windows operating system with 16 GB RAM and Intel CORE i7, 3.7 GHz. Both iterative methods are terminated when the

difference of the objective for the last 5 iterations is less than  $10^{-3}$ .

TABLE I  
COMPARISON OF RUN-TIME (IN SECONDS) BETWEEN THE PROPOSED METHOD AND THE SCA-BASED METHOD. HERE,  $K = 40$  AND  $D = 1$ .

APs	SCA Method	Proposed Method
200	330.84	<b>2.88</b>
400	408.94	<b>9.42</b>
800	1115.18	<b>18.07</b>
1600	1648.09	<b>49.45</b>

We next take advantage of the proposed method to explore the spectral efficiency performance of cell-free massive MIMO for a large metropolitan area. In particular, we investigate the performance for two cases  $D = 1$  and  $D = 10$ . To obtain a fair comparison we keep the AP density, defined as the number of APs per square kilometer, the same for both cases. Note that the AP density of 1000 means 10 000 APs for the case of  $D = 10$ , which has not been studied in the literature previously. To appreciate the proposed method for this large-scale scenario, we compare it with the SCA method and the equal power allocation (EPA) method where the power control coefficient  $\eta_{mk}$  is given by,  $\eta_{mk} = (\sum_{i=1}^k \nu_{mi})^{-1}$ . The results in Figure 2 are interesting. First, increasing the AP density improves the sum spectral efficiency of the system. Second, for the same AP density, a larger area provides a better sum spectral efficiency. The reason is that for a larger area, the users that are served by the APs become far apart each other. As a result, the inter-user interference becomes weaker, leading to an improved sum spectral efficiency. On the other hand, the EPA method yields smaller spectral efficiency as the coverage area is larger because more power should be spent to the users having small path loss. The SCA method produces the same spectral efficiency as the proposed APG method but it is unable to run for  $D = 10$  on the system specifications mentioned above. Thus, the proposed scheme outclasses both SCA-based and EPA methods in terms of improved sum spectral efficiency and larger coverage area.

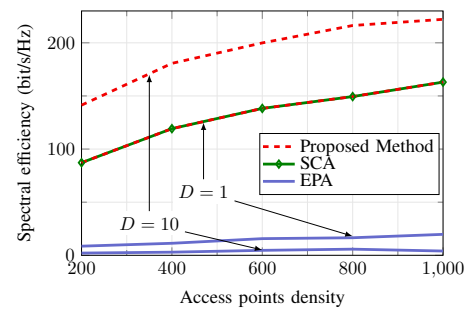


Fig. 2. Total spectral efficiency versus AP density. The number of users is  $K = 40$ .

Finally, we compare the cumulative distribution function (CDF) of the per-user spectral efficiency for the EPA, SEmax, and max-min fairness power controls, see Figure 3. Here, we consider two scenarios: (i) small-scale scenario:  $M = 100$ ,  $K =$

20,  $T_c = 200$ ,  $T_p = 20$ ,  $D = 1$ ; and (ii) large-scale scenario:  $M = 2000$ ,  $K = 500$ ,  $T_c = 1000$ ,  $T_p = 200$ ,  $D = 1$ . As

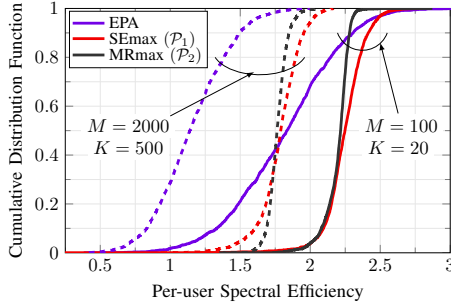


Fig. 3. CDF of per-user spectral efficiency for  $(\mathcal{P}_1)$  and  $(\mathcal{P}_2)$ . The parameters other than the mentioned are kept the same.

expected, the spread of CDF for the SEmax power control is the larger than the max-min fairness power control. On the other hand, the order is reversed in terms of fairness as the CDF curve of the max-min problem has the largest slope which shows that the spectral efficiencies of all the users are very close to each other. The per-user spectral efficiencies in the large-scale scenario are lesser than the ones in the small-scale scenario. This is because when the number users increases, the inter-user interference increases which yields to the reduced performance.

## VI. CONCLUSION

We have considered the downlink of cell-free massive MIMO and aimed to maximize the total and minimum spectral efficiencies, subject to a sum power constraint at each AP. Conjugate beamforming has been adopted, resulting in a power control problem for which an accelerated project gradient method has been proposed. The proposed solution only requires the first order information of the objective and, in particular, can be founded by closed-form expressions. We have numerically shown that the proposed method can achieve the same spectral efficiency as a known SCA-based method but with much lesser run-time. For the first time, we have evaluated the SE performance of cell-free massive MIMO for an area of  $10 \text{ km} \times 10 \text{ km}$ , consisting of up to 10 000 APs. In this case, the achieved sum spectral efficiency can be up to 200 (bit/s/Hz).

