

GEOMETRIC WATER-FILLING AND RADIO RESOURCE MANAGEMENT

by

Zhiming He

Master of Applied Science in Electrical Engineering

McMaster University, Hamilton, Canada, 2009

A Dissertation

Presented to Ryerson University

in partial fulfilment of the

requirements for the degree of

Doctor of Philosophy

in the Program of Electrical and Computer Engineering

Toronto, Ontario, Canada, 2015

© Zhiming He, 2015

Author's Declaration

(AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A DISSERTATION)

I hereby declare that I am the sole author of this dissertation. This is a true copy of the dissertation, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this dissertation to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this dissertation by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

Abstract

GEOMETRIC WATER-FILLING AND RADIO RESOURCE MANAGEMENT

© Zhiming He, 2015

Doctor of Philosophy

Electrical and Computer Engineering

Ryerson University

This thesis considers the radio resource management (RRM) of advanced wireless communication systems. With the emerging of more advanced and more complicated systems, such as cognitive radio, nodes with energy harvesting capacities (green communications), and the application of Multiple-Input Multiple-Output (MIMO) technology, RRM problems introduce more difficulties and challenges to optimize system performances. Due to specific structure of communication systems, water-filling (WF) plays an important role in RRM. This thesis introduces the fundamental theory and development of WF algorithm. The proposed Geometric Water-Filling (GWF) is presented and compared with the conventional WF algorithms. It can break through the limitations of the conventional WF to solve the more complicated optimization problems in the advanced wireless communication systems. For the application of the proposed GWF to solve the RRM problems in the advanced MIMO communication systems, cognitive radio communication systems, green communication systems and the “dual problems”, which are the sum power minimization problems, of the throughput maximization problems is investigated in this thesis. Efficient algorithms are presented to achieve the optimal resource allocation.

Acknowledgements

The thesis is written under the guidance and with the help of my supervisor, Dr. Lian Zhao, whose valuable advices and extended knowledge help me all along. I learn from her the precious research experience, and how it combines with and applies to wireless communications, especially on RRM. I would like to express my sincere gratitude to my supervisor Dr. Lian Zhao, for her continuous guidance and support. I am also grateful to my friends, teachers and classmates.

My special thanks go to all the members of the examination committee: Dr. Beheshti, Dr. Cai, Dr. Fernando, Dr. Misic, and Dr. Zhao (in the alphabetical order).

Finally, I am indebted to thank all the members of my family, especially my wife and son, for their patience, understanding and continuous support.

Contents

Author's Declaration	ii
Abstract	iii
Acknowledgements	iv
List of Figures	x
List of Tables	xi
Notation and Acronyms	xii
1 Introduction	1
1.1 Multiple Input Multiple Output System	1
1.2 Energy Harvesting Technology	2
1.3 Water-Filling Algorithm	2
1.4 Organization of Thesis	3
2 Geometric Water-Filling in RRM	4
2.1 Problem Statement and Water-Filling	4
2.2 Proposed Geometric Water-Filling Approach	5
2.3 Generalization of Weighted Case	8
2.4 Weighted Water-Filling with Individual Peak Power Constraints	12
2.5 Weighted Water-Filling with Group Bounded Power Constraints	14
2.6 Numerical Results and Complexity Analysis	19
2.6.1 Numerical Results	19

2.6.2	Complexity Analysis	24
3	RRM in Wireless Communications with Energy Harvesting Technology	26
3.1	Introduction	26
3.2	Extended GWF for Preparation of EH system	29
3.3	Maximizing Throughput in Fading Channel and Algorithm RGWF	32
3.3.1	Problem Statement and Conventional Approach	32
3.3.2	Recursive Geometric Water-Filling and Its Optimality	33
3.4	Transmission Completion Time Minimization	38
3.5	Numerical Examples for RRM with Energy Harvesting Transmission	42
4	Optimal Recursive Power Allocation for Energy Harvesting System with Multiple Antennas	47
4.1	Introduction	48
4.1.1	Our Work	48
4.2	System Model and Proposed Problems	49
4.3	RGWFM - Recursive GWF for Multiple Antennas	52
4.4	Transmission Completion Time Minimization	55
4.4.1	Discrete Transmission Completion Time Minimization	55
4.4.2	Continuous Transmission Completion Time Minimization	59
4.5	Extension to a Hybrid System Coexisting with Grid Power	60
4.6	Numerical Examples and Computational Complexity	62
4.6.1	Numerical Examples	62
4.6.2	Computational Complexity Analysis	68
5	Exact Solution of Non-Commutative Composite Water-Fillings for EH and Smart Power Grid Coexisting System with Peak Power Constraints	70
5.1	Introduction	71
5.2	Problem Statement	72
5.3	Hybrid Power Allocation Algorithm 1 (HPA1) for Maximum Throughput	74
5.3.1	Two parts of HPA1(K) and Their Non-Commutativity	75
5.3.2	Optimality of HPA1	78

5.4	Numerical Example and Computational Complexity	81
5.4.1	Numerical Example	81
5.4.2	Computational Complexity Analysis	84
6	Using GWF to Solve A Class of Sum Power Minimization Problems	86
6.1	Introduction	86
6.1.1	Our Work for Sum Power Minimization Problem	87
6.2	Generalized Water-Filling for Sum Power Problem	89
6.2.1	Problem Statement and Conventional Approach	89
6.2.2	Illustration of the Proposed Generalized Water-Filling Algorithm (P-GWF) .	90
6.2.3	Extend to Weighted Case	93
6.3	Solving Generalized RRM Problem Using P-GWF	98
6.3.1	Weighted Water-Filling with Individual Peak Power Constraints (P-WFPP) for Sum Power Problems	98
6.3.2	Weighted Water-Filling with Group Peak Power Constraints (P-WFGPP) for Sum Power Problems	99
6.3.3	Weighted Water-Filling with Group Bounded Power Constraints for Sum Power Problem (P-WFGBP)	101
6.4	Numerical Results and Complexity Analysis	104
6.4.1	Complexity Analysis	106
7	Conclusions	108
	Bibliography	110

List of Figures

2.1	Illustration for the proposed Geometric Water-Filling (GWF) algorithm. (a) Illustration of water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = 1/a_i$. (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 $k = 3$. (d) Illustration of the weighted case.	7
2.2	Illustration for Example 1, results for CWF and GWFPP. (a) the CWF: without peak power restriction check ($s_1 = 3, s_2 = 0$); (b) the CWF: s_1 is clipped considering peak power constraint; (c) GWFPP: $s_1 = 2, s_2 = 1$	20
2.3	Illustration for Power Allocation using the CWF and the proposed GWFPP for Example 2. (a) Results for the CWF; (b) the CWF, clipped s_1 to s_4 due to peak power restrictions; (c) Results for GWFPP.	21
2.4	Optimal Power Allocation Results for Example 2. (a) solution from the the CWF; (b) 1st iteration results of GWFPP; (c) 2nd iteration results of GWFPP; (d) 3rd iteration results of GWFPP.	22
2.5	Illustration for Example 3 using GWFPP ($s_1 = 2, s_2 = 1$).	23
2.6	Illustration for Multi-User Power Allocation Results by GWFGBP.	24
3.1	Illustration of simple system instance	27
3.2	Illustration for the proposed Geometric Water-Filling (GWF) algorithm. (a) Illustration of water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = \frac{1}{a_i w_i}$. (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 $k = 3$. (d) Illustration of the weights as the widths.	30

3.3	Illustration for Algorithm RGWF (Line 6-15 for $L = 6$), harvested energy having been allocated up to epoch 5; horizontal-wave shadowed areas denote power allocation for the processing window; (a) $n = 6, k_e^* = 5$; (b) $n = 5, k_e^* = 4$; (c) $n = 4, k_e^* = 3$; (b) $n = 3, k_e^* = 2$;	35
3.4	Procedures to solve Example 1: (a) $s_1 = 1$; (b) $s_1 = 1, s_2 = 1$; (c) $s_1 = 1, s_2 = 1, s_3 = 1$	43
3.5	Procedures to solve Example 2: (a) $s_1 = 1$; (b) $s_1 = 0.75, s_2 = 1.25$; (c) $s_1 = 11/18, s_2 = 20/18, s_3 = 23/18$	43
3.6	Procedures to solve Example 3: (a) $s_1 = 2$; (b) $s_1 = 2, s_2 = 2$; (c) $s_1 = 2, s_2 = 2, s_3 = 1$	45
3.7	Procedures to solve Example 4: (a) $s_1 = 1$; (b) $s_1 = 0.75, s_2 = 1.25$; (c) $s_1 = 11/18, s_2 = 20/18, s_3 = 2399/2523$	46
4.1	Procedures to solve Example 2: (a) $\text{Tr}(\mathbf{S}_1) = 2$; (b) $\text{Tr}(\mathbf{S}_1) = \frac{3}{2}, \text{Tr}(\mathbf{S}_2) = \frac{5}{2}$; (c) $\text{Tr}(\mathbf{S}_1) = \frac{14}{12}, \text{Tr}(\mathbf{S}_2) = \frac{26}{12}, \text{Tr}(\mathbf{S}_3) = \frac{8}{3}$; (d) $\text{Tr}(\mathbf{S}_1) = \frac{7}{6} \doteq 1.17, \text{Tr}(\mathbf{S}_2) = \frac{26}{12} \doteq 2.17, \text{Tr}(\mathbf{S}_3) = 2 \log(4 \cdot (\frac{12}{19})^2) \doteq 1.35$	64
4.2	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=10 and 15 . . .	65
4.3	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=10 and 15 . . .	66
4.4	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=10 and 15 . . .	66
4.5	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=5	68
5.1	Weighted sum-rates (Unit: bits) of HPA1 and PD-IPM, as K=10	82
5.2	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as 15	82
5.3	Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=20	83
6.1	Illustration for the proposed generalized Water-Filling Algorithm for the sum power problems (P-GWF). (a) Water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = 1/a_i$. (b) $ER(k)$ (which is determined by the shadowed area, representing the total water/power, up to, but excluding step k) when $k = 2$. (c) $ER(k)$ when $k = 3$. (d) The weighted case.	91
6.2	Illustration for Examples 1, 3, 4 respectively.	104

6.3	Illustration for Example 2, results for the first, second and third iterations respectively.	105
-----	--	-----

List of Tables

4.1	Comparison of the achieved throughput ($N_r = 4$)	67
5.1	Comparison of the achieved throughput	83

Notation and Acronyms

Notation:

$A(\cdot)$	Mapping
$\mathbb{C}(H, P)$	Channel capacity
\mathbb{C}^n	Unitary space with dimension n
$\mathbb{C}^{m \times n}$	The set of $m \times n$ complex matrices
$\det(B)$ or $ B $	Determinant of the square matrix B
$E(\xi)$	Expected value of the random variable ξ
f_ξ	The probability density function of random variable ξ
H	Channel matrix
H^\dagger	The conjugate transpose of matrix H
H_i	The channel matrix of the i -th user
H_i^\dagger	The conjugate transpose of matrix H_i
H_i^T	The transpose of matrix H_i

Notation (Con't):

I_r	Identity matrix with dimension r
\log	Natural logarithm
$M \succeq 0$	Matrix M being positive semidefinite
$\text{Tr}(M)$	The trace of matrix M
x	The input vector of the channel or a vector
x^\dagger	The conjugate transpose of vector x
x_i	The i -th entry (scalar) or the i -th block of vector x
x^i	The input vector of the i -th user
$(x)_i$	the i -th entry (scalar) of vector x
$\lfloor \cdot \rfloor$	Floor function
$\lceil \cdot \rceil$	Ceiling function

Acronyms:

KKT	Karush-Kuhn-Tucker
WFA	Water-filling algorithm
IWFA	Iterative water-filling algorithm
MIMO	Multiple input multiple output
MAC	Multiple access channel
BC	Broadcast channel
CR	Cognitive radio
EH	Energy harvesting
GWF	Geometric water-filling

Chapter 1

Introduction

The world is demanding more from wireless communication services now than ever before. Many advanced wireless communication techniques, such as multiple-input multiple-output (MIMO), cognitive radio (CR), and energy harvesting (EH) communications have attracted lots of research attention. For a wireless network operating in a fading environment, power and bandwidth are precious radio resources which need careful planning. Reducing power consumption to satisfy the target QoS requirement leads to enlarged system capacity and prolonged battery life. With the evolution to the more complicated wireless communication systems, the issues of optimal radio resource management problems have become more and more important to achieve overall optimal system design.

Water-filling has been an important algorithm for radio resource management (RRM) problem in wireless communication systems. In this thesis, we present a new geometric water-filling (GWF) algorithm, and explore its application in solving RRM problems for the advanced wireless communication systems. In the remaining of this Chapter, we will first review the advances in MIMO, EH communication systems, and water-filling algorithm. In the subsequent chapters, we will further discuss the application of water-filling algorithm to solve RRM problems in these advanced communication systems.

1.1 Multiple Input Multiple Output System

The use of multiple antennas at the transmitter and receiver, *i.e.*, Multiple-Input Multiple-Output (MIMO) technology, constitutes a breakthrough ([5, p. 1]) in the design of wireless communication

systems. MIMO technology is now at the core of several existing and emerging wireless standards ([5, p. 18]). Exploiting multipath scattering, MIMO techniques have delivered significant performance enhancements in terms of data transmission rate and interference reduction on point-to-point links. In this thesis, we focus our attention on MIMO systems. Actually, we have also obtained the design of multi-user SIMO systems [18].

1.2 Energy Harvesting Technology

In recent years, energy harvesting in green communications has attracted a lot of research attention due to its environment friendly features. One possible technique to overcome the limitation of battery lifetime is to harvest energy from the environment. In such systems, harvesting energy has become a preferred choice for supporting green communications. However, harvesting energy depends on natural conditions and thus is random over time. It owns causality of power usage, corresponding to the dependence of harvested energy on time. As a result, the energy from this constraint is often considered to regulate the overall energy flow of the system. For convenience, this system is called the EH system in this thesis. The optimal radio resource allocation problem to maximize system throughput turns out to be more complicated. In this thesis, the proposed algorithms can be used for the maximum throughput problems and/or the minimum transmission completion time problems.

1.3 Water-Filling Algorithm

In many engineering problems, water-filling plays an important role in radio resource management. For communications, it stems from a class of the problems of maximizing the mutual information between the input and the output of a channel with parallel independent sub-channels. With water-filling, more power is allocated to the channels with higher gains to maximize the sum of data rates or the capacity of all the channels. The solution to this class of the problems can be interpreted by a vivid description as pouring limited volume of water into a tank, the bottom of which has the stair levels determined by the inverse of the sub-channel gains.

The conventional way to solve the water-filling problem is to solve the Karush-Kuhn-Tucker

(KKT) conditions [2], and then find the water-level(s) and the solutions. In Chapter 2 of our paper [19], we proposed a water-filling algorithm from geometric approach (GWF). GWF provides the exact solutions, by a finite amount of computation or the basically arithmetic and logic operation, to the power allocation problem and avoids the complexity to solve the KKT conditions with non-linear equations. Due to complexity of solving the KKT conditions of the problem with multiple variables, the GWF is easier to compute than the conventional water-filling and reveals more useful information. GWF has also been extended to solve more general and more complicated power allocation problems, *e.g.*, the minimum sum power with the throughput and more constraints in the following Chapter 6, or [21].

1.4 Organization of Thesis

In this thesis, GWF is proposed to compute the exact solutions to the maximum throughput problems with more constraints in Chapter 2. For the EH system, the coexisting system of EH with MIMO, and the coexisting system of EH with power grid, new algorithms are proposed to compute the exact solutions to the maximum throughput problems and/or the minimum transmission completion time problems in Chapters 3-5, respectively. A set of novel algorithms to compute the exact solutions to the sum power minimization problems with more constraints are proposed in Chapter 6. Chapter 7 closes this thesis.

Chapter 2

Geometric Water-Filling in RRM

In this thesis, we firstly introduce water-filling algorithms to solve power allocation problems. Two water-filling approaches are presented. One is the conventional water-filling (CWF); and the other one is the proposed geometric water-filling (GWF). GWF is further extended to efficiently solve a class of power allocation problems with more complex structure which owns upper bounds of the power variables. Computational complexities are investigated.

2.1 Problem Statement and Water-Filling

The water-filling problem can be abstracted and generalized into the following problem: given $P > 0$, as the total power or volume of the water; the allocated power and the propagation path gain for the i th channel are given as s_i and a_i respectively, $i = 1 \dots K$; and K is the total number of channels. Letting $\{a_i\}_{i=1}^K$ be a sorted sequence, which is positive and monotonically decreasing, find that

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \forall i; \\ & \sum_{i=1}^K s_i = P. \end{aligned} \tag{2.1}$$

Since the constraints are that (i) the allocated power to be nonnegative; (ii) the sum of the power equals P , the problem (2.1) is called the water-filling (problem) with sum power constraint. *Note, according to information theory, that s_i of $\log(1 + a_i s_i)$ in (2.1) is a ratio, i.e., a numerical value without any physical unit, under the assumption that the power of noise is 1. It is seen that this assumption does not lose any generality. Thus, the s_i is the signal noise ration (SNR), although*

this value is equal to the numerical value part of the transmitted signal power, under the assumption that the power of noise is 1. Therefore, it does not depend on choice of physical units, such as Watt or mW.

To find the solution to problem (2.1), we usually start from the Karush-Kuhn-Tucker (KKT) conditions of the problem, as a group of the optimality conditions, and derive the system (2.2) below from the KKT conditions,

$$\begin{cases} s_i = \left(\mu - \frac{1}{a_i}\right)^+, \text{ for } i = 1, \dots, K, \\ \sum_{i=1}^K s_i = P, \\ \mu \geq 0, \end{cases} \quad (2.2)$$

where $(x)^+ = \max\{0, x\}$. μ is the water level chosen to satisfy the power sum constraints with equality ($\sum_{i=1}^K s_i = P$). The solution to (2.2) is referred to as a solution of the CWF problem (2.1).

It can be seen that the implied system (2.2) has been used to find the optimal solution. The existence of its Lagrange multipliers and the implication mentioned above determine that enumeration can be utilized to find the water level μ . In [36], how to solve the problems has been discussed extensively. Complexity of the non-geometric approach to solve the problem (2.1) will be discussed in Section 2.6. In the sequel of the chapter, when water-filling problem is mentioned, the power sum constraint is always included.

2.2 Proposed Geometric Water-Filling Approach

In this chapter, we propose a novel approach to solve problem (2.1) based on geometric view. The proposed Geometric Water-Filling (GWF) approach eliminates the procedure to solve the non-linear system for the water level, and provides explicit solutions and helpful insights to the problem and the solution.

Figs. 2.1(a)-(c) give an illustration of the proposed GWF algorithm, *by analogy*. Suppose there are 4 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 for each stair (water volume above the stair) is solved.

