IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 6, 2017, accepted April 24, 2017, date of publication April 28, 2017, date of current version June 7, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2699198

Fronthaul Load Balancing in Energy Harvesting Powered Cloud Radio Access Networks

CHENG QIN[1], (Student Member, IEEE), WEI NI[2], (Senior Member, IEEE), HUI TIAN[1], (Member, IEEE), AND REN PING LIU[3], (Senior Member, IEEE)

[1]State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Data61, Commonwealth Scientific and Industrial Research Organization, Sydney, NSW 2122, Australia
[3]Global Big Data Technologies Centre, University of Technology Sydney, Sydney, NSW 2007, Australia

Corresponding author: Hui Tian (tianhui@bupt.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61471057, and in part by the Funds for Creative Research Groups of China under Grant 61421061.

ABSTRACT Enhanced with wireless power transfer capability, cloud radio access network (C-RAN) enables energy-restrained mobile devices to function uninterruptedly. Beamforming of C-RAN has potential to improve the efficiency of wireless power transfer, in addition to transmission data rates. In this paper, we design the beamforming jointly for data transmission and energy transfer, under finite fronthaul capacity of C-RAN. A non-convex problem is formulated to balance the fronthaul requirements of different remote radio heads (RRHs). Norm approximations and relaxations are carried out to convexify the problem to second-order cone programming (SOCP). To improve the scalability of the design to large networks, we further decentralize the SOCP problem using the alternating direction multiplier method (ADMM). A series of reformulations and transformations are conducted, such that the resultant problem conforms to the state-of-the-art ADMM solver and can be efficiently solved in real time. Simulation results show that the distributed algorithm can remarkably reduce the time complexity without compromising the fronthaul load balancing of its centralized counterpart. The proposed algorithms can also reduce the fronthaul bandwidth requirements by 25% to 50%, compared with the prior art.

INDEX TERMS Cloud radio access network (C-RAN), energy harvesting, fronthaul, decentralization.

I. INTRODUCTION

Cloud radio access network (C-RAN) enables cooperation among ubiquitously deployed remote radio heads (RRHs) through baseband units (BBUs) aggregated in a centralized BBU pool, hence allowing centralized signal processing, such as joint beamfoming, mitigating interference and improving users' data rate [1]–[3]. By integrating BBUs, C-RAN can save operating expenses, reduce energy expenditure, and provide flexible network management [4]. It is a promising candidate technique in 5G systems. C-RAN has also been recently considered to implement wireless power transfer, thereby allowing mobile devices with depleted battery to operate uninterruptedly for critical missions [5]. Simultaneous wireless information and power transfer (SWIPT) provides a practical means to transfer power while maintaining data transmission, using power splitting (PS) [6], [7]. More specifically, an RRH is connected to the closest Internet port and power socket, retrieving data from a BBU and energy from the grid. Multiple RRHs in vicinity work cooperatively to construct beams to a number of users, dispatching data while transferring energy.

Limited fronthaul connecting the RRH and the BBU is a critical bottleneck of C-RAN, especially in the case where the RRHs are installed on a plug-and-play basis through existing Internet infrastructure (e.g., Ethernet or ADSL) for ubiquitous, dense deployment.It is possible that some of the RRHs in cooperation serves more users than the others due to the near-far effect. However, uneven demands for fronthaul among the RRHs would lead to an inefficient use of fronthaul and increase operational cost [8]. Balancing the load of the fronthaul is also important to increase the transmit rates of the users, by avoiding congestion in the fronthaul links [8]. This paper develops a new efficient and scalable algorithm to balance the fronthaul load.

Existing works which take limited fronthaul into account have been focused on transmit (Tx) beamformer designs to minimize the energy consumption of C-RAN [9]–[12] or maximize the data rate [13], [14]. Load balancing on the

fronthaul of different RRHs is yet to be investigated [8]. Another challenge lies in the beamformer design in wireless powered C-RAN. Although the beamforming in SWIPT has been proposed for different objectives [15]–[20], existing approaches cannot be directly applied to our problem, due to the constraints of limited fronthaul. In addition, most existing beamforming designs on C-RAN or SWIPT are centralized, such as semidefinite relaxation (SDR) [15]–[17], and cannot meet the scalability requirement of the future cellular networks with substantially increased densities.

In this paper, we propose a holistic design of ubiquitous C-RAN in coupling with wireless energy transfer (WET), which allows mobile users to harvest radio frequency (RF) energy while receiving desired signals. We jointly design the beamformers and the power splitting factors (PSFs) to balance fronthaul workload in an energy harvesting powered C-RAN, where multiple-input single-output (MISO) is considered. Under the constraints on the limited fronthaul capacity between BBU pool and each RRH, we formulate a min-max problem of the fronthaul load, while the quality of service (QoS) and harvested energy of each user are guaranteed. We relax the ℓ_0 norm and convexify the problem into second-order cone programming (SOCP).

We further decentralize solving the SOCP problem by taking an alternating direction multipliers method (ADMM). Non-trivial mathematic manipulations and transformations are conducted to comply the problem with a specialized, efficient ADMM solver, including Smith-form transformation and replica copy generations. Extensive simulations show that the proposed algorithm can substantially reduce the time-complexity without compromising the fairness in fronthaul, as compared to its centralized counterpart. Without requesting extra energy from the power grid, the proposed algorithm can save up to 25% - 50% bandwidth in fronthaul, compared to SDR and greedy solutions.

The rest of the paper is organized as follows. In Section II, we review the related works. In Section III, the system model is introduced. In Section IV, the min-max workload balancing problem is formulated and an iterative centralized algorithm is proposed, followed by a decentralized algorithm in Section V. In Section VI, simulation results are provided, followed by a conclusion in Section VII.

Notations: Use upper/lower case boldface to denote a matrix/vector. $(\cdot)^T$, $(\cdot)^\dagger$ and $\|\cdot\|$ stand for transpose, conjugate transpose and Euclidean norm, respectively. $\otimes$ is the Kronecker product. $\|\cdot\|_p$ denotes p-norm. $\mathcal{CN}(a, b)$ stands for a complex Gaussian distribution with mean a and variance b. blkdiag$\{\mathbf{A}, \mathbf{B}\}$ denotes a block diagonal matrix $\mathbf{A}$ and $\mathbf{B}$. $\mathbf{0}_n$ is the all-zero column-vector with dimension n. $\mathbf{0}_{m\times n}$ and $\mathbf{I}_n$ are the $m \times n$ all-zero matrix and the $n \times n$ identity matrix, respectively.

II. RELATED WORK

Earlier works on C-RAN beamforming under limited fronthaul were focused on energy consumption or sum rate. In [9]–[12], the fronthaul energy was minimized under the QoS constraints. The fronthaul consumption was modeled as ℓ_0 norm and relaxed by using weighted ℓ_1/ℓ_2 norm. Different methods, such as uplink-downlink duality, SDR or branch-and-cut method were employed to tackle the non-convexity of the QoS constraints. In [13] and [14], the sum rate was maximized, under limited fronthaul. The problem was relaxed by using ℓ_0-norm relaxation or conjugate functions, and solved by transforming the problem into the minimization of weighted sum mean square error (WSMSE) [21] in an alternating fashion. In [22], the fairness of QoS among users in C-RAN was considered. The problem was solved by using convex approximation and bisectional search. However, all these works do not consider workload balancing in fronthaul.

Recently, joint beamforming has been considered in SWIPT systems with different objectives. In [15]–[17], the Tx power was minimized, where QoS and energy harvesting requirements were considered. The SDR method was employed to solve the problem. In [18], the same problem was transformed into an SOCP problem. The reformulated problem has a lower complexity than the SDR methods. In [19] and [20], the beamforming design in SWIPT was extended to C-RAN, under finite fronthaul. SDR was employed to convexify the problem and centralized solvers were developed. However, most of the works in SWIPT do not consider non-ideal fronthaul in C-RAN, and these problems were also solved in a centralized manner with limited scalability.

More recently, distributed beamforming designs in C-RAN or SWIPT have been proposed. In [23], the joint beamforming in large scale C-RAN was considered to minimize the fronthaul consumption. The problem was solved by using ADMM and a matrix stuffing method was employed to further enhance the efficiency of the algorithm. In [24], the joint beamforming of multicast in C-RAN was considered. Parallel beamforming design was proposed based on ADMM. In [25], the total data rate of C-RAN was maximized under limited fronthaul capacity. Dual decompositions were employed to decentralize the problem. In [26], distributed beamforming in SWIPT was implemented based on the ADMM. However, these works consider C-RAN or SWIPT separately. Their methods cannot be directly applied into the system under consideration. To the best of our knowledge, there are few works targeting on the workload balance of fronthaul links in wireless powered C-RAN. Let alone the distributed algorithm design under this setting.

III. SYSTEM MODEL

Consider an energy harvesting powered C-RAN with M RRHs and K users in the downlink, as shown in Fig. 1. The RRHs are connected to the BBU pool via fronthaul links. Each RRH is equipped with N_T Tx antennas, and each user has a single antenna. In this paper, we assume that the RRHs receive the same power from BBU pool. The users without persistent power supply and/or running out of battery, harvest energy from the RRHs to power their operations. Note that there can be power consumption due to computations at the BBU pool. However, the BBU has access to persistent power

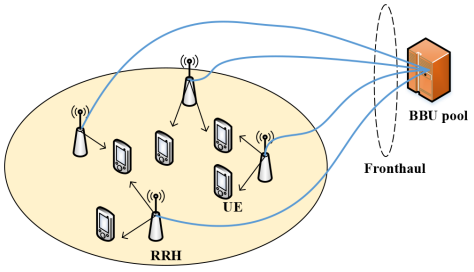


FIGURE 1. An illustration of energy harvesting powered C-RAN. RRH connects to the BBU pool via fronthaul links. Users can receive data while harvesting energy simultaneously.

supplies. For the sake of all users benefit, the amount of power consumed at the BBU pool can be compromised. Each user j has a QoS requirement and a minimum requirement of harvested energy, denoted by Γ_j and E_j, respectively. Particularly, user j employs a PSF $\rho_j \in (0, 1)$ to split the received signal for data recovery and energy harvesting.

The time which is needed to harvest the energy is the same time as the downlink transmission carries on in SWIPT systems, since the energy is harvested from the downlink information transmission. The harvested energy is later used for other operations.

Each RRH i employs a Tx beamformer $\mathbf{v}_{i,j} \in \mathbb{C}^{N_T \times 1}$ to transmit data to user j. Let $\mathbf{v}_j = [\mathbf{v}_{1,j}^\dagger, \cdots, \mathbf{v}_{M,j}^\dagger]^\dagger \in \mathbb{C}^{N \times 1}$ denote the aggregated beamformers of all RRHs for user j, where $N = MN_T$. Let $\mathbf{h}_{i,j} \in \mathbb{C}^{N_T \times 1}$ denote the spatial channel from RRH i to user j, and $\mathbf{h}_j = [\mathbf{h}_{1,j}^\dagger, \cdots, \mathbf{h}_{M,j}^\dagger]^\dagger \in \mathbb{C}^{N \times 1}$ the aggregated channel of user j. Suppose that s_j is the data transmitted for user j, which has zero-mean and unit variance. The data of different users are independent. The received signal at user j can be given by

$$y_j = \sum_{k=1}^{K} \mathbf{h}_j^\dagger \mathbf{v}_k s_k + n_j, \tag{1}$$

where $n_j \sim \mathcal{CN}(0, \sigma_j^2)$ is the additional white Gaussian noise (AWGN) at user j. Here, the signals from multiple RRHs are jointly beamformed to deliver data and energy to the users. As will be shown later in Section VI, joint beamforming can help improve the system gain. More energy can be harvested and fronthaul requirements can be reduced, as the number of RRHs increases.

By using PS, part of the received signal at user j with a factor ρ_j is input for data recovery, while the rest $(1 - \rho_j)$ of the signal is used for energy harvesting. The part of the signal for data recovery is

$$\widetilde{y}_j = \sqrt{\rho_j} y_j + \bar{n}_j, \tag{2}$$

where $\bar{n}_j$ is the additional noise at data recovery and follows $\mathcal{CN}(0, \delta_j^2)$. On the other hand, the harvested energy can be given by

$$P_j^E = (1 - \rho_j)\phi_j \sum_{j=1}^{K} (|\mathbf{h}_j^\dagger \mathbf{v}_k|^2 + \sigma_j^2), \tag{3}$$

where ϕ_j is the power transfer efficiency.

