Admission Control Algorithms for QoS-Constrained Multicell MISO Downlink Systems

K. B. Shashika Manosha¹, *Student Member, IEEE*, Satya Krishna Joshi², *Student Member, IEEE*,
Marian Codreanu, *Member, IEEE*, Nandana Rajatheva, *Senior Member, IEEE*,
and Matti Latva-aho, *Senior Member, IEEE*

Abstract—The problem of admission control in a multicell downlink multiple-input single-output system is considered. The objective is to maximize the number of admitted users subject to the signal-to-interference-plus-noise ratio constraint for each admitted user and a transmit power constraint at each base station. We cast the admission control problem as an ℓ_0 minimization problem. This problem is known to be combinatorial NP-hard. Hence, we have to rely on suboptimal algorithms to solve it. We first approximate the ℓ_0 minimization problem via a non-combinatorial one. Then, we propose centralized and distributed algorithms to solve the non-combinatorial problem. To develop the centralized algorithm, we use the sequential convex programming method. The distributed algorithm is derived by using the alternating direction method of multipliers in conjunction with sequential convex programming. We show numerically that the proposed admission control algorithms achieve a near-to-optimal performance. Next, we extend the admission control problem to provide fairness, where a long term fairness among the users is guaranteed. We focus on proportional and max-min fairness and propose dynamic control algorithms via Lyapunov optimization. It is shown numerically that the proposed fair admission control algorithms guarantee fairness among the users.

Index Terms—Admission control, fairness, stochastic optimization, Lyapunov drift, alternating direction method of multipliers (ADMM), distributed algorithm, ℓ_0 minimization.

I. INTRODUCTION

THE growth in the number of wireless users and their stringent Quality of Service (QoS) requirements make a wireless network congested. Hence, in order to utilize the wireless resources efficiently while ensuring a guaranteed QoS for each user, a network has to find a maximum number of users that can be supported simultaneously. This brings up the problem of user admission control in a wireless network [1, Ch. 3]. Unfortunately, this is a difficult combinatorial optimization problem, and it is known to be NP-hard [2], [3].

This work was supported by the Finnish Funding Agency for Technology and Innovation (Tekes), Academy of Finland, Broadcom, Nokia, Huawei, Anite, and Elektrobit. The associate editor coordinating the review of this paper and approving it for publication was T. Melodia. (*Corresponding author: K. B. Shashika Manosha.*)

The authors are with the Centre for Wireless Communications, University of Oulu, 90014 Oulu, Finland (e-mail: manosha@ee.oulu.fi; sjoshi@ee.oulu.fi; codreanu@ee.oulu.fi; rrajathe@ee.oulu.fi; matti.latva-aho@ee.oulu.fi).

The exhaustive search method is one approach to find the global optimal solution of the admission control problem. However, the computational complexity of the exhaustive search method increases exponentially with the number of users. Systematic approaches like branch and bound has been proposed to optimally solve this problem [3]. Although the solution in [3] is optimal, it is not suitable for practical scenarios due to the complexity of branch and bound method [4]. Hence, fast suboptimal algorithms are desirable in practice [2], [5], [6].

In the case of Single-Input Single-Output (SISO) systems, suboptimal methods for the problem of admission control have been studied in [7]–[13]. Specifically, in [7] this problem has been cast as an ℓ_0 minimization problem and an ℓ_1 -norm relaxation technique has been used to find an approximate solution. In [8] different priority user groups are considered, and by exploiting a relationship between the Signal-to-Interference-plus-Noise Ratio (SINR) and the transmit power of users an iterative algorithm is proposed. Both works in [7] and [8] have proposed centralized algorithms. By using the dual decomposition technique [14] a distributed algorithm is proposed in [9]. Since the dual decomposition method relies on subgradient method [15], the convergence speed of [9] can be slow and highly sensitive to the choice of a step size. By using the Karush-Kuhn-Tucker (KKT) optimality conditions, a distributed algorithm is proposed in [10]. Note that a solution obtained by solving the KKT conditions is only a stationary point [16, Ch. 5], and in general its optimality is not guaranteed for a nonconvex (admission control) problem. Using Foschini-Miljanic power control technique [17], distributed methods for admission control have been studied in [11]–[13]. But, these solutions cannot be directly applied to multi antenna networks.

For Multiple-Input Single-Output (MISO) systems, the centralized implementation of the admission control problem has been studied in [2]. Specifically, the authors in [2] have formulated this problem as an integer non-linear optimization problem [18, Ch. 1] for a single cell. Then, two approximate solutions are proposed via semidefinite-relaxation method [19], [20] and second order cone programming [16, Ch. 4.4]. The work in [2] can be directly applied to a multicell scenario, however, its distributed implementation is not straightforward, and there is no reported work on that. In the context of a multicell system,

this problem has been cast as an ℓ_0 minimization problem in [5]. Then, the ℓ_1 -norm relaxation technique [21], [22] in conjunction with semidefinite-relaxation method is used to provide both centralized and distributed algorithms. In particular, the distributed algorithm is based on block coordinate descent method [23, Ch. 2], where the subproblems associated with the Base Stations (BS) are solved in a cyclic order. Both algorithms in [2] and [5] are derived by using the deflation based approach, which relies on dropping users at each iteration of the algorithm. In a different context (i.e., for a two-tier macrocell and femtocell network) the problem of admission control has been addressed in [6]. By imposing cross-tier interference constraints to protect the macro users, the authors of [6] proposed several suboptimal centralized and distributed algorithms.

In cellular networks, the users who are closer to the BS have a better average SINR compared to the cell edge users. Also some users can be located in a rich scattering environment and others with no scatterers around them [24, Ch. 6]. These differences in the fading statistics of the users make it challenging to provide fairness among the users while they are being dynamically admitted to the system [25, Ch. 12.2]. In [2] and [7], it is mentioned that the fairness can be provided by scheduling the non-admitted users in orthogonal resources. However, the authors in [2] and [7] have not provided any method to obtain fairness while admitting users. Furthermore, since the resources are scarce, orthogonal allocation may not be spectrally efficient even though it can provide fairness [24, Ch. 6], [26].

To the best of our knowledge all the existing admission control algorithms consider a static case (i.e., the admission control problem for a given instance). Thus, when these algorithms are applied to a *dynamic network* over a period of time, they may not provide fairness while admitting users. In this paper, we consider the problem of *fair admission control* in a dynamic network. Specifically, we focus on providing *long term* fairness [27] based on proportional and max-min fairness criteria. To do this, we combine the Lyapunov drift-plus-penalty framework [28] with the recent results in sparse optimization [22], and cast the fair admission control problem as a stochastic optimization problem.

For the static case, inspired by [7] and [10], we cast the classic admission control problem as an ℓ_0 minimization problem. The idea is to minimize the number of non-admitted users, instead of maximizing the number of admitted users. This ℓ_0 minimization problem is known to be NP-hard [21], [29]. Compared to the ℓ_0 based problem formulations in [7] and [10], which are restricted to the SISO case, we have generalized the formulation for a MISO system. This extension is highly nontrivial, because the beamformers are optimization variables in our problem; hence, the methods proposed in [7] and [10] cannot be applied to solve the admission control problem considered in this paper.

The main contributions of this paper are two-fold. The first is to propose both centralized and distributed algorithms to solve the problem of admission control for a *static* network. The second contribution is to provide *dynamic control*

algorithms that guarantee proportional and max-min fairness for a dynamic network.

In the case of static networks, the centralized algorithm is derived by using *sequential convex programming* [30]. Unlike the deflation based algorithm in [2] that relies on dropping a user at each iteration until the maximum number of admitted users is found, the proposed algorithm decides the admissibility of users only when it converges. This behavior of the proposed algorithm, clearly, has an advantage when the number of users in the network is large. Numerically, we have shown that the proposed centralized algorithm achieves near-to-optimal performance. Furthermore, results show that for large networks the proposed algorithm has a lower average execution time compared to the algorithm in [2], while having a similar performance in terms of average number of admitted users. The proposed distributed algorithm is derived by using the consensus-based Alternating Direction Method of Multipliers (ADMM) [31] in conjunction with sequential convex programming. In contrast to the existing distributed algorithm in [5] that solves the subproblems in a cyclic order, our proposed algorithm solves the subproblems independently in parallel at each BS. Thus, compared to the distributed algorithm in [5] our one is desirable for a network with a large number of BSs.

In the case of dynamic networks, we have adopted the Lyapunov drift-plus-penalty framework in [28] to derive dynamic control algorithms that guarantee proportional and max-min fairness. We derive these algorithms by introducing a notion of virtual queue for each user. A virtual queue grows if a user is unserved in the current time slot and reduces otherwise. The proposed dynamic control algorithms use these virtual queues as the scheduling priorities of users. Numerical results show that proposed algorithms guarantee proportional and max-min fairness among the users.

Notations: All boldface lower case and upper case letters represent vectors and matrices, respectively, and calligraphy letters represent sets. The notation $\mathbb{C}^n$ denotes the set of complex n -vectors, $|x|$ denotes the absolute value of scalar x , $\|\mathbf{x}\|_1$ and $\|\mathbf{x}\|_2$ denote the ℓ_1 -norm and ℓ_2 -norm of vector $\mathbf{x}$, respectively. Cardinality of set $\mathcal{A}$ is denoted by $|\mathcal{A}|$. The notation $\mathcal{CN}(\mathbf{m}, \mathbf{C})$ denotes the complex circular symmetric Gaussian vector distribution with mean $\mathbf{m}$ and covariance matrix $\mathbf{C}$. The superscripts $(\cdot)^H$ and $(\cdot)^T$ are used to denote a Hermitian transpose and a transpose of a matrix, and $(\cdot)^*$ is used to denote a solution of an optimization problem. We use parentheses to construct a column vector from a comma separated list, e.g., $(\mathbf{a}, \mathbf{b}, \mathbf{c}) = [\mathbf{a}^T, \mathbf{b}^T, \mathbf{c}^T]^T$.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider the downlink of a multicell network with K BSs. The set of all BSs is denoted by $\mathcal{K}$ and we label them with the integer values $k = 1, \dots, K$. We assume that each BS is equipped with T transmit antennas and each user with a single receive antenna, that is a MISO system is considered. We denote the set of all users associated with k th BS by $\mathcal{U}_k$, and we label them with the integer values $i = 1, \dots, I_k$. Furthermore, we assume that each BS transmits data only to

its associated users. The network is assumed to be operating in slotted time with normalized slots $t \in \{0, 1, \dots\}$.

The antenna signal vector transmitted by k th BS during time slot t is given by

$$\mathbf{x}_k(t) = \sum_{i \in \mathcal{U}_k} \mathbf{m}_{ki}(t) d_{ki}(t), \quad (1)$$

where $d_{ki}(t) \in \mathbb{C}$ and $\mathbf{m}_{ki}(t) \in \mathbb{C}^T$ represent the information symbol and the transmit beamformer associated with i th user of BS k during time slot t , respectively. We assume that $d_{ki}(t)$ is normalized such that $\mathbb{E}[|d_{ki}(t)|^2] = 1$. Furthermore, we assume that the data streams are independent, i.e., $\mathbb{E}\{d_{ki}(t)d_{kj}(t)^*\} = 0$ for all $i \neq j$, where $i, j \in \mathcal{U}_k$, $k \in \mathcal{K}$.

The signal received by i th user of BS k in time slot t can be written as

$$\begin{aligned} y_{ki}(t) &= (\mathbf{h}_{ki}^k(t))^H \mathbf{m}_{ki}(t) d_{ki}(t) \\ &+ \sum_{j \in \mathcal{U}_k, j \neq i} (\mathbf{h}_{ki}^k(t))^H \mathbf{m}_{kj}(t) d_{kj}(t) \\ &+ \sum_{l \in \mathcal{K} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} (\mathbf{h}_{ki}^l(t))^H \mathbf{m}_{lj}(t) d_{lj}(t) + n_{ki}(t), \end{aligned} \quad (2)$$

where $\mathbf{h}_{ki}^l(t) \in \mathbb{C}^T$ is the channel vector from BS l to i th user of BS k and n_{ki} is circularly symmetric complex Gaussian noise with variance σ_{ki}^2 . The received SINR of i th user of BS k is expressed in (3), as shown in the bottom of this page, where we use the notation $\mathbf{m}(t)$ to denote a vector obtained by stacking $\mathbf{m}_{ki}(t)$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, on top of each other, i.e., $\mathbf{m}(t) = [\mathbf{m}_{11}(t)^T, \mathbf{m}_{12}(t)^T, \dots, \mathbf{m}_{KI_K}(t)^T]^T$.

