Optimal Power Allocation for Maximum Throughput of General MU-MIMO Multiple Access Channels With Mixed Constraints

Peter He, *Member, IEEE*, and Lian Zhao, *Senior Member, IEEE*

***Abstract*—Based on the efficient generalized water-filling with group peak power constraints (GWFGP), this paper proposes an iterative algorithm to compute the optimal solutions to system throughput (sum-rate) maximization problems. This class of problems is equipped with the multiuser multiple input multiple output multiple access channels (MU-MIMO MAC) in the general communication systems. The proposed iterative GWFGP algorithm (IGWFGP) has two levels of loops. The inner loop aims at computing the solution to each member in the family, while the outer loop aims at computing the solution to the target problem based on the results obtained by the inner loop. Both GWFGP and the convergence theory of an algorithm are used in the inner loop and the outer loop respectively. Furthermore, by exploiting the concept of variable weighting factor for covariance update, IGWFGP owns fast convergence and provides optimal solutions to the sum rate maximization problems. The usage of the convergence theory in IGWFGP and the algorithm of GWFGP are efficient and novel. To the best of the authors' knowledge, no prior algorithm has been reported in the open literature to solve the targeted problem in this paper. In addition, the proposed algorithm does not require to choose the initial value for computation. This feature is a significant advantage of the algorithm, especially for large and complicated systems.**

***Index Terms*—Maximum throughput, multi-user MIMO (MU-MIMO), multiple access channel (MAC), optimal power distribution, generalized water-filling algorithm with group peak power constraints (GWFGP), iterative GWFGP (IGWFGP).**

I. Introduction

THE multiple-input multiple-output (MIMO) communication system has multiple antennas. These antennas are equipped at either the transmitter or the receiver or both. The purpose of MIMO is to significantly increase data throughput and link range without using additional bandwidth or transmitted power. Thus MIMO plays an important role in wireless communications today, *e.g.*, [1]–[2]. It is a hot topic since some important issues need further solving. As the development of the MIMO system continuously progresses, it now needs to consider the mixed constraints of the individual power and the sum power together. This background of these constraints is from field requirements, such as, avoiding the saturation of powers allocated to the users and avoiding system level power leakage out of band due to the non-linearity generated by the users.

Some earlier investigations have been reported to solve the target problem by solving a roughly approximated problem, *e.g.* [3] (referring to its (13) and (34)) and [4] (referring to its obtaining (6) from (5)). The limitation is that these proposed algorithms, being designed to solve approximated problem instead, cannot guarantee to converge to the optimal solution of the original problem. Together with the differentiability issue, they belong to a separate category and therefore are not chosen as reference algorithms to compare with in this paper. Fundamentally, the convex optimization theory and methods are built up on the basis of the real space [5]. Our approach avoids the problem that the objective function, in several complex optimization variables, is actually not differentiable. At the same time, our novel design principle can solve the target problems more efficiently.

The contributions of this paper are twofold, 1) directly solving the target problem, and 2) correctly utilizing the fundamental concept of the differentiability. We proposed an algorithm with two levels of loops to obtain the optimal solution to the target problem. The target problem is first decomposed into a family of the equivalent optimization sub-problems over real space. The inner loop computes the solution for each member in the family; while the outer loop computes the solution to the target problem based on the results obtained by the inner loop. Both the proposed generalized water-filling with group peak power constraints (GWFGP) and the convergence theory of an algorithm are used in the inner loop and the outer loop respectively, and are formed into the proposed algorithm: iterative GWFGP (IGWFGP). The GWFGP is only used once with a finite amount of computation and a low computational complexity during each of the inner loop. This point leads to IGWFGP being convergent.

The conventional iterative water-filling algorithms [6] and [7] cannot solve the target problem in this paper. The problem that the algorithm in [6] solved does not have peak power constraint; while the problem that the algorithm in [7] solved does not have sum power constraint. These problems are special cases of the target problem in this paper. One is achieved by relaxing the group peak power constraints, and another by relaxing the sum power constraint of the proposed problem. The proposed problem and algorithm are more general. At the same time, the investigated object of General multi-user

Manuscript received June 21, 2015; revised November 5, 2015 and January 14, 2016; accepted January 17, 2016. Date of publication January 21, 2016; date of current version March 15, 2016. The associate editor coordinating the review of this paper and approving it for publication was H. Dai.

The authors are with the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON M5B 2K3, Canada (e-mail: phe@ee.ryerson.ca; lzhao@ee.ryerson.ca).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCOMM.2016.2520458

Multiple Input Multiple Output Multiple Access Channel is briefly referred to as GMU-MIMO MAC in this paper.

The conventional water-fillings [8]–[11] cannot be substituted for GWFGP in the inner loop due to the additional group peak power constraints posed by our general problem, which [8]–[11] do not take into consideration. Other optimization methods, such as the popular primal-dual interior point method (PD-IPM), cannot complete the computation of each inner loop with a finite amount of computation. In addition, they require the initial point selections.

In the remaining of this paper, the system model of the GMU-MIMO MAC and its throughput are described in Section II. Section III proposes GWFGP, a fundamental block of the proposed algorithm. Then, the algorithm (IGWFGP) based on iteration is presented to solve the target problem. Section IV provides the convergence proof of IGWFGP. Section V presents the numerical results to show the effectiveness of the proposed IGWFGP, when IGWFGP is compared with others. Section VI concludes the paper.

Key notations that are used in this paper are as follows: $|\mathbf{A}|$ and $\mathrm{Tr}\,(\mathbf{A})$ give the determinant and the trace of a square matrix $\mathbf{A}$, respectively; $E[X]$ is the expectation of the random variable X; and the capital symbol $\mathbf{I}$ for a matrix denotes the identity matrix with the corresponding size. A square matrix $\mathbf{B} \succeq 0$ means that $\mathbf{B}$ is a positive semi-definite matrix. Further, for arbitrary two positive semi-definite matrices $\mathbf{B}$ and $\mathbf{C}$, the expression $\mathbf{B} \succeq \mathbf{C}$ means the difference of $\mathbf{B} - \mathbf{C}$ is a positive semi-definite matrix. In addition, for any complex matrix, its superscripts $\dagger$ and T denote the conjugate transpose and the transpose of the matrix, respectively. For convenience, a table of some notations or symbols is added here to help the readers keep track of the notations.

II. System of GMU-MIMO MAC and Its Throughput

For a MIMO-MAC, assume that there are one base-station (BS) with N_r antennas, and K users. Each of the users is equipped with N_t antennas. The received signal $\mathbf{y} \in \mathbb{C}^{N_r \times 1}$ at the BS is described as

$$\mathbf{y} = \textstyle\sum_{i=1}^{K} \mathbf{H}_i^{\dagger}\mathbf{x}^i + \mathbf{z}, \text{ where } \mathbf{H}_i \in \mathbb{C}^{N_t \times N_r}, i = 1, 2, \ldots, K, \tag{1}$$

the $\mathbf{x}^i \in \mathbb{C}^{N_t \times 1}$ is a complex input signal vector from the i-th user and it is also assumed to be a Gaussian random vector having zero mean for any i. Further $\{\mathbf{x}^i\}_{i=1}^{K}$ is a set of independent random vectors. The noise term, $\mathbf{z} \in \mathbb{C}^{N_r \times 1}$, is an additive Gaussian noise random vector, i.e., $\mathbf{z}, \sim \mathbb{N}(0, \mathbf{I})$, without loss of generality. The channel input, $\{\mathbf{x}^i\}_{i=1}^{K}$, and the noise $\mathbf{z}$ are also assumed to be mutually independent. Furthermore, the i-th user's transmitted power can be expressed as $\mathrm{Tr}(\mathbf{S}_i)$, where

$$\mathbf{S}_i \triangleq \mathcal{E}\left[\mathbf{x}^i \left(\mathbf{x}^i\right)^{\dagger}\right], i = 1, 2, \ldots, K. \tag{2}$$

Note that $\mathbf{S}_i, \forall i$, is positive semi-definite, *i.e.*, $\mathbf{S}_i \succeq 0$.

A mathematical model of the throughput optimization problem for the MU-MIMO MAC in the general system can be formed. It is stated as follows (refer to [12, (2.16) for the concept of multi-user MIMO MAC]):

$$\begin{aligned} &\max\nolimits_{\{\mathbf{S}_k\}_{k=1}^{K}} \log|\mathbf{I} + \sum_{j=1}^{K} \mathbf{H}_j^{\dagger}\mathbf{S}_j\mathbf{H}_j| \\ &\text{subject to: } \mathbf{S}_k \succeq 0;\ \mathrm{Tr}(\mathbf{S}_k) \le P_k, \forall k; \\ &\sum_{k=1}^{K} g_k \mathrm{Tr}\,(\mathbf{S}_k) \le P_T, \end{aligned} \tag{3}$$

where, for the MAC cases, the peak power constraint on the kth user exists and is denoted by P_k in a group of positive numbers: $P_i, i = 1, \ldots, K$; and the upper bound of the (total) weighted sum power is denoted by a positive number P_T. In the weighted sum power expression, $\{g_k\}_{k=1}^{K}$ is a set of weights or gains. As a general assumption, they are positive numbers. When $g_k = 1, \forall k$, as our special case, this was the assumption in [6]. As a summary, the system of MU-MIMO MAC together with the user peak power and the weighted sum power constraints is the system of GMU-MIMO MAC. The target problem (3) is just to look for the maximum throughout of GMU-MIMO MAC. Solving the target problem (3) needs solving some sub-problems. To distinguish the target problem from the sub-problems, the target problem is also called the original problem, in the following. Note that if $\exists \mathbf{H}_{i_0} = \mathbf{0}$, $1 \le i_0 \le K$, in (3), the user i_0 is removed, and the number of the users is reduced to $K - 1$. In this way, we can assume that $\mathbf{H}_i \neq \mathbf{0}, \forall i$. Note that besides the well known Jensen's inequality, a convex optimization problem requires the objective function to be a mapping from a set of real space to the set of real numbers [5, p. 7]. The objective function of the original problem (3) is a mapping from a set of complex matrix-valued variables to the set of real numbers. Although it satisfies the Jensen's inequality, it does not satisfy the mapping condition in the definition of a convex optimization problem. Therefore, the original problem (3) is not a convex optimization problem.

III. Algorithms for System of GMU-MIMO MAC

In this section, we discuss the proposed algorithms to solve the throughput maximization problem, as the original problem, in the system of GMU-MIMO MAC. In the first subsection, the generalized water-filling problem with the mixed constraints is first investigated, and then its algorithm, GWFGP is proposed. In the second subsection, the proposed algorithm IGWFGP is presented. It treats the algorithm GWFGP as a basic functional block. Then it iteratively utilizes this functional block to compute the solution to the original or target problem. In the final subsection, we introduce the algorithm which is a realization of the Primal-Dual approach for the cases with Group Peak power constraints, referred to as PD-GP, for comparison purpose.