Let us use d_i to denote the “step depth” of the i th stair which is the height of the i th step to

the bottom of the tank, and is given as

$$d_i = \frac{1}{a_i}, \text{ for } i = 1, 2, \dots, K. \quad (2.3)$$

d_i may take the same unit as s_i . Since the sequence a_i is sorted as monotonically decreasing, the step depth of the stairs indexed as $\{1, \dots, K\}$ is monotonically increasing. We further define $\delta_{i,j}$ as the “step depth difference” of the i th and the j th stairs, expressed as

$$\delta_{i,j} = d_i - d_j = \frac{1}{a_i} - \frac{1}{a_j}, \text{ for } i \geq j \text{ and } 1 \leq i, j \leq K, \quad (2.4)$$

where $1 \leq i, j \leq K$ means that $1 \leq i \leq K$ and $1 \leq j \leq K$.

Instead of trying to determine the water level μ , which is a real non-negative number, we aim to determine water level step, which is an integer number from 1 to K , denoted by k^* , as the highest step under water. Based on the result of k^* , we can write out the solutions for power allocation instantly.

Fig. 2.1(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^* .

In the following, we explain how to find the water level step k^* without the knowledge of the water level μ . Let $P_2(k)$ denote the water volume above step k or zero, whichever is greater. The value of $P_2(k)$ can be solved by subtracting the volume of the water under step k from the total power P , as

$$P_2(k) = \left\{ P - \left[\sum_{i=1}^{k-1} \left(\frac{1}{a_k} - \frac{1}{a_i} \right) \right] \right\}^+ = \left\{ P - \left[\sum_{i=1}^{k-1} \delta_{k,i} \right] \right\}^+, \text{ for } k = 1, \dots, K. \quad (2.5)$$

Due to the definition of $P_2(k)$ being the power (water volume) above step k , it cannot be a negative number. Therefore we use $\{\cdot\}^+$ in (2.5) to assign 0 to $P_2(k)$ if the result inside the bracket is negative. The corresponding geometric meaning is that the k th level is above water. Note a reminder of the definition of a special case for the summation is:

$$\sum_{i=m}^n b_i = 0, \text{ as } m > n. \quad (2.6)$$

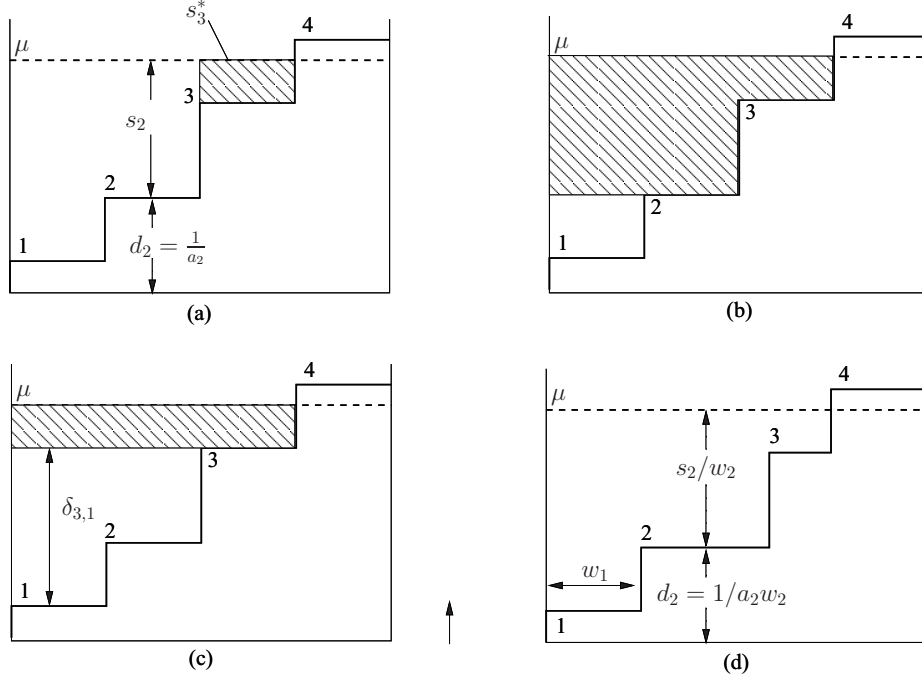


Figure 2.1: Illustration for the proposed Geometric Water-Filling (GWF) algorithm. (a) Illustration of water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = 1/a_i$. (b) Illustration of $P_2(k)$ (shadowed area, representing the total water/power above step k) when $k = 2$. (c) Illustration of $P_2(k)$ when $k = 3$. (d) Illustration of the weighted case.

Fig. 2.1(b) and Fig. 2.1(c) illustrate the concept of $P_2(k)$ for $k = 2$ and $k = 3$ respectively by the shadowed area. As an example of Fig.2.1(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

$$P_2(k = 3) = [P - \delta_{3,1} - \delta_{3,2}]^+$$

and the above result is the shadowed area in Fig. 2.1(c), which is also an expansion of the composite form of (2.5). Then, we are ready to have the following proposition:

Proposition 2.1. The explicit solution to (2.1) is:

$$s_i = \begin{cases} s_{k^*} + (d_{k^*} - d_i) & 1 \leq i \leq k^* \\ 0, & k^* < i \leq K, \end{cases} \quad (2.7)$$

where the water level step k^* is given as

$$k^* = \max \left\{ k \mid P_2(k) > 0, \ 1 \leq k \leq K \right\} \quad (2.8)$$

and the power level for this step is

$$s_{k^*} = \frac{1}{k^*} P_2(k^*). \quad (2.9)$$

We can interpret Proposition 2.1 from Fig. 2.1. The first step of the proposed approach is to find the water level step k^* . From Fig. 2.1, we can find that $k = 3$ is the maximal index that makes $P_2(k)$ greater than zero. Therefore, based on (6.28), $k^* = 3$ can be determined. Then the power at this step s_{k^*} can be determined based on (6.29). For those steps with index higher than k^* , no power is assigned. For those steps with index lower than k^* , their power levels are obtained by adding s_{k^*} with the corresponding level depth difference with the k^* th step as shown in (2.7).

Proposition 2.1 provides an explicit constructed solution rather than the implicit solution. *The procedure eliminates solving the nonlinear equation as shown in (2.2) and the real number water level μ , through finding the index k^* of the highest “step” under the “water”. This idea will be used in the followings, too.* This point is just the distinct characteristic between CWF and GWF. The proof of the optimality of the solution will be left to the next subsection when we discuss the weighted case.

2.3 Generalization of Weighted Case

For the weighted case, the generalized problem can be stated as: given $P > 0$, as the total power or volume of the water; *the weights*, the allocated power and the propagation path gain for the i th antenna are given as w_i , s_i and a_i respectively, $i = 1, \dots, K$; and K is the total number of the transmit antennas. Furthermore, the weighted coefficients $w_i > 0, i \in \{1, \dots, K\}$, and $\{a_i w_i\}_{i=1}^K$ being monotonically decreasing, find that

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \ \forall i; \\ & \sum_{i=1}^K s_i = P. \end{aligned} \quad (2.10)$$

Using the proposed geometric approach, we can extend the geometric relation for the weighted case as shown in Fig. 2.1(d) to obtain the corresponding solution to (2.10). *By the way, since the power distribution is our major concern in this paper and RRM, the weights are not selected as the optimization variables of the problem (2.10), like the most references on RRM. Also, the other investigated problems follow this rule in this thesis.*

In Fig. 2.1(d), the width of the i th stair/step is denoted as w_i , by analogy. The value of $1/a_i$ denotes the volume under the i th step to the bottom of the tank. Hence, the step depth of the i th step is given as

$$d_i = \frac{1}{a_i w_i}, \quad i = 1, \dots, K. \quad (2.11)$$

Then, $P_2(k)$, the water volume above step k , can be obtained using the similar approach as in the previous subsection considering the step depth difference and the width of the stairs as,

$$P_2(k) = \left[P - \sum_{i=1}^{k-1} (d_k - d_i) w_i \right]^+, \quad \text{for } k = 1, \dots, K. \quad (2.12)$$

As an example in Fig. 2.1(d), the water volume above step 1 and below step 3 with the width w_1 can be found as: the step depth difference, $(d_3 - d_1)$ multiplying the width of the step, w_1 . Therefore, the corresponding $P_2(k = 3)$ can be expressed as,

$$P_2(k = 3) = [P - (d_3 - d_1)w_1 - (d_3 - d_2)w_2]^+,$$

which is an expansion of (2.12). Then we have the following proposition.

Proposition 2.2. The explicit solution to (2.10) is:

$$\begin{cases} s_i = \left[\frac{s_{k^*}}{w_{k^*}} + (d_{k^*} - d_i) \right] w_i, & \text{as } 1 \leq i \leq k^*; \\ s_i = 0, & \text{as } k^* < i \leq K, \end{cases} \quad (2.13)$$

where

$$k^* = \max \left\{ k \mid P_2(k) > 0, \quad 1 \leq k \leq K \right\} \quad (2.14)$$

and the power level for this step is

$$s_{k^*} = \frac{w_{k^*}}{\sum_{i=1}^{k^*} w_i} P_2(k^*). \quad (2.15)$$

Proof of Proposition 2.2. System (2.13) implies that

$$\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} = \frac{w_i}{\frac{1}{a_i} + s_i}, \text{ as } 1 \leq i \leq k^*. \quad (2.16)$$

Let

$$\lambda = \frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}}. \quad (2.17)$$

From a geometric view, λ is the reciprocal of water level μ . According to the definitions of k^* and s_{k^*} , for $k^* < i \leq K$, $\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} > \frac{w_i}{\frac{1}{a_i} + s_i}$ and $s_i = 0$.

Let

$$\sigma_i = \frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} - \frac{w_i}{\frac{1}{a_i} + s_i}. \quad (2.18)$$

Then

$$\begin{cases} \sigma_i > 0, & \text{if } k^* < i \leq K \\ \sigma_i = 0, & \text{if } 1 \leq i \leq k^*. \end{cases} \quad (2.19)$$

Therefore, the following system holds:

$$\begin{cases} \frac{w_i}{\frac{1}{a_i} + s_i} - \lambda + \sigma_i = 0, & \text{as } 1 \leq i \leq K \\ s_i \geq 0, & \forall i \\ \sigma_i s_i = 0, & \forall i \\ \sigma_i \geq 0, & \forall i \\ \sum_{i=1}^K s_i = P, & \lambda \in \mathbb{R}. \end{cases} \quad (2.20)$$

By observation, the equation and inequality set above is just a set of the KKT conditions of the problem in Proposition 2.2 and the water level μ is equal to the reciprocal of the Lagrange multiplier λ mentioned above. Note that the Lagrange function of the problem in Proposition 2.2 is

$$L(\{s_i\}, \lambda, \{\sigma_i\}) = \sum_{i=1}^K w_i \log(1 + a_i s_i) - \lambda \left(\sum_{i=1}^K s_i - P \right) + \sum_{i=1}^K \sigma_i s_i. \quad (2.21)$$

Since it 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 the constraint qualification of the problem (2.10) holds. Proposition 2.2 hence is proved.

Similar to the unweighted case, the first step is to calculate $P_2(k)$, then find the water level step, k^* , from (2.14), which is the maximal index making $P_2(k)$ nonnegative. The corresponding power level for this step, s_{k^*} , can be obtained by applying (2.15). Then for those steps with index higher than k^* , the power level is assigned with zero. For those steps below k^* , the power level is assigned as in (2.13). The first term (s_{k^*}/w_{k^*}) inside the square bracket denotes the depth of the k^* th step to the surface of the water. The second term inside the square bracket denotes the step depth difference of the k^* th step and the i th step. Therefore, the sum inside the square bracket means the depth of the i th step to the surface of the water. When this quantity is multiplied with the width of this step, the volume of the water above this step (allocated power) can be then readily obtained.

With the proposed GWF approach, the weighted problem could be solved straightforwardly, avoiding complicated derivation and calculation. When the weighting factors are set to ones, the corresponding unweighted case is obtained. In the following description of algorithm implementation and proof, we only provide weighted case.

From Proposition 2.2, when k^* is obtained, $P_2(k^*)$ is given. Then it is memorized and only multiplied by a constant to compute s_{k^*} . Thus, searching k^* is a key point for the proposed GWF and the procedure is stated as follows:

- 1) Initialize $W_s = 0; P_M = P^* = P; i = 1$.
- 2) Compute $W_s \leftarrow W_s + w_i; P^* \leftarrow P^* - (d_{i+1} - d_i)W_s$. Then $i \leftarrow i + 1$, where the symbol " \leftarrow " represents the assignment operation.
- 3) If $P^* > 0$ and $i \leq K$, $P_M = P^*$, and repeat the step 2); else, output $k^* = i - 1, W_s = W_s - w_i$ and $s_{k^*} = \frac{w_{k^*}}{W_s} P_M$.

We can observe that $\frac{s_{k^*}}{w_{k^*}} + d_{k^*}$ is the water level due to $\frac{s_{k^*}}{w_{k^*}} + d_{k^*} = \frac{s_i}{w_i} + d_i$, for $1 \leq i \leq k^*$ in (2.13).

As an alternative to the enumeration search in the Algorithm GWF, a Fibonacci-like search is possibly used to speed up finding k^* due to (non-increasing) monotonicity of the sequence $\{P_2(k)\}$.

Without loss of generality, let Fibonacci approximation ratios be $\frac{1}{3}$ and $\frac{2}{3}$ for searching k^* . The method can be described as:

1st Step. Assume that $a = 1$ and $b = K$.

2nd Step. If $a = b$, then $k^* = a$ and go to Step 3 of GWF.

Else, $a_1 = \lfloor a + \frac{1}{3}(b - a) \rfloor$, $b_1 = \lceil a + \frac{2}{3}(b - a) \rceil$.

3rd Step. If $P_2(a_1) \leq 0$, then $b = a_1 - 1$ and go to the 2nd Step;

If $P_2(b_1) > 0$, then $a = b_1$ and go to the 2nd Step;

If $P_2(a_1) > 0$ and $P_2(b_1) \leq 0$, then $a = a_1$, $b = b_1 - 1$ and go to the 2nd Step.

The number of loops to search k^* is reduced into a complexity level of $\log_3(K)$.

2.4 Weighted Water-Filling with Individual Peak Power Constraints

In this section, we extend the CWF problem to include individual peak power constraints (WFPP).

The weighted WFPP problem is stated as follows. Given $P > 0$, as the total power or volume of the water; the allocated power and the propagation path gain for the i th antenna are given as s_i and a_i respectively, $i = 1, \dots, K$; and K is the total number of the transmit antennas. Also, the weights $w_i > 0, \forall i$, and without loss of generality, $\{a_i \cdot w_i\}_{i=1}^K$ being positive and monotonically decreasing, find that

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i \leq P_i, \forall i; \\ & \sum_{i=1}^K s_i \leq P. \end{aligned} \tag{2.22}$$

Comparing the problem (2.22) with (2.10), the constraint of $0 \leq s_i$ is extended to $0 \leq s_i \leq P_i$, *i.e.*, additional individual peak power constraints, and $\sum_{i=1}^K s_i = P$ to $\sum_{i=1}^K s_i \leq P$. The problem (2.22) is thus referred to as (weighted) water-filling with sum and individual peak power constraints (WFPP). In this section, we discuss the solution to the WFPP problem. *By the way, the discussed problem above is of multi-user cases. According to the multi-user information theory, the optimal capacity point set determined by the optimal power allocation is located at the boundary of the*

capacity region that is a polygon. Therefore, any optimal capacity point can be found by setting up a group of weights of the target problem, like our (2.22). The detail can refer to [5] and its references therein. As a simple explanation, a group of weights plays a role of adjusting priority of channels to obtain the optimal power allocation.

Proposition 2.2 in section 2.3 provides an explicit solution using geometric view approach. Interestingly, the proposed GWF can be applied to the WFPP problem with some modifications. The following presents an algorithm which is a modification of the above discussed GWF and it is termed as the GWFPP.

For convenience, the expression (2.12) can further be extended into the expression:

$$P_2(i_k) = \left[P - \sum_{t=1}^{|E|-1} (d_{i_k} - d_{i_t}) w_{i_t} \right]^+, \text{ for } k = 1, \dots, |E|, \quad (2.23)$$

where E is a subsequence of the sequence $\{1, 2, \dots, K\}$, $|E|$ is the cardinality of the set E , so E can be expressed as $\{i_1, i_2, \dots, i_{|E|}\}$. Especially, if E is taken as the sequence $\{1, 2, \dots, K\}$, then the extended expression is regressed into the original expression (2.12). Similarly, some corresponding changes in (2.13)-(2.15) are also done (*i.e.*, the subscripts of sequence are replaced with those of the subsequence). For avoiding tediousness, these extended expressions are still labelled as (2.13)-(2.15) in the following statement of Algorithm GWFPP.

Algorithm GWFPP:

Input: vector $\{d_i\}, \{w_i\}, \{P_i\}$ for $i = 1, 2, \dots, K$, the set $E = \{1, 2, \dots, K\}$, and P .

- 1) Utilize (2.13)-(2.15) to compute $\{s_i\}$.
- 2) The set Λ is defined by the set $\{i | s_i > P_i, i \in E\}$. If Λ is the empty set, output $\{s_i\}_{i=1}^K$; else, $s_i = P_i$, as $i \in \Lambda$.
- 3) Update E with $E \setminus \Lambda$ and P with $P - \sum_{t \in \Lambda} P_t$. Then return to 1) of the GWFPP.

Remark 1. Algorithm GWFPP is a dynamic power distribution process. The state of this process is the difference between the individual peak power sequence and the current power distribution sequence obtained by the Algorithm GWF. The control of this process is to use (2.13)-(2.15) of the Algorithm GWF based on the state mentioned above. Thus, a new state for next time stage

appears. Therefore, an optimal dynamic power distribution process, the GWFPP, with the state feedback is formed. Since the finite set E is getting smaller and smaller until the set Λ is empty, Algorithm GWFPP carries out K loops to compute the optimal solution, at most.

Similar to the proof of the Proposition 2.2, we can obtain the following conclusion:

Proposition 2.3: Algorithm GWFPP can provide the optimal solution to the problem (2.22).

Proof of Proposition 2.3. If the final set E in Algorithm GWFPP is empty, it implies that $\sum_{i=1}^K P_i \leq P$. Then it is seen that the optimal solution $s_i = P_i$ holds, for any i .

If it is non-empty, observing the stricture of (2.22), Proposition 2.3 is easily proved, similarly to the previous one.