We assume that the power allocation is subject to a maximum transmit power constraint at each BS, i.e.,

$$\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K}, \quad (4)$$

where $P_k^{\max}$ is the maximum transmit power of k th BS. Furthermore, we assume that the QoS of i th user of BS k is assured if its SINR is greater than a threshold γ_{ki} , i.e.,

$$\Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki}. \quad (5)$$

Let $\tilde{\mathcal{U}}_k(\mathbf{m}(t))$ denote a generic set of admissible users at k th BS. Specifically, $\tilde{\mathcal{U}}_k(\mathbf{m}(t))$ denotes a set of users who satisfy their SINR thresholds under the power constraint, i.e.,

$$\begin{aligned} \tilde{\mathcal{U}}_k(\mathbf{m}(t)) &= \{ki \mid \Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki}, \\ &\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, i \in \mathcal{U}_k\}, \quad k \in \mathcal{K}. \end{aligned} \quad (6)$$

Our goal is to maximize the number of admitted users to the system, i.e., to maximize the sum of the cardinalities of $\tilde{\mathcal{U}}_k(\mathbf{m}(t))$ for all $k \in \mathcal{K}$. We now formulate this design problem as a mathematical optimization problem. To do this, let us introduce the nonnegative auxiliary variables $s_{ki}(t)$ for

all $k \in \mathcal{K}$, $i \in \mathcal{U}_k$, and consider a set of relaxed SINR constraints as follows:

$$\Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, k \in \mathcal{K}. \quad (7)$$

In (7), when $s_{ki}(t) = 0$ we recover constraint (5), i.e., the SINR constraint of i th user of k th BS is satisfied. Furthermore, by making $s_{ki}(t)$ large enough the set of relaxed SINR constraints in (7) can be always made feasible.

Note that maximizing the number of admitted users that satisfy the SINR constraints (5) is equivalent to minimizing the number of users that require a strictly positive value of auxiliary variable $s_{ki}(t)$ that satisfies constraint (7). Hence, by using expressions (4) and (7) the problem of admission control can be expressed as

$$\begin{aligned} &\text{minimize} \quad \sum_{k \in \mathcal{K}} \|\mathbf{s}_k(t)\|_0 \\ &\text{subject to} \quad \Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, k \in \mathcal{K} \quad (8a) \\ &\quad \sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K} \quad (8b) \\ &\quad s_{ki}(t) \geq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K}, \quad (8c) \end{aligned}$$

where $\mathbf{s}_k(t) = [s_{k1}(t), \dots, s_{kI_k}(t)]^T$ for all $k \in \mathcal{K}$ and variables are $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$. In problem (8), constraint (8c) is redundant.¹ However, this constraint is useful while deriving an algorithm to solve problem (8), as will be cleared in the next section.

III. ALGORITHM DERIVATION

Problem (8) is a difficult combinatorial optimization problem [21] due to the ℓ_0 objective function. In fact, this problem is known to be NP-hard [21], [29], and it requires exponential complexity to find a global solution. Therefore, we have to rely on suboptimal algorithms to solve it. In this section, we provide suboptimal centralized and distributed algorithms to solve problem (8). The proposed centralized algorithm is based on ℓ_0 approximation method [32] and sequential convex programming [33]. The distributed algorithm is derived by using the consensus-based ADMM [31] in conjunction with sequential convex programming. In this section, we solve problem (8) for a single time slot, hence we drop the time index t for notational simplicity.

A. Centralized Algorithm

We start by approximating the objective function of problem (8) with a concave function $\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(s_{ki} + \epsilon)$, where ϵ is a small positive constant² and $s_{ki} \geq 0$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$ [32]. Let us define a new variable β_{ki} to denote the interference-plus-noise experienced by

¹We can show by contradiction that negative values of variables $s_{ki}(t)$, $i \in \mathcal{U}_k$, $k \in \mathcal{K}$ are not optimal solutions of problem (8).

²We refer the reader to [32] for a better understanding (with illustrations) of the advantages of approximating an ℓ_0 function with a log function, instead of using ℓ_1 -norm to approximate it.

$$\Gamma_{ki}(\mathbf{m}(t)) = \frac{|\mathbf{h}_{ki}^k(t))^H \mathbf{m}_{ki}(t)|^2}{\sum_{j \in \mathcal{U}_k, j \neq i} |\mathbf{h}_{ki}^k(t))^H \mathbf{m}_{kj}(t)|^2 + \sum_{l \in \mathcal{K} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} |\mathbf{h}_{ki}^l(t))^H \mathbf{m}_{lj}(t)|^2 + \sigma_{ki}^2} \quad (3)$$

i th user of k th BS, i.e., $\beta_{ki} = \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{X} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2 + \sigma_{ki}^2$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$. Then, a solution of problem (8) can be approximated by solving following optimization problem:

$$\begin{aligned} & \text{minimize } \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(s_{ki} + \epsilon) \\ & \text{subject to } \gamma_{ki} - s_{ki} - \frac{|(\mathbf{h}_{ki}^k)^H \mathbf{m}_{ki}|^2}{\beta_{ki}} \leq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \end{aligned} \quad (9a)$$

$$\begin{aligned} & \sum_{\substack{j \in \mathcal{U}_k, \\ j \neq i}} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{X} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2 \\ & + \sigma_{ki}^2 \leq \beta_{ki}, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \end{aligned} \quad (9b)$$

$$\text{constraints (8b), (8c),} \quad (9c)$$

with variables $\{s_{ki}, \mathbf{m}_{ki}, \beta_{ki}\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$. In constraint (9a), we have replaced the interference-plus-noise term of $\Gamma_{ki}(\mathbf{m})$ with β_{ki} . Note that the objective function of problem (9) is increasing in s_{ki} , hence, it can be shown (e.g., by contradiction) that constraint (9b) holds with equality at the optimal point.

Problem (9) is a non-combinatorial optimization problem. However, it is still *nonconvex* because the objective function is concave and constraint function (9a) is not convex. In the sequel, we apply sequential convex programming [30] to solve problem (9). Here, we approximate the objective function and constraint function (9a) with their best convex approximations. Then, we iteratively solve the approximated convex problem to find a solution for problem (9).

Let the objective function of problem (9) is denoted by $f(\mathbf{s}) = \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(s_{ki} + \epsilon)$, where $\mathbf{s} = [s_{11}, \dots, s_{K I_K}]$. Note that $f(\mathbf{s})$ is a concave function [16, Ch. 3]. Hence, the best convex approximation of function $f(\mathbf{s})$ can be obtained by replacing it with its first order approximation [30]. The first order approximation of function $f(\mathbf{s})$ near an arbitrary positive point $\hat{\mathbf{s}} = [\hat{s}_{11}, \dots, \hat{s}_{K I_K}]$ can be expressed as

$$\hat{f}(\mathbf{s}) = f(\hat{\mathbf{s}}) + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} (s_{ki} - \hat{s}_{ki}) / (\hat{s}_{ki} + \epsilon). \quad (10)$$

We now focus on constraint (9a). Let $g_{ki}(\mathbf{m}_{ki}, \beta_{ki})$ be a function defined as $g_{ki}(\mathbf{m}_{ki}, \beta_{ki}) = |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{ki}|^2 / \beta_{ki}$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$. It is easy to see that constraint function (9a) is a difference of the affine function $\gamma_{ki} - s_{ki}$ and the convex function $g_{ki}(\mathbf{m}_{ki}, \beta_{ki})$. The best convex approximation of constraint function (9a) near an arbitrary point $(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})$ can be obtained by replacing $g_{ki}(\mathbf{m}_{ki}, \beta_{ki})$ with its first order approximation [30], and it can be expressed as

$$\begin{aligned} \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) &= g_{ki}(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}) + \nabla g_{ki}(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})^T \\ & \quad \times ((\mathbf{m}_{ki}, \beta_{ki}) - (\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})), \end{aligned} \quad (11)$$

where $\nabla g_{ki}(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})$ is the gradient of $g_{ki}(\mathbf{m}_{ki}, \beta_{ki})$ evaluated at point $(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})$, defined as

$$\nabla g_{ki}(\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}) = \left(2\mathbf{h}_{ki}^H \mathbf{h}_{ki}^H \hat{\mathbf{m}}_{ki} / \hat{\beta}_{ki}, -\hat{\mathbf{m}}_{ki}^H \mathbf{h}_{ki}^H \hat{\mathbf{m}}_{ki} / \hat{\beta}_{ki}^2 \right). \quad (12)$$

Now by using expressions (10) and (11), we approximate problem (9) near an arbitrary positive point $(\hat{s}_{ki}, \hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$, as the following convex optimization problem:

$$\begin{aligned} & \text{minimize } \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} w_{ki} s_{ki} \\ & \text{subject to } \gamma_{ki} - s_{ki} - \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) \leq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \\ & \quad \sum_{\substack{j \in \mathcal{U}_k, \\ j \neq i}} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{X} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2 \\ & \quad + \sigma_{ki}^2 \leq \beta_{ki}, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \\ & \text{constraints (8b), (8c),} \end{aligned} \quad (13)$$

where $w_{ki} = 1/(\hat{s}_{ki} + \epsilon)$, variables are $s_{ki}, \mathbf{m}_{ki}, \beta_{ki}$ for all $k \in \mathcal{K}, i \in \mathcal{U}_k$, and w_{ki} acts as a weight associated with user i of k th BS for all $i \in \mathcal{U}_k, k \in \mathcal{K}$. Note that in the objective function of problem (13) we dropped the constant term $f(\hat{\mathbf{s}}) - \hat{s}_{ki}/(\hat{s}_{ki} + \epsilon)$, since it does not affect the solution of problem (13).

Finally, we summarize the proposed centralized algorithm for problem (9) in *Algorithm 1*.

Algorithm 1 Centralized Algorithm for Solving Problem (9)

- 1: **initialization:** $\{s_{ki}^0, \mathbf{m}_{ki}^0, \beta_{ki}^0\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$, set $w_{ki} = 1$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$, iteration index $p = 0$.
 - repeat**
 - 2: Set $\hat{\mathbf{m}}_{ki} = \mathbf{m}_{ki}^p, \hat{\beta}_{ki} = \beta_{ki}^p$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$. Form $\{\hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki})\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$ using expression (11).
 - 3: Solve problem (13). Denote the solution $\{s_{ki}^*, \mathbf{m}_{ki}^*, \beta_{ki}^*\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$.
 - 4: Update $\{s_{ki}^{p+1} = s_{ki}^*, \beta_{ki}^{p+1} = \beta_{ki}^*, \mathbf{m}_{ki}^{p+1} = \mathbf{m}_{ki}^*, w_{ki} = 1/(s_{ki}^* + \epsilon)\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$. Set $p = p + 1$.
 - until** stopping criterion is satisfied
-

Note that problem (13) approximates a solution for problem (9) near an arbitrary point $\{\hat{s}_{ki}, \hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$. Hence, to obtain the best local solution for problem (9), *Algorithm 1* solves problem (13) repeatedly for different values of $\{\hat{s}_{ki}, \hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$ until the stopping criteria is satisfied (see steps 2-4). Algorithm can be stopped when the difference between the achieved objective values of problem (9) in two successive iterations is less than a given threshold.

B. Distributed Algorithm

In this subsection we extend *Algorithm 1* to derive a distributed algorithm for admission control problem (8). The distributed algorithm is derived by solving step 3 of *Algorithm 1* (i.e., problem (13)) distributively over the BSs using ADMM [31].