A. Algorithm of Generalized Water-Filling With Group Peak Power Constraints (GWFGP)

In the first part of this subsection, the background of GWFGP is stated. In the second part, the statement of GWFGP is

proposed and then optimality of GWFGP, as an algorithm, is provided. In the third part, through a formal definition of the equivalence between two optimization problems, GWFGP is explained as a basic functional block for completing one iteration of IGWFGP.

1) Background of GWFGP: For the original problem (3), it is first assumed that an iterative algorithm can be used to compute an optimal solution. Formally, the iterative algorithm should form a point sequence, through iterations, from a point to the next one. At the same time, any limit point of this sequence would be an optimal solution, *i.e.*, to meet convergence of the iterative algorithm. Thus how to design each of such iterations, from a point to the next point, becomes important. The mentioned next point, however, is generally not an optimal solution to the original problem.

Then, each of the iterations generates the next point from a given point. This generation is stated with three steps as follows.

First Step: Let us denote a point that has been obtained just before an iteration, by $(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_K)$. The objective function of (3) is denoted by f.

Second Step: For the given point: $(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_K)$, let us compute the optimal solution to the following optimization problem:

$$\begin{aligned} \max_{\{\mathbf{S}_k\}_{k=1}^K} \ & \frac{1}{K}\sum_{k=1}^{K} f\left(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_{k-1}, \mathbf{S}_k, \overline{\mathbf{S}}_{k+1}, \ldots, \overline{\mathbf{S}}_K\right) \\ \text{subject to: } & \mathbf{S}_k \succeq 0;\ \mathrm{Tr}\left(\mathbf{S}_k\right) \le P_k, \forall k; \\ & \sum_{k=1}^{K} g_k \mathrm{Tr}\left(\mathbf{S}_k\right) \le P_T, \end{aligned} \tag{4}$$

i.e., this feasible set is the same as that of (3). The optimal solution to this just mentioned optimization problem is denoted by $(\tilde{\mathbf{S}}_1, \ldots, \tilde{\mathbf{S}}_K)$.

Third Step: $(\frac{K-1}{K}\overline{\mathbf{S}}_1 + \frac{1}{K}\tilde{\mathbf{S}}_1, \ldots, \frac{K-1}{K}\overline{\mathbf{S}}_K + \frac{1}{K}\tilde{\mathbf{S}}_K)$ is, as a new point, generated by this step. For convenience, it is written as $\{\overline{\overline{\mathbf{S}}}_k\}_{k=1}^K$.

To compute the solution to (4), while only involving the Second Step, is the motivation or background of GWFGP. With any limit point of the point sequence determined by all the three steps, iteratively, being an optimal solution to the original problem (3), this becomes the motivation or background of the *iterative* GWFGP. *These two points would account for background difference between GWFGP and the iterative GWFGP.* In addition, since the function f mentioned above satisfies the Jensen inequality, it is easy to see that with each iteration, the objective function of (3) increases.

For the Second Step, let us focus on maximizing $\frac{1}{K}\sum_{k=1}^K f(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_{i-1}, \mathbf{S}_i, \overline{\mathbf{S}}_{i+1}, \ldots, \overline{\mathbf{S}}_K)$ in the optimization variable $(\mathbf{S}_1, \ldots, \mathbf{S}_K)$ over the feasible set. Since

$$\begin{aligned} &\sum_{k=1}^{K} \log\left|\mathbf{I} + \mathbf{H}_k^\dagger \mathbf{S}_k \mathbf{H}_k + \sum_{i\in\{1,\ldots,K\}\setminus\{k\}} \mathbf{H}_i^\dagger \overline{\mathbf{S}}_i \mathbf{H}_i\right| \\ &= \sum_{k=1}^{K} \log\left|\mathbf{I} + \mathbf{G}_k^\dagger \mathbf{S}_k \mathbf{G}_k\right| \\ &\quad + \sum_{k=1}^{K} \left|\mathbf{I} + \sum_{i\in\{1,\ldots,K\}\setminus\{k\}} \mathbf{H}_i^\dagger \overline{\mathbf{S}}_i \mathbf{H}_i\right|, \end{aligned} \tag{5}$$

where $\overline{\mathbf{S}}_i, \forall i$, is fixed and

$$\mathbf{G}_k = \mathbf{H}_k \left(\mathbf{I} + \textstyle\sum_{i\in\{1,\ldots,K\}\setminus\{k\}} \mathbf{H}_i^\dagger \overline{\mathbf{S}}_i \mathbf{H}_i\right)^{-\frac{1}{2}}, \forall k, \tag{6}$$

the mentioned optimization problem (4) with the detailed form:

$$\begin{aligned} \max_{\{\mathbf{S}_k\}} \ & \textstyle\sum_{k=1}^{K} \log\left|\mathbf{I} + \mathbf{H}_k^\dagger \mathbf{S}_k \mathbf{H}_k + \sum_{i\in\{1,\ldots,K\}\setminus\{k\}} \mathbf{H}_i^\dagger \overline{\mathbf{S}}_i \mathbf{H}_i\right| \\ \text{subject to: } & \mathbf{S}_k \succeq 0, \forall k; \\ & \mathrm{Tr}\left(\mathbf{S}_k\right) \le P_k, \forall k; \\ & \textstyle\sum_{k=1}^{K} g_k \mathrm{Tr}\left(\mathbf{S}_k\right) \le P_T \end{aligned}$$

is identical to the problem below:

$$\begin{aligned} \max_{\{\mathbf{S}_k\}} \ & \textstyle\sum_{k=1}^{K} \log\left|\mathbf{I} + \mathbf{G}_k^\dagger \mathbf{S}_k \mathbf{G}_k\right| + \mathcal{C} \\ \text{subject to: } & \mathbf{S}_k \succeq 0, \forall k; \\ & \mathrm{Tr}(\mathbf{S}_k) \le P_k, \forall k; \\ & \textstyle\sum_{k=1}^{K} g_k \mathrm{Tr}(\mathbf{S}_k) \le P_T, \end{aligned} \tag{7}$$

where $\mathcal{C}$ is the constant of $\sum_{k=1}^K \left|\mathbf{I} + \sum_{i\in\{1,\ldots,K\}\setminus\{k\}} \mathbf{H}_i^\dagger \overline{\mathbf{S}}_i \mathbf{H}_i\right|$. With or without the constant of $\mathcal{C}$ in the objective function, the optimum point set to this simplified optimization problem (7) is the same as the optimum point set of the optimization problem: (4). A constant $\mathcal{C}$ in the objective function has no effect on the set of the optimal solutions. In addition, according to the aforementioned two motivations above, the optimization problem (7) is not identical to the original problem (3).

The kth term in the summation of the objective function for the simplified optimization problem only contains $\mathbf{S}_k$ without others, for any k. This structure permits us to design an efficient GWFGP to compute the solution to the simplified optimization problem.

2) Statement of GWFGP: As a fundamental block of the optimum resource allocation problem for the GMU-MIMO MAC systems, the generalized water-filling problem is abstracted as follows.

For a multiple receiving antenna system of parallel independent channels, it is divided as K groups. Each group has N_t channels. It is given that $P_T > 0$, as the total power or volume of the water; squares of the channel gain norms, $\{a_j\}_{j=1}^{KN_t}$, are partitioned as the K groups, the index sets of which are labelled as $\{\Lambda_k\}, k = 1, \ldots, K$, such that $\{\Lambda_k\}_{k=1}^K$ is a partition of $\{1, 2, \ldots, KN_t\}$, with $\Lambda_k = \{(k-1)N_t + 1, (k-1)N_t + 2, \ldots, kN_t\}, \forall k$; all the corresponding channels in the group k share a power gain labelled as g_k, i.e., $\{a_j\}_{j\in\Lambda_k}$ corresponds to g_k; and the allocated powers of the kth group of channels are given as $\{s_j\}_{j\in\Lambda_k}, \forall k$.

From the partition mentioned above, a mapping σ is defined, for clarity, as follows: if for channel j, there exists a unique $k, 1 \le k \le K$ such that $j \in \Lambda_k$, then $\sigma(j) = k$. Without loss

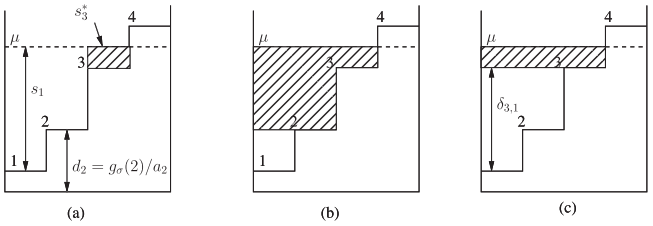

Fig. 1. Illustration for the proposed Generalized Water-Filling (GWF) algorithm ($\sigma(i)$ is assumed to be a constant, for any i). (a) Illustration of water level step $k^* = 3$, allocated power for the third step s_3^*, and step/stair depth $d_i = g_{\sigma(i)}/a_i$. (b) Illustration of $P_2(k)$ (shadowed area, representing the total water/power above step k) when $k = 2$. (c) Illustration of $P_2(k)$ when $k = 3$.

of generality, it is assumed that the sequence $\{a_j/g_{\sigma(j)}\}_{j=1}^{KN_t}$ is monotonically decreasing; else, we would have to tediously utilize permutation of the subscript sequence used by the summation operator. Thus, under the assumptions mentioned above, we can find the problem:

$$\begin{array}{ll} \max_{\{s_j\}_{j=1}^{KN_t}} & \sum_{j=1}^{KN_t} \log\left(1 + a_j s_j\right) \\ \text{subject to:} & 0 \le s_j, \forall j; \\ & \sum_{j\in\Lambda_k} s_j \le P_k, \forall k; \\ & \sum_{k=1}^{K} \sum_{j\in\Lambda_k} g_k s_j \le P_T. \end{array} \tag{8}$$

Note, as $\sum_{k=1}^{K} g_k P_k \le P_T$, the solution to problem (3) is regressed into a trivial case. Hence, $\sum_{k=1}^{K} g_k P_k > P_T$ is assumed. When $P_k \gg 0, \forall k$, then the problem (8) is reduced into the simple case that can be solved by the conventional water-filling problem [8]. In general case, the problem structure in (8) cannot be solved by the conventional water-filling. Our proposed GWFGP is presented to solve this generalized radio resource management problem.