2.5 Weighted Water-Filling with Group Bounded Power Constraints

The weighted WFGBP problem is stated as follows. Considering a cognitive network, given $P \geq 0$, as the total power of the CRs or volume of the water; the allocated power, the propagation path gain and the weight for the i th CR are given as s_k , a_k and $w_k (\geq 0)$ respectively, $k = 1, \dots, K$, where K is the total number of the CRs; and let $\{\chi_i\}_{i=1}^T$ be a partition of the index set: $\{1, \dots, K\}$. For convenience, the elements of χ_i can be listed, monotonically increasing, *i.e.*, $i_1 < i_2 < \dots < i_{(\chi_i)^\#}$. \underline{P}_i and \overline{P}_i , under the assumption of $0 \leq \underline{P}_i \leq \overline{P}_i$, denote the lower bound and the upper bound of the power constraints for the i th group of the CRs, $\forall i$. The generalized weighted water-filling problem with group bounded power constraints under consideration then reads

$$\begin{aligned} \max_{\{s_k\}_{k=1}^K} \quad & \sum_{k=1}^K w_k \log(1 + a_k s_k) \\ \text{subject to:} \quad & 0 \leq s_k, \forall k; \\ & \sum_{k=1}^K s_k \leq P; \\ & \underline{P}_i \leq \sum_{k \in \chi_i} s_k \leq \overline{P}_i, i = 1, \dots, T. \end{aligned} \tag{2.24}$$

Compared the problem (2.24) with (2.22), the constraints of $0 \leq s_i \leq P_i, \forall i$, are generalized to $\underline{P}_i \leq \sum_{k \in \chi_i} s_k \leq \overline{P}_i$, *i.e.*, additional group bounded power constraints, *the lower bounds of which can be used to guarantee the fair transmitted rate from the i th group of CRs, whereas the upper bounds of which can be used to limit interference of the group with the primary users, for any i .*

That is to say, adjusting the just mentioned lower bounds can make the rate more fair, due to the relationship between the power and the rate; and adjusting the just mentioned upper bounds can more reasonably control the interference from SUs, also due to the same cause. The problem (2.24) is thus referred to as (weighted) water-filling with group bounded power constraints (WFGBP). In this subsection, we discuss the solution to the WFGBP problem.

Due to the explicit solution using geometric view approach that is provided in Proposition 2.2, interestingly, the proposed GWF can be applied to the WFGBP problem with some modifications. The following presents an extended algorithm, which is a meaningful modification of the GWF and is termed as the GWFGPB.

Note, as $P \leq \sum_{i=1}^T \underline{P}_i$, it is seen that there does not exist any solution to problem (2.24); whereas, as $\sum_{i=1}^T \overline{P}_i \leq P$, problem (2.24) is regressed into a trivial case without the sum power constraint. Hence, $\sum_{i=1}^T \underline{P}_i \leq P \leq \sum_{i=1}^T \overline{P}_i$ is assumed.

If $\underline{P}_i = 0, \overline{P}_i > 0, \forall i$, and the weights are equal, then problem (2.24) is reduced into the regular case that can be solved by the conventional weighted water-filling problem [44]; and if χ_i is regressed a singleton and $\underline{P}_i = 0, \forall i$, then problem (2.24) is reduced into the WFPP problem. Thus, (2.24) is a more general form of the RRA problem.

To find the solution to (2.24), the generalized geometric water-filling algorithm for the group bounded power constraints (GWFGPB) is presented as follows: Firstly, for integrity of this new algorithm, let us re-visit the four concepts: (i) power (water volume) above level k , $P_2(k)$; (ii) power allocated to the i th group t th channel, s_{it} ; (iii) water level step k^* ; and (vi) power allocated for the water level step $s_{i_{k^*}}$ as below:

$$P_2(k) = \left[P - \sum_{t=1}^{E^\#-1} \left(\frac{1}{a_{i_k} w_{i_k}} - \frac{1}{a_{i_t} w_{i_t}} \right) w_{i_t} \right]^+, \quad \text{for } k = 1, \dots, E^\#, \quad (2.25)$$

where E is a subsequence of the sequence $\{1, 2, \dots, K\}$, $E^\#$ is the cardinality of the set E , so E can be written as $\{i_1, i_2, \dots, i_{E^\#}\}$. Note that k in $P_2(k)$ is a subscript of the subsequence $\{i_t\}_{t=1}^{E^\#}$ under the assumption: $1 \leq i_1 < i_2 < \dots < i_{E^\#} \leq K$ in the given set E , and the sequence $\{1, 2, \dots, K\}$ is a subsequence of itself under the definition of subsequence.

Also, note

$$s_{i_t} = \begin{cases} w_{i_t} \left(\left(\frac{s_{i_{k^*}}}{w_{i_{k^*}}} + \frac{1}{a_{i_{k^*}} w_{i_{k^*}}} \right) - \frac{1}{a_{i_t} w_{i_t}} \right), & 1 \leq t \leq k^* \\ 0, & k^* < t \leq E^\sharp, \end{cases} \quad (2.26)$$

where the water level step k^* is given as

$$k^* = \max \left\{ k \mid P_2(k) > 0, \quad 1 \leq k \leq E^\sharp \right\} \quad (2.27)$$

and the power level for this step is

$$s_{i_{k^*}} = \frac{1}{k^*} P_2(k^*). \quad (2.28)$$

If water-filling is vividly described as pouring the water of volume P into a tank with the bottom of E^\sharp stairs, then $P_2(k)$ is the water volume above the k th stair. Along this analogue understanding, we can interpret the geometric meaning of these four concepts under the simple assumptions of $E = \{1, 2, \dots, K\}$, similarly mentioned above. Using these four concepts, the steps of GWFGBP can be described as below.

Algorithm GWFGBP:

Input: the channel gains $\{a_k\}_{k=1}^K$, the weights $\{w_k\}_{k=1}^K$, the group lower and upper power bounds $\{\underline{P}_i, \overline{P}_i\}_{i=1}^T$, the index set $E = (E_0 =)\{1, 2, \dots, K\}$, the partition $\{\chi_i\}_{i=1}^T$, the sum power constraint P and $i = 1$.

- 1) Initialize $W_{i_s} = 0$; $P_M = P^* = \underline{P}_i$; $j = 1$.
- 2) Update W_{i_s} with $W_{i_s} + w_{i_j}$ and P^* with $P^* - (d_{i_{j+1}} - d_{i_j})W_{i_s}$. Then increase the iteration index j to $j + 1$, where the used symbols are referred in **Proposition 2.2**.
- 3) If $P^* > 0$ and $j \leq (\chi_i)^\sharp$, $P_M = P^*$, and repeat the step 2); else, output $k^* = j - 1$, $s_{i_{k^*}} = \frac{w_{i_{k^*}}}{W_{i_s}} P_M$, increase the iteration index i to $i + 1$, and then repeat the step 1), until $i = T$. Thus, $\{s_k\}_{k=1}^K$ is obtained. Let E be updated with $\{1, \dots, K\}$, P_t with P and $\frac{1}{a_k}$ with $\frac{1}{a_k} + s_k, \forall k$. Finally in this step, let $n = 1$ and $\Lambda = \emptyset$, where \emptyset stands for the empty set.
- 4) Then utilize (2.25)-(2.28) to compute $\{s_i\}$, which appear in the left hand-side (LHS) of (2.26). Successively, assign Δs_k with $s_k, \forall k$.

- 5) The set Λ_n is defined by the set $\{i | \sum_{j \in \chi_i} \Delta s_j > \bar{P}_i - \underline{P}_i, 1 \leq i \leq T\}$. If Λ_n is the empty set, output the solution $\{s_k + \Delta s_k\}_{k=1}^K$ to the problem (2.24); else, let $\sum_{j \in \chi_i} \Delta s_j = \bar{P}_i - \underline{P}_i$, as $i \in \Lambda_n$. Further, continuously utilize similar expressions to (2.25)-(2.28), these similar expressions only changing from P_t to $\bar{P}_i - \underline{P}_i$ and from E^\sharp to χ_i for any $i \in \Lambda_n$, and then obtain $\Delta s_j, j \in \cup_{i \in \Lambda_n} \chi_i$. Let $\Lambda = \Lambda \cup \Lambda_n$.
- 6) Update E with $E \setminus (\cup_{i \in \Lambda_n} \chi_i)$; P_t with $P_t - \sum_{i \in \Lambda_n} (\bar{P}_i - \underline{P}_i)$. Then increase the iteration index n to $n + 1$, and return to 4) of GWFGBP.

Remark 2. GWFGBP is also a dynamic power distribution process. The state of this process is the difference between the group bounded power sequence and current power distribution sequence obtained by (2.25)-(2.28). The control of this process is to use the mentioned similar expressions to (2.25)-(2.28) based on the state mentioned above. Thus, a new state for next time stage appears. Therefore, an optimal dynamic power distribution process, GWFGBP, with the state feedback is formed. Since the finite set E is getting smaller and smaller until there exists n such that the set Λ_n is empty, GWFGBP carries out T loops to compute the optimal solution, at most.

For optimality of the proposed GWFGBP, we can obtain the following conclusion:

Proposition 2.4: Algorithm GWFGBP can provide the optimal solution to the problem (2.24) via finite computation.

Proof of Proposition 2.4. Without loss of generality, assume that the final set Λ in GWFGBP is empty. It is seen that the optimal solutions $\{\Delta s_j\}_{j \in \chi_i}$ hold, for any i , with only requiring to satisfy $\sum_{j \in \chi_i} \Delta s_j \leq \bar{P}_i - \underline{P}_i$ and also satisfy the total sum power constraint for $\{\Delta s_k\}$. Thus, appending all the groups of the solutions from GWFGBP, we can obtain the solution to the problem (2.24) and its optimality is proven as follows.

The final set E , as a non-empty set, implies that

$$\frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*} + \Delta s_{k^*}}{w_{k^*}}} = \frac{1}{\frac{1}{a_j w_j} + \frac{s_j + \Delta s_j}{w_j}}, \text{ as } \{j, k^*\} \subset E, \quad (2.29)$$

and there exists χ_i such that $\sum_{j \in \chi_i} \Delta s_j > 0$. Thus, under $\sum_{j \in \chi_i} \Delta s_j > 0$, let

$$\lambda = \frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*} + \Delta s_{k^*}}{w_{k^*}}} \quad (2.30)$$

and then $\mu_j = 0$ as $\Delta s_j > 0$ and $j \in \chi_i$. Further, according to the definitions of k^* and s_{k^*} , for $j \in E$ and $\Delta s_j = 0$, since

$$\frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*} + \Delta s_{k^*}}{w_{k^*}}} > \frac{1}{\frac{1}{a_j w_j} + \frac{s_j + \Delta s_j}{w_j}} \quad (2.31)$$

holds, then let

$$\mu_j = \frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*} + \Delta s_{k^*}}{w_{k^*}}} - \frac{1}{\frac{1}{a_j w_j} + \frac{s_j + \Delta s_j}{w_j}} > 0 \quad (2.32)$$

and $\underline{\sigma}_i = \bar{\sigma}_i = 0$, as $\chi_i \subset E$. If the set χ_i implies $\sum_{j \in \chi_i} \Delta s_j = 0$, then let

$$\underline{\sigma}_i = \lambda - \frac{1}{\frac{1}{a_{k^*}(\chi_i) w_{k^*}(\chi_i)} + \frac{s_{k^*}(\chi_i)}{w_{k^*}(\chi_i)}} \geq 0, \quad (2.33)$$

and $\bar{\sigma}_i = 0$. Also, if $s_j > 0$, let $\mu_j = 0$; if $s_j = 0$, let $\mu_j = \lambda - \underline{\sigma}_i - a_j w_j \geq 0$, keeping the mentioned values of $\underline{\sigma}_i$ and $\bar{\sigma}_i$.

On the other hand, if $i \in \Lambda$ and $j \in \chi_i$, then

$$0 < \sum_{j \in \chi_i} \Delta s_j = \bar{P}_i - \underline{P}_i, \quad (2.34)$$

$$\bar{\sigma}_i = \frac{1}{\frac{1}{a_{k^*}(\chi_i) w_{k^*}(\chi_i)} + \frac{s_{k^*}(\chi_i) + \Delta s_{k^*}(\chi_i)}{w_{k^*}(\chi_i)}} - \lambda \geq 0, \quad (2.35)$$

let $\underline{\sigma}_i = 0$ and $\mu_j = 0$, as $\Delta s_j > 0$. If $\Delta s_j = 0$, then

$$\mu_j = \lambda + \bar{\sigma}_i - \frac{1}{\frac{1}{a_j w_j} + \frac{s_j}{w_j}} > 0 \quad (2.36)$$

and $\underline{\sigma}_i = 0$.

Therefore, there have been the Lagrange multipliers $\lambda, \{\underline{\sigma}_i, \bar{\sigma}_i\}_{i=1}^T$ and $\{\mu_k\}_{k=1}^K$ obtained above, the Lagrange function of which, for the problem (2.24), is:

$$\begin{aligned} & L(\{s_k\}, \lambda, \{\bar{\sigma}_i\}, \{\underline{\sigma}_i\}, \{\mu_k\}) \\ &= \sum_{k=1}^K w_k \log(1 + a_k s_k) - \lambda \left(\sum_{k=1}^K s_k - P \right) \\ &- \sum_{i=1}^T \bar{\sigma}_i (\sum_{j \in \chi_i} s_j - \bar{P}_i) + \sum_{i=1}^T \underline{\sigma}_i (\sum_{j \in \chi_i} s_j - \underline{P}_i) + \sum_{k=1}^K \mu_k s_k. \end{aligned} \quad (2.37)$$

Further, by observation, they satisfy the KKT conditions. Since the problem (2.24) 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 easily seen that the constraint qualification of the problem (2.24) holds. Proposition 2.4 hence is proved.

Remark 3. If we chose the CWF to solve the problem (2.24), similarly, a non-linear system with non-linear equations and inequalities in multiple dual variables would have had to be solved:

$$\left\{ \begin{array}{l} \sum_{i=1}^T \sum_{j \in \chi_i} \left(\frac{w_j}{\lambda + \bar{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_i} \right)^+ = P, \text{ as } j \in \chi_i, i = 1, 2, \dots, T; \\ \underline{P}_i \leq \sum_{j \in \chi_i} \left(\frac{w_j}{\lambda + \bar{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_i} \right)^+ \leq \bar{P}_i, \text{ as } i = 1, 2, \dots, T; \\ \underline{\sigma}_i (\sum_{j \in \chi_i} \left(\frac{w_j}{\lambda + \bar{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_i} \right)^+ - \underline{P}_i) = 0, \text{ as } i = 1, 2, \dots, T; \\ \bar{\sigma}_i (\sum_{j \in \chi_i} \left(\frac{w_j}{\lambda + \bar{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_i} \right)^+ - \bar{P}_i) = 0, \text{ as } i = 1, 2, \dots, T; \\ \lambda \geq 0; \underline{\sigma}_i \geq 0, \bar{\sigma}_i \geq 0, \text{ as } i = 1, 2, \dots, T. \end{array} \right. \quad (2.38)$$

There seems no existing result that can solve this system.

As a note, the proposed algorithms in this section can output the unique optimal solution to each of the corresponding problems, since the objective function of each of the problems has a negative definite Hessian matrix.

2.6 Numerical Results and Complexity Analysis

2.6.1 Numerical Results

As an illustration for the proposed algorithm, some numerical examples are provided in this subsection. *To clearly account for procedures of the proposed algorithms, the simple and easily checked parameters are used. This way can make readers grasp essence of procedures in the proposed algorithms without being tangled with tedious numerical computation. It stems from loose assumption by our approach.*

Example 1. Instance a case of the water-filling with individual peak power constraints (WFPP) problem:

$$\begin{aligned} & \max_{\{s_i\}_{i=1}^2} \quad \sum_{i=1}^2 \log(1 + a_i s_i) \\ & \text{subject to: } \quad 0 \leq s_i \leq 2, \quad \forall i; \\ & \quad \quad \quad \sum_{i=1}^2 s_i \leq 3. \end{aligned} \quad (2.39)$$

where $a_1 = 1$ and $a_2 = 0.2$. The problem given is a WFPP problem. Utilizing the proposed GWFPP, the result of the first loop is $s_1 = 2$, as part of the solution based on the algorithm. The

remaining of the solution, s_2 , is allocated with zero. The result of the second loop is $s_1 = 2$ and $s_2 = 1$, as full entries of the solution. According to Proposition 2.3, the result of the second loop is guaranteed to be the optimal solution. For comparison purpose, the solution to the CWF is also plotted in Fig. 2.2. In Fig. 2.2, the step depth for channel 1 and channel 2 are 1 and 5 respectively, as the reciprocal of their respective channel gains. Using the CWF, the solution is shown in Fig. 2.2(a): all the power is allocated to the first channel with good channel condition. If consider peak power constraints check, s_1 may be clipped as shown in Fig. 2.2(b). With the proposed GWFPP, we can directly obtain the optimal solution as shown in Fig. 2.2(c).

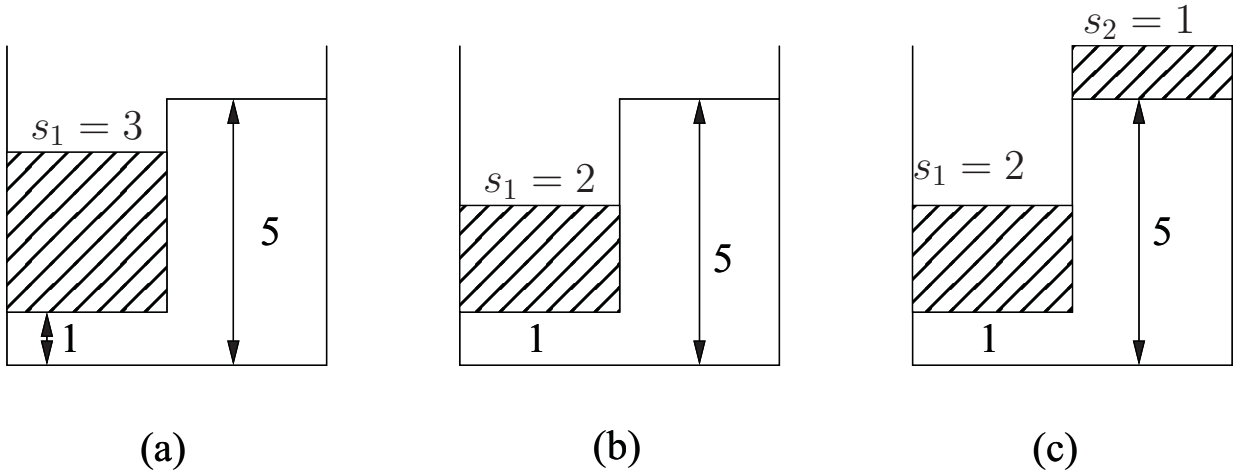


Figure 2.2: Illustration for Example 1, results for CWF and GWFPP. (a) the CWF: without peak power restriction check ($s_1 = 3, s_2 = 0$); (b) the CWF: s_1 is clipped considering peak power constraint; (c) GWFPP: $s_1 = 2, s_2 = 1$.