We start by introducing an auxiliary variable z_{ki}^l to denote the interference generated by l th BS to i th user of BS k , i.e., $z_{ki}^l = \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$, and $l \in \mathcal{X} \setminus \{k\}$. Then problem (13) can be equivalently

Fig. 1. Illustration of BS coupling, and introducing local copies to decouple a problem. BS2 and BS3 are coupled with BS1 by coupling variables z_{1i}^2 and z_{1i}^3 , respectively. To distribute the problem, local copy $x_{1i,1}^2$ of z_{1i}^2 at BS1 and local copy $x_{1i,2}^2$ of z_{1i}^2 at BS 2 are introduced. Similarly, local copy $x_{1i,1}^3$ of z_{1i}^3 at BS1 and local copy $x_{1i,3}^3$ of z_{1i}^3 at BS3 are introduced.

written as

$$\begin{aligned}
& \text{minimize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} w_{ki} s_{ki} \\
& \text{subject to} \quad \gamma_{ki} - s_{ki} - \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) \leq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \\
& \quad \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{K} \setminus \{k\}} z_{ki}^l + \sigma_{ki}^2 \leq \beta_{ki}, \\
& \quad i \in \mathcal{U}_k, k \in \mathcal{K} \\
& \quad \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2 \leq z_{ki}^l, \\
& \quad i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\} \\
& \quad \text{constraints (8b), (8c),}
\end{aligned} \tag{14}$$

with variables $s_{ki}, \mathbf{m}_{ki}, \beta_{ki}$ for all $k \in \mathcal{K}, i \in \mathcal{U}_k$ and $\{z_{ki}^l\}_{i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}}$. Note that problem (13) and (14) are equivalent as it can be easily shown (e.g., by contradiction) that the third inequality holds with equality at the optimal point.

In problem (14), variable z_{ki}^l represents the power of inter-cell interference caused by l th BS to i th user of BS k , and thus variable z_{ki}^l couples the two BSs l and k . We use consensus technique [31, Ch. 7] to distribute problem (14) over the BSs. The method consists of introducing local copies of coupling variables z_{ki}^l for all $i \in \mathcal{U}_k, k \in \mathcal{K}$, and $l \in \mathcal{K} \setminus \{k\}$, at each BS (see Fig. 1).

Let us define $x_{ki,k}^l$ as the local copy of z_{ki}^l saved at k th BS, and $x_{ki,l}^l$ as the local copy of z_{ki}^l saved at l th BS. Thus for each z_{ki}^l we make two local copies (see Fig. 1). Then, problem (14) can be written equivalently in *global consensus* form as

$$\begin{aligned}
& \text{minimize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} w_{ki} s_{ki} \\
& \text{subject to} \quad \gamma_{ki} - s_{ki} - \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) \leq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K}
\end{aligned} \tag{15a}$$

$$\begin{aligned}
& \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{K} \setminus \{k\}} x_{ki,k}^l + \sigma_{ki}^2 \leq \beta_{ki}, \quad i \in \mathcal{U}_k, k \in \mathcal{K}
\end{aligned} \tag{15b}$$

$$\begin{aligned}
& \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l)^H \mathbf{m}_{lj}|^2 \leq x_{ki,l}^l, \\
& i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}
\end{aligned} \tag{15c}$$

$$x_{ki,k}^l = z_{ki}^l, \quad i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\} \tag{15d}$$

$$x_{ki,l}^l = z_{ki}^l, \quad i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\} \tag{15e}$$

$$\text{constraints (8b), (8c),} \tag{15f}$$

with optimization variables $\{s_{ki}, \mathbf{m}_{ki}, \beta_{ki}\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ and $\{x_{ki,k}^l, z_{ki}^l\}_{i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}}$. Note that in constraint (15b) we have replaced z_{ki}^l with the local copy $x_{ki,k}^l$. In constraints (15c), we have replaced z_{ki}^l with the local copy $x_{ki,l}^l$. Constraints (15d) and (15e) are called *consistency constraints* and they enforce the local copies $x_{ki,k}^l$ and $x_{ki,l}^l$ to be equal to the corresponding global variable z_{ki}^l .

We now express problem (15) more compactly. To do this, we first express the consistency constraints of problem (15) more compactly by using vector notation. The set of local variables associated with k th BS includes two components: 1) copies of the interference experienced by all the users associated with k th BS from all the other BSs, i.e., $\{x_{ki,k}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_k}$, and 2) copies of the interference generated by k th BS to all the other users, i.e., $\{x_{li,k}^k\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_l}$. Thus, we can compactly write the local copies of intercell interference terms associated with k th BS as

$$\mathbf{x}_k = \left(\{x_{ki,k}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_k}, \{x_{li,k}^k\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_l} \right). \tag{16}$$

Similarly, we can compactly write the global variables associated with k th BS as

$$\mathbf{z}_k = \left(\{z_{ki}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_k}, \{z_{li}^k\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_l} \right). \tag{17}$$

It is worth noting that some components of $\mathbf{z}_k$ are also in the components of $\mathbf{z}_l$ for all $l \in \mathcal{K} \setminus \{k\}$. For example, components $\{z_{ki}^l\}_{i \in \mathcal{U}_k}$ and $\{z_{lj}^k\}_{j \in \mathcal{U}_l}$ are common to both $\mathbf{z}_k$ and $\mathbf{z}_l$. With the compact notations introduced above we equivalently write the equality constraints (15d) and (15e) as

$$\mathbf{x}_k = \mathbf{z}_k, \quad k \in \mathcal{K}. \tag{18}$$

Now, for the sake of brevity, let us define the following set

$$O_k = \left\{ \mathbf{s}_k, \mathbf{m}_k, \beta_k, \mathbf{x}_k \left| \begin{aligned} & \gamma_{ki} - s_{ki} - \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) \leq 0, \\ & i \in \mathcal{U}_k, k \in \mathcal{K} \\ & \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{K} \setminus \{k\}} x_{ki,k}^l + \sigma_{ki}^2 \leq \beta_{ki}, \quad i \in \mathcal{U}_k \\ & \sum_{i \in \mathcal{U}_k} |(\mathbf{h}_{li}^k)^H \mathbf{m}_{ki}|^2 \leq x_{li,k}^k, \\ & j \in \mathcal{U}_l, l \in \mathcal{K} \setminus \{k\} \\ & \sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}\|_2^2 \leq P_k^{\max}, \\ & s_{ki} \geq 0, \quad i \in \mathcal{U}_k, k \in \mathcal{K} \end{aligned} \right. \right\}, \tag{19}$$

where $\beta_k = [\beta_{k1}, \dots, \beta_{kI_k}]$, and the following function

$$f_k(\mathbf{s}_k, \mathbf{m}_k, \beta_k, \mathbf{x}_k) = \begin{cases} \sum_{i \in \mathcal{U}_k} w_{ki} s_{ki} & (\mathbf{s}_k, \mathbf{m}_k, \beta_k, \mathbf{x}_k) \in O_k \\ \infty & \text{otherwise.} \end{cases} \tag{20}$$

Then by using notations in (16), (17), (19), and (20), consensus problem (15) can be compactly written as

$$\begin{aligned} & \text{minimize } \sum_{k \in \mathcal{K}} f_k(\mathbf{s}_k, \mathbf{m}_k, \boldsymbol{\beta}_k, \mathbf{x}_k) \\ & \text{subject to } \mathbf{x}_k = \mathbf{z}_k, \quad k \in \mathcal{K}, \end{aligned} \quad (21)$$

with variables $\mathbf{s}_k, \mathbf{m}_k, \boldsymbol{\beta}_k, \mathbf{x}_k$, and $\mathbf{z}_k$ for all $k \in \mathcal{K}$.

C. Distributed Algorithm via ADMM

In this subsection, we derive a distributed algorithm for problem (21). The proposed algorithm is based on ADMM [31]. Let $y_{ki,k}^l$ and $y_{li,l}^l$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, and $l \in \mathcal{K} \setminus \{k\}$, be the dual variables associated with constraints (15d) and (15e) of problem (15). Specifically, the dual variables associated with k th BS in constraints (15d) and (15e) can be compactly written as

$$\mathbf{y}_k = \left(\{y_{ki,k}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_k}, \{y_{li,k}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_l} \right). \quad (22)$$

Now we can write the *augmented Lagrangian* for problem (21) as

$$\begin{aligned} & L_\rho(\{\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k, \mathbf{z}_k\}_{k \in \mathcal{K}}) \\ & = \sum_{k \in \mathcal{K}} \left(f_k(\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k) + \mathbf{y}_k^T (\mathbf{x}_k - \mathbf{z}_k) + \frac{\rho}{2} \|\mathbf{x}_k - \mathbf{z}_k\|_2^2 \right), \end{aligned} \quad (23)$$

where $\{\mathbf{y}_k\}_{k \in \mathcal{K}}$ are the dual variables associated with the equality constraint of (21), and $\rho > 0$ is the *penalty parameter* that adds the quadratic penalty term to the standard Lagrangian L_0 when the equality constraints of problem (21) are being violated.

Each iteration of the ADMM algorithm consists of the following three steps [31, Ch. 3]:

$$\begin{aligned} & \mathbf{s}_k^{q+1}, \boldsymbol{\beta}_k^{q+1}, \mathbf{x}_k^{q+1}, \mathbf{m}_k^{q+1} \\ & = \underset{\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k}{\text{argmin}} L_\rho(\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k, \mathbf{z}_k^q, \mathbf{y}_k^q), \quad k \in \mathcal{K} \end{aligned} \quad (24a)$$

$$\begin{aligned} & \{\mathbf{z}_k^{q+1}\}_{k \in \mathcal{K}} = \underset{\mathbf{z}_k \in \mathcal{K}}{\text{argmin}} L_\rho(\{\mathbf{s}_k^{q+1}, \boldsymbol{\beta}_k^{q+1}, \mathbf{x}_k^{q+1}, \mathbf{m}_k^{q+1}\}_{k \in \mathcal{K}}, \\ & \quad \{\mathbf{z}_k\}_{k \in \mathcal{K}}, \{\mathbf{y}_k^q\}_{k \in \mathcal{K}}) \end{aligned} \quad (24b)$$

$$\mathbf{y}_k^{q+1} = \mathbf{y}_k^q + \rho(\mathbf{x}_k^{q+1} - \mathbf{z}_k^{q+1}), \quad k \in \mathcal{K}, \quad (24c)$$

where the superscript q is the ADMM algorithm iteration counter. Note that steps (24a) and (24c) are completely decentralized, and hence, they can be carried out independently in parallel at each BS. Recall that the components of $\mathbf{z}_k$ couple with two local variables that are associated with the interfering BSs (see the consistency constraints (15d) and (15e)). Thus, step (24b) requires gathering the updated local variables and the dual variables from all K BSs. In the sequel, we first explain in detail the way to solve ADMM steps (24a) and (24b). Then, we present the proposed ADMM based distributed algorithm.