The duration of a beamforming cycle depends on the coherence time of the channel, during which the channel state information (CSI) stays unchanged and so do $\mathbf{v}_{i,j}$, $\forall i, j$ and the beams. The beamforming is accomplished at the beginning of a cycle based on the CSI and the minimum data rate requirements. $\mathbf{v}_{i,j}$ are implemented to form beams for the rest of the cycle.

IV. LOAD BALANCING IN THE FRONTHAUL

Consider the fairness of fronthaul among all RRHs. In other words, the fronthaul consumption needs to be balanced to avoid the fronthaul between a particular RRH and the BBU being overloaded. As such, the bandwidth requirement of fronthaul can be reduced and the fronthaul can be efficiently utilized [8]. Let R_j denote the actual data rate of user j, the fronthaul workload (i.e., required bandwidth of fronthaul) between RRH i and the BBU can be given by

$$D_i(\{\mathbf{v}_j\}) = \sum_{j=1}^{K} R_j \| |\mathbf{v}_{i,j}|^2 \|_0. \tag{4}$$

Note that $\| |\mathbf{v}_{i,j}|^2 \|_0 = 0$, if RRH i does not transmit to user j; or $\| |\mathbf{v}_{i,j}|^2 \| = 1$, if RRH i transmits to user j. In this sense, the ℓ_0-norm is an on-off indicator, and (4) gives the workload (in bits per second) that RRH i transmits.

Our target is to balance the fronthaul workload. We propose to minimize the maximum load of all fronthaul links, under the constraints of QoS and harvested energy.[1] A fronthaul link connects an RRH and the BBU pool. All the bandwidth of this fronthaul link serves the RRH, and is not shared among RRHs. However, the bandwidth is shared among the users which the RRH decides to transmit data to. The problem can be formulated as

$$\min_{\{\mathbf{v}_j\},\{\rho_j\}} \max_i \; D_i(\{\mathbf{v}_j\}), \tag{5a}$$

$$\text{s.t.} \sum_{j=1}^{K} |\mathbf{v}_{i,j}|^2 \le P_i^{\max}, \tag{5b}$$

$$\frac{\rho_j |\mathbf{h}_j^\dagger \mathbf{v}_j|^2}{\rho_j \sum_{k=1,k\neq j}^{K} |\mathbf{h}_j^\dagger \mathbf{v}_k|^2 + \rho_j \sigma_j^2 + \delta_j^2} \ge \Gamma_j, \tag{5c}$$

$$(1 - \rho_j)(\sum_{k=1}^{K} |\mathbf{h}_j^\dagger \mathbf{v}_k|^2 + \sigma_j^2) \ge \frac{E_j}{\phi_j}, \tag{5d}$$

$$0 < \rho_j < 1, \tag{5e}$$

where $P_i^{\max}$ is the maximum Tx power of RRH i; Γ_j and E_j are the QoS and energy harvesting thresholds required by each user j, respectively. (5b) is the Tx power constraint

[1] Although fairness models are typically formulated as a max-min problem with the objective being a utility or gain, such as data rate, our fairness model is formulated to be a min-max problem with the object being a cost, i.e., fronthaul bandwidth consumption. To this end, these fairness models share the same ground. These min-max fairness has been adopted in many works with mean square error (MSE) [27] and power consumption [28] as objectives.

at RRH i; (5c) is the QoS requirement at each user; (5d) specifies the energy that a user j needs to harvest to fulfill its QoS or data rate requirement; (5e) specifies the feasible region of the PSF.

Note that a set of K constraints on individual users is far more stringent than a constraint on the total harvested energy of all K users. Particularly, the constraints on individual users provide a sufficient condition of the constraint on all K users. Consider that each user has a stringent requirement on harvested energy, and so the total requirement of all K users can be specified. The solution under the constraints on individual users satisfies the constraint on all K users and fulfills the requirement of each individual user, but not the other way around.

To proceed, we introduce an auxiliary variable t to denote the upper bound of all fronthaul requirements, i.e., $D_i(\{\mathbf{v}_j\}) \leq t, \forall i$. (5) can be rewritten as

$$\min_{\{\mathbf{v}_j\},\{\rho_j\},t} \quad t, \tag{6a}$$

$$\text{s.t. } D_i(\{\mathbf{v}_j\}) \leq t, \quad \forall i, \tag{6b}$$

$$(5b) - (5e), \tag{6c}$$

Apparently, (6) is non-convex due to constraints (5c) and (5d), and is also NP-hard due to the coupling of the discrete ℓ_0-norm constraint (6b). To address this, we first approximate the ℓ_0 norm in (6b) by using a conjugate function and ℓ_1-norm relaxation. Then, we transform the problem to SOCP. To further enhance the sparsity of the solution, i.e., forcing each beamformer to be sparse (with more zero entries), we adopt a link removal method to generate sparser beamformers, which can force more $|\mathbf{v}_{i,j}|$ to be zero.

A. ℓ_0-NORM RELAXATION

Two ℓ_0-norm relaxation techniques can be employed to address the non-smooth constraint (6b). First, we adopt the conjugate method [14] to approximate the ℓ_0-norm. Particularly, the ℓ_0-norm can be approximated by an exponential function [29], as given by

$$\||\mathbf{v}_{i,j}|^2\|_0 \approx 1 - \exp(-\frac{|\Xi_i \mathbf{v}_j|^2}{2\vartheta^2}), \tag{7}$$

where $\Xi_i = [\mathbf{0}_{N_T\times(i-1)N_T}, \mathbf{I}_{N_T}, \mathbf{0}_{N_T\times(M-i)N_T}] \in \mathbb{C}^{N_T\times N}$ is a selection matrix. Provided that $\vartheta \to 0$, (7) approaches to one when $|\Xi_i \mathbf{v}_j|^2 > 0$ and otherwise it approaches to zero [29]. However, the exponential approximation (7) is concave in $\mathbf{v}_j$. Hence, we convexify (7) using its conjugate function [30], as given by

$$g_i(\{\mathbf{v}_j\}) = \inf_{u_{i,j}} \left\{ u_{i,j}|\Xi_i \mathbf{v}_j|^2 - g_i^*(\{u_{i,j}\}) \right\}, \tag{8}$$

where $g_i(\{\mathbf{v}_j\})$ is the right-hand side (RHS) of (7) and $u_{i,j}$ is an introduced weight. $g_i^*(\cdot)$ is the conjugate function of $g_i(\cdot)$, as given by

$$g_i^*(\{u_{i,j}\}) = 2\vartheta^2 u_{i,j}\{1 - \log(2\vartheta^2 u_{i,j})\} - 1. \tag{9}$$

It can be verified that $g_i^*(\{u_{i,j}\})$ is concave, thus the RHS of (8) is convex with respect to (w.r.t.) $u_{i,j}$. Therefore, $u_{i,j}$ can be obtained by using the first-order derivative, as given by

$$u_{i,j} = \frac{\exp\left(-\frac{|\Xi_i \mathbf{v}_j|^2}{2\vartheta^2}\right)}{2\vartheta^2}. \tag{10}$$

As a result, given $\{u_{i,j}\}$, we can rewrite (6), as given by

$$\min_{\{\mathbf{v}_j\},\{\rho_j\},t} \quad t, \tag{11a}$$

$$\text{s.t. } \sum_{j=1}^{K} u_{i,j} R_j |\Xi_i \mathbf{v}_j|^2 - \sum_{j=1}^{K} R_j g_i^*(\{u_{i,j}\}) \leq t, \tag{11b}$$

$$(5b) - (5e), \tag{11c}$$

Given $\{u_{i,j}\}$, we can solve (11) and then update $\{u_{i,j}\}$ using the obtained $\{\mathbf{v}_j\}$ in sequel, until convergence.

We can also employ the weighted ℓ_1-norm relaxation method to convexify the constraint (6b) [10]. To this end, $D_i\{\mathbf{v}_j\}$ can be approximated as given by

$$D_i(\{\mathbf{v}_j\}) = \sum_{j=1}^{K} R_j \theta_{i,j} |\Xi_i \mathbf{v}_j|^2, \tag{12}$$

where $\theta_{i,j}$ is the weight of the ℓ_1-norm of $\Xi_i \mathbf{v}_j$, and can be updated by

$$\theta_{i,j} = \frac{1}{|\Xi_i \mathbf{v}_j|^p + \epsilon}, \tag{13}$$

where $\epsilon \ll 1$ is a small positive parameter to retune the sparsity, and p is a predefined value. However, the convergence of this relaxation cannot be guaranteed [13], [14], since the objective is not monotonic.

Remark 1: Suppose that the weighted ℓ_1-norm relaxation is employed. From (13), we see the weight $\theta_{i,j}$ is small at the beginning of the algorithm, since $\mathbf{v}_{i,j}$ is not zero. This can lead to an inaccurate approximation in (12) and the objective will increase at the first few iterations, as shown in Section VI. Therefore, when $\theta_{i,j}$ becomes large, $\mathbf{v}_{i,j}$ is unnecessarily forced to zero to meet (6b). In this case, the objective may not further be reduced due to the support for QoS and energy harvesting constraints. This can lead to a larger objective value, as compared to using the conjugate function method, as corroborated in simulations in Section VI.

B. SOCP BASED ALGORITHM

For illustration purpose, we consider the conjugate function method to relax (6b). However, the algorithm based on the weighted ℓ_1-norm relaxation can be easily extended. We convexify (5c) and (5d) by reformulating them into second-order cone (SOC) form. Particularly, (5c) can be rearranged as

$$\sum_{k=1,k\neq j}^{K} |\mathbf{h}_j^\dagger \mathbf{v}_k|^2 + \sigma_j^2 \leq \frac{|\mathbf{h}_j^\dagger \mathbf{v}_j|^2}{\Gamma_j} - \frac{\delta_j^2}{\rho_j}, \tag{14}$$

Referring to [18], we substitute $\sum_{k=1,k\neq j}^{K}|\mathbf{h}_j^\dagger\mathbf{v}_k|^2+\sigma_j^2$, using its upper bound, i.e., $\frac{|\mathbf{h}_j^\dagger\mathbf{v}_j|^2}{\Gamma_j}-\frac{\delta_j^2}{\rho_j}$. (11) can be relaxed, as

$$\min_{\{\mathbf{v}_j\},\{\rho_j\},t} \quad t, \tag{15a}$$

$$\text{s.t. } \frac{|\mathbf{h}_j^\dagger\mathbf{v}_j|^2}{\Gamma_j}\geq\sum_{k=1,k\neq j}^{K}|\mathbf{h}_j^\dagger\mathbf{v}_k|^2+\sigma_j^2+\frac{\delta_j^2}{\rho_j}, \tag{15b}$$

$$(1+\frac{1}{\Gamma_j})|\mathbf{h}_j^\dagger\mathbf{v}_j|^2\geq\frac{E_j}{\phi_j(1-\rho_j)}+\frac{\delta_j^2}{\rho_j}, \tag{15c}$$

$$\text{(5b), (5e), (11b).} \tag{15d}$$

The tightness of this relaxation can be proved by extending the discussions in [18]. Moreover, it can be verified that the data rate R_j of each user j in fronthaul must be equal to the QoS requirement, i.e., $R_j=\log_2(1+\Gamma_j)$; otherwise, the consumption of the fronthaul bandwidth would increase. On the other hand, to transform the problem into an SOCP form, we first let $\tilde{t}=\sqrt{t}$, and confirm that the optimality of the problem does not change with this reformulation, i.e., minimizing $\tilde{t}$ is equivalent to the minimization of t. Also note that $g_i^*(\{u_{i,j}\})\leq 0$. Since $u_{i,j}\leq\frac{1}{2\vartheta^2}$ and $g_i^*(\{u_{i,j}\})$ is concave in $u_{i,j}$, we can let $w_i^2=-\sum_{j=1}^{K}R_jg_i^*(\{u_{i,j}\})$.