Example 2. Instance another case of the water-filling with the WFPP problem with multiple channels:

$$\begin{aligned}
& \max_{\{s_i\}_{i=1}^8} \quad \sum_{i=1}^8 \log(1 + a_i s_i) \\
& \text{subject to: } 0 \leq s_i \leq i, \quad \forall i; \\
& \quad \quad \quad \sum_{i=1}^8 s_i \leq 30.
\end{aligned} \tag{2.40}$$

where $a_i = 1/i, \forall i$. The step depth is then monotonically increase from 1 to 8, as shown in Fig. 2.3. For the CWF, without considering the peak power constraints, the water level is solved as 8.25, then the power allocation is shown in Fig. 2.3(a) and Fig. 2.4(a). Considering peak power constraints, the power levels for channels 1-4 are clipped and are set to their peak values as shown in Fig. 2.3(b). The CWF doesn't tell us where to and how to assign the clipped power.

Utilizing the proposed GWFPP, the result of the first loop is $s_i = i$, as $i = 1, \dots, 4$, as part of the solution. The remaining entries of the solution are allocated with zero, as shown in Fig. 2.4(b). The result of the second loop is $s_i = i$, as $i = 1, \dots, 5$, also as part of the solution. The remaining entries of the solution are allocated with zero, as shown in Fig. 2.4(c). The results of the third loop are $s_i = i$, as $i = 1, \dots, 5$; and $s_i = 12 - i$, as $i = 6, 7, 8$, as full entries of the solution, as shown in Fig. 2.4(d) and Fig. 2.3(c). According to Proposition 2.3, the result of the third loop is the optimal solution.

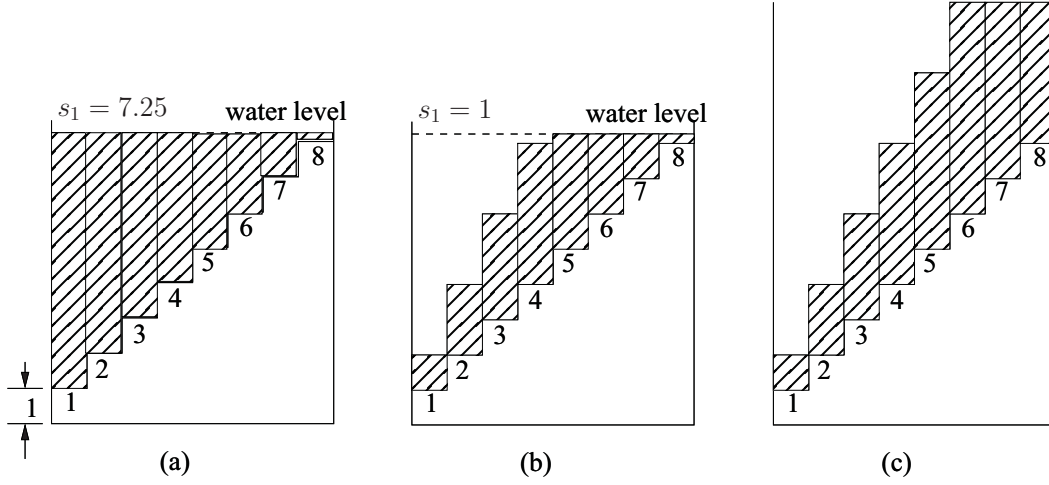


Figure 2.3: Illustration for Power Allocation using the CWF and the proposed GWFPP for Example 2. (a) Results for the CWF; (b) the CWF, clipped s_1 to s_4 due to peak power restrictions; (c) Results for GWFPP.

It is shown that the solution for the problem with sum power constraint only (see for example, Fig. 2.3(a) and Fig. 2.4(a)) is different from the solution of the corresponding problem with added peak power constraints (see for example, Fig. 2.3(c) and Fig. 2.4(d)). From Fig. 2.3, we can observe that for more complicated problems, the conventional water-fill exhibits its limitations. The water level is no longer a unique level. Thus, our approach using the concept of water-fill is more general to solve the RRA problems.

Example 3. Instance a case of the weighted water-filling with individual peak power constraints (WFPP) problem:

$$\begin{aligned}
 & \max_{\{s_i\}_{i=1}^2} \quad \sum_{i=1}^2 w_i \log(1 + a_i s_i) \\
 & \text{subject to: } 0 \leq s_i \leq 2, \quad \forall i; \\
 & \quad \quad \quad \sum_{i=1}^2 s_i \leq 3,
 \end{aligned} \tag{2.41}$$

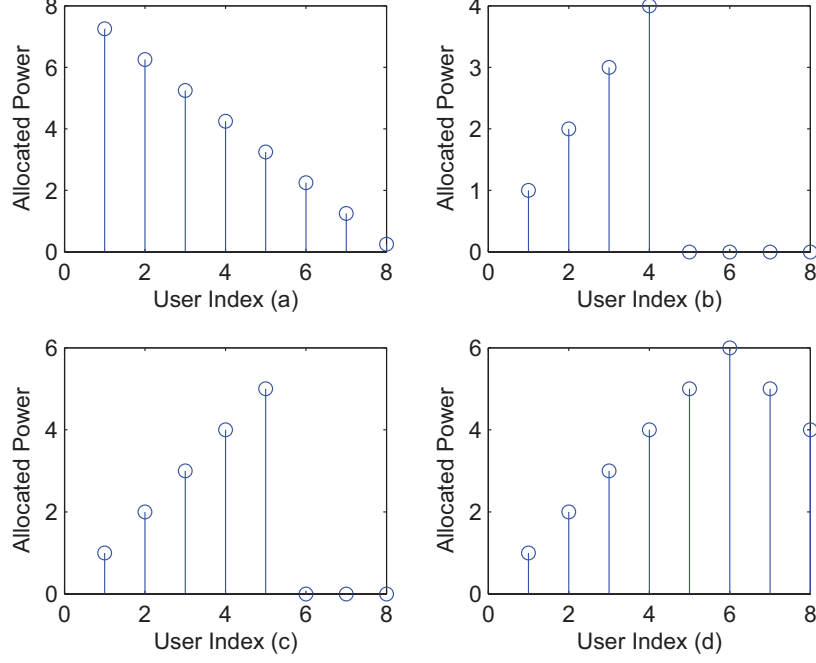


Figure 2.4: Optimal Power Allocation Results for Example 2. (a) solution from the the CWF; (b) 1st iteration results of GWFPP; (c) 2nd iteration results of GWFPP; (d) 3rd iteration results of GWFPP.

where $a_1 = 2, a_2 = 0.1, w_1 = 0.2$ and $w_2 = 0.8$. Utilizing the proposed GWFPP, the result of the first loop is $s_1 = 2$, as part of the solution. The remaining of the solution, s_2 , is allocated with zero. The result of the second loop is $s_1 = 2$ and $s_2 = 1$, as full entries of the solution. From Proposition 2.3, the result of the second loop is guaranteed to be the optimal solution. The result is illustrated in Fig. 2.5. In this figure, for channel 1, the stair width is 0.2, specified by its weight factor. The level depth is $1/(a_1 w_1) = 2.5$. Similarly, for the channel two, the stair width is 0.8 and the level depth is $1/(a_1 w_1) = 12.5$. The power allocated for channel 1 is 2, so the water level for channel 1 is 10. For channel 2, the power is 1, the water level is 1.25. Again, water level is not unique for different channels.

Example 4. As a last example, we instance a case of the weighted water-filling with group

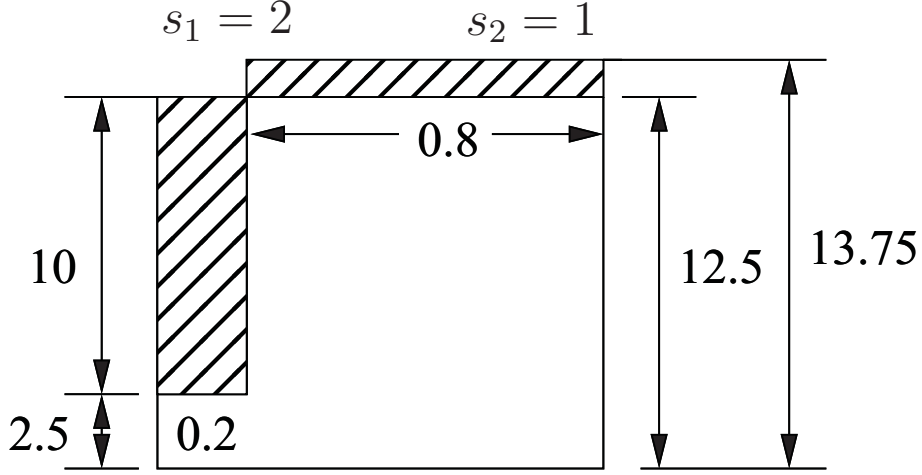


Figure 2.5: Illustration for Example 3 using GWFPP ($s_1 = 2, s_2 = 1$).

bounded power constraints (WFGBP) problem:

$$\begin{aligned}
& \max_{\{s_i\}_{i=1}^3} \quad \sum_{i=1}^3 w_i \log(1 + a_i s_i) \\
& \text{subject to: } 0 \leq s_i, \forall i; \\
& \quad \sum_{i=1}^3 s_i \leq 5; \\
& \quad 1 \leq s_1 + s_2 \leq 2.5; \\
& \quad 1 \leq s_3 \leq 2.5,
\end{aligned} \tag{2.42}$$

where $a_1 = a_2 = a_3 = 1, w_1 = 0.3, w_2 = 0.2$ and $w_3 = 0.5$. Utilizing the proposed GWFGPB, the result from 1)-3) of GWFGPB is: $s_1 = 0.8, s_2 = 0.2$ and $s_3 = 1$. Then continuously using 4)-6) of GWFGPB, the optimal solution is: $s_1^* = 0.8 + 0.9 = 1.7, s_2^* = 0.2 + 0.6 = 0.8$ and $s_3^* = 1 + 1.5 = 2.5$.

The results are shown in Fig. 2.6, where the stair width for the three channels are 0.3, 0.2, 0.5 respectively specified by their weighting factors. The step depth is calculated as $1/(a_i w_i)$, leading to the step depth values as 3.33, 5, and 2 respectively for the three channels. The water level for channel 1 and channel 2 is the same, but different with that of channel 3. It is interesting to observe that with the same path gain of these three channels, the channel with the highest weight factor, channel 3, is not necessarily being assigned as the highest power level due to the structure of the constraints.

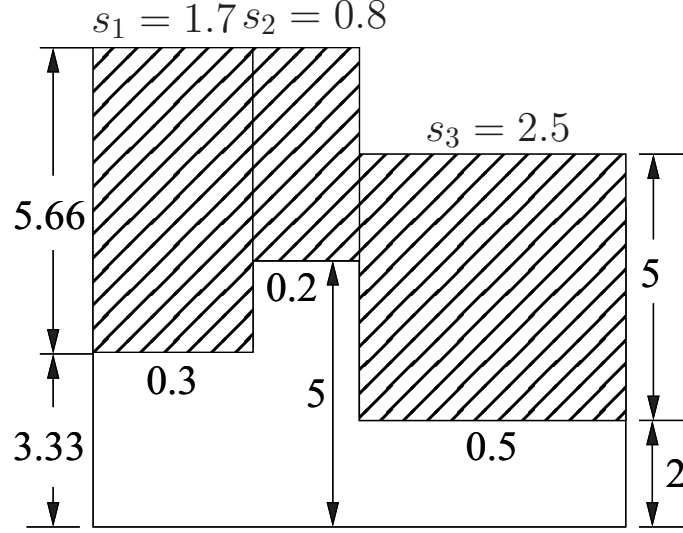


Figure 2.6: Illustration for Multi-User Power Allocation Results by GWFGPB.

2.6.2 Complexity Analysis

As stated in [36, Section 3], the conventional WF algorithm had an exponential worst-case complexity of 2^K , where K is the number of the channels, even though the channel gains had been sorted in decreasing order. Pointing to this case, [36] proposed an improved algorithm with worst-case complexity of K iterations. Since each iteration consists of multiple arithmetic and logical operations, here we use total number of operations as a measure of the complexity level (See [38, Chapter 8]).

The CWF approach has a worst-case complexity of K iterations, *i.e.*, total $O(K^2)$ fundamental arithmetic and logical operations under the $2(K+1)$ memory requirement and the sorted parameters $\{w_k a_k\}_{k=1}^K$ (e.g. see [37, p. 137], for more details).

The proposed GWF algorithm occupies less computational resource. It is seen that it needs K loops at most to search k^* and it needs 4 arithmetic operations and 2 logical operations to complete each loop. Thus, the worst-case computational complexity of the proposed solution is $8K + 2$ (from the operations of $6K + 2 + 2K$) fundamental arithmetical and logical operations under the $2(K+1)$ memory units to store $\{d_i\}$, $\{w_i\}$, W_s , and P_M .

For GWFP, it needs K loops to compute the optimal solution, at most. The required number of operations is, at worst, $\sum_{i=1}^K (8i+2) = 4K^2 + 6K$ fundamental arithmetical and logical operations.

For GWFGPB, it needs T loops to compute the optimal solution, at most. The required number of operations, at worst, is $O(K^2)$ fundamental arithmetical and logical operations.

In this complexity analysis, we didn't take sorting procedure into consideration. It is stated in [36] that the channel gain sequences come from the eigenvalues of a matrix and many of the algorithms to compute the eigenvalues and eigenvectors already produce the eigenvalues sorted. *As a note, CWF and GWF assumed the channel gain sequences to have been sorted.*

Chapter 3

RRM in Wireless Communications with Energy Harvesting Technology

3.1 Introduction

There has been recent research effort on understanding data transmission with an energy harvesting transmitter that has a rechargeable battery for green communications [33, 43, 35, 25]. Recently, based on [35], [14] further investigates the issues of power allocation problems to minimize the grid power consumption with random energy and data arrival. In more detail, for the implied problem that is a convex optimization problem, rather than the original non-convex problem, a solution is computed. For convenience and without loss of generality, the process is considered as a discrete time process. The simplest and useful system model, illustrated in Fig. 3.1, assumes that there are K epochs in the time period or interval $(0, T]$. For each epoch, an event occurs which may be the consequence of channel fading gain variation and new energy harvested. At the beginning of the process, a sequence of the fading gain is denoted by $\{a_i\}$, the value of each term of which depends on epoch i . Thus, a_i corresponds to epoch i . *Similarly, we denote the corresponding sequence of harvested energy by $\{E_{in}(t)\}$, where $E_{in}(t)$ is the observed harvested energy for epoch t , $t = 1, \dots, K$. For convenience, $E_{in}(t)$, where $t = i$, is also expressed by $E_{in}(i), \forall i$.* This setting leads to new design insights in a wireless link with a rechargeable transmitter and fading channels.

Besides the allocated power to be non-negative, sum of successively harvested energy over time determines a more complicated power constraint. Generally, the incoming energy can be stored in

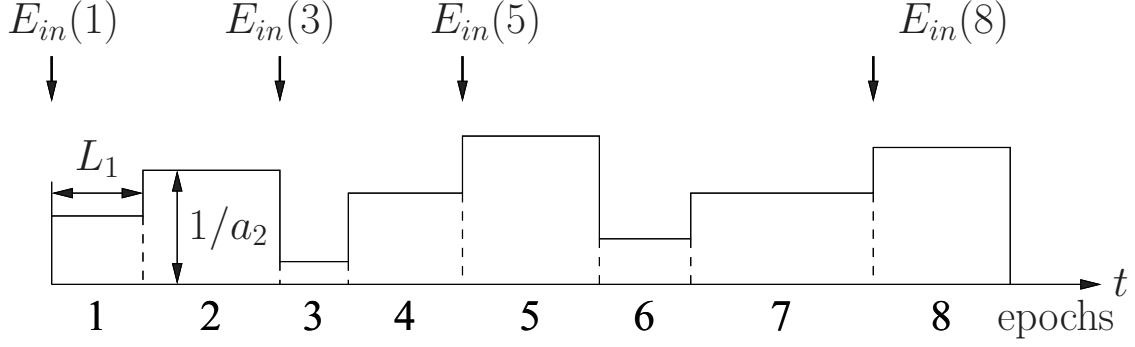


Figure 3.1: Illustration of simple system instance

the battery of the rechargeable transmitter for future use. However, it cannot be used before its arrival. This point is called causality. Often, this battery owns a great storage capacity and it is hardly filled fully from the incoming harvested energy. This assumption is taken into account in this chapter, *i.e.*, the maximum energy capacity of the battery, $E_{max} \gg 0$. This assumption also lays down a solid foundation to solve the cases of finite E_{max} in our incoming research outcome. In this setting, we can compute optimal transmission schemes that adapt the instantaneous transmit power to the variations in the energy and fading levels. Since the proposed optimal dynamic transmission power allocation policy results from the recursive computing, which does not utilize any information in time future, the optimal dynamic power allocation can provide the optimal solution to the maximum throughput for every sub-process or time window from epoch 1 until epoch k , as $k = 1, \dots, K$. This advantage owns more challenge and could be utilized to efficiently solve other problems.

In recent years, energy harvesting green communication has attracted great research attention. In [33], data transmission with energy harvesting sensors is considered, and the optimal online policy for controlling admissions into the data buffer is derived using a dynamic programming framework. Dynamic programming can offer a real-time feedback policy or control, *i.e.*, considering the time constraint of decisions, but it needs to store a family of offline policies. Further, its object is a dynamic optimization problem model, including the dynamic state transition equation(s). To avoid the curse of dimensionality, a “good” dynamic optimization problem model should be needed or set up. In [43], energy management policies stabilizing the data queue are proposed for single-user communication and under a linear approximation, some delay optimality properties are derived. In [12], the optimality of a variant of the back pressure algorithm using energy queues is shown. In [35]

and references therein, optimization approaches are considered to attempt to obtain the maximum throughput over AWGN and fading channels. Successively, in [25], throughput optimal energy allocation is studied for energy harvesting systems in a time constrained slotted setting. Especially in recent work, [35] and [25] investigated the same objective function, in order to attempt to use offline machinery and conventional water-filling approach which come directly from the KKT conditions of the target problem. For the fading channel cases, [35], at the first paragraph on page 1737, used the four sentences to define its “directional water-filling algorithm”. The third sentence is a key point but it still used the term “directional water-filling algorithm” to define the algorithm. Thus, a circular logic or definition seems generated. As a result, optimality of the algorithm is not provided either. Since Algorithm 2, on page 4815 in [25], used its embedded Algorithm 1 to compute the solution to the cases of the full side information, an infinitely iterated algorithm may be required. The proposed algorithm in this chapter could overcome these weaknesses.

With water-filling, more power is allocated to the channels with higher gains to maximize the sum of data rates of all the sub-channels [44]. The conventional way to solve the water-filling problem is to solve the KKT conditions, and then find the water-level(s) and the solutions. In this chapter, we exploit our proposed GWF presented in Chapter 2, and construct a recursive algorithm to solve the target problem and then prove its optimality, referred as RGWF. We have shown that GWF owns less computation. This advantage becomes more significant, especially when GWF is utilized multiple times.