Firstly let us introduce a vivid description of water-filling algorithm from a geometric point of view by pouring the water of volume P_T into a tank with the bottom of K stairs as shown in Fig. 1, for four steps/stairs ($K = 4$) with unit width inside a water tank. For the conventional approach, the dashed horizontal line, which is the water level μ, needs to be determined first and then the power allocated (water volume) above is solved.

In the following, we will introduce four variables used in our approach. The first variable, $P_2(k)$, is defined as the total water volume above the kth stair. The second variable is s_i, as the allocated power for the ith channel. The third variable is water level step, denoted as k^*. It denotes the highest step under water. The fourth variable, s_{k^*} is defined as the optimal power allocated to the water level step. Fig. 1(a) illustrates the concept of k^*. Since the third level is the highest level under water, we have $k^* = 3$. The shaded area denotes the allocated power for the third step by s_3^*. Fig. 1(b) and Fig. 1(c) illustrate the concept of $P_2(k)$ when $k = 2$ and $k = 3$ respectively.

Let us use $g_{\sigma(i)}/a_i$ to denote the "step depth" of the ith stair which is the height of the ith step to the bottom of the tank, and is given as

$$d_i = \frac{g_{\sigma(i)}}{a_i}, \text{ for } i = 1, 2, \ldots, KN_t. \tag{9}$$

Since the sequence $a_i/g_{\sigma(i)}$ is sorted as monotonically decreasing, the step depth of the stairs indexed as $[1, \ldots, KN_t]$ is monotonically increasing. We further define $\delta_{i,j}$ as the "step depth difference" of the ith and the jth stairs, expressed as

$$\delta_{i,j} = d_i - d_j = \frac{g_{\sigma(i)}}{a_i} - \frac{g_{\sigma(j)}}{a_j}, \text{ as } i \ge j \text{ and } 1 \le i, j \le KN_t. \tag{10}$$

Instead of trying to determine the water level μ which is a real nonnegative number, as in the conventional WF algorithm, we aim to determine the water level step, k^*, which is an integer number from 1 to KN_t, as the highest step under water. Based on the result of k^*, we can write out the solutions for power allocation explicitly.

In the following, we explain how to find k^* without the knowledge of the water level μ. The value of $P_2(k)$ can be solved by subtracting the volume of the water under step k from the total power P_T, as

$$\begin{aligned} P_2(k) &= \left\{ P_T - \left[\sum_{i=1}^{KN_t-1} \left(\frac{g_{\sigma(k)}}{a_k} - \frac{g_{\sigma(i)}}{a_i} \right) \right] \right\}^+ \\ &= \left\{ P_T - \left[\sum_{i=1}^{KN_t-1} \delta_{k,i} \right] \right\}^+, \quad k = 1, \ldots, KN_t. \end{aligned} \tag{11}$$

As an example of Fig. 1(c), the water volume under step 3 can be expressed as the sum of the two terms: (i) the step depth difference between the 3rd and the 1st step, $\delta_{3,1}$, and (ii) the step depth difference between the 3rd and the 2nd step, $\delta_{3,2}$. Thus, $P_2(k = 3)$ can be written as $P_2(k = 3) = \left[P_T - \delta_{3,1} - \delta_{3,2}\right]^+$ which is an expansion of the composite form of (11).

Generally, from Fig. 1, we can have

$$P_2(k) = \left[P_T - \sum_{t=1}^{E^\sharp - 1} \left(\frac{g_{\sigma(i_k)}}{a_{i_k}} - \frac{g_{\sigma(i_t)}}{a_{i_t}} \right) \right]^+, \quad \text{for } k = 1, \ldots, E^\sharp, \tag{12}$$

where E is a subsequence of the sequence $\{1, 2, \ldots, KN_t\}$, $E^\sharp$ is the cardinality of the set E, so E can be expressed as $\{i_1, i_2, \ldots, i_{E^\sharp}\}$. As a reminder, the definition of a special case for the summation is: $\sum_{i=m}^{n} b_i = 0$, as $m > n$, where $\{b_i\}$ is a general series of numbers. The water level step k^* is given as

$$k^* = \max\left\{ k \,\middle|\, P_2(k) > 0,\ 1 \le k \le E^\sharp \right\} \tag{13}$$

and the power level for this step is

$$s_{i_{k^*}} = \frac{1}{k^* g_{\sigma(i_{k^*})}} P_2\left(k^*\right). \tag{14}$$

The power levels for all other steps are given as

$$s_{i_t} = \begin{cases} \frac{g_{\sigma(i_{k^*})}}{g_{\sigma(i_t)}} \left(s_{i_{k^*}} + \frac{1}{a_{i_{k^*}}} \right) - \frac{1}{a_{i_t}}, & 1 \le t \le k^* \\ 0, & k^* < t \le E^\sharp. \end{cases} \tag{15}$$

Based on these results, the steps of the GWFGP can be described as below.

Algorithm GWFGP:

Input: the squared channel gain norms $\{a_i\}_{i=1}^{KN_t}$, the power gains $\{g_i\}_{i=1}^{K}$, the individual power peak or upper limit $\{P_i\}_{i=1}^{K}$, the index set $E = (E_0 =)\{1, 2, \ldots, KN_t\}$, the partition $\{\Lambda_i\}_{i=1}^{K}$ and the sum power constraint P_T.

1) Utilize Eqns. (12)–(15) to compute $\{s_i\}$..

2) The set Λ is defined by the set $\{i | \sum_{j\in\Lambda_i} s_j > P_i, 1 \le i \le K\}$. If Λ is the empty set, output $\{s_i\}_{i=1}^{KN_t}$; else, let $\sum_{j\in\Lambda_i} s_j = P_i$, as $i \in \Lambda$. Further, continuously utilize similar Eqns. (12)–(15) as these similar expressions. These similar expressions differ only by changing from P_T in (12)–(15) to $g_i P_i$ and from $E^\sharp$ in (12)–(15) to Λ_i for any $i \in \Lambda$, and then obtain s_j, $j \in \cup_{i\in\Lambda}\Lambda_i$.

3) $E \setminus \cup_{i\in\Lambda}\Lambda_i \to E$, where the symbol "$\to$" means the assignment operation forwarding the value of the LHS (left-hand side) to that of the RHS (right-hand side). $P_T - \sum_{i\in\Lambda} g_i P_i \to P_T$. Then return to 1) of GWFGP.

For **2)** above, the set Λ consists of the indexes, where the index i in the set corresponds to group i. The sum power of this group is strictly greater than P_i, when first compute **1)** of GWFGP. If Λ is empty, then output $\{s_i\}_{i=1}^{KN_t}$, which were just obtained by **1)**. Else, for any $i \in \Lambda$, using Λ_i is to solve the following (sub-)problem:

$$\begin{array}{ll} \max_{\{s_j\}_{j\in\Lambda_i}} & \sum_{j\in\Lambda_i} \log\left(1 + a_j s_j\right) \\ \text{subject to:} & 0 \le s_j, \forall j \in \Lambda_i; \\ & g_i \sum_{j\in\Lambda_i} s_j \le g_i P_i, \end{array} \tag{16}$$

where $g_i > 0$. The mentioned set of formulas is used without introducing additional set of new formulas. The index set E is then replaced with $E \setminus \cup_{i\in\Lambda}\Lambda_i$ in 3) above. This procedure has been carried out to shrink the index set.

Regarding optimality of the proposed GWFGP, we can obtain the following conclusion:

Proposition 3.1: Algorithm GWFGP can provide the exact optimal solution to the problem (8) via a finite amount of computation.

Proof: See Appendix A. ■

From the computational details of the GWFGP, the computational complexity of the proposed algorithm is, at worst, $\sum_{i=1}^{KN_t}(8i + 2) = 4KN_t^2 + 6KN_t$, *i.e.*, a moderate computational complexity, to solve a more generalized case of a weighted water-filling problem with sum and individual peak power constraints. Proposition 3.1 is also applicable. Note that only the most popular PD-IPM can guarantee the computational complexity of $O(\log(\frac{1}{\epsilon})(KN_t)^{3.5})$ to compute an ϵ solution, which is not an optimal solution.

As a side note, the conventional water-filling approach (CWF) attempts to directly solve the KKT conditions for solution. This CWF vastly differs from ours. If we used the CWF to solve the problem (8), a non-linear system with non-linear equations and inequalities in multiple dual variables would have had to be solved. The problem becomes (very difficult),

$$\begin{cases} \sum_{k=1}^{K}\sum_{j\in\Lambda_k}\left(\frac{1}{\lambda\times g_k+\sigma_k} - \frac{1}{a_j}\right)^+ = P_T, \\ \sum_{j\in\Lambda_k}\left(\frac{1}{\lambda\times g_k+\sigma_k} - \frac{1}{a_j}\right)^+ \le P_k, \text{ for } k = 1, 2, \ldots, K; \\ \sigma_k\left(\sum_{j\in\Lambda_k}\left(\frac{1}{\lambda\times g_k+\sigma_k} - \frac{1}{a_j}\right)^+ - P_k\right) = 0, \\ \text{for } k = 1, 2, \ldots, K; \\ \lambda \ge 0; \sigma_k \ge 0, \text{ for } k = 1, 2, \ldots, K. \end{cases} \tag{17}$$

There seems no existing result that can solve such a system.

3) Equivalence of GWFGP: A clear definition for the equivalence between two optimization problems is stated formally next. This definition was only provided in an informal form (mentioned in [5, p. 130]); and many other books have seldom included this formal definition. For clarity, the formal definition of two optimization problems being equivalent is provided as below.

Definition 3.3.1 (Equivalence between Two Optimization Problems): Two optimization problems are said to be equivalent iff there exists a bijection between their optimal solution sets.

Therefore, different objective functions and/or feasible sets needn't lead to non-equivalence. A typical example to this is: $\min_x x^2$, subject to: $|x| \le 2$; and $\min_x |\sin x| + 1$, subject to: $|x| \le 1$. The two optimization problems are equivalent due to existence of the identity mapping between the two optimal solution sets. Also, the definition of equivalence implies that, if both the optimal solution set to the first optimization problem, and the bijection are given, then the optimal solution set to the second optimization problem can be found. While the first optimization problem is typically simpler than the second one, the construction of the bijection is key.

It is seen that the following problem is just an instance of the optimization problem (8):

$$\begin{array}{ll} \max & \sum_{k=1}^{K}\sum_{j=1}^{N_t}\log\left(1 + d_k(j)s_k(j)\right) \\ \text{subject to:} & s_k(j) \ge 0, \text{ for } 1 \le k \le K, 1 \le j \le N_t; \\ & \sum_{j=1}^{N_t} s_k(j) \le P_k, \forall k; \\ & \sum_{k=1}^{K} g_k \sum_{j=1}^{N_t} s_k(j) \le P_T, \end{array} \tag{18}$$

where $d_k(j), 1 \le j \le N_t$, is an eigenvalue of the matrix $\mathbf{G}_k\left(\mathbf{G}_k\right)^\dagger, \forall k$. At the same time, it can be obtained from the eigendecomposition. Thus, the optimization problem (18) can be solved by GWFGP.