Let $\varpi_j=\sqrt{\rho_j}$ and $\varrho_j=\sqrt{1-\rho_j}$. $\mathbf{v}=[\mathbf{v}_1^\dagger,\cdots,\mathbf{v}_K^\dagger]^\dagger\in\mathbb{C}^{KN}$ denotes the aggregated Tx beamforming for all users. Further introduce $\varphi_j\geq\frac{\sqrt{E_j}}{\varrho_j\sqrt{\phi_j}}$ and $\psi_j\geq\frac{\delta_j}{\varpi_j}$. We can reformulate (15) into SOCP [18], as given by[2]

$$\min_{\boldsymbol{\mathcal{X}}} \quad \tilde{t}, \tag{16a}$$

$$\text{s.t. } \|\Phi_i\mathbf{v};w_i\|\leq\tilde{t}, \tag{16b}$$

$$\|\widetilde{\Xi}_i\mathbf{v}\|\leq\sqrt{P_i^{\max}}, \tag{16c}$$

$$\|\mathbf{H}_j^\dagger\mathbf{v};\psi_j;\sigma_j\|\leq\Big(1+\frac{1}{\Gamma_j}\Big)\mathbf{h}_j^\dagger\mathbf{v}_j \tag{16d}$$

$$\|\psi_j;\varphi_j\|\leq\Big(1+\frac{1}{\Gamma_j}\Big)\mathbf{h}_j^\dagger\mathbf{v}_j, \tag{16e}$$

$$\Big\|2\sqrt{\delta_j};(\psi_j-\varpi_j)\Big\|\leq\psi_j+\varpi_j, \tag{16f}$$

$$\Big\|2\sqrt{\sqrt{\frac{E_j}{\phi_j}}};(\varphi_j-\varrho_j)\Big\|\leq\varphi_j+\varrho_j, \tag{16g}$$

$$\|\varpi_j;\varrho_j\|\leq 1, \tag{16h}$$

$$\varpi_j>0,\quad \varrho_j>0, \tag{16i}$$

where $\boldsymbol{\mathcal{X}}=\Big\{\tilde{t},\mathbf{v},\{\varpi_j\},\{\varrho_j\},\{\varphi_j\},\{\psi_j\}\Big\}$ collects all variables. $\Phi_i=\text{blkdiag}\{\sqrt{u_{i,1}R_1}\Xi_i,\cdots,\sqrt{u_{i,K}R_K}\Xi_i\}\in\mathbb{C}^{KN_T\times KN}$, $\widetilde{\Xi}_i=\text{blkdiag}\{\underbrace{\Xi_i,\cdots,\Xi_i}_{K}\}\in\mathbb{C}^{KN_T\times KN}$ and $\mathbf{H}_j=\text{blkdiag}\{\underbrace{\mathbf{h}_j^\dagger,\cdots,\mathbf{h}_j^\dagger}_{K}\}^\dagger\in\mathbb{C}^{KN\times K}$. (16b) is from (11b). (16c) corresponds to the power constraint (5b). (16d) and (16e) are from (15b) and (15c). (16f) and (16g) can be obtained from the transformations in [18]. (16h) and (16i) correspond to the constraints of ρ_j.

[2]In this paper, we follow the notations in MATLAB to form a vector/matrix for simplicity, i.e., $[\mathbf{a};\mathbf{b}]=\begin{pmatrix}\mathbf{a}\\\mathbf{b}\end{pmatrix}$

Algorithm 1 Centralized Algorithm

1: **Initialize** all $u_{i,j}=1$ and $s_{i,j}=1$;
2: **repeat**
3: Solve (17) by using SDPT3 solver;
4: Update $u_{i,j}$ by using (10) and update w_j;
5: Set $s_{i,j}=0$ if $\mathbf{v}_{i,j}<\kappa$;
6: **until convergence.**

Note that (16) can be solved by using the standard convex solvers, e.g., SDPT3 [31]. After (16) is solved, we can then update the $u_{i,j}$ in (10), to sparsify $\mathbf{v}_j$.

C. ITERATIVE LINK REMOVAL

We can generate sparse beamformer $\{\mathbf{v}_j\}$ through the ℓ_0-norm relaxation. In practice, however, the sparsity is not always guaranteed (e.g., some parts of $\mathbf{v}_j$ are small but non-zero). We further adopt a simple link removal method to speed up the sparsification.

Let $\mathbf{S}$ be an $M\times K$ binary matrix. Each entry of $\mathbf{S}$, $s_{i,j}=1$ if the link between RRH i to user j is active, i.e., $|\mathbf{v}_{i,j}|^2>0$; otherwise, $s_{i,j}=0$. We can set $s_{i,j}=0$ if $|\mathbf{v}_{i,j}|<\kappa$, where κ is a predefined threshold to adjust the sparsity; and otherwise $s_{i,j}=1$. As a result, we remove those links which are nearly deactivated.

We introduce a group of constraints on problem (16) to push $\mathbf{v}_{i,j}$ towards zero, as given by

$$\min_{\boldsymbol{\mathcal{X}}} \quad \tilde{t}, \tag{17a}$$

$$\text{s.t. (16b)}-\text{(16i)}, \tag{17b}$$

$$\|\Xi_i\mathbf{v}_j\|\leq\sqrt{s_{i,j}P_i^{\max}},\ \forall i,j. \tag{17c}$$

The problem is still SOCP, which can be solved by using a centralized solver.

D. CONVERGENCE AND COMPLEXITY

The proposed algorithm can be summarized in Algorithm 1. From [14], Algorithm 1 converges since the objective is non-increasing. $\boldsymbol{\mathcal{X}}^{(n)}$ denotes the feasible region of the problem (16) in the n-th iteration. If $\{\mathbf{v}_j^*\}^{(n)}$ is the optimal solution for (16), $\{u_{i,j}\}^{(n)}$ is updated based on $\{\mathbf{v}_j^*\}^{(n)}$ (which are used for the $(n+1)$-th iteration). If we substitute $\{\mathbf{v}_j^*\}^{(n)}$ and $\{u_{i,j}\}^{(n)}$ into (11b), these constraints can be verified to still hold. Therefore, $\{\mathbf{v}_j^*\}^{(n)}\in\boldsymbol{\mathcal{X}}^{(n+1)}$. The output of (16) in each loop is non-increasing.

The complexity of solving each SOCP problem (16) (or (17)) is $\mathcal{O}\Big((KN)^{3.5}\Big)$ [18]. Therefore, the overall complexity of Algorithm 1 is $\mathcal{O}\Big(\log(1/\varkappa)(KN)^{3.5}\Big)$, where $\varkappa$ is the

desired accuracy. Unfortunately, this centralized algorithm does not scale well as the network increases. A distributed algorithm is required to enhance the scalability of the algorithm. ADMM [32], [33] has been widely employed in convex optimizations for decentralization. Therefore, we apply ADMM to (16) and propose to solve this problem in a parallel fashion.

V. DECENTRALIZED ALGORITHM BASED ON ADMM

ADMM is an efficient method to decentralize a convex problem into parallel subproblems. The details of ADMM is provided in Appendix A. In this section, we use the method in [23] to transform the problem (16) into an ADMM-compliant form, which can be solved by extending the recently developed, advanced homogeneous self-dual embedding (HSE) method in [32].

A. ADMM FORM REFORMULATION

In this section, we convert the inequalities constraints of (16) to equalities, one by one, to comply with the input to ADMM. We define a unified variable $\mathbf{x} = [\mathbf{v}; \boldsymbol{\psi}; \boldsymbol{\varphi}; \boldsymbol{\varpi}; \boldsymbol{\varrho}] \in \mathbb{C}^L$ containing all variables, where $\boldsymbol{\psi} = [\psi_1, \cdots, \psi_K]^T$, $\boldsymbol{\varphi} = [\varphi_1, \cdots, \varphi_K]^T$, $\boldsymbol{\varpi} = [\varpi_1, \cdots, \varpi_K]^T$, $\boldsymbol{\varrho} = [\varrho_1, \cdots, \varrho_K]^T$, and $L = KN + 4K$. We use $\mathcal{Q}^m$ $(m > 1)$ to denote a SOC, $\left\{(t, \mathbf{z}) \in \mathcal{Q}^m \middle| \|\mathbf{z}\| \le t, \mathbf{z} \in \mathbb{C}^{m-1}\right\}$,[3] where $(t, \mathbf{z}) \in \mathcal{Q}^m$ is known as the Smith form of a SOC [23]. For the fronthaul constraints (16b), we can rewrite the SOC form of (16b) in the Smith form, as given by

$$\left(\tilde{t}, \widetilde{\Phi}_i \mathbf{x} + \mathbf{w}_i\right) \in \mathcal{Q}^{KN_T+2}, \tag{18}$$

where $\widetilde{\Phi}_i = \left[\text{blkdiag}\{\widetilde{\Phi}_i^1, \cdots, \widetilde{\Phi}_i^K\}, \mathbf{0}_{KN_T \times 4K}; \mathbf{0}_L^T\right] \in \mathbb{C}^{(KN_T+1)\times L}$, $\widetilde{\Phi}_i^j = \sqrt{u_{i,j} R_j}\, \Xi_i$ and $\mathbf{w}_j = [\mathbf{0}_{KN_T}; w_j]$. Then we define an auxiliary cone $\boldsymbol{\lambda}_1^i \in \mathcal{Q}^{KN_T+2}$ as a replicate of the cone (18), and transform (16b) into the following equality constraints:

$$\text{C1: } \mathbf{M}_i \begin{bmatrix} \tilde{t} \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\lambda}_1^i = \begin{bmatrix} 0 \\ \mathbf{w}_i \end{bmatrix}, \tag{19}$$

where $\mathbf{M}_i = \text{blkdiag}\{-1, -\widetilde{\Phi}_i\}$.

For the per-RRH power constraint (16b), we introduce $y_0^i = \sqrt{P_i^{\max}}$, and the SOC form of (16c) can be written as

$$\left(y_0^i, \mathbf{B}_i \mathbf{x}\right) \in \mathcal{Q}^{KN_T+1}, \tag{20}$$

where $\mathbf{B}_i = \left[\widetilde{\Xi}_i, \mathbf{0}_{KN_T \times 4K}\right] \in \mathbb{C}^{KN_T \times L}$. By introducing a cone denoted by $\boldsymbol{\lambda}_2^i \in \mathcal{Q} \times \mathcal{Q}^{KN_T+1}$, which is the Cartesian of two SOCs (i.e., y_0^i and (20)), the power constraint (16c) can be rewritten as

$$\text{C2: } \mathbf{P}_i \begin{bmatrix} y_0^i \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\lambda}_2^i = \begin{bmatrix} \sqrt{P_i^{\max}} \\ \mathbf{0}_{KN_T+1} \end{bmatrix}, \tag{21}$$

where $\mathbf{P}_i = \left[[1, \mathbf{0}_L^T]; \text{blkdiag}\{-1, -\mathbf{B}_i\}\right]$.

[3] If $m = 1$, then it becomes a non-negative constraint

The QoS constraint (16d) can be transformed into

$$\left(t_0^j, \mathbf{C}_j \mathbf{x} + \mathbf{g}_j\right) \in \mathcal{Q}^{K+3}, \tag{22}$$

where $t_0^j = \beta_j \bar{\mathbf{h}}_j^\dagger \mathbf{x}$, $\beta_j = \sqrt{1 + \frac{1}{\Gamma_j}}$, $\mathbf{C}_j^1 = [\mathbf{H}_j^\dagger, \mathbf{0}_{K\times 4K}] \in \mathbb{C}^{K\times L}$, and $\mathbf{C}_j = [\mathbf{C}_j^1; \mathbf{e}_L^{NK+j}; \mathbf{0}_L^T] \in \mathbb{C}^{(K+2)\times L}$. Here, we define $\mathbf{e}_L^i$ as an element row vector, which indicates that the i-th element of a vector $\mathbf{e} \in \mathbb{C}^{1\times L}$ is 1, while the remaining parts are all zero. $\mathbf{g}_j = [\mathbf{0}_{K+1}; \sigma_j]$. $\bar{\mathbf{h}}_j = [\mathbf{0}_{(j-1)N}; \mathbf{h}_j; \mathbf{0}_{(K-j)N+4K}] \in \mathbb{C}^L$. Likewise, we further let $\boldsymbol{\pi}_1^j \in \mathcal{Q} \times \mathcal{Q}^{K+3}$ as a copy of t_0^j and the SOC of QoS constraint, and reformulate the new constraint:

$$\text{C3: } \mathbf{Q}^j \begin{bmatrix} t_0^j \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\pi}_1^j = \mathbf{q}^j, \tag{23}$$

where $\mathbf{Q}^j = \left[[1, -\beta_j \bar{\mathbf{h}}_j^\dagger]; \text{blkdiag}\{-1, -\mathbf{C}_j\}\right]$ and $\mathbf{q}^j = [\mathbf{0}_2; \mathbf{g}_j]$.