Compared with the existing results on energy harvesting, the proposed RGWF owns three distinct characteristics: 1) for the fading channel cases, the algorithm is clearly defined; 2) it provides the exact optimal solution via finite computation recursively; 3) its optimality is strictly proven. Therefore, following the proposed algorithm, exact solution can be obtained for any sub-process from time $(0, T]$. Numerical examples provide detailed procedures for determining the optimal solution by the proposed RGWF. The contents in this chapter corresponds to that in [20]. *In fact, our proposed algorithm RGWF can lay down the solid fundamental of the “on-line” optimal power or energy allocation for our future energy harvesting study. To the best knowledge of ours, the existing “on-line” algorithms on energy harvesting are all suboptimal, without setting up the solid fundamental.*

In the remaining of the chapter, the proposed GWF is discussed in Section 3.2 with sum power

constraint. The proposed power allocation problem and RGWF is further investigated in Section 3.3. The transmission completion time minimization problems and their solution are investigated in Section 3.4. Numerical examples are presented in Section 3.5.

3.2 Extended GWF for Preparation of EH system

This section is only preparation for the incoming recursive GWF, in this chapter, to maximize the throughput under the EH system. *It is a somewhat extended version of GWF, mentioned in the last chapter. If the readers are familiar with GWF and the extended index of the starting channel and the ending channel, this section can be skipped over.* For Let L and K be two positive integers and $L \leq K$ to denote the index of the starting channel and the ending channel respectively. Often, L is assigned to be 1. The water-filling problem can be abstracted and generalized into the following problem: given $P > 0$, as the total power or volume of the water; the allocated power and the propagation path gain for the i th channel are given as s_i and a_i respectively, $i = L \dots K$; and $K - L + 1$ is the total number of channels. Furthermore, the weighted coefficient $w_i > 0, \forall i$, and $\{a_i w_i\}_{i=L}^K$ being monotonically decreasing, find that

$$\begin{aligned} \max_{\{s_i\}_{i=L}^K} \quad & \sum_{i=L}^K w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \forall i; \\ & \sum_{i=L}^K s_i = P. \end{aligned} \tag{3.1}$$

Since the constraints are that (i) the allocated power to be nonnegative; (ii) the sum of the power equals P , the problem (3.1) is called the water-filling (problem) with sum power constraint.

In this chapter, we propose a novel approach to solve problem (3.1) based on geometric view. The proposed Geometric Water-Filling (GWF) approach eliminates the procedure to solve the non-linear system for the water level, and provides explicit solutions and helpful insights to the problem and the solution.

Similarly, Figs. 3.2(a)-(d) give an illustration of the proposed GWF algorithm. Suppose there are 4 steps/stairs ($L = 1, K = 4$) 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 the step is solved.

Let us use d_i to denote the “step depth” of the i th stair which is the height of the i th step to the bottom of the tank, and is given as

$$d_i = \frac{1}{a_i w_i}, \text{ for } i = L, L+1, \dots, K. \quad (3.2)$$

Since the sequence $a_i w_i$ is sorted as monotonically decreasing, the step depth of the stairs indexed as $\{L, \dots, K\}$ is monotonically increasing.

Instead of trying to determine the water level μ , which is a real nonnegative number, we aim to determine water level step, which is an integer number from L to K , denoted by k^* , as the highest step under water. Based on the result of k^* , we can write out the solutions for power allocation instantly.

Fig. 3.2(a) illustrates the concept of k^* . Since the third level is the highest level under water, we have $k^* = 3$. The shadowed area denotes the allocated power for the third step by s_3^* .

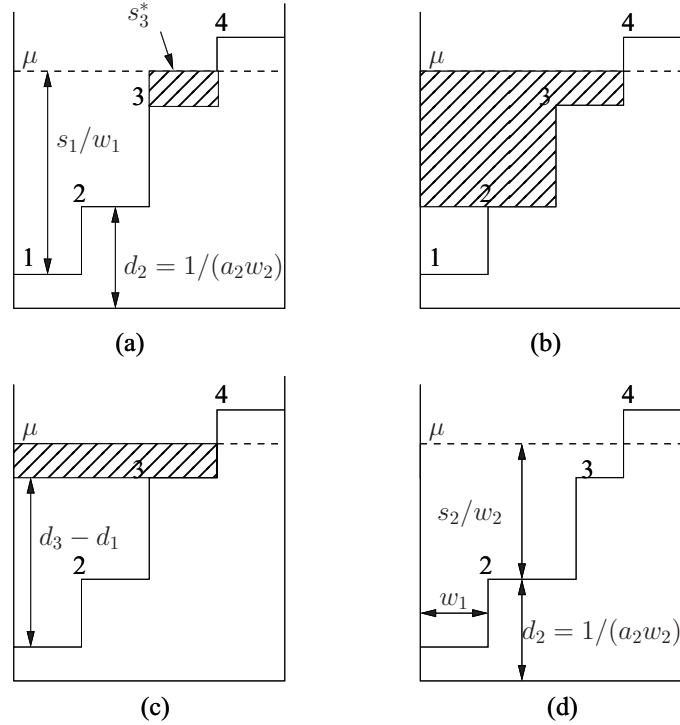


Figure 3.2: Illustration for the proposed Geometric Water-Filling (GWF) algorithm. (a) Illustration of water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = \frac{1}{a_i w_i}$. (b) Illustration of $P_2(k)$ (shadowed area, representing the total water/power above step k) when $k = 2$. (c) Illustration of $P_2(k)$ when $k = 3$. (d) Illustration of the weights as the widths.

In the following, $P_2(k)$, the water volume above step k , can be obtained considering the step

depth difference and the width of the stairs as,

$$\begin{aligned} P_2(k) &= \left[P - \sum_{i=L}^{k-1} (d_k - d_i) w_i \right]^+, \text{ for} \\ k &= L, \dots, K. \end{aligned} \quad (3.3)$$

where $(x)^+ = \max\{0, x\}$.

As an example in Fig. 3.2(c), the water volume above step 1 and below step 3 with the width w_1 can be found as: the step depth difference, $(d_3 - d_1)$ multiplying the width of the step, w_1 . Therefore, the corresponding $P_2(k = 3)$ can be expressed as,

$$P_2(k = 3) = [P - (d_3 - d_1)w_1 - (d_3 - d_2)w_2]^+,$$

which is an expansion of (6.14). Then we have the following proposition for integrity.

Proposition 3.1. The explicit solution to (3.1) is:

$$\begin{cases} s_i = \left[\frac{s_{k^*}}{w_{k^*}} + (d_{k^*} - d_i) \right] w_i, & L \leq i \leq k^* \\ s_i = 0, & k^* < i \leq K, \end{cases} \quad (3.4)$$

where

$$k^* = \max \left\{ k \mid P_2(k) > 0, \quad L \leq k \leq K \right\} \quad (3.5)$$

and the power level for this step is

$$s_{k^*} = \frac{w_{k^*}}{\sum_{i=L}^{k^*} w_i} P_2(k^*). \quad (3.6)$$

Proposition 3.1 has been proven above.

GWF can be regarded as a mapping from the point of parameters

$$\{L, K, \{w_i\}_{i=L}^K, \{a_i\}_{i=L}^K, P\}$$

to the solution $\{s_i\}_{i=L}^K$ and the important water level step index: k^* . That is to say, it can be

written as a formal expression:

$$\{\{s_i\}_{i=L}^K, k^*\} = GWF(L, K, \{w_i\}_{i=L}^K, \{a_i\}_{i=L}^K, P).$$

Note that, for concision and without confusion from context, we may write the right hand side of the expression, mentioned above, as $GWF(L, K)$ to emphasize time stages from L to K .

3.3 Maximizing Throughput in Fading Channel and Algorithm RGWF

In this section, we firstly introduce the maximizing throughput problem in fading channel and the conventional approach from its KKT conditions. Then, we present the proposed online geometric algorithm.

3.3.1 Problem Statement and Conventional Approach

As shown in Fig. 3.1, we consider the time period from $(0, T]$. The channel state changes or/and energy arrives K times in this time period. Hence, we have K epochs, with L_i being the time length of the i th epoch. As a recall, $\{a_i\}$ and $\{E_{in}(i)\}$ have been claimed above. Our objective is to maximize the number of bits transmitted by the deadline T . The optimal power management strategy is such that the transmit power is constant in each event epoch. Therefore, let us again denote the transmit power in epoch i by s_i , for $i = 1, \dots, K$.

We have causality constraints due to energy arrivals and an E_{max} constraint due to finite battery size. Hence, the optimization problem in this fading case becomes [35]:

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K \frac{L_i}{2} \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \quad \forall i; \\ & \sum_{i=1}^l L_i s_i \leq \sum_{i=1}^l E_{in}(i), \quad \text{as } l = 1, \dots, K; \\ & \sum_{i=1}^l E_{in}(i) - \sum_{i=1}^l L_i s_i \leq E_{max}, \quad \forall l, \end{aligned} \tag{3.7}$$

where if we interpret the observed properties of the optimal power allocation scheme as a water-filling scheme mentioned above, $E_{in}(i)$ units of water is filled into a rectangle of bottom width

$L_i, \forall i$. With the assumption of $E_{max} \gg 0$, the last constraint of (3.7) disappears. Note that the last power sum constraint in this narrowed problem is of equality. Furthermore, for unifying parameter notation, through a change of variables, we can obtain an equivalent problem to the narrowed one by $E_{max} \gg 0$ as follows:

$$\begin{aligned} & \max_{\{s_i\}_{i=1}^K} \quad \sum_{i=1}^K w_i \log(1 + a_i s_i) \\ & \text{subject to: } \quad 0 \leq s_i, \quad \forall i; \\ & \quad \quad \quad \sum_{i=1}^l s_i \leq \sum_{i=1}^l E_{in}(i), \forall l, \end{aligned} \tag{3.8}$$

where $w_i \leftarrow \frac{L_i}{2}, a_i \leftarrow \frac{a_i}{L_i}$ and $s_i \leftarrow L_i s_i$. Note that the symbol \leftarrow has been used at last section as assignment operator.

To find the solution to problem (3.8), the parameters of which are given with off-line, the conventional water-filling approach usually starts from the Karush-Kuhn-Tucker (KKT) conditions of the problem as a group of the optimality conditions, then the following system in the variables $\{s_i\}$ and the dual variables can be written as

$$\begin{cases} s_i = \left(\frac{w_i}{\sum_{j=i}^K \lambda_j} - \frac{1}{a_i} \right)^+, \text{ for } i = 1, \dots, K, \\ \sum_{i=1}^l w_i \left(\frac{1}{\sum_{j=i}^K \lambda_j} - \frac{1}{a_i w_i} \right)^+ \leq \sum_{i=1}^l E_{in}(l), \forall l \\ \lambda_j \geq 0, \forall j \end{cases} \tag{3.9}$$

where λ_j is the dual variable corresponding to the j th sum power constraint, for any j . The solution to (3.9) is the solution of the problem (3.8). However, it is not easy to solve (3.9).

3.3.2 Recursive Geometric Water-Filling and Its Optimality

In this section, we propose a novel approach to solve problem (3.8) using our proposed GWF approach. *Similar to those assumptions in [25, 35], the parameters of problem (3.8) are given.* The constraint in (3.8) can be expanded into a matrix form as

$$\begin{pmatrix} 1 & & & \\ 1 & 1 & & \\ \cdot & \cdot & \cdot & \\ 1 & 1 & \cdots & 1 \end{pmatrix} \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_K \end{pmatrix} \leq \begin{pmatrix} \sum_{i=1}^1 E_{in}(i) \\ \sum_{i=1}^2 E_{in}(i) \\ \vdots \\ \sum_{i=1}^K E_{in}(i) \end{pmatrix}. \tag{3.10}$$

Note the coefficients matrix forms a triangle matrix, as a different mathematical structure from that conventional Water-Filling deals with, mentioned before.

The proposed RGWF(K) is stated as in the following Algorithm description:

Algorithm 1 Pseudocode for RGWF

```

1: Initialize:  $L = 1, K, E_{in}(1), w_1, a_1$ ;
2: Output the result for epoch 1:
   RGWF(1) =  $s_1^* = E_{in}(1)$ ;
3: for  $L = 2 : 1 : K$  do
4:   Input:  $\{E_{in}(L), w_L, a_L\}$ ;
5:    $\{s'_k\}_{k=1}^{L-1} = \text{RGWF}(L-1)$ ;
6:   for  $n = L : -1 : 1$  do
7:      $W = \{w_j\}_{j=n}^L$ ;
8:      $A = \{a_j\}_{j=n}^L$ ;
9:      $S_T = \sum_{j=n}^{L-1} s'_j + E_{in}(L)$ ;
10:     $\{\{s_k^*\}_{k=n}^L, k^*\} = \text{GWF}(n, L, W, A, S_T)$ ;
11:    if  $\frac{1}{a_k^* w_k^*} + \frac{s_k^*}{w_k^*} \geq \frac{1}{a_{n-1} w_{n-1}} + \frac{s'_{n-1}}{w_{n-1}}$  and  $s'_{n-1} > 0$ , or  $n == 1$  then
12:      Output:  $\text{RGWF}(L) = \{s'_1, \dots, s'_{n-1}, s_n^*, \dots, s_L^*\}$ ,
13:      Move to next epoch, i.e., go to Line 16;
14:    end if
15:  end for
16: end for

```

RGWF is illustrated as follows. Based on Lines 1-2 of RGWF, as the base case of the recursive definition, the inner loop (Lines 6-15) can be illustrated in Fig. 3.3 where it is assumed that the current processing epoch $L = 6$. The optimal power allocation for the first 5 epochs has been completed as shown in the shadowed area in Fig. 3.3(a). Epoch 6 is now under processing. Based on Line 9, since there is no harvested energy input in epoch 6, the power level for epoch 6 is zero and the water level is just the fading level. Line 10 calculates that $k_e^* = 5$ and then Line 11 compares the water level of current processing window with that of k_e^* th epoch. Since the comparison in Line 11 does not hold, the algorithm goes back to Line 6 by decreasing n to 5 and then the processing window is extended to include epochs 5 and 6 as shown in Fig. 3.3(b). Fig. 3.3(b) also shows

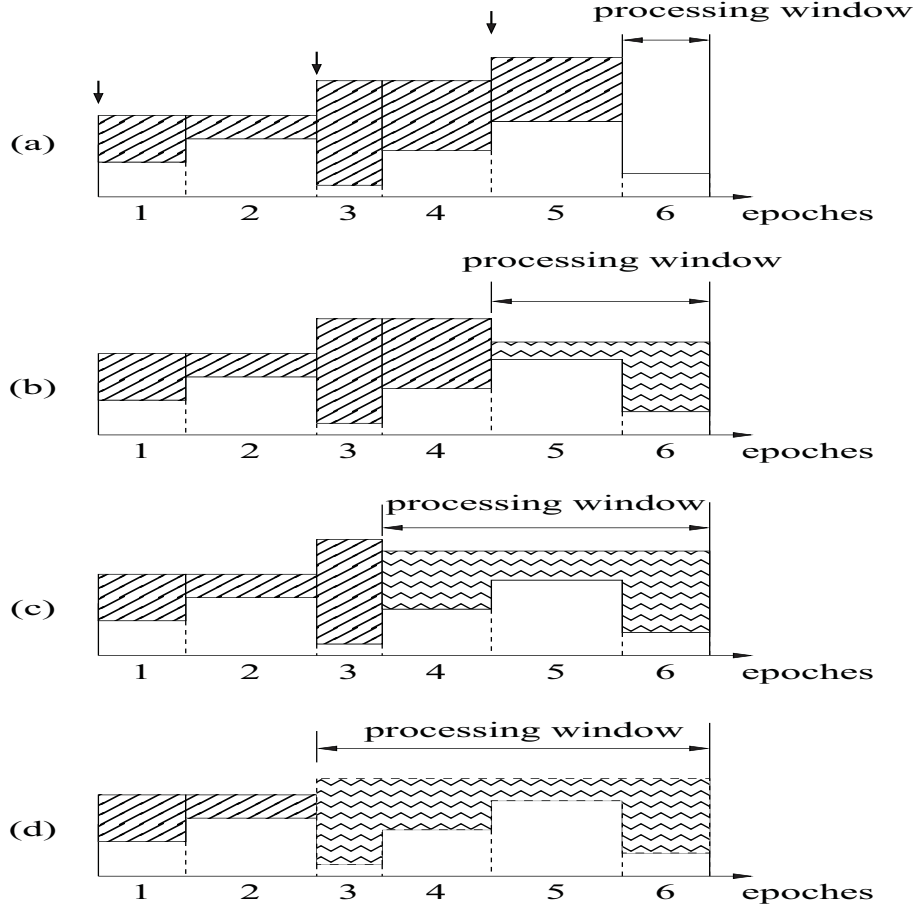


Figure 3.3: Illustration for Algorithm RGWF (Line 6-15 for $L = 6$), harvested energy having been allocated up to epoch 5; horizontal-wave shadowed areas denote power allocation for the processing window; (a) $n = 6, k_e^* = 5$; (b) $n = 5, k_e^* = 4$; (c) $n = 4, k_e^* = 3$; (d) $n = 3, k_e^* = 2$;

the power allocation from GWF(5,6) in Line 9 as the horizontal-wave shadowed areas. Still, the comparison of the water level non-decreasing in Line 11 does not hold, the algorithm returns to Line 6 again by decreasing $n = 4$. As shown in Fig. 3.3(c), the processing window is epochs 4 to 6. The water level non-decreasing condition still is not satisfied. The processing window is extended from epochs 3 to 6 as shown in Fig. 3.3(d). With the new water level in the processing window, the water level non-decreasing condition up to epoch 6 is satisfied. As a result, RGWF($L = 6$) is solved which is recursively obtained from RGWF($L - 1 = 5$) as illustrated in Fig. 3.3.

A summation is used in Line 8. If the lower limit of the summation is greater than the upper limit, the result of this summation is defined as zero, as well known. Through this mechanism, the solution $\{s_i^*\}_{i=1}^K$ is obtained as RGWF(K) within finite loops.

The proposed algorithm eliminates the procedure to solve the non-linear system (3.9) in multiple variables and dual variables, provides online and exact solutions via finite computation steps, and offers helpful insights to the problem and the solutions. To guarantee optimality of RGWF, we have the following proposition:

Proposition 3.2. RGWF can compute the optimal exact solution to problem (3.8) within finite loops.