Proposition 3.2: The optimization problems (18) and (7) are equivalent.

Proof: Resorting to the well known Hadamard's inequality on positive definite matrices, the equivalence may be proven. Given a family of the positive semi-definite matrices: $\{G_k G_k^\dagger\}_{k=1}^{K}$, there exists a family of the unitary matrices:

$\{U_k\}_{k=1}^K$ such that $U_k^\dagger G_k G_k^\dagger U_k$ is a diagonal matrix, denoted by D_k, through the eigendecomposition, for $k = 1, \ldots, K$. Further, the major diagonal entries of this diagonal matrix are the eigenvalues of $G_k G_k^\dagger$. That is to say, the diagonal matrix and $G_k G_k^\dagger$ are unitarily similar. Thus, we have a bijection from the set of feasible solutions to (18) to that to (7): $(s_k(1), \ldots, s_k(N_t)) \mapsto U_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) U_k^\dagger, \forall k$. The following will prove that the bijection is the mentioned one in the definition of equivalence between two optimization problems.

Let us take any optimum point to (18), denoted by $\{(s_k(1), \ldots, s_k(N_t))\}$. It is seen that $\{U_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) U_k^\dagger\}$ is a feasible point to (7). Evaluate the objective function of (7) at this feasible point. Thus we obtain:

$$\begin{aligned}
&\sum_{k=1}^K \log \left| \mathbf{I} + \mathbf{G}_k^\dagger U_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) U_k^\dagger \mathbf{G}_k \right| \\
&\quad = \sum_{k=1}^K \log \left| \mathbf{I} + U_k^\dagger \mathbf{G}_k \mathbf{G}_k^\dagger U_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) \right| \\
&\quad = \sum_{k=1}^K \log \left| \mathbf{I} + D_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) \right| \\
&\quad = \sum_{k=1}^K \sum_{j=1}^{N_t} \log(1 + D_k(j,j) s_k(j)). \qquad (19)
\end{aligned}$$

Assume, to the contrary, that this feasible point is not an optimal solution to (7). There is another feasible solution to (7), denoted by $\{S_k\}_{k=1}^K$ (a set of the matrices) such that

$$\sum_{k=1}^K \sum_{j=1}^{N_t} \log(1 + D_k(j,j) s_k(j)) < \sum_{k=1}^K \log \left| \mathbf{I} + \mathbf{G}_k^\dagger S_k \mathbf{G}_k \right|.$$

Since

$$\begin{aligned}
\sum_{k=1}^K \log \left| \mathbf{I} + \mathbf{G}_k^\dagger S_k \mathbf{G}_k \right| &= \sum_{k=1}^K \log \left| \mathbf{I} + \mathbf{G}_k \mathbf{G}_k^\dagger S_k \right| \\
&= \sum_{k=1}^K \log \left| \mathbf{I} + U_k^\dagger \mathbf{G}_k \mathbf{G}_k^\dagger U_k U_k^\dagger S_k U_k \right|, \qquad (20)
\end{aligned}$$

according to the Hadamard's inequality,

$$\begin{aligned}
&\sum_{k=1}^K \log \left| \mathbf{I} + U_k^\dagger \mathbf{G}_k \mathbf{G}_k^\dagger U_k U_k^\dagger S_k U_k \right| \\
&\quad \le \sum_{k=1}^K \sum_{j=1}^{N_t} \log(1 + D_k(j,j)(U_k^\dagger S_k U_k)(j,j)). \qquad (21)
\end{aligned}$$

Thus,

$$\begin{aligned}
&\sum_{k=1}^K \sum_{j=1}^{N_t} \log(1 + D_k(j,j) s_k(j)) \\
&\quad < \sum_{k=1}^K \sum_{j=1}^{N_t} \log(1 + D_k(j,j)(U_k^\dagger S_k U_k)(j,j)), \qquad (22)
\end{aligned}$$

where $(U_k^\dagger S_k U_k)(j,j)$ denotes the jth diagonal entry of the matrix $U_k^\dagger S_k U_k$, for any j and k. Hence, since it is seen that $\{(U_k^\dagger S_k U_k)(j,j)\}$, *i.e.*, $\cup_{k=1}^K \{(U_k^\dagger S_k U_k)(j,j)\}_{j=1}^{N_t}$, is a feasible solution to (18), $\{(s_k(1), \ldots, s_k(N_t))\}$ is not an optimum point to (18). A contradiction is obtained. Therefore, $\{U_k \mathrm{diag}(s_k(1), \ldots, s_k(N_t)) U_k^\dagger\}$ is an optimum point to (7) under the bijection.

Similarly, the bijection can project an optimum point to (7) to an optimum point to (18).

As a result, equivalence between (7) and (18) holds. ■

Since the solution to (18) can be computed rapidly and exactly, and a bijection has been constructed, (7) can be solved rapidly and exactly as well.

B. Algorithm IGWFGP and Its Implementation

The proposed IGWFGP is based on the combination of the MU-MIMO MAC with the mixed power constraints. The algorithm is listed below.

Note that the new algorithm employs variable weighting factors or innovation, which are obtained to maximize the objective function and to update the covariance.

From the process mentioned above, it is seen that the $(\tilde{\mathbf{S}}_1^{(n)}, \ldots, \tilde{\mathbf{S}}_K^{(n)})$ is a preparation for computation of the $(\mathbf{S}_1^{(n)}, \ldots, \mathbf{S}_K^{(n)})$. The $(\mathbf{S}_1^{(n)}, \ldots, \mathbf{S}_K^{(n)})$ is also briefly written into $p^{(n)}$. The $(\tilde{\mathbf{S}}_1^{(n)}, \ldots, \tilde{\mathbf{S}}_K^{(n)})$ corresponds to the non-iterative GWFGP; while the $(\mathbf{S}_1^{(n)}, \ldots, \mathbf{S}_K^{(n)})$, corresponds to the iterative IGWFGP.

Remark 3.1: IGWFGP iteratively uses GWFGP in its 2), until it converges. Due to the objective function $f\left(\beta\gamma^{(n)} + (1-\beta)p^{(n-1)}\right)$ in Step 3) of Algorithm IGWFGP being (upper) convex, *i.e.*, being concave, in the scalar variable β, for computing the maximum solution to the corresponding optimization problem, we can choose finite searching steps with even fewer evaluations of the objective function. The objective function in step 3) is evaluated at the four points $\left\{\beta = \frac{1}{K}, \frac{1}{K} + \frac{1}{3}\left(1 - \frac{1}{K}\right), \frac{1}{K} + \frac{2}{3}\left(1 - \frac{1}{K}\right) \text{ and } 1\right\}$ by parallel computation to determine β^*. For example to set the the four points, if $K = 4$, β takes $\frac{1}{4}, \frac{2}{4}, \frac{3}{4}$ and 1. That is to say, this **parallel operation** can be **distributed** to and carried out by the multiple processors (for example, 4 processors) at the base station in order to expedite convergence of the proposed algorithm.

We may replace the number "3" in $\beta = \frac{1}{K} + \frac{i}{3}(1 - \frac{1}{K})$ for $i = 0, 1, 2, 3$, with $m - 1$ for $\beta = \frac{1}{K} + \frac{i}{m-1}(1 - \frac{1}{K})$, for $i = 0, 1, 2, m - 1$. Further, the integer $m \ge 4$ is assumed. As long as $m \ge 4$, the conclusion in this paper still holds.

In Remark 3.1 above, β_1 only takes four points, including the point of $\beta_1 = 1/K$. We evaluated f in the finite points of β_1, and then picked up the maximum over the finite points as β^*. Without loss of generality, we can utilized different number of points of β_1. If we only choose one point, $\beta_1 = 1/K$, the proposed algorithm is still convergent, but its convergent rate is slower than that of multiple points. Since a quad-core CPU is popular for parallel processing, four points are chosen. The more cores a CPU has, the faster the convergent

Algorithm IGWFGP:

Input: Matrices $\mathbf{H}_i, \mathbf{S}_i^{(0)} = 0, i = 1, \ldots, K; n = 1.$

1) Generate effective channels

$$\mathbf{G}_i{}^{(n)} = \mathbf{H}_i \left(\mathbf{I} + \sum_{k \in \{1,\ldots,K\} \setminus \{i\}} \mathbf{H}_k^{\dagger} \mathbf{S}_k^{(n-1)} \mathbf{H}_k \right)^{-\frac{1}{2}}, \forall i, \tag{23}$$

where the superscript with a pair of bracket, (n), represents the number of iterations.

2) Treating these effective channels as parallel, noninterfering channels, the new covariances $\left\{\tilde{\mathbf{S}}_i^{(n)}\right\}_{i=1}^{K}$ are generated by the GWFGP under the sum power constraint P_T. That is to say, $\left\{\tilde{\mathbf{S}}_i^{(n)}\right\}_{i=1}^{K}$ is the optimal solution to (24):

$$\begin{aligned} &\max_{\{\mathbf{S}_i\}_{i=1}^{K}} \sum_{i=1}^{K} \log |\mathbf{I} + \left(\mathbf{G}_i^{(n)}\right)^{\dagger} \mathbf{S}_i \mathbf{G}_i^{(n)}| \\ &\text{subject to: } \mathbf{S}_i \succeq 0, \forall i; \\ &\qquad \mathrm{Tr}(\mathbf{S}_i) \le P_i, \forall i; \\ &\qquad \sum_{i=1}^{K} g_i \mathrm{Tr}(\mathbf{S}_i) \le P_T. \end{aligned} \tag{24}$$

Note that (24) is just the expression of (7).

3) Update step: Let $\gamma^{(n)}$ and $p^{(n-1)}$ denote the newly obtained covariance set and the immediate past covariance set respectively,

$$\gamma^{(n)} \triangleq \left(\tilde{\mathbf{S}}_1^{(n)}, \tilde{\mathbf{S}}_2^{(n)}, \ldots, \tilde{\mathbf{S}}_K^{(n)}\right) \text{ and}$$
$$p^{(n-1)} \triangleq \left(\mathbf{S}_1^{(n-1)}, \mathbf{S}_2^{(n-1)}, \ldots, \mathbf{S}_K^{(n-1)}\right).$$

Let $\beta^* =$

$$\max \left\{ \beta_1 | \begin{array}{c} \beta_1 \in \arg\max_{\beta \in \{\frac{1}{K} + \frac{i}{3}(1-\frac{1}{K})\}_{i=0}^{3}} f \\ \left(\beta \gamma^{(n)} + (1-\beta)\, p^{(n-1)}\right) \end{array} \right\}, \tag{25}$$

as the innovation or spacer step (refer to [13] and [14]). In this paper, for emphasizing variability of β^*, β^* is also called the variable weighting factor, where the function f means the objective function of (3), mentioned before. Then, the covariance update step is

$$p^{(n)} = \left(\mathbf{S}_1^{(n)}, \mathbf{S}_2^{(n)}, \ldots, \mathbf{S}_K^{(n)}\right) = \beta^* \gamma^{(n)} + \left(1 - \beta^*\right) p^{(n-1)}. \tag{26}$$

The updated covariance is a convex combination of the newly obtained covariance and the immediate past covariance.