We start by providing a method to compute step (24a). The local variables update $(\mathbf{s}_k^{q+1}, \boldsymbol{\beta}_k^{q+1}, \mathbf{x}_k^{q+1}, \mathbf{m}_k^{q+1})$ in (24a) is a solution of the following optimization problem:

$$\begin{aligned} & \text{minimize } f_k(\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k) + \mathbf{y}_k^{qT} (\mathbf{x}_k - \mathbf{z}_k^q) \\ & \quad + \frac{\rho}{2} \|\mathbf{x}_k - \mathbf{z}_k^q\|_2^2, \end{aligned} \quad (25)$$

with variables $\mathbf{s}_k, \boldsymbol{\beta}_k, \mathbf{x}_k, \mathbf{m}_k$. Here we use $\mathbf{y}_k^{qT}$ instead of $(\mathbf{y}_k^q)^T$ to lighten the notation. Let $\mathbf{v}_n$ be a scaled dual variable, i.e., $\mathbf{v}_n = (1/\rho)\mathbf{y}_n$. By using the notations (19) and (20), problem (25) can be equivalently written as

$$\begin{aligned} & \text{minimize } \sum_{i \in \mathcal{U}_k} w_{ki} s_{ki} + \frac{\rho}{2} \|\mathbf{x}_k - \mathbf{z}_k^q + \mathbf{v}_k^q\|_2^2 \\ & \text{subject to } \gamma_{ki} - s_{ki} - \hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki}) \leq 0, \quad i \in \mathcal{U}_k \\ & \quad \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k)^H \mathbf{m}_{kj}|^2 + \sum_{l \in \mathcal{K} \setminus \{k\}} x_{ki,k}^l \\ & \quad + \sigma_{ki}^2 \leq \beta_{ki}, \quad i \in \mathcal{U}_k \\ & \quad \sum_{i \in \mathcal{U}_k} |(\mathbf{h}_{ij}^k)^H \mathbf{m}_{ki}|^2 \leq x_{ij,k}^k, \quad j \in \mathcal{U}_l, l \in \mathcal{K} \setminus \{k\} \\ & \quad \sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}\|_2^2 \leq P_k^{\max}, \\ & \quad s_{ki} \geq 0, \quad i \in \mathcal{U}_k, \end{aligned} \quad (26)$$

with variables $s_k, \boldsymbol{\beta}_k, \mathbf{x}_k$, and $\mathbf{m}_k$. Note that the second term of the objective function of problem (26) is obtained by: 1) combining the linear and quadratic terms of the objective function of problem (25) as $\mathbf{y}_k^{qT} (\mathbf{x}_k - \mathbf{z}_k^q) + \frac{\rho}{2} \|\mathbf{x}_k - \mathbf{z}_k^q\|_2^2 = \frac{\rho}{2} \|\mathbf{x}_k - \mathbf{z}_k^q + \mathbf{v}_k^q\|_2^2 - \frac{\rho}{2} \|\mathbf{v}_k^q\|_2^2$, and 2) dropping the constant term $-\frac{\rho}{2} \|\mathbf{v}_k^q\|_2^2$. This constant term is dropped since it does not affect the solution of the problem. Let us denote by $\mathbf{s}_k^*, \boldsymbol{\beta}_k^*, \mathbf{x}_k^*$, and $\mathbf{m}_k^*$ a solution of problem (26) and we denote $\mathbf{s}_k^{q+1} = \mathbf{s}_k^*$, $\boldsymbol{\beta}_k^{q+1} = \boldsymbol{\beta}_k^*$, $\mathbf{x}_k^{q+1} = \mathbf{x}_k^*$, and $\mathbf{m}_k^{q+1} = \mathbf{m}_k^*$ as the updates.

We now consider the second step of ADMM algorithm, i.e., (24b), and provide a solution for global variable update. The update $\{\mathbf{z}_k^{q+1}\}_{k \in \mathcal{K}}$ is a solution of the following optimization problem:

$$\text{minimize } \sum_{k \in \mathcal{K}} \mathbf{y}_k^{qT} (\mathbf{x}_k^{q+1} - \mathbf{z}_k) + \frac{\rho}{2} \|\mathbf{x}_k^{q+1} - \mathbf{z}_k\|_2^2, \quad (27)$$

with variables $\{\mathbf{z}_k\}_{k \in \mathcal{K}}$. By observing notations in (16), (17), and (22), and also noting the equality constraint in (18) is equivalent to constraints (15d) and (15e), problem (27) can be expressed as

$$\begin{aligned} & \text{minimize } \sum_{k \in \mathcal{K}} \left(\sum_{l \in \mathcal{K} \setminus \{k\}} \sum_{i \in \mathcal{U}_k} (y_{ki,k}^{l[q]} (x_{ki,k}^{l[q+1]} - z_{ki}^l) \right. \\ & \quad + \frac{\rho}{2} (x_{ki,k}^{l[q+1]} - z_{ki}^l)^2) \\ & \quad + \sum_{l \in \mathcal{K} \setminus \{k\}} \sum_{i \in \mathcal{U}_l} (y_{li,k}^{k[q]} (x_{li,k}^{k[q+1]} - z_{li}^k) \\ & \quad \left. + \frac{\rho}{2} (x_{li,k}^{k[q+1]} - z_{li}^k)^2) \right), \end{aligned} \quad (28)$$

with variables $\{z_{ki}^l\}_{i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}}$. Note that problem (28) is an expanded version of problem (27), in which we have substituted elements for each vector using (16), (17), and (22). Here, the square brackets in superscript represent the ADMM algorithm iteration index.

Problem (28) decouples across z_{ki}^l , as the objective function is separable in $\{z_{ki}^l\}_{i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}}$. Moreover, the objective function of problem (28) is quadratic in z_{ki}^l . Hence, by setting the gradient of (28) with respect to z_{ki}^l equal to zero, we obtain the solution of problem (28) as

$$\begin{aligned} z_{ki}^{l*} = & \left(x_{ki,l}^{l[q+1]} + x_{ki,k}^{l[q+1]} + \frac{1}{\rho} (y_{ki,l}^{l[q]} + y_{ki,k}^{l[q]}) \right) / 2, \\ & i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad l \in \mathcal{K} \setminus \{k\}. \end{aligned} \quad (29)$$

Therefore, the update for $z_{ki}^{[lq+1]}$ is $z_{ki}^{[lq+1]} = z_{ki}^{l*}$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, and $l \in \mathcal{K} \setminus \{k\}$. Now we further simplify expression (29) as follows. We can write (24c) in the components of $\mathbf{y}_k$, $\mathbf{x}_k$, $\mathbf{z}_k$ for all $k \in \mathcal{K}$ as $y_{ki,k}^{[lq+1]} = y_{ki,k}^{[lq]} + \rho(x_{ki,k}^{[lq+1]} - z_{ki,k}^{[lq+1]})$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, and $l \in \mathcal{K} \setminus \{k\}$. Then, by adding the two expressions related to $y_{ki,l}^{[lq+1]}$ and $y_{ki,k}^{[lq+1]}$, we obtain the following expression

$$y_{ki,l}^{[lq+1]} + y_{ki,k}^{[lq+1]} = y_{ki,l}^{[lq]} + y_{ki,k}^{[lq]} + \rho(x_{ki,l}^{[lq+1]} + x_{ki,k}^{[lq+1]} - 2\rho z_{ki}^{[lq+1]}). \quad (30)$$

Recall that $z_{ki}^{[lq+1]} = z_{ki}^{l*}$, hence, by substituting the expression z_{ki}^{l*} into (30) we can show that the sum of the dual variables $(y_{ki,l}^{[lq+1]} + y_{ki,k}^{[lq+1]})$ is equal to zero. Thus, $z_{ki}^{[lq+1]}$ update simplifies to

$$z_{ki}^{[lq+1]} = (x_{ki,l}^{[lq+1]} + x_{ki,k}^{[lq+1]})/2 \quad i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}. \quad (31)$$

The global variables update in (31) requires gathering local copies from the two BSs. For example, to update global variable $z_{ki}^{[lq+1]}$ the local copy $x_{ki,l}^{[lq+1]}$ from l th BS and $x_{ki,k}^{[lq+1]}$ from k th BS should be collected. Hence, a coordination between the BSs is required to update (31).

Finally, we summarize the proposed distributed algorithm for problem (9) in *Algorithm 2*.

Algorithm 2 Distributed Algorithm for Solving Problem (9)

- 1: **initialization:** $\{s_{ki}^0, \mathbf{m}_{ki}^0, \beta_{ki}^0\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$, $w_{ki} = 1$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$, iteration index $q = 0$.
repeat
 - 2: Set $\hat{\mathbf{m}}_{ki} = \mathbf{m}_{ki}^q, \hat{\beta}_{ki} = \beta_{ki}^q$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}$. Form $\{\hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki})\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$ using expression (11).
 - 3: ADMM Algorithm:
repeat
 - a). Each BS $k \in \mathcal{K}$ updates the local variables $(s_k^{q+1}, \beta_k^{q+1}, \mathbf{x}_k^{q+1}, \mathbf{m}_k^{q+1})$ by solving (26).
 - b). BS k and l exchange local copies $x_{ki,k}^l$ and $x_{ki,l}^l$ for all $i \in \mathcal{U}_k, k \in \mathcal{K}, l \in \mathcal{K} \setminus \{k\}$.
 - c). Each BS k updates global variable z_k^{q+1} using (31).
 - d). Each BS k updates dual variable $\mathbf{y}_k^{q+1}$ by solving (24c). Set $q = q + 1$.**until** stopping criterion is satisfied
 - 4: Update $\{w_{ki} = 1/(s_{ki}^q + \epsilon)\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$.
until stopping criterion is satisfied
-

We refer to each execution of the steps 2-4 as an outer iteration and to each execution of the ADMM algorithm (i.e., steps 3a-3d) as an inner iteration. In each outer iteration, the algorithm approximates problem (9) as a convex function at step 2, and this can be done individually in parallel at each BS. In the inner iteration, steps 3a, 3c, and 3d are decentralized over the BSs. Step 3b requires a coordination between the BSs to exchange the updated values of the local variables. In ADMM algorithm, the standard stopping criteria is to check the primal and dual residuals [31, Ch. 3.3]. But, as ADMM can produce acceptable results of practical use within a few

iterations, a finite number of iterations is used to stop the ADMM algorithm in step 3 [31].

In many practical applications, we have to stop the distributed algorithm after a finite number of iterations before converging the algorithm. On the other hand, an intermediate solution provided by *Algorithm 2* at each inner iteration does not necessarily result in a feasible solution for problem (15). That is, constraints (15d) and (15e) of problem (15) may be violated by an intermediate solution provided by *Algorithm 2*. However, at the cost of solving one additional subproblem by each BS, we can find a feasible solution for problem (15). To do this, we make the local copies $x_{ki,k}^l$ and $x_{ki,l}^l$ to be equal, by setting them to be equal to the consensus value z_{ki}^l , i.e., $x_{ki,k}^l = z_{ki}^l$ and $x_{ki,l}^l = z_{ki}^l$ for all $i \in \mathcal{U}_k, l \in \mathcal{K} \setminus \{k\}, k \in \mathcal{K}$. Then, each BS can solve problem (26) independently for variables $\mathbf{s}_k$, β_k , and $\mathbf{m}_k$ in order to find a feasible solution for problem (15).

D. The Signalling Overhead of Algorithm 1 and Algorithm 2

Proposed *Algorithm 1* is a centralized algorithm. Hence, to solve problem (13) in step 3 of the algorithm, it is required to collect the Channel State Information (CSI) of all the users in the network at a Central Controller (CC). That is, k th BS needs to send $\{\mathbf{h}_{ki}^k\}_{i \in \mathcal{U}_k}$ and $\{\mathbf{h}_{lj}^k\}_{j \in \mathcal{U}_l, l \in \mathcal{K} \setminus \{k\}}$ to the CC. Let U denote the total number of users in the network, i.e., $U = \sum_{k \in \mathcal{K}} |\mathcal{U}_k|$. Then, the number of real scalars that has to be sent from K BSs to the CC is $2TKU$, where the factor 2 is considered to take into account the real and complex parts of a channel coefficient. Next, the CC needs to send the computed beamformers of the admitted users to their respective BSs. This requires $2T \sum_{k \in \mathcal{K}} |\tilde{\mathcal{U}}_k|$ scalars to be sent to K BSs. Thus, the total number of real scalars required to be exchanged between the BSs and the CC in *Algorithm 1* is $2T(KU + \sum_{k \in \mathcal{K}} |\tilde{\mathcal{U}}_k|)$. Following a similar approach, we can show that the algorithm proposed in [2] requires the same amount of signalling overhead if it is applied in a multicell network.

In the case of distributed *Algorithm 2*, coordination between the BSs is required at step 3b, only to exchange auxiliary variables that are used to achieve the consensus on the out-of-cell interference values. Specifically, k th BS sends $\{x_{ki,k}^l\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_k}$ and $\{x_{li,k}^k\}_{l \in \mathcal{K} \setminus \{k\}, i \in \mathcal{U}_l}$ to BS l for all $l \in \mathcal{K} \setminus \{k\}$. Thus, the number of scalars that k th BS has to send to all the other BSs per iteration is $\omega_k = (K-1)|\mathcal{U}_k| + \sum_{l \in \mathcal{K} \setminus \{k\}} |\mathcal{U}_l|$. Hence, the total number of scalars that K BSs exchange per iteration is $\sum_{k \in \mathcal{K}} \omega_k = 2(K-1)U$. The signalling overhead of the algorithm in [5] per iteration is U . The additional signalling overhead of the proposed *Algorithm 2* is worth, as our algorithm provides a better objective value compared to the algorithm in [5], which is illustrated in the Section V.