As for the harvested energy constraint (16e), we also define $\tilde{t}_0^j = \beta_j \bar{\mathbf{h}}_j^\dagger \mathbf{x}$. The Smith form of (16e) can be given by

$$\left(\tilde{t}_0^j, \mathbf{D}_j \mathbf{x}\right) \in \mathcal{Q}^3, \tag{24}$$

where $\mathbf{D}_j = [\mathbf{e}_L^{NK+j}; \mathbf{e}_L^{(N+1)K+j}] \in \mathbb{C}^{2\times L}$. Introduce $\boldsymbol{\pi}_2^j \in \mathcal{Q} \times \mathcal{Q}^3$. We can rewrite (16e) as given by

$$\text{C4:} \widetilde{\mathbf{Q}}^j \begin{bmatrix} \tilde{t}_0^j \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\pi}_2^j = \mathbf{0}_4, \tag{25}$$

where $\widetilde{\mathbf{Q}}^j = \left[[1, -\beta_j \bar{\mathbf{h}}_j^\dagger]; \text{blkdiag}\{-1, -\mathbf{D}_j\}\right]$.

For (16f) and (16g), we let $\bar{s}_0^j = \bar{\mathbf{e}}_j^T \mathbf{x}$ and $\tilde{s}_0^j = \tilde{\mathbf{e}}_j^T \mathbf{x}$, respectively, where $\bar{\mathbf{e}}_j = [\mathbf{0}_{NK+j-1}; 1; \mathbf{0}_{2K-1}; 1; \mathbf{0}_{2K-j}]$ and $\tilde{\mathbf{e}}_j = [\mathbf{0}_{(N+1)K+j-1}; 1; \mathbf{0}_{2K-1}; 1; \mathbf{0}_{K-j}]$. Then (16f) and (16g) can be rewritten, respectively, as

$$\left(\bar{s}_0^j, \bar{\mathbf{E}}_j \mathbf{x} + \bar{\mathbf{g}}_j\right) \in \mathcal{Q}^3, \tag{26}$$

and

$$\left(\tilde{s}_0^j, \widetilde{\mathbf{E}}_j \mathbf{x} + \tilde{\mathbf{g}}_j\right) \in \mathcal{Q}^3, \tag{27}$$

where $\bar{\mathbf{E}}_j = [\mathbf{0}_{NK+(j-1)}^T, 1, \mathbf{0}_{2K-1}^T, -1, \mathbf{0}_{2K-j}^T; \mathbf{0}_L^T] \in \mathbb{C}^{2\times L}$, $\bar{\mathbf{g}}_j = [0; 2\sqrt{\delta_j}]$, $\widetilde{\mathbf{E}}_j = [\mathbf{0}_{(N+1)K+j-1}^T, 1, \mathbf{0}_{2K-1}^T, -1, \mathbf{0}_{K-j}^T; \mathbf{0}_L^T] \in \mathbb{C}^{2\times L}$, and $\tilde{\mathbf{g}}_j = [0; 2\sqrt{\sqrt{\frac{E_j}{\phi_j}}}]$. Also introduce two groups of auxiliary cones, $\boldsymbol{\pi}_3^j \in \mathcal{Q} \times \mathcal{Q}^3$ and $\boldsymbol{\pi}_4^j \in \mathcal{Q} \times \mathcal{Q}^3$, then we can rewrite (16f) and (16g) into the following new constraints:

$$\text{C5: } \bar{\mathbf{Y}}^j \begin{bmatrix} \bar{s}_0^j \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\pi}_3^j = \bar{\mathbf{y}}^j, \tag{28}$$

where $\bar{\mathbf{Y}}^j = \left[[1, -\bar{\mathbf{e}}_j]; \text{blkdiag}\{-1, -\bar{\mathbf{E}}_j\}\right]$, $\bar{\mathbf{y}}^j = [0; \bar{\mathbf{g}}_j]$, and

$$\text{C6: } \widetilde{\mathbf{Y}}^j \begin{bmatrix} \tilde{s}_0^j \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\pi}_4^j = \tilde{\mathbf{y}}^j, \tag{29}$$

where $\widetilde{\mathbf{Y}}^j = \left[[1, -\tilde{\mathbf{e}}_j]; \text{blkdiag}\{-1, -\widetilde{\mathbf{E}}_j\}\right]$ and $\tilde{\mathbf{y}}^j = [0; \tilde{\mathbf{g}}_j]$.

For the PSF constraint (16h), let $\hat{s}_0^j = 1$, and it can be transformed into

$$\left(\hat{s}_0^j, \widehat{\mathbf{E}}_j \mathbf{x}\right) \in \mathcal{Q}^3, \tag{30}$$

where $\widehat{\mathbf{E}}_j = [\mathbf{e}_L^{(N+2)K+j}; \mathbf{e}_L^{(N+3)K+j}] \in \mathbb{C}^{2\times L}$. Further let $\boldsymbol{\pi}_5^j \in \mathcal{Q} \times \mathcal{Q}^3$, and we can rewrite (16h) as

$$\text{C7: } \widehat{\mathbf{Y}}^j \begin{bmatrix} \hat{s}_0^j \\ \mathbf{x} \end{bmatrix} + \boldsymbol{\pi}_5^j = \hat{\mathbf{y}}^j, \tag{31}$$

where $\widehat{\mathbf{Y}}^j = \left[[1, \mathbf{0}_L^T]; \text{blkdiag}\{-1, -\widehat{\mathbf{E}}_j\}\right]$ and $\hat{\mathbf{y}}^j = [1; \mathbf{0}_3]$.

For the constraints (16i), we define two group replicates of variables $\{\varpi_j\}$ and $\{\varphi_j\}$, i.e., $\{\pi_6^j\}$ and $\{\tilde{\pi}_6^j\}$, and rewrite (16i), as given by

$$\text{C8: } \begin{cases} -\mathbf{e}_L^{(N+2)K+j}\mathbf{x} + \pi_6^j = 0, \\ -\mathbf{e}_L^{(N+3)K+j}\mathbf{x} + \tilde{\pi}_6^j = 0. \end{cases} \tag{32}$$

As a result of the above reformulations, we now have the conic equality constraints C1-C8. The problem (16) can be reformulated into an ADMM compliant form, i.e.,

$$\begin{aligned} &\min_{\widetilde{\mathbf{x}}, \boldsymbol{\mu}} \ \mathbf{c}^T \widetilde{\mathbf{x}}, \\ &\text{s.t. } \mathbf{A}\widetilde{\mathbf{x}} + \boldsymbol{\mu} = \mathbf{b}, \\ &\quad (\widetilde{\mathbf{x}}, \boldsymbol{\mu}) \in \mathbb{C}^n \times \mathcal{K}. \end{aligned} \tag{33}$$

where $\mathbf{A}$ collects the coefficients on the LHS of C1–C8, $\mathbf{b}$ collects the parameters on the RHS of C1–C8, $\widetilde{\mathbf{x}} = [\tilde{t}; \mathbf{y}_0; \hat{\mathbf{s}}_0; \mathbf{t}_0; \tilde{\mathbf{t}}_0; \bar{\mathbf{s}}_0; \tilde{\mathbf{s}}_0; \mathbf{x}] \in \mathbb{C}^n$, $\mathbf{c} = [1; \mathbf{0}_{n-1}]$, $n = 1 + M + 5K + L$. The introduced auxiliary variables are denoted in vectors, as given by

$$\begin{aligned} \mathbf{y}_0 &= [y_0^1, \cdots, y_0^M]^T, \quad \mathbf{t}_0 = [t_0^1, \cdots, t_0^K]^T, \\ \tilde{\mathbf{t}}_0 &= [\tilde{t}_0^1, \cdots, \tilde{t}_0^K]^T, \quad \hat{\mathbf{s}}_0 = [\hat{s}_0^1, \cdots, \hat{s}_0^K]^T, \\ \bar{\mathbf{s}}_0 &= [\bar{s}_0^1, \cdots, \bar{s}_0^K]^T, \quad \tilde{\mathbf{s}}_0 = [\tilde{s}_0^1, \cdots, \tilde{s}_0^K]^T. \end{aligned}$$

In (33), $\boldsymbol{\mu}$ aggregates all the auxiliary cones in C1 to C8, which is a cone $\mathcal{K}$ with larger dimensions (here, $\mathcal{K}$ stands for the feasible conic set of $\boldsymbol{\mu}$) consists of the following SOCs:

$$\begin{aligned} \mathcal{K} = &\underbrace{\mathcal{Q}^1 \times \cdots \times \mathcal{Q}^1}_{M+7K} \times \underbrace{\mathcal{Q}^{KN_T+2} \times \cdots \times \mathcal{Q}^{KN_T+2}}_{M} \\ &\times \underbrace{\mathcal{Q}^{KN_T+1} \times \cdots \times \mathcal{Q}^{KN_T+1}}_{M} \times \underbrace{\mathcal{Q}^3 \times \cdots \times \mathcal{Q}^3}_{K} \\ &\times \underbrace{\mathcal{Q}^{K+3} \times \cdots \times \mathcal{Q}^{K+3}}_{K} \times \underbrace{\mathcal{Q}^3 \times \cdots \times \mathcal{Q}^3}_{3K}, \end{aligned} \tag{34}$$

the length of the cone is

$$\begin{aligned} m = &M + 7K + (KN_T + 2)M + (KN_T + 1)M \\ &+ K(K + 3) + 4K \times 3. \end{aligned} \tag{35}$$

B. REFORMULATION OF A AND b

In (33), $\mathbf{A}$ and $\mathbf{b}$ depend on the coefficients in the equality constraints C1 to C8. Some of the coefficients need to be updated while solving (33), e.g., $\widetilde{\Phi}_i^j$ in (C1), slowing down the convergence of the algorithm. We propose to reformulate $\mathbf{A}$ and $\mathbf{b}$ based on the method in [23], to speed up updating these coefficients and in turn, the convergence of the entire algorithm. Let $\mathbf{F}_1 \triangleq [\beta_1 \bar{\mathbf{h}}_1^\dagger; \cdots; \beta_K \bar{\mathbf{h}}_K^\dagger]$, $\mathbf{F}_2 \triangleq [\bar{\mathbf{e}}_1; \cdots; \bar{\mathbf{e}}_K]$ and $\mathbf{F}_3 \triangleq [\tilde{\mathbf{e}}_1; \cdots; \tilde{\mathbf{e}}_K]$, we define $\mathbf{F} \triangleq [\mathbf{0}_{M+K}; \mathbf{F}_1; \mathbf{F}_2; \mathbf{F}_3]$. Let $\Omega_j = [\mathbf{e}_L^{(N+2)K+j}; \mathbf{e}_L^{(N+3)K+j}]$ and $\Omega = [\Omega_1; \cdots; \Omega_K]$. Let $\tilde{\mathbf{f}} \triangleq [-1; \mathbf{0}_{KN_T+1}] \otimes \mathbf{1}_M$ and $\bar{\mathbf{F}}_i \triangleq [\mathbf{0}_L^T; \widetilde{\Phi}_i]$, we define $\bar{\mathbf{F}} \triangleq [\bar{\mathbf{F}}_1; \cdots; \bar{\mathbf{F}}_M]$. We further let $\mathbf{G}_1 \triangleq [1; \mathbf{0}_{KN_T}] \otimes \mathbf{I}_M$, $\mathbf{G}_2 \triangleq [1; \mathbf{0}_2] \otimes \mathbf{I}_K$ and $\mathbf{G}_3 \triangleq [1; \mathbf{0}_{K+2}] \otimes \mathbf{I}_K$, and define a new matrix as given by

$$\mathbf{G} \triangleq \text{blkdiag}\{\mathbf{G}_1, \mathbf{G}_2, \mathbf{G}_3, \mathbf{G}_2, \mathbf{G}_2, \mathbf{G}_2\}.$$