4) Let $n + 1 \to n$. Go to **1)** until convergence.

rate of the proposed algorithm is. Hence, $f(p^{(n)}) = f(\beta^*) \ge \max\{f(\beta_1)\} \ge f(\beta_1 = 1/K) \ge f(p^{(n-1)})$. The monotonicity holds for the sequence $\{f(p^{(n)})\}$. At the same time, if we have one more maximum, we take the maximum with the greatest value from β^*s for innovation, *i.e.*, to speed up the convergent rate [14]. Hence, we utilize this four-point method of $\{\beta_1 = 1/K, \frac{1}{K} + \frac{1}{3}(1 - \frac{1}{K}), \frac{1}{K} + \frac{2}{3}(1 - \frac{1}{K}), 1\}$ which are independent on the iteration number n.

C. Algorithm Based on Primal-Dual Approach

In the following, for comparison, we introduce a reference algorithm, PD-GP, which instantiates the primal dual approach (refer to [13]) to attempt to solve the target problem. The essence of the primal dual approach is: firstly we should evaluate exactly a dual function, and then optimize the dual function. Often, we cannot exactly evaluate the dual functions by a finite amount of computation, one of which is the dual function of the target problem. This point leads to the fact that PD-GP cannot obtain the optimal solution, unlike the proposed IGWFGP.

As well known, given $\lambda \ge 0$ and the optimization problem:

$$\begin{aligned} &\max_{\{\mathbf{S}_k\}_{k=1}^{K}} \log |\mathbf{I} + \sum_{j=1}^{K} \mathbf{H}_j \mathbf{S}_j \mathbf{H}_j^{\dagger}| - \\ &\qquad \lambda \left(\sum_{k=1}^{K} g_k \mathrm{Tr}(\mathbf{S}_k) - P_T \right) \\ &\text{subject to } \mathbf{S}_k \succeq 0, \mathrm{Tr}(\mathbf{S}_k) \le P_k, \forall k, \end{aligned} \tag{27}$$

an optimization algorithm may be used here and *the optimal objective function value, as a value of the dual function* of the problem (27) is denoted by $f_d(\lambda)$. It is seen that $f_d(\lambda)$ is a convex function over $\lambda \ge 0$, and λ is a scalar. Thus, the subgradient algorithm or a line search is often used to look for the optimal solution λ^* to the minimum value problem, as the outer loop of the primal dual approach, of the dual function.

Note that the problem (27) has decoupled constraints. Therefore, the block coordinate ascend algorithm (BCAA) or the cyclic coordinate ascend algorithm (CCAA) (refer to [15]) can be used to attempt to solve the problem. The iterative algorithm BCAA works as follows. In each step, the objective function is maximized over a single matrix-valued variable $\mathbf{S}_k$, while keeping all other $\mathbf{S}_k$s fixed, $k = 1, \ldots, K$ and then repeating this process. Without loss of generality, let us consider an optimization problem below over $\mathbf{S}_k, k = 1$, with respect to all other $\mathbf{S}_k$s being fixed, as follows:

$$\begin{aligned} &\max_{\{\mathbf{S}_1\}} \log |\mathbf{I} + \sum_{j=1}^{K} \mathbf{H}_j \mathbf{S}_j \mathbf{H}_j^{\dagger}| - \\ &\qquad \lambda \left(\sum_{k=1}^{K} g_k \mathrm{Tr}(\mathbf{S}_k) - P_T \right) \\ &\text{subject to } \mathbf{S}_k \succeq 0, \mathrm{Tr}(\mathbf{S}_k) \le P_k. \end{aligned} \tag{28}$$

Similar to the problem which GWFGP handled, we may use the eigendecomposition with the unitarily similar meaning. As a

Algorithm PD-GP:

1) Given $\varepsilon > 0$, initialize
$$\left\{\mathbf{S}_1^{(0)} = 0, \ldots, \mathbf{S}_K^{(0)} = 0\right\}, \lambda_{\min} \text{ and } \lambda_{\max}.$$
2) Set $\lambda = (\lambda_{\min} + \lambda_{\max})/2$.
3) Compute $\left\{\mathbf{S}_k^{(n+1)}\right\}_{k=1}^K = f_1\left(\left\{\mathbf{S}_k^{(n)}\right\}_{k=1}^K\right)$. Then $n+1 \to n$. Repeat the above process until the optimal solution $\mathbf{S}_k^*\}_{k=1}^K$ to the problem (27) is reached, where the **BCAA** mentioned before is used. For each iteration of BCAA that takes S_k as the optimization variable with respect to others being fixed, the most popular PD-IPM can be used.
4) If $\sum_{k=1}^K g_k \mathrm{Tr}(\mathbf{S}_k^*) - P_T > 0$, then $\lambda_{\min}$ is assigned by λ; if $\sum_{k=1}^K g_k \mathrm{Tr}(\mathbf{S}_k^*) - P_T < 0$, then $\lambda_{\max}$ is assigned by λ; and if $\sum_{k=1}^K g_k \mathrm{Tr}(\mathbf{S}_k^*) - P_T = 0$, stop.
5) If $|\lambda_{\min} - \lambda_{\max}| \le \varepsilon$, stop. Otherwise, goto step 2).

note, for a matrix, denoted by $\mathbf{A}$, if there exist a diagonal matrix $\mathbf{B}$ and a unitary matrix $\mathbf{U}$ such that $\mathbf{A} = \mathbf{UBU}^\dagger$, then $\mathbf{UBU}^\dagger$ is called the eigendecomposition of $\mathbf{A}$ with the unitarily similar meaning. Further, it is known that the diagonal entries of $\mathbf{B}$ are the eigenvalues of $\mathbf{A}$. Thus, we can obtain the real optimization problem:

$$\begin{aligned} &\max_{x_i\}_{i=1}^{N_t}} \sum_{i=1}^{N_t} \log(1+\lambda_i x_i) - \lambda\left(g_1 \sum_{i=1}^{N_t} x_i - P_T\right) \\ &\text{subject to } x_i \ge 0, \forall i, \quad \sum_{i=1}^{N_t} x_i \le P_1, \end{aligned} \tag{29}$$

where the matrix $\mathrm{diag}\left(\lambda_1, \ldots, \lambda_{N_t}\right)$, with $\{\lambda_i\}$ being decreasing ordered, is unitarily equivalent to the matrix $\mathbf{G}^\dagger\mathbf{G}$ by the unitary matrix $\mathbf{U}$ that is a matrix expression of the similarity transformation. It is seen that we can compute the optimal solution $\{x_i^*\}$ to the problem (29) and then obtain the optimal solution $\mathbf{U}\mathrm{diag}\left(x_1^*, \ldots, x_{N_t}^*\right)\mathbf{U}^\dagger$ to the problem (28). Thus, the proposed PD-GP algorithm, which is based on the primal-dual approach, is concisely described as follows.

The designed PD-GP can indeed avoid the differentiability issue from several complex optimization variable, and it can be used to compare with the proposed algorithm: IGWFGP. However, note that 3) of PD-GP cannot get the optimal solution to (27), even if the PD-IPM is used.

IV. Convergence of Algorithm IGWFGP

Convergence of the proposed algorithm can be proven by a method similar to that of [16, Section II]. However, IGWFGP taking the step 3), as the update step of the innovation or spacer step, is different from that in [16]. Thus, its convergence proof has different subtleties. The differences are accounted for here, but the similarities are omitted.

Lemma 4.1: The optimization problem in (3) is equivalent to a convex optimization problem over the field of real numbers.

Proof of Lemma 4.1 can refer to [16, Proposition II.3]. The explicit real form of the function f_r is $f_r = \frac{1}{2}\log|\hat{\mathbf{I}} + \sum_{j=1}^K \hat{\mathbf{H}}_j^\dagger \hat{\mathbf{S}}_j \hat{\mathbf{H}}_j|$. The corresponding real matrix-valued optimization variable can take on the form $\hat{\mathbf{S}_k} = \begin{pmatrix} \mathbb{R}e(\mathbf{S}_k) & -\mathbb{I}m(\mathbf{S}_k) \\ \mathbb{I}m(\mathbf{S}_k) & \mathbb{R}e(\mathbf{S}_k) \end{pmatrix}$ with four sub-blocks, for any k. $\hat{\mathbf{I}}$ and $\hat{\mathbf{H}}_j\hat{\mathbf{H}}_j^\dagger$, $\forall j$, can be treated similarly.

For any convergent subsequence, whose limit is denoted by $(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_K)$, generated by Algorithm IGWFGP, we may use the following lemma to prove that the limit is a fixed point under Algorithm IGWFGP, when Algorithm IGWFGP is regarded as a mapping.

Lemma 4.2: A point is the limit of a convergent subsequence of the point sequence generated by Algorithm IGWFGP if and only if this point is a fixed point under Algorithm IGWFGP.

Proof: See Appendix B. ■

Lemma 4.3: The feasible solution $(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_K)$ is a fixed point under Algorithm IGWFGP if and only if this feasible solution $(\overline{\mathbf{S}}_1, \ldots, \overline{\mathbf{S}}_K)$ is one of the optimal solutions to the problem in (3).

Proof: Please refer to [16, Lemma II.5]. ■

As a side note, revisit the previous typical example of the optimization problems: $\min_x x^2$, subject to: $|x| \le 2$, and $\min_x |\sin x| + 1$, subject to: $|x| \le 1$. Since the two optimization problems have the same optimal solution set, they are equivalent according to either the informal definition of the equivalence [5, p. 130] or our formal definition. However, the first optimization problem is convex (referring to [5]) but the second one is not. Therefore, two optimization problems being equivalent needn't guarantee they are all convex.

Lemma 4.3 states that a point is an optimal solution to (3) iff this point is a fixed point, where the optimal solution is under the global optimality. Thus, if (3) has multiple optimal solutions, evaluating the objective function at all the solutions would yield the same value.