E. Complexity of Algorithm 1 and Algorithm 2

Algorithm 1 and *Algorithm 2* are iterative algorithms, and both of them solve convex problems at each iteration (i.e., problem (13) is solved in *Algorithm 1* and problem (26) is solved in *Algorithm 2*). Thus, these problems can be

efficiently solved by using interior-point method that relies on the Newton's method [16, Ch. 11]. It is easy to see that problem (13) consists of $U(T + 2)$ variables and $3U + K$ inequality constraints. Hence, the complexity of *Algorithm 1* per iteration is $O(\sqrt{3U + K}U^3(T + 2)^3 \log(1/\epsilon))$, where ϵ is the required relative accuracy of the duality gap at termination³ [16, Ch. 11]. Furthermore, when T and K are much smaller than U , the complexity of *Algorithm 1* per iteration can be approximated as $O(U^{3.5} \log(1/\epsilon))$. For a multicell scenario as considered in this paper, the complexity of the algorithm in [2] per iteration is the same as in *Algorithm 1*. To get an insight into the overall complexity of *Algorithm 1* and the one in [2], we compare the average execution times of these two algorithms in the numerical results in Section V.

In *Algorithm 2*, each BS solves problem (26), which consists of $|\mathcal{U}_k|(K + T) + U$ variables and $2|\mathcal{U}_k| + U + 1$ inequality constraints. Hence, the complexity of solving problem (26) at k th BS is $O(\sqrt{2|\mathcal{U}_k| + U + 1}(|\mathcal{U}_k|(K + T) + U)^3 \log(1/\epsilon))$. By following a similar approach (i.e., by counting the total number of variables and the inequality constraints), the complexity of solving the BS specific problem in [5] can be expressed as $O(\sqrt{|\mathcal{U}_k| + 3U + 1}(2U + T^2|\mathcal{U}_k|)^3 \log(1/\epsilon))$. To compare the complexities of these two algorithms, assume that the number of users associated with each BS is equal, i.e., let $|\mathcal{U}_k| = \tilde{U}$ for all $k \in \mathcal{K}$. Then, the complexity of solving a BS specific problem in *Algorithm 2* can be expressed as $O(\sqrt{(K + 2)\tilde{U} + 1}(2K + T)^3 \tilde{U}^3 \log(1/\epsilon))$ and that in [5] is $O(\sqrt{(3K + 1)\tilde{U} + 1}(2K + T^2)^3 \tilde{U}^3 \log(1/\epsilon))$. Thus, it is easy to see that *Algorithm 2* has low complexity than the one in [5] for solving a BS specific problem.

F. Convergence of Algorithm 1 and Algorithm 2

Recall that problem (8) is an NP-hard problem. Hence, the optimal solution of problem (8) can only be obtained by using some sort of exhaustive search methods like branch and bound [4]. In practice, the quality of the solution achieved within the first few iterations is a more important performance criterion than the asymptotic results, as we have time to perform only a finite (usually small) number of iterations. The goal of this paper is to provide polynomial-time algorithms that find suboptimal solutions for problem (8). Therefore, the convergence of our proposed polynomial-time algorithms to the optimal value of problem (8) cannot be established. Nonetheless, a weaker form of convergence of *Algorithm 1* and *Algorithm 2* can be established.

Note that *Algorithm 1* obtains an approximate solution of problem (8) by solving problem (9), which is a difference of convex problem. Hence, it can be shown that the sequence of objective values generated by *Algorithm 1* is nonincreasing [30]. Moreover, the optimal objective value of problem (9) is bounded. Thus, the monotonic convergence of *Algorithm 1* can be guaranteed.

To establish the convergence of *Algorithm 2*, we first note that the steps of *Algorithm 2* are similar to those of centralized *Algorithm 1*, except step 3 of both algorithms. At step 3 of *Algorithm 1*, it solves problem (13) in a central controller. However, in *Algorithm 2* the same problem is solved distributively using ADMM algorithm. Note that problem (13) is convex, and hence, by following the approach in [34, Proposition 4.2], [31], we can show that ADMM algorithm converges. Since, *Algorithm 2* inherits the properties of *Algorithm 1*, a sequence of the objective values of problem (13) that is produced upon ADMM convergence is guaranteed to be convergent.

IV. FAIR ADMISSION CONTROL

In Section III, we proposed a centralized and a distributed algorithm to find a set of admissible users during a single time slot. We know that the provided solution cannot always admit all the users within a single time slot due to limited resources. Hence, some users remain unserved, and thus fairness among the users has to be considered as well.

In this section, we first consider the problem of proportional fair admission control. We model this problem as a stochastic optimization problem, and propose a dynamic control algorithm via Lyapunov optimization method [28] to solve it. Finally, we modify this algorithm to obtain max-min fairness.

Recall from Section III that i th user of BS k is admitted to the network if $s_{ki} = 0$, i.e., when $\Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki}$ (see problem (8)). Based on this observation, we define an indicator function $a_{ki}(t)$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, to model the admission status of i th user of BS k in time slot t as

$$a_{ki}(t) = \begin{cases} 1 & s_{ki}(t) = 0 \\ 0 & s_{ki}(t) > 0. \end{cases} \quad (32)$$

It is easy to see from (32) that admission status $a_{ki}(t) = 1$ means $\Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki}$.

We follow the approach in [28, Ch. 3-5] to formulate the problem of proportional fair admission control. The idea is to proportionally maximize a time average admissions of each user, subject to the SINR targets of each user and the power constraints of each BS. To formulate this problem, let us define the time average expectation of $a_{ki}(t)$ over t time slots as

$$\bar{a}_{ki}(t) \triangleq \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{a_{ki}(\tau)\}, \quad (33)$$

where the expectation depends on the control policy and is with respect to the random channel states and the control actions made in reaction to these channel states [28], [35].⁴ Then, by following the approach in [28, Ch. 5], [36] the problem of proportional fair admission control can be expressed, as the following optimization problem:

$$\begin{aligned} & \text{maximize } \liminf_{t \rightarrow \infty} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(\bar{a}_{ki}(t)) \\ & \text{subject to } \Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad \forall t \end{aligned} \quad (34a)$$

³To be consistent with the complexity analysis presented in [2], we use a similar bound to express for the complexities of the proposed algorithms.

⁴Note that we use a commonly used procedure and express the long term average as the time average of expectation, which leads to a tractable algorithm [28], [35]–[38].

$$\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K}, \quad \forall t \quad (34b)$$

$$s_{ki}(t) \geq 0, \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad \forall t, \quad (34c)$$

with variables $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$.

A. Transformed Problem via Auxiliary Variables

In this section we use the Lyapunov optimization technique [28], [35] to find a solution of problem (34). To do this we transform problem (34) such that it conforms to the structure required by the drift-plus-penalty minimization method in [28] and [35]. Note that the $\log(\cdot)$ function is a monotonically increasing function, and hence, we can move $\liminf_{t \rightarrow \infty}$ inside the $\log(\cdot)$ function. Then, by introducing an auxiliary variable $\bar{\mu}_{ki} = \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t)$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, we can equivalently reformulate problem (34) as follows:

$$\begin{aligned} & \text{maximize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(\bar{\mu}_{ki}) \\ & \text{subject to} \quad \bar{\mu}_{ki} \leq \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \\ & \quad \text{constraints (34a) - (34c),} \end{aligned} \quad (35)$$

with variables $\{\bar{\mu}_{ki}\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ and $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$. In problem (35), the first set of inequality constraints holds with equality at the optimal solution due to the monotonic increasing property of the objective function.

Note that problem (35) involves the statistics of random channel states, and that we do not even know. Our goal is to provide a dynamic control algorithm that takes a decision in each time slot and solves problem (35). This we achieve by using the drift-plus-penalty minimization method [28], [35], which converts a long term objective of problem (35) into a series of myopic optimizations. To do this we assume that the auxiliary variable $\bar{\mu}_{ki}$ is a time average of auxiliary variable $\mu_{ki}(t)$ for all $t = \{0, 1, \dots\}$, i.e., $\bar{\mu}_{ki} \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{\mu_{ki}(\tau)\}$. Then, by following the approach in [28, Ch. 5.0.5], we modify problem (35) as the following optimization problem:

$$\begin{aligned} & \text{maximize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \overline{\log(\mu_{ki})} \\ & \text{subject to} \quad \bar{\mu}_{ki} \leq \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \quad (36a) \\ & \quad \text{constraints (34a) - (34c),} \end{aligned} \quad (36b)$$

with variables $\{s_{ki}(t), \mathbf{m}_{ki}(t), \mu_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$, where $\overline{\log(\mu_{ki})}$ is defined as

$$\overline{\log(\mu_{ki})} \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{\log(\mu_{ki}(\tau))\}. \quad (37)$$

Note that by using Jensen's inequality we can easily verify that $\overline{\log(\mu_{ki})}$ is a lower bound on $\log(\bar{\mu}_{ki})$. Thus, a solution of problem (36) is also feasible for the original problem (34), and hence problem (36) provides a reasonable lower bound for the original problem (34).

B. Solving the Transformed Problem

In this subsection we apply the Lyapunov drift-plus-penalty minimization method introduced in [28] to solve problem (36). In particular for this method, the time average inequality constraint (36a) is enforced by transforming problem (36) into a queue stability problem. In other words, for each time average inequality constraint in (36a), a *virtual queue* is associated; the stability of these virtual queues implies the feasibility of constraint (36a).

Let $\{G_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ be the virtual queues associated with constraint (36a). We update virtual queue $G_{ki}(t)$ for all $k \in \mathcal{K}$, $i \in \mathcal{U}_k$, in every time slot as

$$G_{ki}(t+1) = \max[G_{ki}(t) + \mu_{ki}(t) - a_{ki}(t), 0]. \quad (38)$$

Note that $G_{ki}(t)$ can be viewed as a backlog in a virtual queue with an input process $\mu_{ki}(t)$ and a service process $a_{ki}(t)$. Here, we adopt the notion of *strong stability*,⁵ and we say that the queues are strongly stable if [28, Ch. 2]

$$\bar{G}_{ki} \triangleq \limsup_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{G_{ki}(\tau)\} < \infty, \quad k \in \mathcal{K}, \quad i \in \mathcal{U}_k. \quad (39)$$

Intuitively, (39) means that a queue is strongly stable if its time average backlog is finite.

We next define the Lyapunov function and its drift, which will be used to define the queue stability problem. Let $\mathbf{G}(t)$ be a vector of the virtual queues, i.e., $\mathbf{G}(t) = [G_{11}(t), \dots, G_{K I_K}(t)]^T$, then a quadratic Lyapunov function $L(\mathbf{G}(t))$ can be defined as [28, Ch. 3]

$$L(\mathbf{G}(t)) = \frac{1}{2} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t)^2. \quad (40)$$

Note that if $L(\mathbf{G}(t))$ is small then all queues are small, and if $L(\mathbf{G}(t))$ is large then at least one queue is large. Hence, by minimizing a drift in the Lyapunov function (i.e., by minimizing a difference in the Lyapunov function from one slot to the next), queues $\{G_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ can be stabilized [28, Ch. 4]. Furthermore, by using expression (40), the drift in the Lyapunov function from one slot to the next can be defined as

$$\Delta(\mathbf{G}(t)) = \mathbb{E}\{L(\mathbf{G}(t+1)) - L(\mathbf{G}(t)) | \mathbf{G}(t)\}. \quad (41)$$

Now we use the drift-plus-penalty minimization method [28] to solve problem (36). In this method, a control policy that solves problem (36) is obtained by minimizing an upper bound on the following expression [28]:

$$\Delta(\mathbf{G}(t)) - V \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \mathbb{E}\{\log(\mu_{ki}(t)) | \mathbf{G}(t)\}, \quad (42)$$

where $V \geq 0$, subject to the constraints (34a)-(34c) in each time slot, i.e.,

$$\Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \quad (43a)$$

$$\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K} \quad (43b)$$

⁵A definition of strong stability is general, and it also implies other forms of stabilities [28, Th. 2.8].

$$s_{ki}(t) \geq 0, \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}. \quad (43c)$$

Note that the expression (42) has two terms. The first term $\Delta(\mathbf{G}(t))$ is the drift; by minimizing this we ensure the inequality constraint (36a) [28, Ch. 4]. Furthermore, by minimizing the second term $-\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} E\{\log(\mu_{ki}(t))|\mathbf{G}(t)\}$, the objective function of problem (36) is maximized.