We further stuff the following matrices as given by

$$\begin{aligned} \mathbf{U}_1^i &= [\mathbf{0}_L^T; \mathbf{B}_i]; \quad \mathbf{U}_1 = [\mathbf{U}_1^1; \cdots; \mathbf{U}_1^M]; \\ \mathbf{U}_2^j &= [\mathbf{0}_L^T; \widehat{\mathbf{E}}_j]; \quad \mathbf{U}_2 = [\mathbf{U}_2^1; \cdots; \mathbf{U}_2^K]; \\ \mathbf{U}_3^j &= [\mathbf{0}_L^T; \mathbf{C}_j]; \quad \mathbf{U}_3 = [\mathbf{U}_3^1; \cdots; \mathbf{U}_3^K]; \\ \mathbf{U}_4^j &= [\mathbf{0}_L^T; \mathbf{D}_j]; \quad \mathbf{U}_4 = [\mathbf{U}_4^1; \cdots; \mathbf{U}_4^K]; \\ \mathbf{U}_5^j &= [\mathbf{0}_L^T; \bar{\mathbf{E}}_j]; \quad \mathbf{U}_5 = [\mathbf{U}_5^1; \cdots; \mathbf{U}_5^K]; \\ \mathbf{U}_6^j &= [\mathbf{0}_L^T; \widetilde{\mathbf{E}}_j]; \quad \mathbf{U}_6 = [\mathbf{U}_6^1; \cdots; \mathbf{U}_6^K]; \\ \mathbf{U} &= [\mathbf{U}_1; \mathbf{U}_2; \mathbf{U}_3; \mathbf{U}_4; \mathbf{U}_5; \mathbf{U}_6]. \end{aligned}$$

As a result, we can stuff the matrix $\mathbf{A} \in \mathbb{C}^{m\times n}$ as

$$\mathbf{A} = \begin{pmatrix} \mathbf{0}_{M+5K} & \mathbf{I}_{M+5K} & -\mathbf{F} \\ \mathbf{0}_{2K} & \mathbf{0}_{2K\times(M+5K)} & -\Omega \\ -\tilde{\mathbf{f}} & \mathbf{0}_{(KN+2M)\times(M+5K)} & -\bar{\mathbf{F}} \\ \mathbf{0}_{m_1} & -\mathbf{G} & -\mathbf{U} \end{pmatrix}, \tag{36}$$

where $m_1 = (KN_T + 1)M + K(K + 3) + 4K \times 3$.

Similarly, to stuff the vector $\mathbf{b}$, we also define the following vectors:

$$\begin{aligned} \mathbf{p} &\triangleq [\sqrt{P_1^{\max}}; \cdots; \sqrt{P_M^{\max}}]; \\ \widehat{\mathbf{w}}^i &= [0; \mathbf{w}_i]; \quad \widehat{\mathbf{w}} = [\widehat{\mathbf{w}}^1; \cdots; \widehat{\mathbf{w}}^M]; \\ \widehat{\mathbf{w}}_1^j &= [0; \mathbf{g}_j]; \quad \widehat{\mathbf{w}}_1 = [\widehat{\mathbf{w}}_1^1; \cdots; \widehat{\mathbf{w}}_1^K]; \\ \widehat{\mathbf{w}}_2^j &= [0; \bar{\mathbf{g}}_j]; \quad \widehat{\mathbf{w}}_2 = [\widehat{\mathbf{w}}_2^1; \cdots; \widehat{\mathbf{w}}_2^K]; \\ \widehat{\mathbf{w}}_3^j &= [0; \tilde{\mathbf{g}}_j]; \quad \widehat{\mathbf{w}}_3 = [\widehat{\mathbf{w}}_3^1; \cdots; \widehat{\mathbf{w}}_3^K]; \end{aligned}$$

Then, the new vector $\mathbf{b}$ can be written as given by

$$\mathbf{b} = [\mathbf{p}; \mathbf{1}_K; \mathbf{0}_{6K}; \widehat{\mathbf{w}}; \mathbf{0}_{3K+KN+M}; \widehat{\mathbf{w}}_1; \mathbf{0}_{3K}; \widehat{\mathbf{w}}_2; \widehat{\mathbf{w}}_3]. \tag{37}$$

C. ALGORITHM IMPLEMENTATION

We note that (33) can be efficiently solved by applying the decentralized ADMM techniques. A recent, specialized ADMM solver, namely, the HSE method [32], [34] can be used, given the inputs $\mathbf{A}$, $\mathbf{b}$ and $\mathbf{c}$ to the solver. The details on the method are provided in Appendix B. Particularly, the HSE method combines the primary and dual problems, takes the KKT conditions of the combined problem, and formulates

a set of linear equations. The method can decentralize solving the equations by projecting the variables onto a Cartesian set which be can decoupled into subsets, such that the variables can be decoupled and solved in parallel. The method can also decentralize solving the equations by taking a sparse permuted LDL$^{\mathrm{T}}$ factorization of the projector, such that different parts of the projector can be inverted or multiplied separately and different parts of the variables can be projected efficiently in parallel with significantly reduced dimensions and time-complexity.

Exploiting the HSE method, we propose a new decentralized algorithm to balance the fronthaul load of the energy harvesting powered C-RAN with a significantly reduced time-complexity. The proposed decentralized algorithm can be implemented in three steps, as summarized in Algorithm 2. In Step 1, the BBU collects the parameters such as channel vectors and QoS threshold, initializes the weights $u_{i,j}$ for the convexification of ℓ_0 norm, constructs $\mathbf{A}$ and $\mathbf{b}$ by (36) and (37), and formulates problem (33). In Step 2, given $u_{i,j}$, $\mathbf{A}$ and $\mathbf{b}$, the BBU solves (33) to optimize the beamforming coefficients of the RRHs and users in a distributed manner, using the HSE method [32], [34].

Given the beamforming results of Step 2, the BBU updates $u_{i,j}$ in parallel and $\mathbf{A}$ and $\mathbf{b}$ in Step 3, before restarting Step 2. Specifically, we extract the beamformers $\{\mathbf{v}_j\}$ and update the weights $u_{i,j}$ using (10). Then the coefficients of the fronthaul constraints C1, i.e., Φ_i and $\mathbf{w}_j$, can be updated. Note that after Step 2, only the matrix $\bar{\mathbf{F}}$ in $\mathbf{A}$ and the vector $\widehat{\mathbf{w}}$ in $\mathbf{b}$ change. We can stuff $\bar{\mathbf{F}}$ and $\widehat{\mathbf{w}}$ by extending the method proposed in [23], thereby speeding up constructing $\mathbf{A}$ and $\mathbf{b}$.

The conjugate function method is taken to approximate the ℓ_0 norm in our formulated SOCP problem, as the difference between the weighted ℓ_1 norm and the conjugate function of the ℓ_1-norm approximation of the ℓ_0 norm; see Step 3. The weights of the approximation, $u_{i,j}$, are recursively updated based on the beamforming results of Step 2 to improve the approximation accuracy. Despite the specialized ADMM solver, i.e., the HSE method, is used to decentralize and accelerate updating the beamforming coefficients in Step 2, our transformation of the SOCP problem to comply with the solver in Step 1 is critical and new. Specifically, we transform the constraints into Smith-form and design new auxiliary replica cones to reformulate the inequality constraints as equalities, thereby complying with the input of the solver. In addition, a new link removal method is developed to accelerate the sparsification and convergence of the solution; see Step 3.

In Step 3, we can also incorporate the link removal described in Section IV-C to speed up sparsifying. Specifically, let $\widetilde{\mathbf{A}} = [\mathbf{A}; \widehat{\mathbf{T}}]$ and $\widetilde{\mathbf{b}} = [\mathbf{b}; \mathbf{0}_{n+1}]$, where $\widehat{\mathbf{T}} = [\mathbf{0}_n^T; \mathrm{blkdiag}\{\mathbf{0}_{(1+M+5K)\times(1+M+5K)}, \mathbf{T}, \mathbf{0}_{4K\times 4K}\}] \in \mathbb{C}^{(n+1)\times n}$. $\mathbf{T} \in \mathbb{C}^{KN\times KN}$ adjusts the sparsity of each beamformer, where the entry $t_{ij} = \{0, 1\}$. At the beginning, $t_{ij} = 0, \forall i, j$. If $s_{\alpha,\beta} = 0$, i.e., $\mathbf{v}_{\alpha,\beta}$ is smaller than the threshold, then we update $t_{ij} = 1$, where $(\beta-1)N + (\alpha-1)N_T < i = j \leq (\beta-1)N + \alpha N_T$. As a result, $\widetilde{\mathbf{A}}$ can be updated based on the change of $\widehat{\mathbf{T}}$.

Algorithm 2 Distributed Parallel Algorithm

Step 1:
1.1) **Initialize** all $u_{i,j} = 1$ and form w_j;
1.2) Stuff the matrix $\mathbf{A}$ and $\mathbf{b}$ by using (36) and (37);
1.3) Formulate $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{b}}$ when link removal is adopted;
repeat
Step 2:
2.1) Solve (33) by using HSE method;
Step 3:
3.1) Update $u_{i,j}$ by using (10) and form w_j;
3.2) Update $\mathbf{A}$ and $\mathbf{b}$ with $u_{i,j}$ and w_j;
3.3) When link removal is adopted, update $s_{i,j}$ and $\widehat{\mathbf{T}}$. Then update $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{b}}$;
until convergence.

D. CONVERGENCE AND COMPLEXITY ANALYSIS

In Algorithm 2, the convergence of the inner loop, i.e., the procedures in ADMM, is guaranteed with $\mathcal{O}(1/k)$ convergence rate [23], where k is the number of iterations required in ADMM. On the other hand, the weights $\{u_{i,j}\}$ can be updated by using Algorithm 1, which proves to be convergent in Section IV-D. Therefore, the overall convergence of Algorithm 2 is guaranteed. The main complexity of the ADMM method, i.e., the HSE method, lies in solving a linear system and a cone projection. More details of this complexity can be found in [23] and [32]. In the simulation, we also testify that the proposed distributed algorithm is much faster than its centralized counterpart.

VI. SIMULATION RESULTS

In our simulations, the RRHs and users are uniformly distributed in a square area with sizes of d meters. The system bandwidth is 1 MHz. Each RRH has an identical Tx power budget, denoted by $P^{\max}$ dBm. We assume that each user has the same QoS and energy harvesting requirement, i.e., Γ and E. The large scale fading is $10^{-PL(d_{ij}/20)}\sqrt{\zeta_{ij}\iota_{ij}}$ [35], where $PL(d_{ij}) = 148.1 + 37.6\log_2(d_{ij})$ is the pathloss over the distance of d_{ij} (in kilometers); $\zeta_{ij} = 9$ dBi is the antenna gain. ι_{ij} is the shadowing fading which follows the log-normal distribution with standard deviance of 8 dB. The small scale fading is flat Rayleigh. Let $\sigma_j^2 = \delta_j^2 = -70$ dBm. The energy transfer efficiency $\phi_j = 0.8, \forall j$. $\vartheta = 0.03$ in (10) and $\kappa = 10^{-5}$ in (17). Our simulations run in a 64-bit Windows 7 operating system with a Intel Core i5 2.40 GHz CPU and 8 GB RAM.

Apart from the proposed Algorithms 1 and 2, we also simulate four other algorithms. The first one is the centralized method with weighted ℓ_1-norm relaxation, where the weights of the ℓ_1 norm are updated as (13). The difference of this algorithm and Algorithm 1 is that it relaxes the fronthaul

constraints of (6b) as (12). The new problem can be solved by reformulating SOCP, as done in Algorithm 1. The second algorithm is a heuristic greedy algorithm, where the fronthaul link with the minimum power is removed from every RRH, i.e., $\arg\min_j |\mathbf{v}_{i,j}|$. This repeats until the problem becomes infeasible, and the greedy algorithm terminates. The third algorithm is based on SDR, where we let $\mathbf{V}_j = \mathbf{v}_j\mathbf{v}_j^\dagger$ in (6), and carry out SDR. This method also adopts the conjugate method, as described in Section IV-A, to relax the ℓ_0 norms. However, this algorithm cannot guarantee the Rank-1 solution. Moreover, the sparsity of the solution cannot be improved [11]. As the forth benchmark algorithm, we minimize the total Tx power without the fronthaul constraints.