Proof of Proposition 3.2. From the algorithm RGWF(K), there exists n_1 , where $1 \leq n_1 \leq K - 1$, and $\{s_i^*\}_{i=1}^{n_1} = [\text{RGWF}(K)]|_{\{1, \dots, n_1\}}$. Thus, there are the non-negative Lagrange dual variables $\{\lambda_i\}_{i=1}^{n_1}$ and $\{\mu_i\}_{i=1}^{n_1}$ such that KKT conditions, of the restriction of the optimization problem RGWF(K) to the set $\{1, \dots, n_1\}$, hold. This restriction means a sub-problem:

$$\max_{\{s_i\}_{i=1}^{n_1}} \sum_{i=1}^{n_1} w_i \log(1 + a_i s_i) \quad (3.11)$$

$$\text{subject to: } \sum_{i=1}^l s_i \leq \sum_{i=1}^l E_{in}(i), \quad (3.12)$$

$$0 \leq s_i, \text{ as } 1 \leq l \leq n_1 - 1; \quad (3.13)$$

$$\sum_{i=1}^{n_1} s_i = \sum_{i=1}^{n_1} s_i^*. \quad (3.14)$$

Further, $\{\lambda_i, \mu_i\}$ correspond to the i th sum power constraint and power non-negativeness constraint,

respectively. On the other hand,

$$\{\{s_i^*\}_{i=n_1+1}^K, k^*\} = \text{GWF}(n_1 + 1, K, \{w_j\}, \{a_j\}, \sum_{j=n_1+1}^K s_j^*).$$

Thus, there are also the non-negative Lagrange dual variables λ and $\{\mu_i\}_{i=n_1+1}^K$ that are the KKT conditions of the following sub-problem:

$$\max_{\{s_i\}_{i=n_1+1}^K} \sum_{i=n_1+1}^K w_i \log(1 + a_i s_i) \quad (3.15)$$

$$\text{subject to: } 0 \leq s_i, \forall i; \quad (3.16)$$

$$\sum_{i=n_1+1}^K s_i = \sum_{i=n_1+1}^K s_i^*. \quad (3.17)$$

Since $\text{GWF}(n_1 + 1, K)$ has one sum power constraint and specific finite loop operations, we can assign

$$\lambda_K = \lambda = \frac{1}{\frac{1}{a_{k^*} + w_{k^*}} + \frac{s_{k^*}}{w_{k^*}}} \quad (3.18)$$

$$\lambda_{K-1} = \dots = \lambda_{n_1+1} = 0 \quad (3.19)$$

with the fine k^* , as the minimum positive step index of the set: $\{n_1 + 1, \dots, K\}$, where the adjective “fine” expresses that k^* can be used to clarify whether the allocated power to be positive or zero, and determine the water-level, at once. Also due to characteristics of the loop transition from $n_1 + 1$ to n_1 during carrying out $\text{RGWF}(n_1)$, and the points mentioned above, it is seen that the feasible solution of $\{s_i^*\}_{i=1}^K$, computed by $\text{RGWF}(K)$, is indeed the optimal solution to (3.8).

Therefore, Proposition 3.2 is proved.

Remark 1. RGWF is a recursive algorithm with the characteristics of optimal dynamic online power distribution. Dynamics of this algorithm but that for the target problem is shown by the generalized varying structure state equation on dynamics:

$$\begin{aligned} \text{RGWF}(L + 1) &= [[\text{RGWF}(L)]|_{\Lambda_1}, [\text{GWF}(n, L + 1)]|_{\Lambda_2}], \\ \text{for } L &= 1, \dots, K - 1, \end{aligned} \quad (3.20)$$

where n is the index of the starting epoch of the currently processing window (*i.e.*, it satisfies Line

11 of RGWF), the set Λ_1 denotes $\{s'_k\}_{k=1}^{n-1}$ and this set is referred to in RGWF mentioned above, and the set Λ_2 denotes $\{s_k\}_{k=n}^{L+1}$ and this set is referred to in (3) mentioned before. Thus, we extend the concept of algorithm [53], as a static mapping to a dynamic mapping as a new concept of algorithm to efficiently solve the problems. In this process, RGWF(L) can be regarded as the generalized system state at the time stage (or epoch) L ; GWF($n, L+1$) can be regarded as the generalized system control at the time stage (or epoch) L ; and then RGWF($L+1$), as a state at the next time stage, can be derived or determined from the previous state and control. Due to the optimality of RGWF(L) from Proposition 3.2 for any L , the proposed algorithm is indeed an optimal dynamic water-filling algorithm with high efficiency. *In addition, the proposed algorithm RGWF(K) in this section can output the unique optimal solution to the corresponding problem, since the objective function of the problem has a negative definite Hessian matrix.*

3.4 Transmission Completion Time Minimization

In previous section, RGWF was discussed to efficiently solve the throughput maximization problem. In this section, RGWF is extended to solve the transmission completion time minimization problem.

Now assume that the transmitter has B bits to be transmitted to the receiver. Our objective now is to minimize the time required to transmit these B bits. This problem is called the transmission completion time minimization problem. In [46] and [50], this problem is formulated and solved for an energy harvesting system in a non-fading environment. In [35], the problem is attempted to be solved offline in a fading channel. In this chapter, we use the proposed RGWF to solve the problem in a fading channel with recursive computation, referred to as RGWF_n.

The transmission completing time minimization can be stated as follows, with assuming N to be a positive integer and $N \leq K$.

$$\min_{\{\{s_i\}_{i=1}^N, N\}} N \quad (3.21)$$

$$\text{subject to: } \sum_{i=1}^N w_i \log(1 + a_i s_i) = B; \quad (3.22)$$

$$0 \leq s_i, \quad \forall i; \quad (3.23)$$

$$\sum_{i=1}^l s_i \leq \sum_{i=1}^l E_{in}(i), \quad l = 1, \dots, N. \quad (3.24)$$

The proposed RGWF_n is presented in the algorithm description shown below.

Compared with the steps of RGWF, in the first line of the algorithm, RGWF_n introduces B as a parameter while others being the same. In line 8, RGWF_n sequentially processes from the second epoch to the K th epoch to output the optimal value N^* and its optimal solution: $\{\text{RGWF}_n(N^*), N^*\}$. Similarly, the inner “**for**” loop updates power levels for the current processing epoch (L) and its previous $L - n$ epochs to form a processing window. GWF algorithm is also applied to this window to find a common water level. Note that a new “**if**” clause is inserted into the outer level “**if**” clause (for the water level non-decreasing condition check). The function of this inner “**if**” clause is to check whether the transmitted bits reach B . Therefore, it is the normal exit of the algorithm (lines 23-28). For convenience, the condition of this new “**if**” clause is called the criterion of RGWF_n. This is also due to the importance of the criterion in the following proposition.

Algorithm 2 RGWF_n, Based on RGWF

```
1: Initialize:  $L = 1, K, B, E_{in}(1), w_1, a_1$ ;  
2: Output the result for epoch 1:  
    $s'_1 = \text{RGWF}(1) = E_{in}(1)$ ,  
3: if  $\sum_{i=1}^1 w_i \log(1 + a_i s'_i) \geq B$  then  
4:    $\text{RGWF}_n(1) = s_1^* = \frac{1}{a_L} (2^{\frac{B}{w_L}} - 1)$ ;  
5:    $N^* = 1$ ;  
6:   Exit the algorithm;  
7: end if  
8: for  $L = 2 : 1 : K$  do  
9:   Input:  $\{E_{in}(L), w_L, a_L\}$ ;  
10:   $\{s'_k\}_{k=1}^{L-1} = \text{RGWF}(L-1)$ ;  
11:  for  $n = L : -1 : 1$  do  
12:     $W = \{w_j\}_{j=n}^L; A = \{a_j\}_{j=n}^L$ ;  
13:     $S_T = \sum_{j=n}^{L-1} s'_j + E_{in}(L)$ ;  
14:     $\{\{s_k^*\}_{k=n}^L, k^*\} = \text{GWF}(n, L, W, A, S_T)$ ;  
15:    if  $n > 1$  then  
16:       $k_e^* = \max\{k | s'_k > 0, 1 \leq k \leq n-1\}$ ,  
      else  $k_e^* = 1$ ;  
17:    end if  
18:    if  $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}^*} \geq \frac{1}{a_{k_e} w_{k_e}} + \frac{s_{k_e}^*}{w_{k_e}}$  then  
19:       $\text{RGWF}(L) = \{s'_1, \dots, s'_{n-1}, s_n^*, \dots, s_L^*\}$ ;  
20:       $T_1 = \sum_{i=1}^{n-1} w_i \log(1 + a_i s'_i)$ ;  
21:       $T_2 = \sum_{i=n}^L w_i \log(1 + a_i s_i^*)$ ;  
22:       $T_3 = \sum_{i=n}^{L-1} w_i \log(1 + a_i s_i^*)$ ;  
23:      if  $T_1 + T_2 \geq B$  then  
24:         $B_1 = B - T_1 - T_3$ ;  
25:         $s_L^* = \frac{1}{a_L} (2^{\frac{B_1}{w_L}} - 1)$ ;  
26:         $\text{RGWF}_n(L) = \{s'_i\}_{i=1}^{n-1} \cup \{s_j^*\}_{j=n}^L$ ;  
27:         $N^* = L$ ;  
28:        Exit the algorithm;  
29:      end if  
30:      Move to next epoch, i.e., go to Line 31;  
31:    end if  
32:  end for  
33: end for
```

Proposition 3.3. If there does not exist L such that the criterion in RGWF_n:

$$\sum_{i=1}^{n-1} w_i \log(1 + a_i s'_i) + \sum_{i=n}^L w_i \log(1 + a_i s_i^*) \geq B \quad (3.25)$$

holds, where the symbols in (3.25) keep the same meaning as those in the statement of RGWF_n, then there is no solution to problem (3.21). If the criterion holds, then the obtained N^* is the optimal value and $\{\text{RGWF}_n(N^*), N^*\}$ is the exact optimal solution.

Proof of Proposition 3.3. For the given B , if there does not exist L such that the criterion in RGWF_n:

$$\sum_{i=1}^{n-1} w_i \log(1 + a_i s'_i) + \sum_{i=n}^L w_i \log(1 + a_i s_i^*) \geq B$$

holds, it implies that the optimal value of problem (3.8) is strictly less than B , corresponding to Proposition 3.2. Thus, the first constraint of problem (3.21) never holds. Then there is no solution to problem (3.21).

Then, assume that there exist N^* and $\text{RGWF}(N^*)$ such that

$$\sum_{i=1}^{n-1} w_i \log(1 + a_i s'_i) + \sum_{i=n}^{N^*} w_i \log(1 + a_i s_i^*) \geq B,$$

where

$$\text{RGWF}(N^*) = \{s'_1, \dots, s'_{n-1}, s_n^*, \dots, s_{N^*}^*\}.$$

According to the obtained N^* from the RGWF_n algorithm, the optimal value of the problem

$$\begin{aligned} & \max_{\{s_i\}_{i=1}^N} \sum_{i=1}^N w_i \log(1 + a_i s_i) \\ \text{subject to: } & 0 \leq s_i, \forall i; \\ & \sum_{i=1}^l s_i \leq \sum_{i=1}^l E_{in}(i) \text{ for } l = 1, \dots, N, \end{aligned}$$

is less than B , where $\text{RGWF}(N)$ is the optimal solution to this problem, for $N = 1, \dots, N^* - 1$. Hence, the optimal value of problem (3.21) is not less than N^* . Stemming from the statement of RGWF_n, $\{\text{RGWF}_n(N^*), N^*\}$ is a feasible solution to problem (3.21) and further N^* is the evaluated objective value of problem (3.21) at $\{\text{RGWF}_n(N^*), N^*\}$. Thus, N^* is a feasible value. Together

with the mentioned fact that the optimal value of problem (3.21) is not less than N^* , as a result, N^* is the optimal value and $\{\text{RGWFn}(N^*), N^*\}$ is the exact optimal solution to problem (3.21).

Therefore, Proposition 3.3 is proved.

Remark 2. RGWFn is an optimal dynamic progressive process to compute the transmission completion time minimization. This progressive process is ended at current epoch and then output the minimum completing time once the criterion is satisfied. Hence, it doesn't need the information/solution of the entire process of the problem(s).

In addition, given the lengths of epochs: $\{L_i\}$, due to $w_i = \frac{L_i}{2}, \forall i$, the minimum transmission completion time duration is $2 \sum_{k=1}^{N^*} w_i$, which can be computed by finding N^* .

3.5 Numerical Examples for RRM with Energy Harvesting Transmission

The system model is static. However, our computation does not wait full information input but it can compute the exact optimal solution through finite computation for every sub-process that starts from epoch 1 and ends at epoch k , as $k = 1, \dots, K$, including the entire process. This point can also lean toward designing other efficient algorithms, such as the algorithm to compute the minimum transmission completion time to avoid a tedious and huge backlog of offline or static computation. The minimum transmission completion time problem can be referred to in [35] for details. Due to the limit of pages, its discussion is omitted.

For simple illustration, we assume only three epochs, each with unit weight ($w_i = 1, i = 1, 2, 3$). At the beginning of each epoch, unit energy is harvested ($E_{in}(i) = 1, i = 1, 2, 3$).

Example 1. Suppose the fading profile for the three epochs is $a_1 = 1, a_2 = \frac{1}{2}$ and $a_3 = \frac{1}{3}$.

Epoch 1 is first scanned to output $\text{RGWF}(1)=s_1 = 1$ as shown in Fig. 3.4(a). Now we move to epoch 2 and apply $\text{GWF}(2,2)$ and output $s_2 = 1$. Check if the water level of epoch 2 ($2+1=3$) is greater than the water level of epoch 1 ($1+1=2$). It is true then output the optimal solution at epoch 2: $s_1 = 1; s_2 = 1$ as shown in Fig. 3.4(b). Similarly, for epoch 3, by applying $\text{GWF}(3,3)$, we have $s_3 = 1$. Check the water level, it satisfies non-decreasing condition. So the algorithm outputs the completed solution as shown in Fig. 3.4(c).

Example 1 is calculated out without power level adjustment. In the following example, we

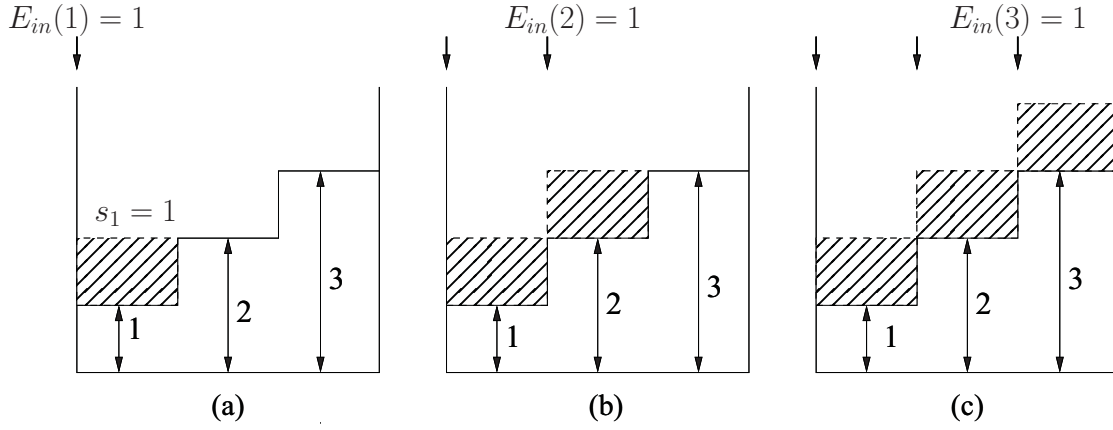


Figure 3.4: Procedures to solve Example 1: (a) $s_1 = 1$; (b) $s_1 = 1, s_2 = 1$; (c) $s_1 = 1, s_2 = 1, s_3 = 1$.

illustrate the power level adjustment procedure.

Example 2. Suppose the fading profile for the three epochs is $a_1 = 1, a_2 = 2$ and $a_3 = 3$. For this example, the proposed RGWF is illustrated in the following fig. 3.5.

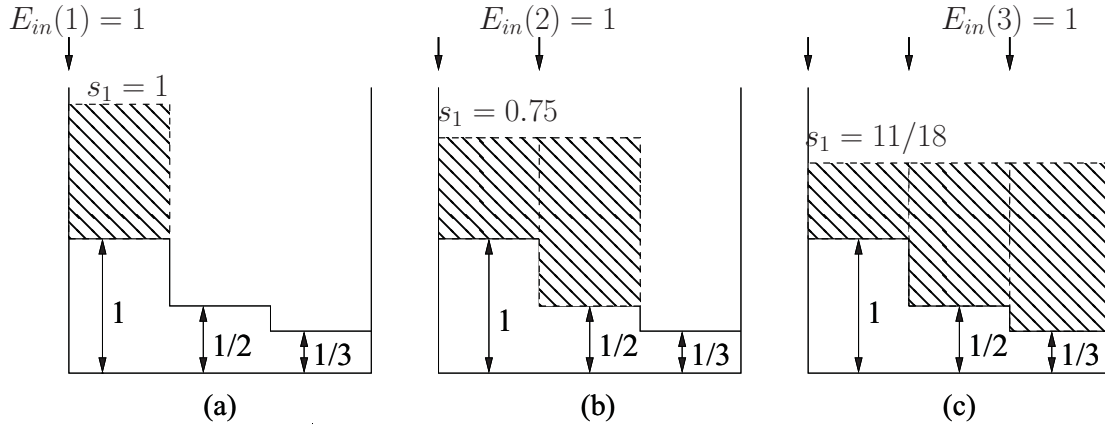


Figure 3.5: Procedures to solve Example 2: (a) $s_1 = 1$; (b) $s_1 = 0.75, s_2 = 1.25$; (c) $s_1 = 11/18, s_2 = 20/18, s_3 = 23/18$.

First, we scan the first epoch and RGWF(1) output $s_1 = 1$, as shown in Fig. 3.5(a). Then move to the second epoch, by applying GWF(2,2), it gives $s_2 = 1$. Now check the water level of epoch 2 is $1+1/2=1.5$ and the water level for epoch 1 is $1+1=2$. Water level non-decreasing condition is violated. Power level adjustment procedure is triggered. By applying GWF to the first two epochs, we have GWF(1,2)={ $s_1 = 0.75, s_2 = 1.25$ }. With this power adjustment, the new water level for both epochs is 1.75, satisfying non-decreasing condition. The output for RGWF(2) is then $s_1 = 0.75, s_2 = 1.25$ as shown in Fig. 3.5(b).

Now we move to epoch 3, the output of $\text{GWF}(3,3)=s_3 = 1$. The corresponding water level for epoch 3 is $1+1/3$, which is lower than the water level of the previous epoch ($=1.75$). Then the power adjustment is triggered. The algorithm calculates the power allocation for current epoch (epoch 3) and its previous epoch (epoch 2) to have output $\text{GWF}(2,3)=\{s_2 = \frac{25}{24}, s_3 = \frac{29}{24}\}$. We move to water level check step. The new water level of epoch 2 ($\frac{1}{2} + \frac{25}{24} = \frac{37}{24}$) is lower than the water level of epoch 1 (1.75). Therefore, power adjustment needs to include epoch 1 as well. We then compute $\text{GWF}(1,3)$, the output is $\{s_1 = \frac{11}{18}, s_2 = \frac{11}{18} + \frac{1}{2}, s_3 = \frac{11}{18} + \frac{2}{3}\}$, which is the completed output for the optimal solution as shown in Fig. 3.5(c).