Based on the lemmas above, we obtain the conclusion that Algorithm IGWFGP is convergent. At the same time, step 3) of Algorithm IGWFGP is regarded as a computation for a point. With these being stated, Algorithm IGWFGP generates a point sequence and every point of the point sequence consists of the K matrices, *e.g.* $(\mathbf{S}_1^{(n)}, \ldots, \mathbf{S}_K^{(n)})$. The details are described below.

Theorem IV.1: Algorithm IGWFGP is convergent. At the same time, the sequence of objective values, obtained by evaluating the objective function at the point sequence, monotonically increases to the optimal objective value.

Proof: Please refer to [16, Theorem II.6]. ■

To reduce the cost of computation, Eq. (25) in Section III may utilize the Fibonacci search. To improve the performance of the algorithm and reduce the cost of the computation, the objective function in step 3) of IGWFGP can be evaluated at the four points mentioned in Remark 3.1, via parallel computation to find the estimate of β^* of (25).

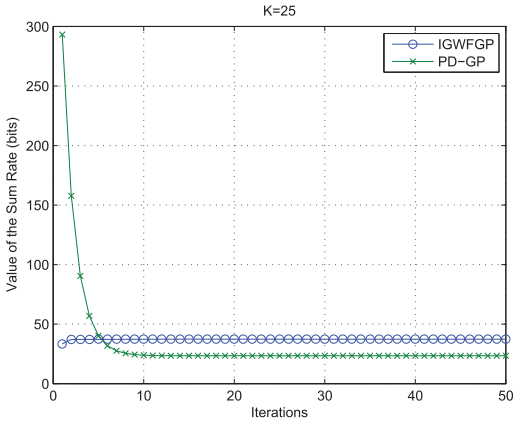

Fig. 2. Different levels between performances of IGWFGP and PD-GP, as K = 25.

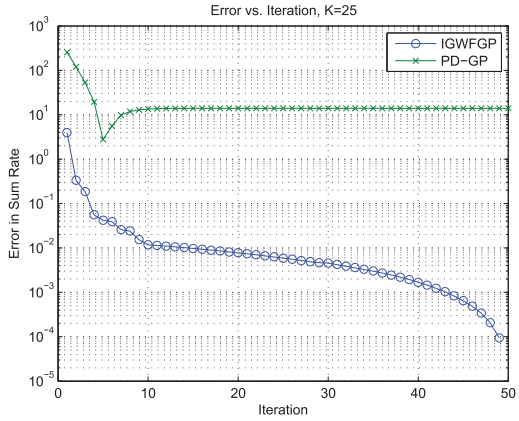

Fig. 3. Error functions of IGWFGP and PD-GP, as K = 25.

V. Numerical Results

In this section, numerical results are provided to illustrate the effectiveness of the proposed algorithm. For fair comparison, the proposed IGWFGP and PD-GP in their inner loop employ the same computational amount. In the presented figures, the circle markers and the cross markers represent the results of the proposed IGWFGP and PD-GP, respectively.

Let the numbers of the antennas at the base station (N_r) and at each mobile station (N_t) be set as 5 respectively. The number of user, $K = 25$. Channel gain matrices are generated randomly using random $N_r \times N_t$ matrices with each entry drawn independently from the standard Gaussian distribution. $\{P_k = k|k = 1, \ldots, K.\}$ is the set of the chosen positive numbers. The sum power constraint is $P_T = 60$, *i.e.*, about 17.8 dB. The presented results are the averaging of 100 independent simulation runs. This averaging obtains the average of the performances in the samples, where each of the samples is $(\mathbf{H}_1, \ldots, \mathbf{H}_K)$. As the sample size grows, the average of the performances can approximate above or below the mean of the performance which is a random variable.

Initial value for PD-GP is referred to in 1) of Algorithm PD-GP. Further, we may take $\lambda_{\min} = 0$ and $\lambda_{\max} = 1$, *i.e.*, the domain of the dual function is the interval of [0, 1]. This result comes from the KKT conditions of a derived model of the dual function $f_d(\lambda)$ minimization problem. The proof is simple and ignored here since IGWFGP is the main concern of this paper.

Fig. 2 compares the sum rate vs. the number of iterations for the proposed algorithm IGWFGP with the reference algorithm PD-GP. The proposed algorithm achieves higher objective function values with less number of iterations for convergence. Algorithm PD-GP under the primal-dual approach cannot achieve the optimal solution.

Let f^* be the maximum throughput, $f^{(n)}$ the throughput at the n-th iteration and $|f^{(n)} - f^*|$ the error in the throughput. Fig. 3 shows the corresponding error in the throughput versus the number of iterations. Note that using the fixed-point theory from the convergence proof of the proposed algorithm can determine the maximum throughput f^*. PD-GP algorithm exhibits an error floor even with the increasing of the iteration.

In addition, if we replace GWFGP of IGWFGP with PD-IPM. This newly formed algorithm, called the iterative PD-IPM (or IPD-IPM), under the same comparison approach, is taken as another comparison reference. Fig. 4 investigates the impact of the number of users/channels, K, on the sum rate of these three algorithms. It is shown that the proposed IGWFGP achieves the highest sum rate.

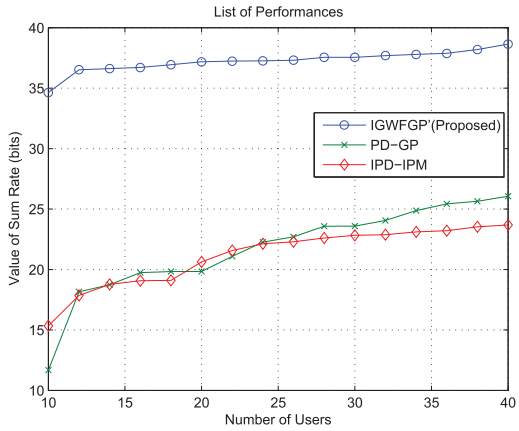

Fig. 4. Throughput (Unit: bits) of IGWFGP, PD-GP and IPD-IPM, as K = 10, 12, 14, . . ., 38 and 40.

Next, we define the required number of iteration to achieve ϵ precision as

$$N_{\text{IGWFGP}} \triangleq \min\{n||f^{(j)} - f^*| < \epsilon, \text{ as } j \geq n\}. \tag{30}$$

It is seen that the point $\{(j, f^{(j)})\}$ is generated by the corresponding algorithm and $\epsilon = 10^{-3}$ without loss of generality. Replacing the subscript of N_{IGWFGP} with "PD-GP", $N_{\text{PD-GP}}$ is similarly defined, too. For different selection of K, we have the corresponding results in Table II. Due to the fact that all $N_{\text{PD-GP}}$ being infinity, $N_{\text{PD-GP}}$ is not listed. With different number of users, the achieved throughput gain of IGWFGP over PD-GP is in the range of 48% to 197%. When K is small, the gain is more significant.

Fig. 5 shows the sum rate as a function in the number of antennas (where $N_t = N_r$) for the three algorithms when $K = 10$. With the increasing of the number of antennas, the achieved sum rate by using IGWFGP improves more quickly than other two algorithms.

Fig. 6 shows the sum rate as a function in the sum power P_T. To avoid the trivial case of the individual peak power upper

TABLE I
LIST OF VARIABLES AND ABBREVIATIONS

Key variables & abbreviations	Representations or interpretations
k^*	water level step (highest step under water)
$\mathcal{E}$	expectation on probability
$S_k(j, j)$	the jth entry at the major diagonal for matrix S_k
$a_k(j)$	the jth entry at the kth array
K	total number of users
$\mathbf{H}_i$	channel representation of user i
$\mathbf{x^i}$	column vector, signal transmitted by user i
$\mathbf{S_i}$	$E[\mathbf{x^i}\mathbf{x^i}^\dagger]$, a matrix
$\mathrm{Tr}(S_i)$	trace to represent power used by user i
$\mathbf{G}_i$	effective channel for user i (refer to [6])
$\mathcal{D}_i$	diagonal matrix, unitarily similar to $[\mathbf{G}_i\ (\mathbf{G}_i)^\dagger]$
Λ	set of the indexes. If a group accepts its sum power greater than upper bound, the index of this group is a member in the set
Λ_i	set of indexes for group i, for $1 \le i \le K$

TABLE II
COMPARISON OF THE ACHIEVED THROUGHPUT ($N_t = N_r = 5$)

parameter	N_{IGWFGP}	PD-GP	IGWFGP	Gain (%)
$K = 10$	7	11.6758	34.6412	197%
$K = 12$	8	18.1563	36.5232	101%
$K = 14$	8	18.7412	36.6152	95%
$K = 16$	8	19.7391	36.7092	86%
$K = 18$	8	19.8335	36.9401	86%
$K = 20$	8	19.8396	37.1791	87%
$K = 22$	8	21.0835	37.2418	77%
$K = 24$	8	22.2663	37.2614	67%
$K = 26$	9	22.7063	37.3102	64%
$K = 28$	10	23.5805	37.5489	59%
$K = 30$	11	23.5929	37.5501	59%
$K = 32$	18	24.0552	37.6840	57%
$K = 34$	26	24.8757	37.7979	52%
$K = 36$	30	25.4256	37.8786	49%
$K = 38$	30	25.6462	38.1890	49%
$K = 40$	32	26.0662	38.6371	48%

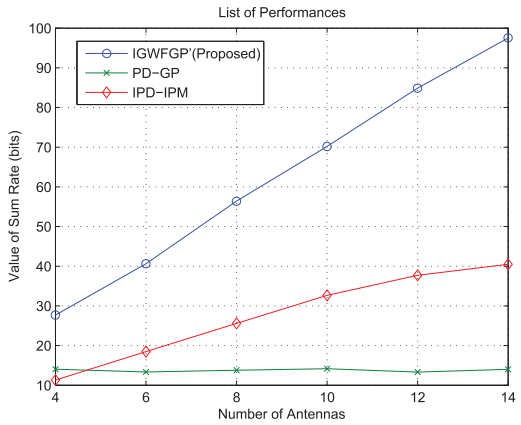

Fig. 5. Throughput (Unit: bits) of IGWFGP, PD-GP and IPD-IPM, as $N_t = 4$, 6, 8, 10, 12, and 14.

bound sum being less than P_T, let $K = 12$. For each sample of the one hundred random experiments on $(\mathbf{H}_1, \ldots, \mathbf{H}_K)$, such a sample keeps the same for varying P_T. According to such random experiment arrangements, IGWFGP, PD-GP and IPD-IPM can lift the obtained sum rates respectively, as the value of P_T increases, for the non-trivial case.