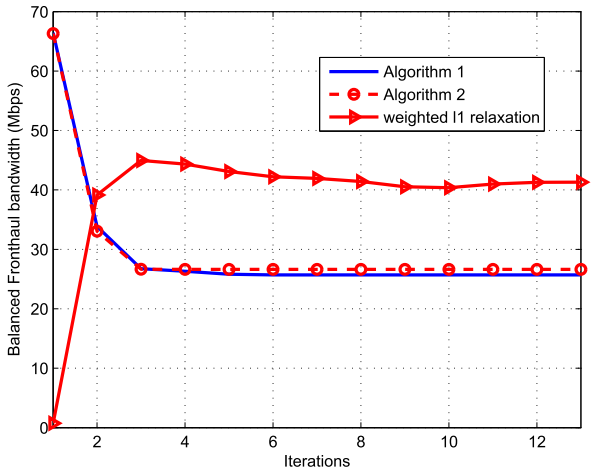


FIGURE 2. Convergence performance of the algorithms. $M = 6$, $K = 10$, $N_T = 2$. $d = 100$ m, $\Gamma = 20$ dB, $E = -30$ dBm. $P^{\max} = 30$ dBm.

Fig. 2 shows the convergence of the proposed algorithms, where the different ℓ_0-norm relaxations are compared. We see that both Algorithms 1 and 2 converge fast within 10 iterations. This is because the objective of the algorithms decreases monotonically. However, the objective under the weighted ℓ_1-norm relaxation increases first and then drops, because the weights $\{\theta_{i,j}\}$ are small at the beginning but increase fast at later stages. The convergence under this relaxation cannot be guaranteed since the objective is not monotonic. Moreover, we see the conjugate method is better than weighted ℓ_1-norm relaxation in terms of approximating ℓ_0 norm, since links are switched off aggressively in weighted ℓ_1-norm relaxation and the fronthaul bandwidth consumption cannot be further reduced in order to support other constraints.

Fig. 3 shows the balanced fronthaul bandwidth with increasing QoS thresholds, where the required amount of harvested energy is set to be -30 dBm. We see that the fronthaul bandwidth grows when QoS becomes tight in the proposed algorithms, because more RRHs needs to participate in joint transmission to meet the high QoS. Hence, the numbers of serving users increase for RRHs. Algorithms 1 and 2 achieve a close performance, since the duality gap between (16) and

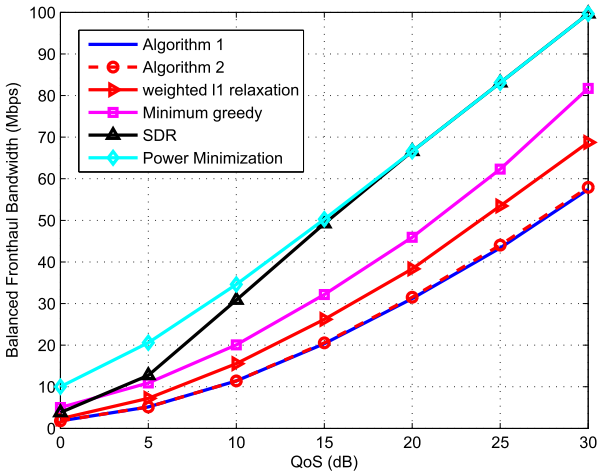


FIGURE 3. Balanced fronthaul bandwidth with increasing QoS requirements. $M = 6$, $K = 10$, $N_T = 2$, $d = 100$ m, $E = -30$ dBm. $P^{\max} = 30$ dBm.

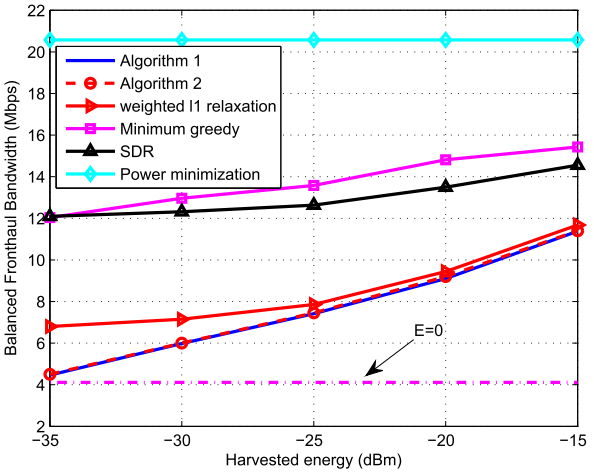


FIGURE 4. Balanced fronthaul bandwidth with increasing harvested energy. $M = 6$, $K = 10$, $N_T = 2$, $d = 100$ m, $\Gamma = 5$ dB. $P^{\max} = 30$ dBm.

its dual problem is zero. ADMM is based on solving the partial dual problem. When the weighted ℓ_1-norm relaxation is adopted, we see that it requires more fronthaul bandwidth than the conjugate relaxation. We also see the greedy algorithm results in higher fronthaul bandwidth requirements, since it minimizes the fronthaul locally and does not jointly optimize beamforming. The SDR method also increases the fronthaul bandwidth requirement, because the sparsity of the solution cannot be improved.

In Fig. 4, we show the balanced fronthaul bandwidth under different energy harvesting requirements, where the QoS is set to be 10 dB. We see the fronthaul bandwidth consumption increases when E grows, since the RRHs have to serve more users and provide the energy for harvesting. Algorithm 2 again has a close performance, as compared with Algorithm 1. The greedy algorithm and SDR method consume much higher fronthaul bandwidth. Interestingly, the gap between the conjugate relaxation and the weighted

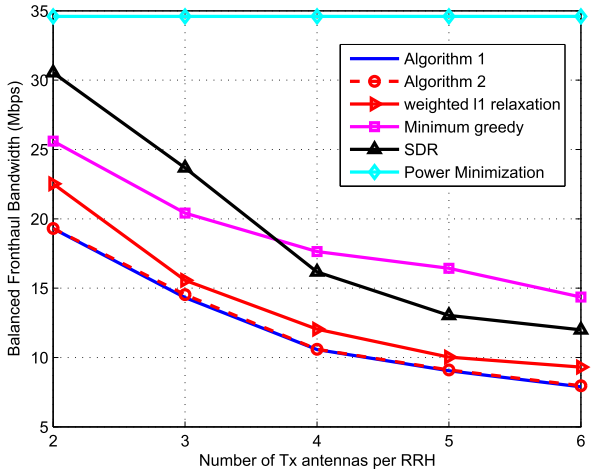


FIGURE 5. Balanced fronthaul bandwidth with growing Tx antennas per RRH. $M = 6$, $K = 10$, $d = 100$ m, $\Gamma = 10$ dB, $E = -20$ dBm. $P^{\max} = 30$ dBm.

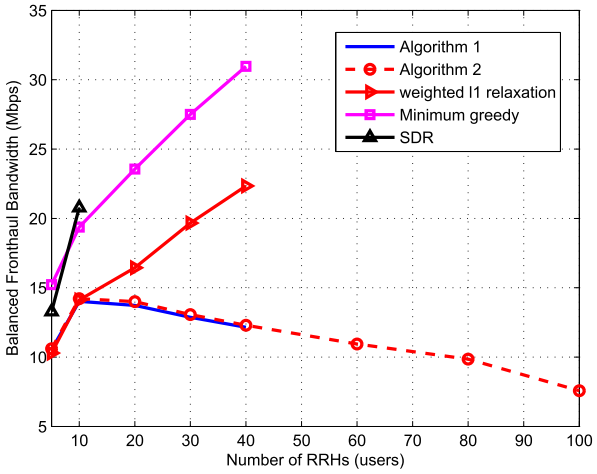


FIGURE 6. Balanced fronthaul bandwidth with increasing scale of the network, where $M = K$. $N_T = 2$, $d = 200$ m, $\Gamma = 10$ dB, $E = -20$ dBm. $P^{\max} = 30$ dBm.

ℓ_1-norm relaxation reduces when E increases. This is because the energy harvesting requirement can allow more links to be activated. Fewer links are switched off unnecessarily. Therefore, the fronthaul consumption can be reduced in a moderate way, similar to the conjugate method employed in Algorithms 1 and 2.

In Fig. 5, we compare the fronthaul bandwidth requirements with the growing number of Tx antennas at each RRH, where $\Gamma = 10$ dB and $E = -20$ dBm. We can see the fronthaul consumption decreases when each RRH has more Tx antennas, because the degree-of-freedom (DoF) of the space increases and the requirements of QoS and harvested energy can be more easily satisfied. In this case, RRHs can switch off those unnecessary links by forcing the corresponding beamfomrers to zero, and the fronthaul workload can be reduced. Algorithms 1 and 2 with the conjugate relaxation again achieve the lowest fronthaul bandwidth.

Next, we increase the scale of the network to examine the scalability of the algorithms. In Fig. 6, we present the

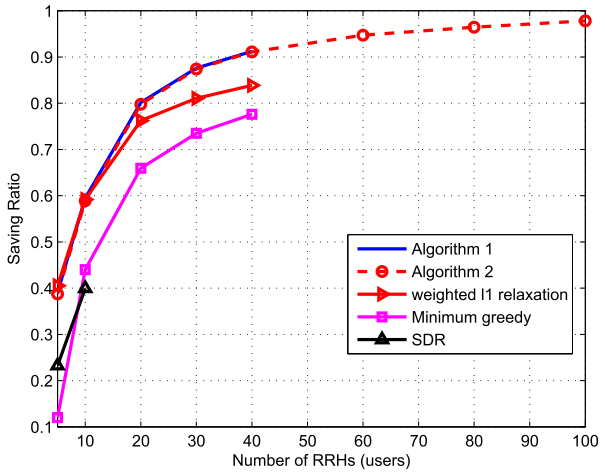


FIGURE 7. Saving ratio with increasing scale of the network, where $M = K$. $N_T = 2$, $d = 200$ m, $\Gamma = 10$ dB, $E = -20$ dBm. $P^{\max} = 30$ dBm.

balanced fronthaul bandwidth, as the scale of the network grows, where we set $M = K$ and $d = 200$ meters. From the figure, we see that the fronthaul bandwidth of the proposed algorithms keep low, i.e., below 15 Mbps. This reveals that the proposed algorithms can efficiently reduce the fronthaul loads and the required bandwidth, as the network enlarges. The QoS and energy requirement can be satisfied in dense networks, and each RRH does not serve too many users. In contrast, the fronthaul bandwidth of other three algorithms are much larger than the proposed algorithms, because the solutions of these algorithms have much lower sparsity, thus serving more links per RRH. We also notice that the weighted ℓ_1-norm relaxation has a close performance to the conjugate method, when $M = K < 10$, i.e., the small-scale network. This is because the weighted ℓ_1-norm relaxation can reduce the fronthaul consumption in a moderate manner, as discussed in Fig. 4. We also see the centralized algorithms, including Algorithm 1, cannot support the network when $M = K > 40$, due to their prohibitive complexities, as will be shown shortly. Algorithm 2 has a better scalability, as the result of parallel implementation.

In Fig. 7, we show the ratio of the fronthaul bandwidth saving, as the scale of the network increases, corresponding to Fig. 6. The ratio is defined between the saved fronthaul bandwidth and the bandwidth in the case that all users are served per RRH. We see that, the fronthaul bandwidth consumption in proposed algorithms is substantially lower (as shown in Fig. 6), while the ratios of the proposed algorithms increases remarkably when the network is large. This demonstrates that the proposed algorithms with the sparse beamforming can reduce the overall fronthaul consumption of the network. The ratios of the other algorithms also increase, as the network scales up. This is because users can be served by more RRHs and the load of each RRH can be reduced.