Example 3. Suppose the fading profile for the three epochs is $a_1 = 1, a_2 = \frac{1}{2}$ and $a_3 = \frac{1}{3}$; the energy harvesting at the beginning of each epoch is $E_{in}(1) = E_{in}(2) = E_{in}(3) = 2$. The information required for transmission is $B = 3$ bits (strictly speaking, B bits/ H_z). Then the minimizing transmission completion time problem is

$$\min_{\{\{s_i\}_{i=1}^N, N\}} N \quad (3.26)$$

$$\text{subject to: } \sum_{i=1}^N \log(1 + a_i s_i) = 3; \quad (3.27)$$

$$0 \leq s_i, \forall i; \quad (3.28)$$

$$\sum_{i=1}^l s_i \leq 2l, l = 1, \dots, N. \quad (3.29)$$

Epoch 1 is first scanned to output $\text{RGWF}(1)=s_1 = 2$ as shown in Fig. 3.6(a). Since $\log(1+2) < B(=3)$, now we move to epoch 2 and apply $\text{GWF}(2,2)$ and output $s_2 = 2$. Check if the water level of epoch 2 ($2+2=4$) is greater than the water level of epoch 1 ($1+2=3$). It is true then output the temporary optimal solution at epoch 2: $s_1 = 2; s_2 = 2$ as shown in Fig. 3.6(b). Since $\log(1+2) + \log(1+1) < B(=3)$, now we move to epoch 3 and apply $\text{GWF}(3,3)$ and output $s_3 = 2$. Check if the water level of epoch 3 ($3+2=5$) is greater than the water level of epoch 2 ($2+2=4$). It is true then output the temporary optimal solution at epoch 3: $s_1 = 2; s_2 = 2; s_3 = 2$. Since $\log 3 + \log 2 + \log \frac{5}{3} > B$, $B_1 = 3 - (\log 3 + \log 2) = \log \frac{4}{3}$. Then, $s_3 = \frac{1}{3}(2^{\log \frac{4}{3}} - 1) = 3 \times (\frac{1}{3}) = 1$. Therefore, The optimal solution is $\{\{s_1^* = 2, s_2^* = 2, s_3^* = 1\}, N^* = 3\}$ and the optimal value is $N^* = 3$ as shown in Fig. 3.6(c).

Example 4. Suppose the fading profile for the three epochs is $a_1 = 1, a_2 = 2$ and $a_3 = 3$;

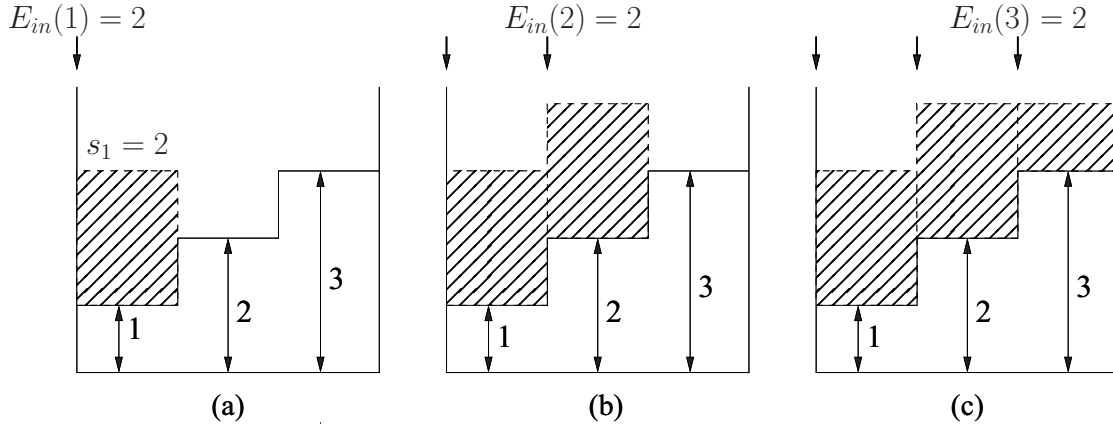


Figure 3.6: Procedures to solve Example 3: (a) $s_1 = 2$; (b) $s_1 = 2, s_2 = 2$; (c) $s_1 = 2, s_2 = 2, s_3 = 1$.

the energy harvesting at the beginning of each epoch is $E_{in}(1) = E_{in}(2) = E_{in}(3) = 1$. The information required for transmission is $B = \log(20)$ bits. The minimizing transmission completion time problem is described as

$$\min_{\{\{s_i\}_{i=1}^N, N\}} N \quad (3.30)$$

$$\text{subject to: } \sum_{i=1}^N \log(1 + a_i s_i) = \log(20); \quad (3.31)$$

$$0 \leq s_i, \quad \forall i; \quad (3.32)$$

$$\sum_{i=1}^l s_i \leq l, \quad l = 1, \dots, N. \quad (3.33)$$

Epoch 1 is first scanned to output $\text{RGWF}(1) = s_1 = 1$ as shown in Fig. 3.7(a). Since $\log(1 + 1) < B (= \log(20))$, now we move to epoch 2 and apply $\text{GWF}(2, 2)$. It gives $s_2 = 1$. Check the water level of epoch 2 which is $1 + 1/2 = 1.5$ and the water level for epoch 1 is $1 + 1 = 2$. Water level non-decreasing condition is violated. Power level adjustment procedure is triggered. By applying GWF to the first two epochs, we have $\text{GWF}(1, 2) = \{s_1 = 0.75, s_2 = 1.25\}$. With this power adjustment, the new water level for both epochs is 1.75, satisfying non-decreasing condition. The output for $\text{RGWF}(2)$ is then $s_1 = 0.75, s_2 = 1.25$ as shown in Fig. 3.7(b). Since $\log(1 + 0.75) + \log(1 + 2 \times 1.25) < B$, we move to epoch 3 and apply $\text{GWF}(3, 3)$. Similarly as in Example 2, the output of $\text{RGWF}(3)$ is $\{s_1 = \frac{11}{18}, s_2 = \frac{11}{18} + \frac{1}{2}, s_3 = \frac{11}{18} + \frac{2}{3}\}$. Since $\log(1 + s_1) + \log(1 + 2s_2) + \log(1 + 3s_3) > B$, $B_1 = \log 20 - \log(1 + \frac{11}{18}) - \log(1 + 2 \times \frac{20}{18}) = \log(10 \times (\frac{18}{29})^2)$. Then, $s_3 = \frac{1}{3}(2^{\log(10 \times (\frac{18}{29})^2)} - 1) = \frac{2399}{2523}$. Therefore, the completed optimal solution is $\{\{s_1^* = \frac{11}{18}, s_2^* = \frac{20}{18}, s_3^* = \frac{2399}{2523}\}, N^* = 3\}$ as shown in

Fig. 3.7(c).

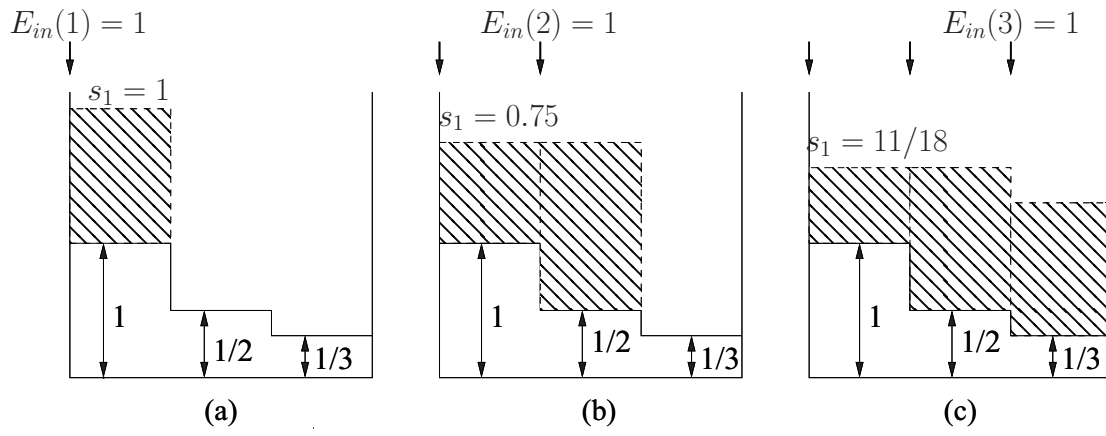


Figure 3.7: Procedures to solve Example 4: (a) $s_1 = 1$; (b) $s_1 = 0.75, s_2 = 1.25$; (c) $s_1 = 11/18, s_2 = 20/18, s_3 = 2399/2523$.

Chapter 4

Optimal Recursive Power Allocation for Energy Harvesting System with Multiple Antennas

In this chapter, we investigate the optimal recursive power allocation policies with energy harvesting wireless nodes equipped with multiple antennas in a fading channel.

This optimization problem includes several complex matrices as optimization variables. As a difference, existing optimization theory and methods have been designed to solve these problems over real space. Naturally, the optimization variables have been assumed to be points in the real space. We proposed a transform approach and designed the algorithms for solving the throughput maximization problem and transmission time minimization problem for a MIMO (multiple-input multiple-output) system. The algorithms were further extended to solve throughput maximization problem of a hybrid system with both harvesting energy and grid power. Numerical results illustrated the algorithms steps and significant efficiency of the proposed algorithms.

To the best knowledge of the authors, there is no existing algorithms reported in the open literature to obtain exact solutions to the proposed problems. Significant features of the proposed algorithms include *i*) they provide the exact optimal solutions via efficient finite computation; *ii*) optimality of the proposed algorithms is strictly proven.

With the property from the aid of the EH and the utilization of MIMO technology, a cooperative MIMO EH communication system (MIMO EH) is formed. The introduction is stated in Section

4.1. System models and proposed maximum throughput problems are mentioned in Section 4.2. In order to find the solution to the maximum throughput problems, the recursive geometric water-filling (RGWF) is then presented in Section 4.3. Then, for minimizing the transmission completion time problems of the MIMO EH, the algorithms based on RGWF is presented in Section 4.4. Extending the MIMO EH to the hybrid system of MIMO, EH and power grid is carried out in Section 4.5. Numerical results and computational complexity are investigated in Section 6.

4.1 Introduction

Prolonging the lifetime of the batteries in wireless communication systems is extremely important. One possible technique to overcome the limitation of battery lifetime is to harvest energy from the environment, such as vibration absorption devices, solar energy, wind energy, thermal energy, and other clean energy. In such systems, energy harvesting has become a preferred choice for supporting “green communication”. Furthermore, the multiple-input multiple-output (MIMO) technology [5] uses multiple antennas at either the transmitter or the receiver or both sides to significantly increase data throughput and link range without additional bandwidth or transmitted power. Thus it plays an important role in advanced wireless communication systems. The focus of this chapter is to investigate the optimal power allocation policies to enhance transmission efficiency for wireless communications with multiple antennas and energy harvesting in fading environment.

4.1.1 Our Work

In our recently published paper [19], we proposed an efficient geometric approach (GWF) to solve water-filling problems. The proposed GWF, as a functional block, was recursively used to solve the power allocation problem with energy harvesting under both fading and single input single output (SISO) channels, *i.e.*, single antenna cases, in [20]. In this chapter, we extend the algorithms in [20] to the MIMO system to solve the problems with energy harvesting causal constraints. Note that [20] cannot directly be used to solve the MIMO cases. Furthermore, it did not consider the related problems under a hybrid system with the coexistence of harvested energy and grid power.

Due to the recursive feature and the repeatedly application of the GWF [19], the proposed algorithms are referred to as RGWFM (Recursive GWF for MIMO) for throughput maximization

and RGWFMn/RGWFMt for transmission minimization problems. They can handle more general cases of the multiple antennas than the cases of the single antenna [20].

Compared with the existing results, the proposed algorithms own two significant and distinguished features: i) it provides the exact optimal solution via finite computation with cubic polynomial computational complexity; ii) its optimality is strictly proven. Due to the usage of recursion, the solutions to a family of the maximum throughput problems for any of the sub-processes starting from epoch 1 to epoch k , for $k = 1, \dots, K$, can be obtained (where K is the index of the last epoch in the process).

Our major contribution in this chapter is: Using our earlier geometric water-filling theory, the maximum throughput and the minimum transmission completion time problems for energy harvesting with multiple antennas are solved with optimal and exact solutions. The throughput maximization problem is further extended to a hybrid system including grid power and optimal solutions being provided. To the best knowledge of the authors, no existing algorithms reported in the open literature could provide exact solutions to the target problem.

4.2 System Model and Proposed Problems

In this section, since system model can be referred to in that of last chapter, we directly introduce the transmission throughput maximization problem with energy harvesting in a fading channel. For convenience and without loss of generality, the process is assumed to be a discrete time process.

For a MIMO-enhanced channel, assume that there are one receiver with N_r antennas, and one transmitter or user, which is equipped with N_t antennas. The received signal at the i th epoch, $\mathbf{y}_i \in \mathbb{C}^{N_r \times 1}$, at the receiver is described as

$$\mathbf{y}_i = \mathbf{H}_i^\dagger \mathbf{x}^i + \mathbf{Z}, \quad \mathbf{H}_i \in \mathbb{C}^{N_t \times N_r}, i = 1, 2, \dots, K, \quad (4.1)$$

where \mathbf{H}_i is the channel gain matrix; $\mathbf{x}^i \in \mathbb{C}^{N_t \times 1}$ is the complex input signal vector transmitted at the i th epoch and is assumed to be a Gaussian random vector, having zero mean for any i , and $\{\mathbf{x}^i\}_{i=1}^K$ are independent on the meaning of probability theory. The noise term, $\mathbf{Z} \in \mathbb{C}^{N_r \times 1}$ is an additive Gaussian noise random vector, *i.e.*, $\mathbf{Z} \sim \mathcal{N}(0, \mathbf{I})$. The channel input, $\{\mathbf{x}^i\}_{i=1}^K$ and \mathbf{Z} are also assumed to be mutually independent. Furthermore, the covariance matrix of the transmit power at

the i th epoch can be expressed as

$$\mathbf{S}_i \triangleq E [\mathbf{x}^i (\mathbf{x}^i)^\dagger], i = 1, 2, \dots, K. \quad (4.2)$$

The total transmit power for the i th epoch is, therefore, $\text{Tr}(\mathbf{S}_i)$. Note that $\mathbf{S}_i, \forall i$, is positively semi-definite, *i.e.*, $\mathbf{S}_i \succeq 0$.

The objective is to maximize the number of bits transmitted by the deadline T , *i.e.*, within the K epochs. Thus, the proposed problem is:

$$\begin{aligned} \min_{\{\mathbf{S}_k\}_{k=1}^K} \quad & - \sum_{i=1}^K \frac{L_i}{2} \log |\mathbf{I} + \mathbf{H}_i^\dagger \mathbf{S}_i \mathbf{H}_i| \\ \text{Subject to:} \quad & \mathbf{S}_i \succeq 0, \forall i; \\ & \sum_{k=1}^l L_i \text{Tr}(\mathbf{S}_k) \leq \sum_{k=1}^l E_{in}(k), \\ & \text{for } l = 1, \dots, K. \end{aligned} \quad (4.3)$$

Furthermore, for unifying parameter notation, through a change of variables, we can obtain an equivalent problem, as follows:

$$\begin{aligned} \min_{\{\mathbf{S}_i\}_{i=1}^K} \quad & - \sum_{i=1}^K w_i \log |\mathbf{I} + \mathbf{H}_i^\dagger \mathbf{S}_i \mathbf{H}_i| \\ \text{subject to:} \quad & \mathbf{S}_i \succeq 0, \forall i; \\ & \sum_{k=1}^l \text{Tr}(\mathbf{S}_k) \leq \sum_{k=1}^l E_{in}(k), \forall l, \end{aligned} \quad (4.4)$$

where $w_i \leftarrow \frac{L_i}{2}, H_i \leftarrow \frac{H_i}{\sqrt{L_i}}$ and $\mathbf{S}_i \leftarrow L_i \mathbf{S}_i$. Note that the symbol \leftarrow is the assignment operator. Since the objective function of problem (4.4) with constraints has complex matrices \mathbf{S}_i s as the optimized variables, and the existing optimization theory and method only solve the real space problems, we need to transform problem (4.4) to an equivalent real form problem, which is stated as follows.

$$\begin{aligned} \min_{\{s_i\}_{i=1}^{N_t \times K}} \quad & - \sum_{i=1}^{N_t \times K} w_{[\frac{i-1}{N_t}] + 1} \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \forall i; \\ & \sum_{i=1}^{N_t \times l} s_i \leq \sum_{i=1}^l E_{in}(i), \forall l, \end{aligned} \quad (4.5)$$

where $[\]$ denotes the integral part of a real number. For simplicity, we write $\{w_{[\frac{i-1}{N_t}] + 1}\}$ as $\{w_i\}$. The transformation for the mentioned equivalence between problem (4.4) and problem (4.5) is stated as

follows. For problem (4.4), since

$$w_i \log |\mathbf{I} + \mathbf{H}_i^\dagger \mathbf{S}_i \mathbf{H}_i| = w_i \log |\mathbf{I} + \mathbf{H}_i \mathbf{H}_i^\dagger \mathbf{S}_i|, \forall i, \quad (4.6)$$

there is the eigenvalue decomposition such that $\mathbf{H}_i \mathbf{H}_i^\dagger$ can be transformed into a diagonal matrix. This procedure is that there exists a unitary matrix \mathbf{U}_i such that

$$\Lambda_i = \mathbf{U}_i \mathbf{H}_i \mathbf{H}_i^\dagger \mathbf{U}_i^\dagger = \begin{pmatrix} a_{(i-1) \times N_t + 1} & & \\ & \ddots & \\ & & a_{(i-1) \times N_t + N_t} \end{pmatrix}, \quad (4.7)$$

where Λ_i is a diagonal matrix and its diagonal element set $\{a_{(i-1) \times N_t + j}\}_{j=1}^{N_t}$, as a sequence, is monotonically decreasing. Thus,

$$\begin{aligned} & w_i \log |\mathbf{I} + \mathbf{H}_i \mathbf{H}_i^\dagger \mathbf{S}_i| \\ &= w_i \log (|\mathbf{U}_i| |\mathbf{I} + \mathbf{H}_i \mathbf{H}_i^\dagger \mathbf{S}_i| |\mathbf{U}_i^\dagger|) \\ &= w_i \log |\mathbf{U}_i \mathbf{U}_i^\dagger + \Lambda_i \mathbf{S}_i'| \\ &= w_i \log |\mathbf{I} + \Lambda_i^{\frac{1}{2}} \mathbf{S}_i' \Lambda_i^{\frac{1}{2}}|, \forall i, \end{aligned} \quad (4.8)$$

where $\mathbf{S}_i' = \mathbf{U}_i \mathbf{S}_i \mathbf{U}_i^\dagger$ for simplifying notation. Furthermore, problem (4.4) is equivalent to the following problem:

$$\begin{aligned} & \min_{\{\mathbf{S}_i'\}_{i=1}^K} - \sum_{i=1}^K w_i \log |\mathbf{I} + \Lambda_i^{\frac{1}{2}} \mathbf{S}_i' \Lambda_i^{\frac{1}{2}}| \\ & \text{subject to: } \mathbf{S}_i' \succeq 0, \forall i; \\ & \sum_{k=1}^l \text{Tr}(\mathbf{S}_k') \leq \sum_{k=1}^l E_{in}(k), \forall l. \end{aligned} \quad (4.9)$$

From the well known Hadamard's inequality on positive definite matrices and some matrix operations, problem (4.9) is equivalent to problem (4.5). Note that, since problem (4.4) earlier mentioned is equivalent to problem (4.9) from the shown derivation above, therefore, problem (4.4) is equivalent to problem (4.5). As a result, we only need to compute the solution to problem (4.5),

and the solution is denoted by $\{s_i^*\}$. Successively, we can obtain

$$\mathbf{U}_i^\dagger \begin{pmatrix} s_{(i-1) \times N_t + 1}^* & & \\ & \ddots & \\ & & s_{(i-1) \times N_t + N_t}^* \end{pmatrix} \mathbf{U}_i, \forall i, \quad (4.10)$$

as the solution to problem (4.4). As a note, equivalence between two optimization problems can be guaranteed only by their having the same optimal value. Further, it needn't require their objective values to be the same for any independent variable.