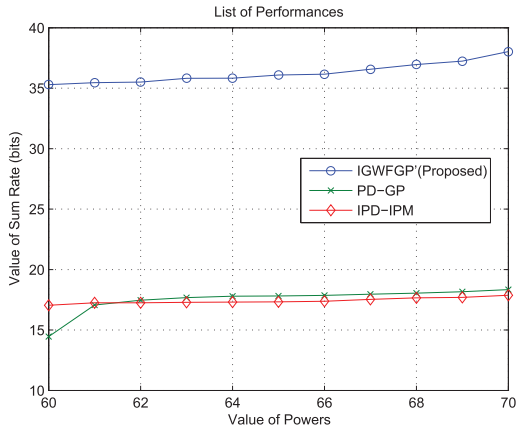

Fig. 6. Throughput (Unit: bits) of IGWFGP, PD-GP and IPD-IPM, as $P_T = 60$, 61, . . ., 69 and 70.

Fig. 7. Throughput (Unit: bits) of IGWFGP, PD-GP and IPD-IPM, as $\{P_k\} = \{k\}_{k=1}^6, \{k+2\}_{k=1}^6, \{k+4\}_{k=1}^6, \{k+6\}_{k=1}^6, \{k+8\}_{k=1}^6$ and $\{k+10\}_{k=1}^6$.

Fig. 7 shows the sum rate as a function of the individual peak power constraint, where $\{P_k\} \in \{\{k+2n\}_{k=1}^6\}_{n=0}^5$. That is to say, $\{P_k\}_{k=1}^6 = \{k+2n\}_{k=1}^6$, for $n = 0, 1, \ldots, 5$. Further, $\{P_1, P_2, \ldots, P_6\} = \{1, 2, \ldots, 6\}$; $\{1+2, 2+2, \ldots, 6+2\}$; $\{1+4, 2+4, \ldots, 6+4\}$; $\{1+6, 2+6, \ldots, 6+6\}$; $\{1+8, 2+8, \ldots, 6+8\}$; and $\{1+10, 2+10, \ldots, 6+10\}$, respectively. According to this meaning, the assignment of $\{P_k\}_{k=1}^6$ can be interpreted based on the caption of Fig. 7. The sum rate increases with the increase of the peak power for two reference algorithms, but no significant increase for IGWFGP.

From these results, we can observe that the proposed IGWFGP based on GWFGP is convergent; the comparison reference algorithms, PD-GP based on BACC, and IPD-IPM based on PD-IPM are not convergent. These results demonstrate efficiency of the proposed IGWFGP.

VI. CONCLUSION

In this paper, IGWFGP is proposed based on GWFGP to solve the throughput maximization problem in the networks of GMU-MIMO MAC. It is formally proven to be convergent through the utilization of the designed fixed point theory. To the best knowledge of the authors, there is no existing algorithm reported in the open literature to have solved the target problem,

under the merit of convergence of an algorithm, while utilizing the differentiability correctly. By exploiting the concept of variable weighting factor for covariance update, IGWFGP achieves fast convergence of the throughput maximization computation. For the target problem, the proposed result shows that our algorithm uses less number of iterations and achieves the optimal system throughput, especially for large scale systems, while the existing optimization methods cannot handle the target problem, even including the most efficient primal-dual interior point method.

Appendix A
Proof of Proposition 3.1

If the final set E in GWFGP is empty, it implies that $\sum_{i=1}^{K} g_i P_i \le P_T$. Successively, since the optimal solution $\{s_j\}_{j\in\Lambda_i}$, for group i, satisfies $\sum_{j\in\Lambda_i} s_j \le P_i$, through appending such solutions of all the groups, we can obtain the solution to the problem (8). It is easy to see that optimality of the mentioned $\{s_j\}_{j\in\Lambda_i}$ above is guaranteed, due to no peak power constraint being considered for each of the groups (referring to (16)). Also, it is easy to see that optimality of the appended solution comes from two facts: the empty final set E; and structure of the problem (8).

Assume the final set E is not empty. For constructing the optimal dual variables for the problem (8), the symbols: $\{\lambda_E, \overline{\sigma}_E, \{\underline{\sigma}_j | j \in E\}\}$, $\{\lambda_{\Lambda_i}, \overline{\sigma}_{\Lambda_i}, \{\underline{\sigma}_j | j \in \Lambda_i\}\}$ and $\{\lambda_{E_0}, \overline{\sigma}_{E_0}, \{\underline{\sigma}_j | j \in E_0\}\}$ are introduced, here. Then symbols will be assigned values by the following.

Since the final set E is non-empty, it implies that

$$\frac{1}{g_{\sigma(k^*)}\left(\frac{1}{a_{k^*}} + s_{k^*}\right)} = \frac{1}{g_{\sigma(j)}\left(\frac{1}{a_j} + s_j\right)}, \text{ as } \{j, k^*\} \subset E$$

and $s_j > 0$. Let $\lambda_E = \frac{1}{g_{\sigma(k^*)}\left(\frac{1}{a_{k^*}} + s_{k^*}\right)}$. According to the definitions of k^* and s_{k^*}, for $j \in E$ and $s_j = 0$,

$$\frac{1}{g_{\sigma(k^*)}\left(\frac{1}{a_{k^*}} + s_{k^*}\right)} > \frac{1}{g_{\sigma(j)}\left(\frac{1}{a_j} + s_j\right)},$$

let $\underline{\sigma}_j = \frac{1}{g_{\sigma(k^*)}\left(\frac{1}{a_{k^*}} + s_{k^*}\right)} - \frac{1}{g_{\sigma(j)}\left(\frac{j}{a_j} + s_j\right)} > 0$ and it is seen that $\overline{\sigma}_E = 0$; for $j \notin E$, let $j \in \Lambda_i$. Then

$$\sum_{j\in\Lambda_i} s_j = P_i, \text{ and}$$

$$\lambda_{\Lambda_i} = \frac{1}{g_{\sigma(k^*(\Lambda_i))}\left(\frac{1}{a_{k^*(\Lambda_i)}} + s_{k^*(\Lambda_i)}\right)} = \frac{1}{g_{\sigma(j)}\left(\frac{1}{a_j} + s_j\right)},$$

as $s_j > 0$. If $s_j = 0$, then

$$\underline{\sigma}_j = \frac{1}{g_{\sigma(k^*(\Lambda_i))}\left(\frac{1}{a_{k^*(\Lambda_i)}} + s_{k^*(\Lambda_i)}\right)} - \frac{1}{g_{\sigma(j)}\left(\frac{1}{a_j} + s_j\right)} > 0$$

and $\overline{\sigma}_{\Lambda_i} = 0$. Similarly, (12)–(14) in the initial utilization can obtain s_{k^*}, which can be written as $s_{k^*(E_0)}$. Thus we can obtain

$$\lambda_{E_0} = \frac{1}{g_{\sigma(k^*(E_0))}\left(\frac{1}{a_{k^*(E_0)}} + s_{k^*(E_0)}\right)},$$

$\underline{\sigma}_j$ and $\overline{\sigma}(E_0)$. (12)–(14) lead to that $\lambda_{E_0} \le \lambda_{\Lambda_i}$, where $\forall \Lambda_i \cap E = \emptyset$, and $\lambda_E \le \Lambda_{E_0}$. Hence, we have obtained $\{\lambda_E, \overline{\sigma}_E, \{\underline{\sigma}_j | j \in E\}\}$, $\{\lambda_{\Lambda_i}, \overline{\sigma}_{\Lambda_i}, \{\underline{\sigma}_j | j \in \Lambda_i\}\}$ and $\{\lambda_{E_0}, \overline{\sigma}_{E_0}, \{\underline{\sigma}_j | j \in E_0\}\}$.

Therefore, there exist the Lagrange multipliers $\lambda, \{\underline{\sigma}_j\}_{j=1}^{KN_t}$ and $\{\overline{\sigma}_i\}_{i=1}^{K}$, the Lagrange function of which, for the problem (8), is:

$$\begin{aligned} &L\left(\{s_i\}, \lambda, \{\overline{\sigma}_i\}, \{\underline{\sigma}_i\}\right) \\ &= \sum_{k=1}^{KN_t} \log\left(1 + a_k s_k\right) - \lambda\left(\sum_{j=1}^{KN_t} g_{\sigma(j)} s_j - P_T\right) \\ &\quad - \sum_{i=1}^{K} \overline{\sigma}_i \left(\sum_{j\in\Lambda_i} s_j - P_i\right) + \sum_{k=1}^{KN_t} \underline{\sigma}_k s_k, \end{aligned}$$

where $\lambda = \lambda_E$; $\overline{\sigma}_j = \overline{\sigma}_E$, as $j \in E$; $\overline{\sigma}_j = \overline{\sigma}_{\Lambda_i} + \left(\lambda_{\Lambda_i} - \lambda_E\right) g_j$, as $j \in \Lambda_i$ and $\forall i$; and the other Lagrange multipliers have been assigned above. By observation, they satisfy the KKT conditions. Since the problem (8) is a differentiable convex optimization problem with linear constraints, not only are the KKT conditions mentioned above sufficient, but they are also necessary for optimality. Note that it is seen that the constraint qualification of the problem (8) holds. Proposition 3.1 hence is proved.

Appendix B
Proof of Lemma 4.2

Note that in the following proof, we use the notation n to stand for number of the iterations for convenience.

The necessity is proved first. For any convergent subsequence of the points generated by IGWFGP, letting $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ be the limit of the convergent subsequence, there is a convergent subsequence. This sequence is labelled as $\left\{(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)})\right\}_{k=0}^{\infty}$ $(\subset \left\{\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right)\right\}_{n=0}^{\infty})$ where $\left\{(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)})\right\}_{n=0}^{\infty}$ is the point sequence generated by IGWFGP, such that

$$\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) = \lim_{k\to\infty}\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right).$$

It is seen that

$$\begin{aligned} &\left(\mathbf{S}_1^{(n_k+1)}, \ldots, \mathbf{S}_K^{(n_k+1)}\right) \in \arg\max_{(\mathbf{S}_1,\ldots,\mathbf{S}_K)\in V} \\ &\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right). \end{aligned}$$

This relationship comes from the definition of IGWFGP. The definition of IGWFGP implies that

$$\begin{aligned} &\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{S}_i^{(n+1)}, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) \\ &\quad \ge \sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{S}_i, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right), \end{aligned} \quad (31)$$

for any n and $(\mathbf{S}_1, \ldots, \mathbf{S}_K) \in V$. Replacing n with n_k, we obtain:

$$\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i^{(n_k+1)}, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right) \geq \sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right). \quad (32)$$

We have the following relationships:

$$\begin{aligned} & f\left(\mathbf{Q}_1^{(n+1)}, \ldots, \mathbf{Q}_{i-1}^{(n+1)}, \mathbf{Q}_i^{(n+1)}, \mathbf{Q}_{i+1}^{(n+1)}, \ldots, \mathbf{Q}_K^{(n+1)}\right) \\ & \geq f\left(\frac{K-1}{K}(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}) + \frac{1}{K}\left(\mathbf{S}_1^{(n+1)}, \ldots, \mathbf{S}_K^{(n+1)}\right)\right) \\ & = f\left(\sum_{i=1}^{K} \frac{1}{K}\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{S}_i^{(n+1)}, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right)\right) \\ & \geq \frac{1}{K}\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{S}_i^{(n+1)}, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) \\ & \geq \frac{1}{K}\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{Q}_i^{(n)}, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) \\ & = f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right). \end{aligned} \quad (33)$$

Between relationships mentioned above, the first inequality and the first equality hold due to step 3) of IGWFGP; the second inequality results from the function f being concave; the third inequality and the second equality are true because of step 2) of IGWFGP, i.e., the definition of $(\mathbf{S}_1^{(n+1)}, \ldots, \mathbf{S}_K^{(n+1)})$.