Fig. 8 plots the relative saving of the fronthaul bandwidth, as the average number of RRHs increases, where the RRHs and users yield two independent spatial Poisson point pro-

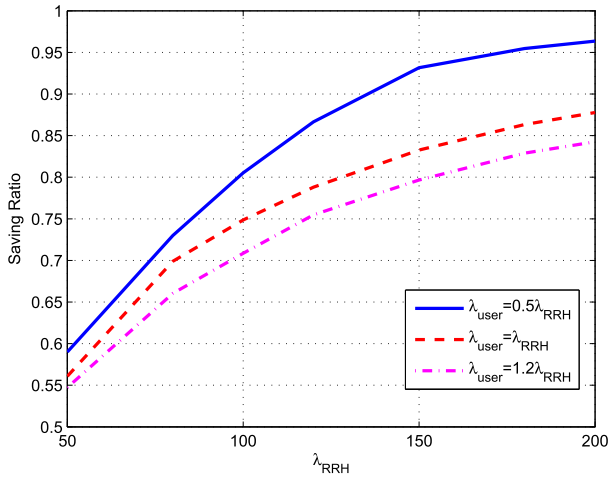


FIGURE 8. Saving ratio with increasing λ_{BS} in Poisson generation, where $N_T = 1$, $d = 500$ m, $\Gamma = 10$ dB, $E = -20$ dBm. $P^{\max} = 30$ dBm.

cesses (SPPPs) with the parameters λ_{RRH} and λ_{user}, respectively. We see that the saving increases with the growth of λ_{RRH}, since the number of RRHs increases and each RRH serves fewer users than it does in Fig. 7. In this sense, a dense deployment of RRHs can help reduce the fronthaul requirement at an RRH. We also see that, when λ_{user} is large, the relative saving of fronthaul bandwidth decreases. This is because an increased number of users to be served per RRH would require higher bandwidth requirements in fronthaul. Moreover, the saving grows much faster in the case of $\lambda_{\text{user}} = 0.5\lambda_{\text{RRH}}$, than it does in the case of $\lambda_{\text{user}} = \lambda_{\text{RRH}}$. This is because the absolute number of RRHs increases twice faster than that of users, thereby increasingly reducing the fronthaul bandwidth requirements.

The proposed decentralized algorithm has the potential to be applied to a large number of RRHs and users, since it decouples computations among different RRHs and users. The running time would grow, as an $(m + n) \times (m + n)$ matrix needs to be inverted for cone projection (as shown in (42) Appendix B) which would increase the complexity as the numbers of RRHs and users increase. Nevertheless, the running time would not substantially grow. The matrix to be inverted becomes increasingly sparse and the non-zero entries become increasingly localized, because there are an increasing number of RRHs with non-overlapping coverage areas (as the network gets bigger). Sparse techniques, such as sparse permute LDL^T decomposition [32], can significantly speed up inverting the matrix through inverting localized sub-matrices in parallel. A localized sub-matrix corresponds to a set of RRHs with an overlapping coverage area and typically involves tens of RRHs.

Our algorithm can be applied to the large-scale C-RAN with hundreds to thousands of RRHs and users, since the decentralized algorithm can be time-efficiently conducted in parallel. Up to 200 RRHs and users are simulated in our laptop based MATLAB platform, given the limited capability of MATLAB and the laptop on which the simulations are

TABLE 1. Average execution time for solving (16) per cycle (in seconds).

M (K)	10	20	40	60	80
SDR	55.1514	N/A	N/A	N/A	N/A
Algorithm 1	17.1294	65.3075	425.8913	N/A	N/A
Algorithm 2	0.9544	4.7430	38.8150	124.8382	371.3327

carried out. It is noteworthy that even though the decentralized algorithm is meant to be implemented in parallel, however MATLAB and the laptop can only support serial computations. The parallel tasks are pipelined and executed in serial, taking far more time than required in a BBU pool.

Finally, we compare the time complexity of the centralized and distributed algorithms in Table 1. Since the ℓ_0-norm approximation is implemented in an iterative manner, without loss of generality, we only select the execution time per cycle, i.e., solving problem (16). From the table, we see Algorithm 2 has a much lower time complexity than the centralized Algorithm 1. As the network continues to grow, Algorithm 2 can support much more RRHs and users than the centralized algorithms.

Table I shows the running time of the proposed algorithms, using MATLAB in the 2.4 GHz laptop. The purpose of the table is to demonstrate the orders-of-magnitude relative reduction of the proposed decentralized algorithm in the running time, as compared to the centralized benchmark. As shown in the table, the running time of the proposed algorithm takes up to hundreds of seconds. This is reasonable since MATLAB is known to be neither optimized nor efficient in terms of processing speed. More importantly, the running time shown here was recorded directly from MATLAB, where the decoupled computing tasks, meant to be accomplished in parallel, were executed in serial, as mentioned earlier. In this sense, the running time here is far higher than it should be, if implemented in parallel.

Note that the 2.4 GHz laptop has a typical processing capability of up to 900 million floating point instructions per second (MFLOPS) [36]. In contrast, specialized BBU hardware, such as digital signal processor (DSP), can have a processing speed of up to 1.6×10^5 MFLOPS (e.g., TI multicore DSP). In this sense, the average running time of the proposed decentralized algorithm would be up to 40 milliseconds (e.g., for 20 RRHs and 20 users), if implemented in specialized BBU hardware. There is still room to further speed up by arithmetically optimizing the algorithms.

VII. CONCLUSION

In this paper, we considered workload balancing between fronthaul links in an energy harvesting powered C-RAN. We balanced the fronthaul load and the required bandwidth consumption under the constraints of QoS and harvested energy. A conjugate method was used to relax the ℓ_0 norm in the problem, and an iterative centralized algorithm was developed based on the reformulation of an SOCP form. We also proposed an ADMM based distributed solver to decentralize the optimization process and reduce the time complexity.

Simulation results show that the distributed algorithm can achieve a close performance to its centralized counterpart, while the time complexity can be remarkably reduced.

APPENDIX A
REVIEW OF ADMM METHOD

For a convex problem as given by

$$\begin{aligned} &\min_{\mathbf{x},\mathbf{z}} f(\mathbf{x}) + g(\mathbf{z}), \\ &\text{s.t. } \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{c}, \\ &\quad \mathbf{x} \in \mathcal{X}, \quad \mathbf{z} \in \mathcal{Z}, \end{aligned} \tag{38}$$

where $\mathbf{x} \in \mathbb{C}^{n_1}, \mathbf{z} \in \mathbb{C}^{n_2}, \mathbf{A} \in \mathbb{C}^{m\times n_1}, \mathbf{B} \in \mathbb{C}^{m\times n_2}$ and $\mathbf{c} \in \mathbb{C}^m$. $f(\cdot)$ and $g(\cdot)$ are convex functions w.r.t. $\mathbf{x}$ and $\mathbf{z}$ respectively. $\mathcal{X}$ and $\mathcal{Z}$ are all convex sets.

The augmented Lagrangian can be given by

$$\begin{aligned} \mathcal{L}_\mu\Big(\mathbf{x}, \mathbf{z}, \mathbf{y}\Big) &= f(\mathbf{x}) + g(\mathbf{z}) + \mathbf{y}^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}) \\ &\quad + \frac{\mu}{2}\|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} - \mathbf{c}\|_2^2, \end{aligned} \tag{39}$$

where μ is a introduced augmented variable. $\mathbf{y} \in \mathbb{C}^m$ denotes the dual variable w.r.t. the coupled constraint.

The Lagrangian can be decomposed to optimize $\mathbf{x}$ and $\mathbf{z}$ independently in a parallel manner, and the dual variable can then be updated by using a subgradient method [30]. As a result, the ADMM is implemented by updating the following variables in an alternative manner:

$$\mathbf{x}^{(n)} = \arg\min_{\mathbf{x}\in\mathcal{X}} \quad \mathcal{L}_\mu\Big(\mathbf{x}, \mathbf{z}^{(n-1)}, \mathbf{y}^{(n-1)}\Big), \tag{40a}$$

$$\mathbf{z}^{(n)} = \arg\min_{\mathbf{z}\in\mathcal{Z}} \quad \mathcal{L}_\mu\Big(\mathbf{x}^{(n)}, \mathbf{z}, \mathbf{y}^{(n-1)}\Big), \tag{40b}$$

$$\mathbf{y}^{(n)} = \mathbf{y}^{(n-1)} + \mu(\mathbf{A}\mathbf{x}^{(n)} + \mathbf{B}\mathbf{z}^{(n)} - \mathbf{c}). \tag{40c}$$

APPENDIX B
A BRIEF INTRODUCTION OF HOMOGENEOUS SELF-DUAL EMBEDDING METHOD

The dual problem of (33) can be given by [32]

$$\begin{aligned} &\max_{\mathbf{y},\boldsymbol{\nu}} \quad -\mathbf{b}^\dagger\mathbf{y}, \\ &\text{s.t. } -\mathbf{A}^\dagger\mathbf{y} + \boldsymbol{\nu} = \mathbf{c}, \\ &\quad (\mathbf{y}, \boldsymbol{\nu}) \in \{0\}^n \times \mathcal{K}^\star, \end{aligned} \tag{41}$$

where $\mathbf{y}$ and $\boldsymbol{\nu}$ are dual variables. $\{0\}^n$ is the dual cone of $\mathbb{C}^n$ and $\mathcal{K}^\star$ is the dual cone of $\mathcal{K}$.

Based on the KKT conditions of (33) and its dual problem, we can introduce two auxiliary variables τ and κ. Let $\mathbf{s} = [\widetilde{\mathbf{x}}; \mathbf{y}; \tau]$ and $\mathbf{t} = [\boldsymbol{\mu}; \boldsymbol{\nu}; \kappa]$, we only need to solve the following problem [32]:

$$\begin{aligned} &\text{find } (\mathbf{s}, \mathbf{t}), \\ &\text{s.t. } \mathbf{t} = \mathbf{Q}\mathbf{s}, \\ &\quad (\mathbf{s}, \mathbf{t}) \in \mathcal{C} \times \mathcal{C}^\star, \end{aligned} \tag{42}$$

where $\mathcal{C} = \mathbb{C}^n \times \mathcal{K}^\star \times \mathbb{C}_+$ and $\mathcal{C}^\star = \{0\}^n \times \mathcal{K} \times \mathbb{C}_+$. And

$$\mathbf{Q} = \begin{pmatrix} \mathbf{0} & \mathbf{A}^\dagger & \mathbf{c} \\ -\mathbf{A} & \mathbf{0} & \mathbf{b} \\ -\mathbf{c}^\dagger & -\mathbf{b}^\dagger & 0 \end{pmatrix}. \tag{43}$$

In order to apply ADMM, (42) is further transformed as given by

$$\begin{aligned} &\min_{\mathbf{s},\mathbf{t},\widetilde{\mathbf{s}},\widetilde{\mathbf{t}}} \mathcal{I}_{\mathcal{C}\times\mathcal{C}^\star}(\mathbf{s}, \mathbf{t}) + \mathcal{I}_{\widetilde{\mathbf{t}}=\mathbf{Q}\widetilde{\mathbf{s}}}(\widetilde{\mathbf{s}}, \widetilde{\mathbf{t}}), \\ &\text{s.t. } (\mathbf{s}, \mathbf{t}) = (\widetilde{\mathbf{s}}, \widetilde{\mathbf{t}}), \end{aligned} \tag{44}$$

where $\mathcal{I}_{\mathcal{Z}}(x)$ is a set indicator function, where $\mathcal{I}_{\mathcal{Z}}(x) = 0$ if $x \in \mathcal{Z}$; otherwise $\mathcal{I}_{\mathcal{Z}}(x) = +\infty$.

By applying ADMM to (44), in the $(k+1)$-th iteration, the variables can be updated as follows [32]:

$$\widetilde{\mathbf{s}}^{k+1} = (\mathbf{I} + \mathbf{Q})^{-1}(\mathbf{s}^{(k)} + \mathbf{t}^{(k)}); \tag{45a}$$

$$\mathbf{s}^{(k+1)} = \Pi_{\mathcal{C}}(\widetilde{\mathbf{s}}^{(k+1)} - \mathbf{t}^{(k)}); \tag{45b}$$

$$\mathbf{t}^{(k+1)} = \mathbf{t}^{(k)} - \widetilde{\mathbf{s}}^{(k+1)} + \mathbf{s}^{(k+1)}, \tag{45c}$$

where $\Pi_{\mathcal{C}}(\cdot)$ is the Euclidean projection on the set $\mathcal{C}$.