In the rest of this section, to simplify algorithm development, we first find an upper bound of expression (42). Then, we present a dynamic control algorithm to solve problem (36) that, at each time slot, minimizes an upper bound of expression (42) subject to constraints (43a)-(43c).

To obtain an upper bound on the objective function of problem (42), using expression (38), we first note that $G_{ki}(t+1)$ can be bounded as

$$G_{ki}(t+1)^2 \leq G_{ki}(t)^2 + a_{ki}(t)^2 + \mu_{ki}(t)^2 + 2G_{ki}(t)[\mu_{ki}(t) - a_{ki}(t)]. \quad (44)$$

In order to find expression (44), we have used the fact that $(\max[G + \mu - a, 0])^2 \leq G^2 + \mu^2 + a^2 + 2G(\mu - a)$ for any $G \geq 0$, $\mu \geq 0$, and $a \geq 0$. Then, by using expression (41) and inequality (44), an upper bound of expression (42) can be expressed as

$$\begin{aligned} \Delta(\mathbf{G}(t)) - V \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} E\{\log(\mu_{ki}(t))|\mathbf{G}(t)\} \\ \leq D - V \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} E\{\log(\mu_{ki}(t))|\mathbf{G}(t)\} \\ + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) E\{\mu_{ki}(t)|\mathbf{G}(t)\} \\ - \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) E\{a_{ki}(t)|\mathbf{G}(t)\}, \end{aligned} \quad (45)$$

where D is a finite positive constant that satisfies the following condition for all t :

$$D \geq \frac{1}{2} \left[\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} E\{\mu_{ki}(t)^2|\mathbf{G}(t)\} + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} E\{a_{ki}(t)^2|\mathbf{G}(t)\} \right]. \quad (46)$$

We now summarize the proposed proportional fair admission control algorithm to solve problem (36) in *Algorithm 3*. The proposed algorithm observes the virtual queues $\{G_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ and the channel states $\{\mathbf{h}_{ki}^l(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ in every time slot t , and takes a control action to minimize the righthand side of expression (45) subject to constraints (43a)-(43c). The minimization of the righthand side of expression (45) can be decoupled across variables $\{\mu_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ and $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$, resulting in subproblems as shown in *Algorithm 3*. Note that the drift-plus-penalty minimization method uses the concept of *opportunisticly minimizing an expectation* [28, Ch. 1.8] to solve the subproblems.

The auxiliary variables are updated in step 2 of *Algorithm 3*. Specifically, in problem (47) the objective function is maximized when the derivative with respect to $\mu_{ki}(t)$ is zero, i.e., $\mu_{ki}^*(t) = \min(V/G_{ki}(t), \mu^{\max})$ for all $k \in \mathcal{K}$, $i \in \mathcal{U}_k$.

Algorithm 3 Proportional Fair Dynamic Admission Control Algorithm for Problem (36)

- 1: **initialization:** Set $t = 0$.
- foreach** time slot t **do**
- 2: Auxiliary variable: obtain $\{\mu_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ by solving following optimization problem:

$$\begin{aligned} & \text{maximize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} (V \log(\mu_{ki}(t)) - G_{ki}(t) \mu_{ki}(t)) \\ & \text{subject to} \quad 0 \leq \mu_{ki}(t) \leq \mu^{\max}, \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \end{aligned} \quad (47)$$

with variables $\{\mu_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$, $\mu^{\max} > 0$ is an algorithm parameter as given in [28, Ch. 5].

- 3: Resource allocation: obtain $\{\mathbf{m}_{ki}(t), s_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ by solving following problem:

$$\begin{aligned} & \text{maximize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) a_{ki}(t) \\ & \text{subject to} \quad \Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \end{aligned} \quad (48a)$$

$$\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K} \quad (48b)$$

$$s_{ki}(t) \geq 0, \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad (48c)$$

with variables $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$.

- 4: Queue update: update $\{G_{ki}(t+1)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ using (38). Set $t = t + 1$, and go to step (2).

end for

Step 3 solves the resource allocation problem (48). A solution of it is found by adopting *Algorithm 1*, and is explained in subsection IV-C. Step 4 updates the virtual queues.

C. Solving the Resource Allocation Problem of Dynamic Control Algorithm

In this subsection, we show how *Algorithm 1* can be adopted to find a solution of problem (48). To do this, we first compactly write expression (32) as [21]

$$a_{ki}(t) = 1 - \|s_{ki}(t)\|_0. \quad (49)$$

By substituting expression (49) into the objective function of problem (48), we simplify it as

$$\begin{aligned} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) a_{ki}(t) &= \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) \\ &\quad - \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) \|s_{ki}(t)\|_0. \end{aligned} \quad (50)$$

Then, by using expression (50) as the objective function, we can equivalently reformulate problem (48) as follows:

$$\begin{aligned} & \text{minimize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t) \|s_{ki}(t)\|_0 \\ & \text{subject to constraints (48a) - (48c)}, \end{aligned} \quad (51)$$

with variables $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$. Note that in the objective function of problem (51) we have removed the constant

term $\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} G_{ki}(t)$, since it does not affect the optimal solution. Furthermore, by changing the sign of the objective function we have converted it to a minimization problem. The objective function of problem (51) is combinatorial, and constraint (48a) is nonconvex; therefore, this problem is a difficult optimization problem.

To solve problem (51), we follow a similar approach that we have carried out in subsection III-A. That is, we approximate the objective function of problem (51) and the constraint function (48a) with their best convex approximations. Using the aforementioned technique, a solution of problem (51) can be approximated near an arbitrary positive point $(\hat{s}_{ki}, \hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki})$ for all $i \in \mathcal{U}_k$, $k \in \mathcal{K}$, by solving the following optimization problem:

$$\begin{aligned} & \text{minimize} \quad \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \tilde{w}_{ki}(t) s_{ki}(t) \\ & \text{subject to} \quad \gamma_{ki} - s_{ki}(t) - \hat{g}_{ki}(\mathbf{m}_{ki}(t), \beta_{ki}(t)) \leq 0, \\ & \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \\ & \quad \sum_{j \in \mathcal{U}_k, j \neq i} |(\mathbf{h}_{ki}^k(t))^H \mathbf{m}_{kj}(t)|^2 \\ & \quad + \sum_{l \in \mathcal{K} \setminus \{k\}} \sum_{j \in \mathcal{U}_l} |(\mathbf{h}_{ki}^l(t))^H \mathbf{m}_{lj}(t)|^2 + \sigma_{ki}^2 \leq \beta_{ki}(t), \\ & \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \\ & \quad \text{constraints (48b), (48c),} \end{aligned} \quad (52)$$

where $\tilde{w}_{ki}(t) = G_{ki}(t)/(\hat{s}_{ki}(t) + \epsilon)$ and variables are $\{s_{ki}(t), \mathbf{m}_{ki}(t), \beta_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$, $\hat{g}_{ki}(\mathbf{m}_{ki}, \beta_{ki})$ is as defined in expression (11). Here, the weight $\tilde{w}_{ki}(t)$ of each user is associated with the virtual queue of that user (compare with weight $w_{ki}(t)$ in problem (13)). We can see that problem (52) is similar to problem (13), and hence we use proposed *Algorithm 1* to find a solution of problem (48).

D. Max-Min Fair Admission Control

In this subsection, we briefly discuss the method for providing max-min fairness while users are being admitted to the network. This notion of fairness can be achieved by following similar steps that we have presented in subsection IV-B.

The problem of max-min fair admission control can be expressed, by modifying the objective function of problem (34), as the following optimization problem [28, Ch. 5]:

$$\begin{aligned} & \text{maximize} \quad \min_{k \in \mathcal{K}, i \in \mathcal{U}_k} \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t) \\ & \text{subject to} \quad \Gamma_{ki}(\mathbf{m}(t)) \geq \gamma_{ki} - s_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad \forall t \end{aligned} \quad (53a)$$

$$\sum_{i \in \mathcal{U}_k} \|\mathbf{m}_{ki}(t)\|_2^2 \leq P_k^{\max}, \quad k \in \mathcal{K}, \quad \forall t \quad (53b)$$

$$s_{ki}(t) \geq 0, \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K}, \quad \forall t, \quad (53c)$$

where $\bar{a}_{ki}(t)$ is as defined in (33), and variables are $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$.

We now present a method to find a solution of problem (53). Let $\bar{v}$ be an auxiliary variable. Then, problem (53) can be

equivalently reformulated as the following optimization problem:

$$\begin{aligned} & \text{maximize} \quad \bar{v} \\ & \text{subject to} \quad \bar{v} \leq \min_{k \in \mathcal{K}, i \in \mathcal{U}_k} \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t), \end{aligned} \quad (54a)$$

$$\text{constraints (53a) - (53c),} \quad (54b)$$

with variables $\bar{v}$ and $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$.

Next, we use the drift-plus-penalty minimization method [28], [35] to find a solution of problem (54). To do this we assume that the auxiliary variable $\bar{v}$ is a time average of auxiliary variable $v(t)$ for all $t = \{0, 1, \dots\}$, i.e., $\bar{v} \triangleq \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathbb{E}\{v(\tau)\}$. Then, by expressing the inequality constraint (54a) as a set of separate inequalities, we modify problem (54) as

$$\begin{aligned} & \text{maximize} \quad \bar{v} \\ & \text{subject to} \quad \bar{v} \leq \liminf_{t \rightarrow \infty} \bar{a}_{ki}(t), \quad i \in \mathcal{U}_k, \quad k \in \mathcal{K} \end{aligned} \quad (55a)$$

$$\text{constraints (53a) - (53c),} \quad (55b)$$

with variables $v(t)$ and $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ for all $t \in \{0, 1, \dots\}$. Next, we follow the steps that we have used in subsection IV-B to find a solution of problem (55).

Let $\{Z_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ be the virtual queues associated with constraint (55a). We update virtual queues $Z_{ki}(t)$ for all $k \in \mathcal{K}$, $i \in \mathcal{U}_k$, in every time slot as

$$Z_{ki}(t+1) = \max[Z_{ki}(t) + v(t) - a_{ki}(t), 0]. \quad (56)$$

Note that the queue update equation (56) is different from the one given in expression (38). The stability of virtual queues $\{Z_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ implies the feasibility of constraint (55a). Then, a quadratic Lyapunov function $L(\mathbf{Z}(t))$ can be defined as [28, Ch. 3]

$$L(\mathbf{Z}(t)) = \frac{1}{2} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t)^2, \quad (57)$$

where $\mathbf{Z}(t) = [Z_{11}(t), \dots, Z_{K I_K}(t)]^T$. Furthermore, the drift in the Lyapunov function from one slot to the next can be defined, using expression (57), as follows:

$$\Delta(\mathbf{Z}(t)) = \mathbb{E}\{L(\mathbf{Z}(t+1)) - L(\mathbf{Z}(t)) | \mathbf{Z}(t)\}. \quad (58)$$

By following the approach in Section IV-B, it is easy to see that drift-plus-penalty expression satisfies

$$\begin{aligned} & \Delta(\mathbf{Z}(t)) - \mathbb{E}\{v(t) | \mathbf{Z}(t)\} \\ & \leq \tilde{D} - \mathbb{E}\{v(t) | \mathbf{Z}(t)\} + \mathbb{E}\{v(t) | \mathbf{Z}(t)\} \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t) \\ & \quad - \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t) \mathbb{E}\{a_{ki}(t) | \mathbf{Z}(t)\}, \end{aligned} \quad (59)$$

where $\tilde{D}$ is a finite positive constant that satisfies the following condition for all t :

$$\begin{aligned} \tilde{D} \geq & \frac{1}{2} \left[\sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \mathbb{E}\{v(t)^2 | \mathbf{Z}(t)\} \right. \\ & \left. + \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \mathbb{E}\{a_{ki}(t)^2 | \mathbf{Z}(t)\} \right]. \end{aligned} \quad (60)$$

We now summarize the proposed dynamic control algorithm for problem (55) in *Algorithm 4*. The proposed algorithm takes a control action to minimize the righthand side of expression (59) subject to constraints (53a)-(53c).