## APPENDIX A

### CONVERGENCE PROOF OF ALGORITHM 1

The proof is due to [7]. We begin with by recalling an important inequality of a  $L$ -smooth function. Specifically, for a function  $f(x)$  has the Lipschitz continuous gradient with a constant  $L$ , the following inequality holds

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle - \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|^2. \quad (28)$$

The projection in Step 6 of Algorithm 1 can be written as

$$\begin{aligned} \mathbf{v}^{n+1} &= \arg \min_{\boldsymbol{\mu} \in \mathcal{S}} \|\boldsymbol{\mu} - \boldsymbol{\mu}^n - \alpha \nabla f(\boldsymbol{\mu}^n)\|^2 \\ &= \arg \max_{\boldsymbol{\mu} \in \mathcal{S}} \langle \nabla f(\boldsymbol{\mu}^n), \boldsymbol{\mu} - \boldsymbol{\mu}^n \rangle - \frac{1}{2\alpha} \|\boldsymbol{\mu} - \boldsymbol{\mu}^n\|^2, \quad (29) \end{aligned}$$

and thus we have

$$\langle \nabla f(\boldsymbol{\mu}^n), \mathbf{v}^{n+1} - \boldsymbol{\mu}^n \rangle - \frac{1}{2\alpha} \|\mathbf{v}^{n+1} - \boldsymbol{\mu}^n\|^2 \geq 0. \quad (30)$$

Applying (28) yields

$$\begin{aligned} f(\mathbf{v}^{n+1}) &\geq f(\boldsymbol{\mu}^n) + \langle \nabla f(\boldsymbol{\mu}^n), \mathbf{v}^{n+1} - \boldsymbol{\mu}^n \rangle \\ &\quad - \frac{L}{2} \|\mathbf{v}^{n+1} - \boldsymbol{\mu}^n\|^2 \\ &\geq f(\boldsymbol{\mu}^n) + \left( \frac{1}{2\alpha} - \frac{L}{2} \right) \|\mathbf{v}^{n+1} - \boldsymbol{\mu}^n\|^2. \quad (31) \end{aligned}$$

It is easy to see that  $f(\mathbf{v}^{n+1}) \geq f(\boldsymbol{\mu}^n)$  if  $\alpha < \frac{1}{L}$ . From Step 7, if  $f(\mathbf{z}^{n+1}) \geq f(\mathbf{v}^{n+1})$ , then

$$\boldsymbol{\mu}^{n+1} = \mathbf{z}^{n+1}, f(\boldsymbol{\mu}^{n+1}) = f(\mathbf{z}^{n+1}) \geq f(\mathbf{v}^{n+1}). \quad (32)$$

Similarly, if  $f(\mathbf{z}^{n+1}) < f(\mathbf{v}^{n+1})$ , then

$$\boldsymbol{\mu}^{n+1} = \mathbf{v}^{n+1}, f(\boldsymbol{\mu}^{n+1}) = f(\mathbf{v}^{n+1}). \quad (33)$$

From (31), (32), and (33) we have

$$f(\boldsymbol{\mu}^{n+1}) \geq f(\mathbf{v}^{n+1}) \geq f(\boldsymbol{\mu}^n). \quad (34)$$

Since the feasible set of the considered problems is compact convex, the iterates  $\{\mathbf{v}^n\}$  and  $\{\boldsymbol{\mu}^n\}$  are both bounded and thus,  $\{\boldsymbol{\mu}^n\}$  has accumulation points. We can prove that each accumulation point of Algorithm 1 is indeed a critical point of  $(\mathcal{P}_1)$ , following the same arguments as those in [7] but skip the details due to the space limitation.

## ACKNOWLEDGMENT

This publication has emanated from research supported in part by a grant from Science Foundation Ireland under grant number 17/CDA/4786.

## REFERENCES

- [1] H. Q. Ngo, A. Ashikhmin, H. Yang, E. G. Larsson, and T. L. Marzetta, "Cell-free massive MIMO versus small cells," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1834–1850, March 2017.
- [2] H. Q. Ngo, L.-N. Tran, T. Q. Duong, M. Matthaiou, and E. G. Larsson, "On the total energy efficiency of cell-free massive MIMO," *IEEE Transactions on Green Communications and Networking*, vol. 2, no. 1, pp. 25–39, March 2018.
- [3] L. D. Nguyen, T. Q. Duong, H. Q. Ngo, and K. Tourki, "Energy efficiency in cell-free massive MIMO with zero-forcing precoding design," *IEEE Commun. Lett.*, vol. 21, no. 8, pp. 1871–1874, Aug 2017.
- [4] S. Buzzi and A. Zappone, "Downlink power control in user-centric and cell-free massive MIMO wireless networks," in *IEEE PIMRC*, Oct 2017, pp. 1–6.
- [5] G. Interdonato, H. Q. Ngo, P. Frenger, and E. G. Larsson, "Downlink training in cell-free massive MIMO: A blessing in disguise," *IEEE Trans. Wireless Commun.*, 2019, in Press.
- [6] L.-N. Tran and H. Q. Ngo, "First-order methods for energy-efficient power control in cell-free massive mimo," in *53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2019, pp. 848–852.
- [7] H. Li and Z. Lin, "Accelerated proximal gradient methods for nonconvex programming," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 379–387.
- [8] H. H. Bauschke, M. N. Bui, and X. Wang, "Projecting onto the intersection of a cone and a sphere," *SIAM Journal on Optimization*, vol. 28, no. 3, pp. 2158–2188, Jan 2018.
- [9] Y. Nesterov, "Smooth minimization of non-smooth functions," *Math. Program., Ser. A*, vol. 103, pp. 127–152, 2005.
- [10] M. ApS, *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28)*, 2015.
- [11] J. Löfberg, "YALMIP : A toolbox for modeling and optimization in MATLAB," in *Proc. the CACSD Conference*, 2004.