REFERENCES

[1] P. Rost *et al.*, "Cloud technologies for flexible 5G radio access networks," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 68–76, May 2014.

[2] V. Jungnickel *et al.*, "The role of small cells, coordinated multipoint, and massive MIMO in 5G," *IEEE Commun. Mag.*, vol. 52, no. 5, pp. 44–51, May 2014.

[3] N. Li, Z. Fei, C. Xing, D. Zhu, and M. Lei, "Robust low-complexity MMSE precoding algorithm for cloud radio access networks," *IEEE Commun. Lett.*, vol. 18, no. 5, pp. 773–776, May 2014.

[4] "C-RAN: The road towards green RAN,"İChina Mobile Research Inst., Beijing, Ver. 2.5, White Paper, 2011.

[5] X. Huang and N. Ansari, "Energy sharing within EH-enabled wireless communication networks," *IEEE Wireless Commun.*, vol. 22, no. 3, pp. 144–149, Jun. 2015.

[6] X. Lu, P. Wang, D. Niyato, D. I. Kim, and Z. Han, "Wireless charging technologies: Fundamentals, standards, and network applications," *IEEE Commun. Surv. Tuts.*, vol. 18, no. 2, pp. 1413–1452, 2nd Quart., 2016.

[7] R. Zhang and C. K. Ho, "MIMO broadcasting for simultaneous wireless information and power transfer," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 1989–2001, May 2013.

[8] C. Ran, S. Wang, and C. Wang, "Balancing backhaul load in heterogeneous cloud radio access networks," *IEEE Wireless Commun.*, vol. 22, no. 3, pp. 42–48, Jun. 2015.

[9] J. Zhao, T. Q. S. Quek, and Z. Lei, "Coordinated multipoint transmission with limited backhaul data transfer," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2762–2775, Jun. 2013.

[10] S. Luo, R. Zhang, and T. J. Lim, "Downlink and uplink energy minimization through user association and beamforming in C-RAN," *IEEE Trans. Wireless Commun.*, vol. 14, no. 1, pp. 494–508, Jan. 2015.

[11] Y. Shi, J. Cheng, J. Zhang, B. Bai, W. Chen, and K. B. Letaief, "Smoothed L_p-minimization for green cloud-RAN with user admission control," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 1022–1036, Apr. 2016.

[12] Y. Cheng, M. Pesavento, and A. Philipp, "Joint network optimization and downlink beamforming for CoMP transmissions using mixed integer conic programming," *IEEE Trans. Signal Process.*, vol. 61, no. 16, pp. 3972–3987, Aug. 2013.

[13] B. Dai and W. Yu, "Sparse beamforming and user-centric clustering for downlink cloud radio access network," *IEEE Access*, vol. 2, pp. 1326–1339, Oct. 2014.

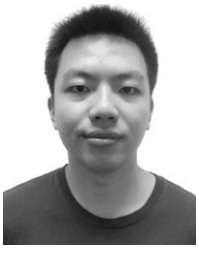
[14] V. N. Ha, L. B. Le, and N. D. Dào, "Coordinated multipoint transmission design for cloud-RANs with limited fronthaul capacity constraints," *IEEE Trans. Veh. Technol.*, vol. 65, no. 9, pp. 7432–7447, Sep. 2016.

[15] Q. Shi, L. Liu, W. Xu, and R. Zhang, "Joint transmit beamforming and receive power splitting for MISO SWIPT systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3269–3280, Jun. 2014.

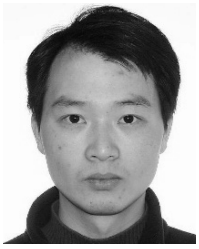
[16] F. Wang, T. Peng, Y. Huang, and X. Wang, "Robust transceiver optimization for power-splitting based downlink MISO SWIPT systems," *IEEE Signal Process. Lett.*, vol. 22, no. 9, pp. 1492–1496, Sep. 2015.

[17] Z. Zong, H. Feng, F. R. Yu, N. Zhao, T. Yang, and B. Hu, "Optimal transceiver design for SWIPT in K-user MIMO interference channels," *IEEE Trans. Wireless Commun.*, vol. 15, no. 1, pp. 430–445, Jan. 2016.

[18] Q. Shi, W. Xu, T. H. Chang, Y. Wang, and E. Song, "Joint beamforming and power splitting for MISO interference channel with SWIPT: An SOCP relaxation and decentralized algorithm," *IEEE Trans. Signal Process.*, vol. 62, no. 23, pp. 6194–6208, Dec. 2014.
[19] Y. Ma, M. Peng, Z. Zhao, and Z. Zhou, "Optimization of simultaneous wireless information and power transfer in cloud radio access networks," in *Proc. IEEE 83rd Veh. Technol. Conf. (VTC Spring)*, May 2016, pp. 1–5.
[20] W. N. S. F. W. Ariffin, X. Zhang, and M. R. Nakhai, "Sparse beamforming for real-time resource management and energy trading in green C-RAN," *IEEE Trans. Smart Grid*, to be published.
[21] S. S. Christensen, R. Agarwal, E. D. Carvalho, and J. M. Cioffi, "Weighted sum-rate maximization using weighted MMSE for MIMO-BC beamforming design," *IEEE Trans. Wireless Commun.*, vol. 7, no. 12, pp. 4792–4799, Dec. 2008.
[22] L. Liu and R. Zhang, "Downlink SINR balancing in C-RAN under limited fronthaul capacity," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 3506–3510.
[23] Y. Shi, J. Zhang, B. O'Donoghue, and K. B. Letaief, "Large-scale convex optimization for dense wireless cooperative networks," *IEEE Trans. Signal Process.*, vol. 63, no. 18, pp. 4729–4743, Sep. 2015.
[24] J. Cheng, Y. Shiy, B. Bai, W. Chen, J. Zhangy, and K. B. Letaief, "Group sparse beamforming for multicast green Cloud-RAN via parallel semidefinite programming," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 1886–1891.
[25] V. N. Ha, D. H. N. Nguyen, and L. B. Le, "Sparse precoding design for cloud-RANs sum-rate maximization," in *Proc. Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2015, pp. 1648–1653.
[26] R. Feng, M. Dai, and H. Wang, "Distributed beamforming in MISO SWIPT system," *IEEE Trans. Veh. Technol.*, to be published.
[27] M. M. Zhao, Y. Cai, B. Champagne, and M. Zhao, "Min-max mse transceiver with switched preprocessing for MIMO interference channels," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2014, pp. 198–202.
[28] J. Choi, "Power allocation for max-sum rate and max-min rate proportional fairness in NOMA," *IEEE Commun. Lett.*, vol. 20, no. 10, pp. 2055–2058, Oct. 2016.
[29] F. Zhuang and V. K. N. Lau, "Backhaul limited asymmetric cooperation for MIMO cellular networks via semidefinite relaxation," *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 684–693, Feb. 2014.
[30] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[31] M. Grant and S. Boyd. (Mar. 2014). *CVX: MATLAB Software for Disciplined Convex Programming, Version 2.1*. [Online]. Available: http://cvxr.com/cvx
[32] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "Conic optimization via operator splitting and homogeneous self-dual embedding," *J. Optim. Theory Appl.*, vol. 169, no. 3, pp. 1042–1068, Jun. 2016. [Online]. Available: http://stanford.edu/ boyd/papers/scs.html
[33] 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.
[34] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd. (Apr. 2016). *SCS: Splitting Sonic Solver, Version 1.2.6*. [Online]. Available: https://github.com/cvxgrp/scs
[35] Y. Shi, J. Zhang, and K. B. Letaief, "Group sparse beamforming for green cloud-RAN," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2809–2823, May 2014.
[36] R. Longbottom. (Feb. 2016). *Roy Longbottom's PC Benchmark Collection*. [Online]. Available: http://www.roylongbottom.org.uk/



CHENG QIN (S'14) received the B.S. degree in communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2011, where he is currently pursuing the Ph.D. degree in information and communications engineering. His research interests include cooperative systems, MIMO systems, energy harvesting, and convex optimizations.



WEI NI (M'09–SM'15) received the B.E. and Ph.D. degrees in electronic engineering from Fudan University, Shanghai, China, in 2000 and 2005, respectively. He was a Post-Doctoral Research Fellow with Shanghai Jiaotong University from 2005 to 2008, a Research Scientist and a Deputy Project Manager with the Bell Labs R&I Center, Alcatel/Alcatel-Lucent, from 2005 to 2008, and a Senior Researcher with Devices Research and Development, Nokia, from 2008 to 2009. He is currently a Senior Scientist, a Team Leader, and a Project Leader with CSIRO, Australia. He also holds adjunct positions with the University of New South Wales, Macquarie University, and the University of Technology Sydney. His research interests include optimization, game theory, graph theory, and their applications to network and security.

Dr. Ni serves as an Editor of *Journal of Engineering* (Hindawi) since 2012, a Secretary of the IEEE NSW VTS Chapter since 2015, a Track Chair of the VTC-Spring 2017, a Track Co-Chair of the IEEE VTC-Spring 2016, and a Publication Chair of the BodyNet 2015. He also served as a Student Travel Grant Chair of WPMC 2014, a Program Committee Member of CHINACOM 2014, a TPC Member of the IEEE ICC'14, the ICCC'15, the EICE'14, and the WCNC'10.

HUI TIAN (M'03) received the M.S. degree in microelectronics and the Ph.D. degree in circuits and systems from the Beijing University of Posts and Telecommunications in 1992 and 2003, respectively. She is currently a Professor with BUPT and the Director of the State Key Laboratory of Networking and Switching Technology. She is also a Committee Member of the Beijing Key Laboratory of Wireless Communication Testing Technology, a Core Member of Innovation Group of the National Natural Science Foundation of China, a member of the China Institute of Communications, and an Expert in the Unified Tolling and Electronic Toll Collection Working Group of the China National Technical Committee on ITS Standardization. Her research interests include LTE and 5G system design, MAC protocols, resource scheduling, cross-layer design, cooperative relaying in cellular systems, and ad hoc and sensor networks.

She was a co-recipient of the National Award for Technological Invention, the Science and Technology Award of China Communications, and ten major scientific and technological progresses Award of China's colleges and Universities for her contribution in the field of wireless communication. She was a Lead Guest Editor of EURASIP *Journal on Wireless Communications and Networking*. She has been a TPC Member for IEEE conferences (GlobalCom, WCNC, WPMC, PIMRC, VTC, ICCC, and so on) and the Reviewer of the IEEE TVT, the IEEE CL, the *IET Communications*, the *Transactions on Emerging Telecommunications Technologies*, the *EURASIP Journal on Wireless Communications and Networking*, the *Journal of Networks*, the *Majlesi Journal of Electrical Engineering*, *The Journal of China University of Posts and Telecommunications*, the *Chinese Journal of Electronics*, the *Journal of Electronics and Information Technology*, and the *Chinese Journal of Aeronautics*.

REN PING LIU (M'09–SM'14) was a Principal Scientist of CSIRO, where he led wireless networking research activities. He is currently a Professor with the School of Computing and Communications, University of Technology Sydney, where he leads Network Security Laboratory. He specializes in protocol design and modeling, and has delivered networking solutions to a number of government agencies and industry customers. He has over 100 research publications, and has supervised over 30 Ph.D. students. His research interests include Markov analysis and QoS scheduling in WLAN, VANET, IoT, LTE, 5G, SDN, and network security.

He received the B.E. (Hons.) and M.E. degrees from the Beijing University of Posts and Telecommunications, China, and the Ph.D. degree from the University of Newcastle, Australia. He is the Founding Chair of IEEE NSW VTS Chapter. He served as a TPC Chair of BodyNets2015, ISCIT2015, and WPMC2014, as an OC Co-Chair of the VTC2017-Spring, BodyNets2014, ICUWB2013, ISCIT2012, SenSys2007, and in Technical Program Committee in a number of IEEE Conferences.