Algorithm 4 Max-Min Fair Dynamic Admission Control Algorithm for Problem (55)

1: **initialization:** Set $t = 0$.

foreach time slot t **do**

2: Auxiliary variable: obtain $v(t)$ by solving following optimization problem:

$$\begin{aligned} & \text{maximize } Vv(t) - v(t) \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t) \\ & \text{subject to } 0 \leq v(t) \leq v^{\max}, \end{aligned} \quad (61)$$

with variables $v(t)$, and $v^{\max} > 0$ is an algorithm parameter as given in [28, Ch. 5].

3: Resource allocation: obtain $\{\mathbf{m}_{ki}(t), s_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ by solving following problem:

$$\begin{aligned} & \text{maximize } \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t) a_{ki}(t) \\ & \text{subject to constraints (53a) – (53c),} \end{aligned} \quad (62)$$

with variables $\{s_{ki}(t), \mathbf{m}_{ki}(t)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$.

4: Queue update: $\{Z_{ki}(t+1)\}_{k \in \mathcal{K}, i \in \mathcal{U}_k}$ using (56). Set $t = t + 1$, and go to step (2).

end for

At step 2 of *Algorithm 4* the auxiliary variables are updated, where we obtain the solution as $v^*(t) = v^{\max}$ if $V \geq \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} Z_{ki}(t)$, and $v^*(t) = 0$ otherwise. Resource allocation is done at step 3 of the algorithm by solving problem (62). Note that problem (62) is identical to problem (48), and hence, we use the method described in Section IV-C to update step 3 of this algorithm.⁶

V. NUMERICAL RESULTS

In this section, we evaluate the performance of the proposed *Algorithm 1*, *Algorithm 2*, *Algorithm 3*, and *Algorithm 4*. In our simulations, we consider two multicell network setups, with the number of BSs, users, and antennas as given in Table I. In each network, the BSs are placed in such a way that the distance between any two adjacent BSs is equal, and we denote this distance by D_{BS} . We assume circular cells, where the radius of each one is denoted by R_{BS} . Furthermore, we consider that each BS serves four users.

We use an exponential path loss model, where the channel gains between BS l and i th user of BS k is modeled as $\mathbf{h}_{ki}^l(t) = (d_{ki}^l(t)/d_0)^{-\eta/2} \mathbf{c}_{ki}^l(t)$, where $d_{ki}^l(t)$ is the distance from l th BS to i th user of BS k , d_0 is the *far field reference distance* [39], η is the path loss exponent, and

⁶The dynamic control *Algorithm 3* and *Algorithm 4* use *Algorithm 1* to solve the optimization problem in step 3 of these algorithms. Compared to the computational complexity of step 3, the complexity of solving the problem in step 2 of both these algorithms is negligible. Hence, the computational complexity of *Algorithm 3* and *Algorithm 4* per time slot is the same as that of *Algorithm 1*. Furthermore, as signalling is required only at step 3 of the dynamic control algorithms, the signalling overhead of these algorithms per time slot is the same as that of *Algorithm 1*.

TABLE I
NETWORK CONFIGURATIONS

Multicell network	Number of BS (K)	Total number of users ($\sum_{k \in \mathcal{K}} \mathcal{U}_k $)	Number of antennas (T)
Network 1 (NW1)	3	12	4
Network 2 (NW2)	7	28	4

Fig. 2. Objective value F_C^p versus iterations, and the number of admitted users versus iterations: *Algorithm 1* for NW1 with $\gamma = 9\text{dB}$.

$\mathbf{c}_{ki}^l(t) \in \mathbb{C}^T$ is arbitrarily chosen from the distribution $\mathcal{CN}(\mathbf{0}, \mathbf{I})$ (i.e., frequency-flat fading with uncorrelated antennas). Note that the term $(d_{ki}^l(t)/d_0)^{-\eta/2}$ denotes large scale fading, and the term $\mathbf{c}_{ki}^l(t)$ denotes small scale fading. We refer to an arbitrarily generated set of fading coefficients $\mathcal{C}(t) = \{\mathbf{c}_{ki}^l(t) | l, k \in \mathcal{K}, i \in \mathcal{U}_k\}$ as a *single fading realization*, and an arbitrarily generated set of distances $\mathcal{D}(t) = \{d_{ki}^l(t) | l, k \in \mathcal{K}, i \in \mathcal{U}_k\}$ as a *single topology realization*.

We assume that $\{P_k^{\max} = P_0^{\max}\}_{k \in \mathcal{K}}$, $\{\sigma_{ki} = \sigma, \gamma_{ki} = \gamma\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$. We define the Signal-to-Noise Ratio (SNR) operating point at a distance d as $\text{SNR}(d) = (P_0^{\max}/\sigma^2) (d/d_0)^{-\eta}$. In our simulations, we set $\text{SNR}(R_{BS}) = 5\text{dB}$, $d_0 = 1$, $\eta = 4$, $P_0^{\max}/\sigma^2 = 45\text{dB}$, and $D_{BS} = 1.6R_{BS}$.

We now evaluate the performance of proposed *Algorithm 1* and *Algorithm 2* for *Network 1* (NW1) and *Network 2* (NW2). In all our simulations, *Algorithm 1* is stopped if the improvement in the objective value is less than some κ [30], where we set $\kappa = 0.01$. In *Algorithm 2*, the ADMM algorithm (i.e., the inner iterations) is stopped after running a fixed number of iterations Q [31, Ch. 3.2]. The outer iteration is stopped when the improvement in the objective function of problem (9) is less than 0.01 or when algorithm runs for a maximum 20 iterations. Furthermore, parameter ρ is set to 0.1. To illustrate the convergence behavior of *Algorithm 1* and *Algorithm 2*, we consider an arbitrarily chosen single fading realization and single topology realization.

Fig. 2 shows the convergence behavior of the proposed *Algorithm 1* in the case of NW1. Note that we have also used the same figure to plot the number of admitted users in

Fig. 3. Objective value F_D^q versus iterations, and the number of admitted users versus iterations: *Algorithm 2* for NW1 with $\gamma = 9\text{dB}$.

each iteration. We denote by $F_C^p = \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(s_{ki}^p + \epsilon)$ the objective value problem (9) calculated at p th iteration of *Algorithm 1*, where s_{ki}^p is the auxiliary variable of i th user of BS k in p th iteration. We evaluate F_C^p after step 2 of *Algorithm 1*. At each iteration, the number of admitted users are counted by checking how many users have satisfied condition (5). Results show that the proposed *Algorithm 1* converges within the first few iterations.

Fig. 3 shows the convergence behavior of the distributed *Algorithm 2*, along with the results obtained by centralized *Algorithm 1* for NW1. It also shows the number of admitted users in each iteration of the distributed *Algorithm 2* compared to that of *Algorithm 1*. Here, we run ADMM algorithm for $Q = 10$ iterations. We denote by $F_D^q = \sum_{k \in \mathcal{K}} \sum_{i \in \mathcal{U}_k} \log(s_{ki}^q + \epsilon)$ the objective value of problem (9) calculated at iteration q of *Algorithm 2*, where s_{ki}^q is the auxiliary variable of i th user of BS k in q th iteration. The “square” and “circle” markers in Fig. 3 are to indicate F_D^q and the number of admitted users, respectively. Furthermore, these markers represent the start of ADMM algorithm for a new point $\{\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$ which is set at step 2 of the algorithm. Results show that the proposed distributed algorithm converges to the objective value obtained via centralized *Algorithm 1*, and it also admits the same number of users as *Algorithm 1*.

We now discuss the average behavior of *Algorithm 1* and *Algorithm 2*. We run both algorithms on NW1 and NW2 over 500 fading and topology realizations. In Fig. 4, we compare the average number of admitted users versus the SINR target γ of the proposed *Algorithm 1* with an exhaustive search algorithm, the D-SOC algorithm in [2] (we name as Alg. [2] in this paper⁷), and the centralized algorithm in [5] (we name as Alg. [5]-centralized). Results in Fig. 4(a) are for NW1, and they show that the average number of admitted users of *Algorithm 1* is close to that of the exhaustive search algorithm. In addition, results show that when γ is low our *Algorithm 1*

(a)

(b)

Fig. 4. Centralized algorithm comparison for average number of admitted users versus the SINR target: (a) NW1; (b) NW2.

slightly outperforms Alg. [2], and when γ is high Alg. [2] marginally performs better than *Algorithm 1*. Furthermore, it is shown that proposed *Algorithm 1* is better than Alg. [5]-centralized for both high and low SINR targets. In the case of NW2 (see Fig. 4(b)), results show that the performance obtained by the proposed *Algorithm 1* and D-SOC algorithm show a similar behavior as in NW1. However, *Algorithm 1* outperforms Alg. [5]-centralized for all considered SINR targets.⁸

After that, we evaluate the average number of admitted users versus SINR target γ of the proposed distributed *Algorithm 2* in Fig. 5. As benchmarks, we consider an exhaustive search algorithm, proposed centralized *Algorithm 1*, and the distributed algorithm in [5] (we name as Alg. [5]-distributed). For NW1 (see Fig. 5(a)), the average number of admitted users obtained by distributed *Algorithm 2* is plotted by running ADMM algorithm for $Q = 1, 5, 10$, and 50 iterations. Results show that for $Q = 5, 10$, and 50 proposed

⁷Note that we compare our results with one of the best algorithms, i.e., Algorithm 2: Deflation based on SOC programming (D-SOC), proposed in [2]. Furthermore, the algorithm parameter M in the D-SOC algorithm is set to zero.

⁸In the case of NW2, the number of combinations required to be checked using an exhaustive search algorithm is very large. Therefore, we have not used the exhaustive search method to find the optimal solution for NW2. It is worth noting that, it may not be practical to use exhaustive search method when the number of users are greater than fifteen [2], [21].

(a)

(b)

Fig. 5. Distributed algorithm comparison for average number of admitted users versus the SINR target: (a) NW1; (b) NW2.

Algorithm 2 outperforms Alg. [5]-distributed for all the considered SINR targets. When $Q = 1$, our *Algorithm 2* performs slightly better than Alg. [5]-distributed at low SINR targets. However, for large SINR targets, proposed *Algorithm 2* outperforms Alg. [5]-distributed when $Q = 1$. Furthermore, results show that when γ is low, proposed centralized *Algorithm 1* admits more users on average than that of the distributed *Algorithm 2*. But for all the simulated cases, when γ is high the average number of admitted users obtained by using *Algorithm 2* is closer to that of the centralized *Algorithm 1*, for the considered values of Q . For NW2 (see Fig. 5(b)), the average number of admitted users obtained by *Algorithm 2* is plotted by varying $Q = 1$ and 10. Results show that our *Algorithm 2* outperforms Alg. [5]-distributed, when $Q = 1$ and 10, for all considered SINR targets. Furthermore, results show that when the number of iterations that we run the ADMM algorithm are increased, the performance of *Algorithm 2* is closer to that of *Algorithm 1*; but it needs more iterations to reach the centralized solution. Fig. 6 shows the average number of admitted users per BS versus iteration q for *Algorithm 2*. We have set $Q = 10$ for ADMM algorithm, i.e., step 3 of *Algorithm 2*. In each graph a marker is drawn to show the

Fig. 6. Iterations versus average number of admitted users per BS for NW1 with $\gamma = 9\text{dB}$.

start of ADMM algorithm for a new point $\{\hat{\mathbf{m}}_{ki}, \hat{\beta}_{ki}\}_{i \in \mathcal{U}_k, k \in \mathcal{K}}$ that is set at step 2 of *Algorithm 2*. Results show that once *Algorithm 2* approximates problem (9) as a convex problem at step 2, (i.e., after each 10th iteration) the average number of admitted users per BS increases.