Thus, $f(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)})$ is monotonically increasing with respect to n increasing, and

$$\begin{aligned} & f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) \\ & \leq \frac{1}{K}\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_{i-1}^{(n)}, \mathbf{S}_i^{(n+1)}, \mathbf{Q}_{i+1}^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) \\ & \leq f\left(\mathbf{Q}_1^{(n+1)}, \ldots, \mathbf{Q}_K^{(n+1)}\right). \end{aligned} \quad (34)$$

From (34), we obtain:

$$\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i^{(n_k+1)}, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right) \leq K f\left(\mathbf{Q}_1^{(n_k+1)}, \ldots, \mathbf{Q}_K^{(n_k+1)}\right).$$

From (32), we acquire:

$$\sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i^{(n_k+1)}, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right) \geq \sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right).$$

Hence, it is true that

$$Kf\left(\mathbf{Q}_1^{(n_{k+1})}, \ldots, \mathbf{Q}_K^{(n_{k+1})}\right) \geq \sum_{i=1}^{K} f\left(\mathbf{Q}_1^{(n_k)}, \ldots, \mathbf{Q}_{i-1}^{(n_k)}, \mathbf{S}_i, \mathbf{Q}_{i+1}^{(n_k)}, \ldots, \mathbf{Q}_K^{(n_k)}\right).$$

Letting k approach to the infinity, we may acquire that

$$\begin{aligned} & \sum_{i=1}^{K} f\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) = Kf\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) \\ & \geq \sum_{i=1}^{K} f\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_{i-1}, \mathbf{S}_i, \overline{\mathbf{Q}}_{i+1}, \ldots, \overline{\mathbf{Q}}_K\right), \end{aligned}$$

where $\forall(\mathbf{S}_1, \ldots, \mathbf{S}_K) \in V$. Thus,

$$\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) \in \arg \max_{(\mathbf{S}_1, \ldots, \mathbf{S}_K) \in V} \sum_{i=1}^{K} f\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_{i-1}, \mathbf{S}_i, \overline{\mathbf{Q}}_{i+1}, \ldots, \overline{\mathbf{Q}}_K\right).$$

Note that the set

$$\arg \max_{(\mathbf{S}_1, \ldots, \mathbf{S}_K) \in V} \sum_{i=1}^{K} f\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_{i-1}, \mathbf{S}_i, \overline{\mathbf{Q}}_{i+1}, \ldots, \overline{\mathbf{Q}}_K\right)$$

is not guaranteed to be a single-point set. However, we may choose $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ is an optimal solution to the problem $\max_{(\mathbf{S}_1, \ldots, \mathbf{S}_K) \in V} \sum_{i=1}^{K} f\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_{i-1}, \mathbf{S}_i, \overline{\mathbf{Q}}_{i+1}, \ldots, \overline{\mathbf{Q}}_K\right)$. This corresponds to step 2) of IGWFGP. Further, $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K) = \beta^*(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K) + (1-\beta^*)(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$, based on the choice of the optimal solution mentioned above. This corresponds to step 3) of IGWFGP.

Therefore, resulting from the two correspondences mentioned above and the definition of IGWFGP, it is true that $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ is a fixed point under IGWFGP, which is viewed as a mapping.

The sufficiency will be proved as follows:

Let $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ represent a fixed point under IGWFGP. Assume that $(\mathbf{Q}_1^{(0)}, \ldots, \mathbf{Q}_K^{(0)})$ to be a fixed point. It is denoted by $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$. Then $(\mathbf{Q}_1^{(1)}, \ldots, \mathbf{Q}_K^{(1)}) = (\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$, i.e., the former is assigned by the latter, due to $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ being a fixed point under IGWFGP. If it is assumed that $(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}) = (\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$, then $(\mathbf{Q}_1^{(n+1)}, \ldots, \mathbf{Q}_K^{(n+1)}) = (\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ due to $(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K)$ being a fixed point under IGWFGP. According to the principle of mathematical induction,

$$\left(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}\right) = \left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) \in V, \forall n.$$

Furthermore, $\lim_{n\to\infty}(\mathbf{Q}_1^{(n)}, \ldots, \mathbf{Q}_K^{(n)}) = \left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) \in V$. Therefore, the sufficiency is true.

Note that in the proving process above, we do not have the following assumption:

$$\left(\overline{\mathbf{Q}}_1, \ldots, \overline{\mathbf{Q}}_K\right) = \lim_{k\to\infty}\left(\mathbf{Q}_1^{(n_k+1)}, \ldots, \mathbf{Q}_K^{(n_k+1)}\right).$$

ACKNOWLEDGMENT

The authors are grateful to the editor and the anonymous reviewers for their valuable comments and suggestions to improve quality of this article.

REFERENCES

[1] R. Liao, B. Bellalta, T. C. Minh, J. Barceló, and M. Oliver, "Uni-MUMAC: A unified down/up-link MU-MIMO MAC protocol for IEEE 802.11ac WLANs," *Wireless Netw.*, vol. 21, no. 5, pp. 1457–1472, 2015.

[2] J. Xu, S. Bi, and R. Zhang, "Multiuser MIMO wireless energy transfer with coexisting opportunistic communication," *IEEE Wireless Commun. Lett.*, vol. 4, no. 3, pp. 273–276, Jun. 2015.

[3] Y. Zhang, E. Dall'Anesei, and G. Giannakis, "Distributed optimal beamformers for cognitive radios robust to channel uncertainties," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6495–6508, Dec. 2012.

[4] S. Kim and G. Giannakis, "Optimal resource allocation for MIMO ad hoc cognitive radio networks," *IEEE Trans. Inf. Theory*, vol. 57, no. 5, pp. 3117–3131, May 2011.

[5] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[6] N. Jindal, W. Rhee, S. Vishwanath, S. A. Jafar, and A. Goldsmith, "Sum power iterative water-filling for multi-antenna Gaussian broadcast channels," *IEEE Trans. Inf. Theory*, vol. 51, no. 4, pp. 1570–1580, Apr. 2005.

[7] W. Yu, W. Rhee, S. Boyd, and J. M. Cioffi, "Iterative water-filling for Gaussian vector multi-access channels," *IEEE Trans. Inf. Theory*, vol. 50, no. 1, pp. 145–152, Jan. 2004.

[8] E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecommun.*, vol. 10, pp. 585–596, 1999.

[9] D. Palomar and J. R. Fonollosa, "Practical algorithms for a family of waterfilling solutions," *IEEE Trans. Signal Process.*, vol. 53, no. 2, pp. 686–695, Feb. 2005.

[10] C. Hsu, H. Su, and P. Lin, "Joint subcarrier pairing and power allocation for OFDM transmission with decode-and-forward relaying," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 399–414, Jan. 2011.

[11] Y. Rong, X. Tang, and Y. Hua, "A unified framework for optimizing linear non-regenerative multicarrier MIMO relay communication systems," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 4837–4852, Dec. 2009.

[12] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2007.

[13] D. Luenberger, *Optimization by Vector Space Methods*, 1st ed. New York, NY, USA: Wiley-Interscience, 1997.

[14] W. Zangwill, *Nonlinear Programming: A Unified Approach*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1969.

[15] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA, USA: Athena Scientific, 1999.

[16] P. He and L. Zhao, "Correction of convergence proof for iterative water-filling in Gaussian MIMO broadcast channels," *IEEE Trans. Inf. Theory*, vol. 57, no. 4, pp. 2539–2543, Apr. 2011.

Peter He (S'15–M'15) received the M.A.Sc. degree in electrical engineering from McMaster University, Hamilton, ON, Canada, in 2009, the Ph.D. degree in electrical engineering from Ryerson University, Toronto, ON, Canada, in 2015. His research interests include the union of large-scale optimization, information theory, communications, and smart power grid. He was the recipient of the Graduate Research Excellence Award in electrical and computer engineering at Ryerson University (continuously), from 2010–present; and the Best Paper Award of the 2013 International Conference on Wireless Communications and Signal Processing (WCSP).

Lian Zhao (S'99–M'03–SM'06) received the Ph.D. degree in electrical and computer engineering (ELCE) from the University of Waterloo, Waterloo, ON, Canada, in 2002. She joined the Department of Electrical and Computer Engineering, Ryerson University, Toronto, ON, Canada, as an Assistant Professor in 2003, and became an Associate Professor in 2007, and then a Professor in 2014. Her research interests include wireless communications, radio resource management, power control, cognitive radio and cooperative communications, design and applications of the energy efficient wireless sensor networks. She has served as the symposium Co-Chair for the IEEE Global Communications Conference (GLOBECOM) 2013 Communication Theory Symposium; Web-conference Co-Chair for the IEEE Toronto International Conference Science and Technology for Humanity (TIC-STH) 2009; the Guest Editor for *International Journal on Communication Networks and Distributed Systems*, special issue on Cognitive Radio Networks in 2011; the IEEE Ryerson Student Branch Counselor since 2005; the Associate Chair for Graduate Studies at ELCE Department from 2013 to 2015. She has been an Associate Editor for the IEEE TRANSACTION ON VEHICULAR TECHNOLOGY since 2013. She has also served as a technical program committee (TPC) Member for numerous IEEE flagship conferences, and reviewer for the IEEE Transaction journals and for various NSERC (Natural Sciences and Engineering Research Council) proposals. She is a Licensed Professional Engineer in Ontario. She was the recipient of the Canada Foundation for Innovation (CFI) New Opportunity Research Award in 2005; Ryerson Faculty Merit Award in 2005 and 2007; the Faculty Research Excellence Award from ELCE Department of Ryerson University in 2010, 2012, and 2014; and the Best Student Paper Award (with her student) from Chinacom in 2011, the Best Paper Award (with her student) from 2013 International Conference on Wireless Communications and Signal Processing (WCSP).