We now evaluate the average execution times and the average number of admitted users for proposed centralized *Algorithm 1*, for $\gamma = 9\text{dB}$ and 15dB . We consider a network with 7 BSs. Then, the simulations are carried out by varying the total number of users in the network as $U = 28, 35, \dots, 56$, such that each BS has an equal number of users (i.e., each BS serves $U/7$ users). Fig. 7(a) shows the average execution time (in seconds) versus total number of users in the network. Results show that for a small number of users (i.e., $U \leq 35$), the average execution time of proposed *Algorithm 1* is slightly higher than that of Alg. [2]. However, for a large number of users (i.e., $U > 35$) the average execution time of Alg. [2] is higher compared to the proposed algorithm. In order to show the gain in the objective value (i.e., the total admitted users), we also plot the average number of admitted users versus total number of users in the network in Fig. 7(b). Results show that for all simulated cases both algorithms show similar performance. However, it is worth noting that the proposed algorithm has low execution time for larger networks.

Next, we evaluate the average behavior of the proposed proportional fair admission control *Algorithm 3* and the max-min fair admission control *Algorithm 4* for NW1 and NW2. In our simulations, we consider that two of the users in each cell are located within a circular region with $2R_{\text{BS}}/3$ radius around the BS (we call them *in-cell users*). The other two users are located at the cell boundary, and we name them as *cell-edge users*. We label the in-cell users in each cell with the integer value 1 and 2 and the cell-edge users with the integer value 3 and 4. We set the SINR target $\gamma = 9\text{dB}$, and the obtained results are averaged over $T_{\text{max}} = 15000$ time slots. As we want to check the performance of the cell-edge users, the simulations are run by changing only the fading realizations while keeping the topology realization fixed. We compare both *Algorithm 3* and *Algorithm 4* with the

(a)

(b)

Fig. 7. Centralized algorithms performance comparison: (a) Average execution time (b) Average number of admitted users.

Fig. 8. Objective value of problem (35) versus V , obtained by using Algorithm 3.

average performance obtained by running proposed Algorithm 1 (which does not account for fairness) for $T_{\max}$ time slots.

Fig. 8 shows the objective value of problem (35) versus parameter V for NW1. Results show that the objective value

Fig. 9. Average number of admissions of users for $V = 25$.

Fig. 10. Objective value of problem (53) versus V , obtained by using Algorithm 4.

Fig. 11. Average number of admissions of users for $V = 25$.

increases with V , and it converges to its maximum value. Fig. 9 shows the average number of admissions of users in each BS for NW1. It can be seen that average number

Fig. 12. Average number of admissions of users for NW2 when $V = 25$: (a) *Algorithm 3*, proportional fairness; (b) *Algorithm 4*, max-min fairness.

of admissions of the cell-edge users (i.e., users 3 and 4 of each BS) have been proportionally increased in proportional fair admission control *Algorithm 3*, compared to that of *Algorithm 1*. However, it can be seen that this performance gain of the cell-edge users is obtained at the expense of the performance of in-cell users (i.e., user 1 and 2 of each BS).

In Fig. 10 and Fig. 11 we see the average behavior of *Algorithm 4* in the case of NW1. The results in Fig. 10 show that the objective value improves as V is increased. Furthermore, Fig. 11 illustrates the average number of admissions of each user, and it shows *Algorithm 4* has admitted all the users nearly equally within $T_{\max}$, compared to *Algorithm 1*. Note that the gains of the cell-edge users are obtained at the cost of performance reduction for in-cell users.

For NW2, the average performances of *Algorithm 3* and *Algorithm 4* are shown in Fig. 12(a) and Fig. 12(b), respectively. Results show that *Algorithm 3* and *Algorithm 4* show a similar performance for NW2 as we have seen in NW1 (compare with Fig. 9 and Fig. 11), and hence, these two

algorithms provide the required fairness for a network with a higher number of users.

VI. CONCLUSIONS

We have considered the admission control problem for the downlink of a multicell multiple-input single-output system. We have cast this problem as an ℓ_0 minimization problem, which is combinatorial, NP-hard. This ℓ_0 minimization problem has then been approximated as a non-combinatorial one. Then, we have proposed suboptimal centralized and distributed algorithms based on sequential convex programming and alternating direction method of multipliers to solve it. Numerically, we have shown that the proposed centralized and distributed admission control algorithms achieve the near-to-optimal performance. Next, we have extended the admission control problem to provide fairness, where a long term fairness among the users has been guaranteed. We have focused on proportional and max-min fairness, and have proposed dynamic control algorithms via Lyapunov optimization to solve them. It has been shown that the proposed fair admission control algorithms guarantee fairness among the users. The study of fairness guaranteed distributed dynamic admission control algorithms is left as a future interesting work.

REFERENCES

- [1] M. Chiang, P. Hande, T. Lan, and C. W. Tan, "Power control in wireless cellular networks," *Found. Trends Netw.*, vol. 2, no. 4, pp. 381–533, Apr. 2008.
- [2] E. Matskani, N. D. Sidiropoulos, Z.-Q. Luo, and L. Tassiulas, "Convex approximation techniques for joint multiuser downlink beamforming and admission control," *IEEE Trans. Wireless Commun.*, vol. 7, no. 7, pp. 2682–2693, Jul. 2008.
- [3] D. I. Evangelinakis, N. D. Sidiropoulos, and A. Swami, "Joint admission and power control using branch & bound and gradual admissions," in *Proc. IEEE Workshop Signal Adv. Wireless Commun.*, Jun. 2010, pp. 1–5.
- [4] S. Boyd and J. Mattingley, "Branch and bound methods," Stanford Univ., Stanford, CA, USA, Tech. Rep. EE364b, 2011. [Online]. Available: https://stanford.edu/class/ee364b/lectures/bb_notes.pdf
- [5] H.-T. Wai and W.-K. Ma, "A decentralized method for joint admission control and beamforming in coordinated multicell downlink," in *Proc. Conf. Rec. 56th Asilomar Conf. Signals, Syst. Comput. (ASILOMAR)*, Nov. 2012, pp. 559–563.
- [6] D. H. N. Nguyen, L. B. Le, and T. Le-Ngoc, "Multiuser admission control and beamforming optimization algorithms for MISO heterogeneous networks," *IEEE Access*, vol. 3, pp. 759–773, Jun. 2015.
- [7] Y.-F. Liu, Y.-H. Dai, and Z.-Q. Luo, "Joint power and admission control via linear programming deflation," *IEEE Trans. Signal Process.*, vol. 61, no. 6, pp. 1327–1338, Mar. 2013.
- [8] M. Monemi, M. Rasti, and E. Hossain, "Low-complexity SINR feasibility checking and joint power and admission control in prioritized multitier cellular networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 2421–2434, Mar. 2016.
- [9] I. Mitliagkas, N. D. Sidiropoulos, and A. Swami, "Joint power and admission control for ad-hoc and cognitive underlay networks: Convex approximation and distributed implementation," *IEEE Trans. Wireless Commun.*, vol. 10, no. 12, pp. 4110–4121, Dec. 2011.
- [10] X. Zhai, L. Zheng, and C. W. Tan, "Energy-infeasibility trade-off in cognitive radio networks: Price-driven spectrum access algorithms," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 3, pp. 528–538, Mar. 2014.
- [11] N. Bambos, S. C. Chen, and G. J. Pottie, "Channel access algorithms with active link protection for wireless communication networks with power control," *IEEE/ACM Trans. Netw.*, vol. 8, no. 5, pp. 583–597, Oct. 2000.

- [12] M. Andersin, Z. Rosberg, and J. Zander, "Gradual removals in cellular PCS with constrained power control and noise," in *Proc. IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun.*, vol. 1, Sep. 1995, pp. 56–60.
- [13] M. Xiao, N. B. Shroff, and E. K. P. Chong, "Distributed admission control for power-controlled cellular wireless systems," *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 790–800, Dec. 2001.
- [14] S. Boyd, L. Xiao, A. Mutapcic, and J. Mattingley, "Notes on decomposition methods," Stanford Univ., Stanford, CA, USA, Tech. Rep. EE364B, 2007. [Online]. Available: http://stanford.edu/class/ee364b/lectures/decomposition_notes.pdf
- [15] S. Boyd, L. Xiao, A. Mutapcic, and J. Mattingley, "Sub-gradient methods," Stanford Univ., Stanford, CA, USA, Tech. Rep. EE364b, 2014. [Online]. Available: http://stanford.edu/class/ee364b/lectures/subgrad_method_notes.pdf
- [16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [17] G. J. Foschini and Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," *IEEE Trans. Veh. Technol.*, vol. 42, no. 4, pp. 641–646, Nov. 1993.
- [18] D. Z. Du and P. M. Pardalos, *Handbook of Combinatorial Optimization*, 1st ed. Dordrecht, Netherlands: Kluwer, 1998.
- [19] Z.-Q. Luo, W.-K. Ma, A. M.-C. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
- [20] A. B. Gershman, N. D. Sidiropoulos, S. Shahbazpanahi, M. Bengtsson, and B. Ottersten, "Convex optimization-based beamforming," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 62–75, May 2010.
- [21] S. Boyd, " ℓ_1 -norm methods for convex-cardinality problems," Stanford Univ., Stanford, CA, USA, Tech. Rep. EE364b, 2007. [Online]. Available: http://www.stanford.edu/class/ee364b/lectures/l1_slides.pdf
- [22] L. Rosasco. (2010). *Statistical Learning Theory and Applications: Sparsity Based Regularization*. [Online]. Available: <http://www.mit.edu/~9.520/spring10/scribe-notes/class10-sparsity-scribe.pdf>
- [23] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.
- [24] D. Tse and P. Vishwanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [25] S. Sesia, I. Toufik, and M. Baker, *LTE the UMTS Long Term Evolution: From Theory to Practice*, 2nd ed. Chichester, U.K.: Wiley, 2011.
- [26] L. Dai, B. Wang, Y. Yuan, S. Han, C.-I. I, and Z. Wang, "Non-orthogonal multiple access for 5G: Solutions, challenges, opportunities, and future research trends," *IEEE Wireless Commun. Mag.*, vol. 53, no. 9, pp. 74–81, Sep. 2015.
- [27] P. Viswanath, D. N. C. Tse, and R. Laroia, "Opportunistic beamforming using dumb antennas," *IEEE Trans. Inf. Theory*, vol. 48, no. 6, pp. 1277–1294, Jun. 2002.
- [28] M. J. Neely, *Stochastic Network Optimization With Application to Communication and Queuing Systems, Synthesis Lectures on Communication Networks*, 1st ed. Belmont, MA, USA: Morgan and Claypool, 2010.
- [29] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, 1995.
- [30] T. Lipp and S. Boyd, "Variations and extensions of the convex-concave procedure," *Opt. Eng.*, vol. 17, no. 2, pp. 263–287, Jun. 2016.
- [31] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
- [32] E. J. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *J. Fourier Anal. Appl.*, vol. 14, nos. 5–6, pp. 877–905, 2008.
- [33] S. Boyd, "Sequential convex programming," Stanford Univ., Stanford, CA, USA, Tech. Rep. EE364b, 2007. [Online]. Available: http://www.stanford.edu/class/ee364b/lectures/seq_slides.pdf
- [34] D. P. Bertsekas and J. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1997.
- [35] L. Georgiadis, M. J. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Found. Trends Netw.*, vol. 1, no. 1, pp. 1–144, 2006.
- [36] M. J. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *IEEE/ACM Trans. Netw.*, vol. 16, no. 2, pp. 396–409, Apr. 2008.
- [37] Y. Mao, J. Zhang, and K. B. Letaief, "A Lyapunov optimization approach for green cellular networks with hybrid energy supplies," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 12, pp. 2463–2477, Dec. 2015.
- [38] S. Li, J. Huang, and S.-Y. R. Li, "Dynamic profit maximization of cognitive mobile virtual network operator," *IEEE Trans. Mobile Comput.*, vol. 13, no. 3, pp. 526–540, Mar. 2014.
- [39] A. Kumar, D. Manjunath, and J. Kuri, *Wireless Networking*. Burlington, MA, USA: Elsevier, 2008.