To find the solution to problem (4.5), the conventional water-filling approach usually starts from the Karush-Kuhn-Tucker (KKT) conditions of the problem, and try to solve a system of equations and inequalities in many optimization variables $\{s_i\}$ and the dual variables. Unlike the conventional approach, our proposed algorithm directly and efficiently solves the target problem by recursion and repeatedly application of our earlier proposed GWF. Due to our constructive solution, it also solves the KKT conditions.

4.3 RGWFM - Recursive GWF for Multiple Antennas

In this section, we propose a novel approach to solve problem (4.5) using our proposed GWF approach.

Since the proposed RGWFM is based on a generalized algorithm of GWF [19], this generalized GWF is still termed as GWF that is concisely introduced as follows. GWF can be regarded as a mapping from the point of parameters $\{L', K', \{w_i\}_{i=L'}^{K'}, \{a_i\}_{i=L'}^{K'}, P\}$ to the solution $\{s_i\}_{i=L'}^{K'}$ and the important water level step index: k^* (which is defined as the highest channel/level index under water), where $\{a_i\}$ and $\{w_i\}$ are respectively fading gain vector and weight vector [19]; L' and K' are two positive integers, to denote respectively the index of the starting channel and the ending channel of a set of channels (for power allocation) sorted according to their channel gains, *i.e.*, $L' \leq K'$; and then $K' - L' + 1$ is the total number of channels. Often, L' is assigned to be 1. That is to say, it can be written as a formal expression:

$$\{\{s_i\}_{i=L'}^{K'}, k^*\} = \text{GWF}(L', K', \{w_i\}_{i=L'}^{K'}, \{a_i\}_{i=L'}^{K'}, P). \quad (4.11)$$

Thus, if $L' = 1$ and $K' = K$, this GWF is regressed into the original GWF. Since we often use the first part, $\{s_i\}_{i=L'}^{K'}$ from GWF, we also write

$$\{s_i\}_{i=L'}^{K'} = \text{GWF}(L', K', \{w_i\}_{i=L'}^{K'}, \{a_i\}_{i=L'}^{K'}, P). \quad (4.12)$$

Note that, for concision and without confusion from context, we may write the right hand side of the expression as $\text{GWF}(L', K')$ to emphasize time stages from L' to K' . Furthermore, the detailed definition, discussion and optimality proof of GWF can be referred to in [19].

Through the mechanism of recursion, the solution $\{s_i^*\}_{i=1}^{KN_t}$ is obtained as $\text{RGWFM}(K)$ within finite loops. Note that in Line 8 shown, we used a summation. If the lower limit of the summation index is greater than the upper limit, the result of this summation is defined as zero.

Algorithm 3 Pseudocode of RGWFM

1. Initialize: $L = 1, N_t, K, P = E_{in}(1), \{w_k\}_{k=1}^{N_t}, \{a_k\}_{k=1}^{N_t}$;
 2. Output the result for epoch 1:
 $\text{RGWFM}(1) = \text{GWF}(1, N_t)|_I = \{s_k^*\}_{k=1}^{N_t}$.
 3. For $L = 2 : 1 : K$,
 4. Input: $\{E_{in}(L), \{w_{(L-1)N_t+j}, a_{(L-1)N_t+j}\}_{j=1}^{N_t}\}$;
 5. $\{s'_k\}_{k=1}^{(L-1)N_t} = \text{RGWFM}(L-1)$;
 6. For $n = L : -1 : 1$,
 7. $W = \{w_j\}_{j=(n-1)N_t+1}^{LN_t}; A = \{a_j\}_{j=(n-1)N_t+1}^{LN_t}$;
 8. $S_T = \sum_{j=(n-1)N_t+1}^{(L-1)N_t} s'_j + E_{in}(L)$;
 9. $\{\{s_k^*\}_{k=(n-1)N_t+1}^{LN_t}, k^*\}$
 $= \text{GWF}((n-1)N_t + 1, LN_t, W, A, S_T)$;
 10. $k_e^* = \max\{k | s'_k > 0, 1 \leq k \leq (n-1)N_t\}$;
 11. If $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}'}{w_{k^*}^*} \geq \frac{1}{a_{k_e} w_{k_e}} + \frac{s_{k_e}'}{w_{k_e}^*}$,
 12. output: $\text{RGWFM}(L)$
 $= \{s'_1, \dots, s'_{(n-1)N_t}, s_{(n-1)N_t+1}^*, \dots, s_{LN_t}^*\}$;
 13. Move to next epoch, *i.e.*, go to Line 16;
 14. End If
 15. End For
 16. End For
-

For algorithm *RGWFM*, all the parameters in problem (4.5) are given except optimization variables. The proposed algorithm eliminates the procedure to solve the non-linear system from the KKT conditions in multiple variables and dual variables, provides exact solutions via finite computation steps, and offers helpful insights to the problem and the solutions. To guarantee optimality of *RGWFM*, we have the following proposition:

Proposition 4.1. *RGWFM* can compute the optimal exact solution to problem (4.5) within

finite loops.

Proof of Proposition 4.1. Problem (4.5) being given implies all the parameters being given except optimization variables. From the algorithm RGWFM(K), there exists n_1 , where $1 \leq n_1 \leq K - 1$, and $\{s_i^*\}_{i=1}^{n_1 N_t} = [\text{RGWFM}(K)]|_{\{1, \dots, n_1 N_t\}} = \text{RGWFM}(n_1)$. Thus, there are the non-negative Lagrange dual variables $\{\lambda_i\}_{i=1}^{n_1 N_t}$ and $\{\mu_j\}_{j=1}^{n_1 N_t}$ such that KKT conditions, of the restriction of the optimization problem RGWFM(K) to the epoch set $\{1, \dots, n_1\}$, hold. This restriction means a sub-problem:

$$\begin{aligned} \min_{\{s_i\}_{i=1}^{n_1 N_t}} \quad & \sum_{i=1}^{n_1 N_t} -w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & \sum_{i=1}^{l N_t} s_i \leq \sum_{i=1}^l E_{in}(i), \text{ as } 1 \leq l \leq n_1 - 1; \\ & 0 \leq s_i, \forall i; \\ & \sum_{i=1}^{n_1 N_t} s_i = \sum_{i=1}^{n_1 N_t} s_i^*. \end{aligned} \tag{4.13}$$

Furthermore, $\{\lambda_i, \{\mu_j\}\}$ correspond to the i th sum power constraint and power non-negativeness constraint, respectively. On the other hand, $\{s_i^*\}_{i=n_1 N_t+1}^{K N_t} = \text{GWF}(n_1 N_t + 1, K N_t, \{w_j\}, \{a_j\}, \sum_{j=n_1 N_t+1}^{K N_t} s_j^*)$. Thus, there are also the non-negative Lagrange dual variables $\{\lambda_i\}_{i=n_1+1}^K$ and $\{\mu_j\}_{j=n_1 N_t+1}^{K N_t}$ that satisfy the KKT conditions of the following sub-problem:

$$\begin{aligned} \min_{\{s_i\}_{i=n_1 N_t+1}^{K N_t}} \quad & - \sum_{i=n_1 N_t+1}^{K N_t} w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \forall i; \\ & \sum_{i=n_1 N_t+1}^{K N_t} s_i = \sum_{i=n_1 N_t+1}^{K N_t} s_i^*. \end{aligned} \tag{4.14}$$

Since $\text{GWF}(n_1 N_t + 1, K N_t)$ has one sum power constraint and specific finite loop operations, we can assign

$$\lambda_K = \frac{1}{\frac{1}{a_{k^*} + w_{k^*}} + \frac{s_{k^*}}{w_{k^*}}} \tag{4.15}$$

$$\lambda_{K-1} = \dots = \lambda_{n_1+1} = 0 \tag{4.16}$$

with the fine k^* , as the minimum positive step index of the set: $\{n_1 N_t + 1, \dots, K N_t\}$, where the adjective “fine” expresses that k^* can be used to clarify whether the allocated power to be positive or zero, and determine the water-level. Also due to characteristics of the loop transition from $n_1 N_t + 1$ to $n_1 N_t$ during carrying out $\text{RGWFM}(n_1)$ and $\text{GWF}(n_1 N_t + 1, K N_t, \{w_j\}, \{a_j\}, \sum_{j=n_1 N_t+1}^{K N_t} s_j^*)$, and the points mentioned above, it is seen that the feasible solution of $\{s_i^*\}_{i=1}^{K N_t}$, computed by

RGWFM(K), is indeed the optimal solution to (4.5) due to the facts that $\{\{\lambda_i\}_{i=1}^K, \{\mu_j\}_{j=1}^{KN_t}\}$ are the dual variables; they, together with $\{s_i^*\}_{i=1}^{KN_t}$, satisfy the KKT conditions of (4.5); and the qualification of (4.5) holds.

Therefore, Proposition 4.1 is proved.

4.4 Transmission Completion Time Minimization

In previous section, RGWFM was discussed as a recursive water-filling to efficiently solve the throughput maximization problem. In this section, RGWFM is used to solve the transmission completion time minimization problem.

Now assume that the transmitter has B bits to be transmitted to the receiver. Our objective now is to minimize the time required to transmit these B bits. This problem is called the transmission completion time minimization problem. In [46] and [50], this problem is formulated and solved for an energy harvesting system in a non-fading environment. In [35], the problem is discussed offline in a fading channel by offering a condition the solution should meet. Single antenna problems are investigated in those works. In this chapter, we use RGWFM to solve the target problem in a fading channel with multiple antennas applying the recursion feature of the computation.

The transmission completion time minimization is categorized into two classes. The first class assumes that the completion time is taken at the ends of the epochs as discrete time points. Since T and $\{L_i\}$ are given, this class of problems just finds the minimum index of the epochs for transmission. The second class of the problem assumes that the completion time is taken at a time point which is continuously located in the interval $[0, T]$, as a continuous straight segment.

4.4.1 Discrete Transmission Completion Time Minimization

The discrete transmission completion time minimization problem can be stated as follows: assume N to be a positive integer and $N \leq K$, noting that the two notations N and N_t stand for the different meanings that the former is the index of an epoch; but the latter is the number of the

antennas equipped by a user. The problem can be written as

$$\begin{aligned}
& \min_{\{\{s_i\}_{i=1}^{NN_t}, N\}} && N \\
& \text{subject to:} && \sum_{i=1}^{NN_t} w_i \log(1 + a_i s_i) = B; \\
& && 0 \leq s_i, \forall i; \\
& && \sum_{i=1}^{lN_t} s_i \leq \sum_{i=1}^l E_{in}(i), l = 1 \dots, K.
\end{aligned} \tag{4.17}$$

We use the RGWFM machinery to design a recursive algorithm to solve (4.17), referred to as RGWFMn. The steps of the RGWFMn is presented in the algorithm description shown below.

Compared with the steps of RGWFM, in the first line of the algorithm, RGWFMn introduces B as a parameter while others being kept unchanged. In Line 6, RGWFMn sequentially processes from the second epoch to the K th epoch to output the optimal value N^* and its optimal solution: RGWFM(N^*) with the target rate: B bits. Similarly, the inner “For” loop updates power levels for the current processing epoch (L) and its previous ($L - n + 1$) epochs to form a processing window. GWF algorithm is also applied to this window to find a common water level. Note that a new “If” clause is inserted into the outer level “If” clause (for the water level non-decreasing condition check). The function of this inner “If” clause is to check whether and how the transmitted bits reach B . Therefore, it is the normal exit of the algorithm (lines 19-20). For convenience, the condition of this new “If” clause is called the criterion of RGWFMn. This is also due to the importance of the criterion in the following proposition. As a note, truth of Line 4 is easily to test. Also, Line 4 of RGWFMn provided a solution but this solution is not unique. However, this proposed solution uses the least power to guarantee the rate requirement ($= B$). As a note, however, the mentioned algorithm of RGWFM(K) can output the unique optimal solution to problem (4.5), since the objective function of problem (4.5) has a negative definite Hessian matrix.

To guarantee optimality of RGWFMn, the proposition is stated as follows:

Proposition 4.2. If there does not exist L such that the criterion in RGWFMn:

$$\sum_{i=1}^{(n-1)N_t} w_i \log(1 + a_i s'_i) + \sum_{i=(n-1)N_t+1}^{LN_t} w_i \log(1 + a_i s_i^*) \geq B \tag{4.18}$$

holds, where the symbols in (4.18) keep the same meaning as those in the statement of RGWFMn, then there is no solution to problem (4.17). If the criterion holds, then the obtained N^* is the

Algorithm 4 Algorithm RGWFMn, Based on RGWFM

1. Initialize: $L = 1, N_t, K, E_{in}(1), \{w_k\}_{k=1}^{N_t}, \{a_k\}_{k=1}^{N_t}, B$;
 2. Output the result for epoch 1:

$$\text{RGWFM}(1) = \text{GWF}(1, N_t, \{w_i\}, \{a_i\}, E_{in}(1))|_I$$

$$= \{s_k^*\}_{k=1}^{N_t};$$
 3. If $\sum_{i=1}^{N_t} w_i \log(1 + a_i s_i^*) \geq B$,
 4. $\{s_i^*\}_{i=1}^{N_t} = \text{GWF}(1, N_t, \{w_i\}, \{a_i\}, P_1)$,
 where $P_1 = \sum_{i=1}^{k_1^*} [\frac{1}{a_{k_1^*}} (\frac{2^{\frac{B}{w_1}}}{\prod_{i=1}^{k_1^*-1} (\frac{a_i}{a_{k_1^*}})})^{\frac{1}{k_1^*}} - \frac{1}{a_i}]$,

$$k_1^* = \max \{k | \prod_{i=1}^{k-1} (\frac{a_i}{a_k}) < 2^{\frac{B}{w_1}}, 1 \leq k \leq N_t\},$$

$$N^* = 1, \text{ and then exit the algorithm};$$
 5. End If
 6. For $L = 2 : 1 : K$,
 7. Input: $\{E_{in}(L), \{w_{(L-1)N_t+j}, a_{(L-1)N_t+j}\}_{j=1}^{N_t}\}$;
 8. $\{s'_k\}_{k=1}^{(L-1)N_t} = \text{RGWFM}(L-1)$;
 9. For $n = L : -1 : 1$,
 10. $W = \{w_j\}_{j=(n-1)N_t+1}^{LN_t}; A = \{a_j\}_{j=(n-1)N_t+1}^{LN_t}$;
 11. $S_T = \sum_{j=(n-1)N_t+1}^{(L-1)N_t} s'_j + E_{in}(L)$;
 12. $\{\{s_k^*\}_{k=(n-1)N_t+1}^{LN_t}, k^*\}$
 $= \text{GWF}((n-1)N_t + 1, LN_t, W, A, S_T)$;
 13. $k_e^* = \max \{k | s'_k > 0, 1 \leq k \leq (n-1)N_t\}$;
 14. If $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}^*}{w_{k^*}} \geq \frac{1}{a_{k_e} w_{k_e}} + \frac{s_{k_e}'}{w_{k_e}}$,
 15. $\text{RGWFM}(L) =$
 $\{s'_1, \dots, s'_{(n-1)N_t}, s_{(n-1)N_t+1}^*, \dots, s_{LN_t}^*\}$,
 16. If $\sum_{i=1}^{(n-1)N_t} w_i \log(1 + a_i s'_i) +$
 $\sum_{i=(n-1)N_t+1}^{LN_t} w_i \log(1 + a_i s_i^*) \geq B$,
 17. $B_1 = B - \sum_{i=1}^{(n-1)N_t} w_i \log(1 + a_i s'_i) -$
 $\sum_{i=(n-1)N_t+1}^{(L-1)N_t} w_i \log(1 + a_i s_i^*)$,
 18. $\{s_i^*\}_{i=(L-1)N_t+1}^{LN_t}$ can be obtained, similar to Line 4,
 19. $\text{RGWFM}(L) =$
 $\{s'_i\}_{i=1}^{(n-1)N_t} \cup \{s_j^*\}_{j=(n-1)N_t+1}^{LN_t}$,
 20. $N^* = L$, and then exit the algorithm;
 21. End If
 22. Move to next epoch, *i.e.*, go to Line 25;
 23. End If
 24. End For
 25. End For
-

optimal value and the $\{\text{RGWFM}(N^*), N^*\}$ is the exact optimal solution.

Proof of Proposition 4.2. For the given B , if there does not exist L and n such that the criterion in RGWFM_n ,

$$\sum_{i=1}^{(n-1)N_t} w_i \log(1 + a_i s'_i) + \sum_{i=(n-1)N_t+1}^{LN_t} w_i \log(1 + a_i s_i^*) \geq B, \quad (4.19)$$

holds, it implies that the optimal value of problem (4.5) is strictly less than B , corresponding to Proposition 4.1. Thus, the first constraint in problem (4.17) never holds. Then there is no solution to problem (4.17).

Then, assume that there exist N^* and $\text{RGWFM}(N^*)$ such that

$$\sum_{i=1}^{(n-1)N_t} w_i \log(1 + a_i s'_i) + \sum_{i=(n-1)N_t+1}^{N^*N_t} w_i \log(1 + a_i s_i^*) \geq B, \quad (4.20)$$

where

$$\text{RGWFM}(N^*) = \{s'_1, \dots, s'_{(n-1)N_t}, s_{(n-1)N_t+1}^*, \dots, s_{N^*N_t}^*\}. \quad (4.21)$$

According to the obtained N^* from the RGWFM_n algorithm, the optimal value of the problem

$$\begin{aligned} \min_{\{s_i\}_{i=1}^{NN_t}} \quad & \sum_{i=1}^{NN_t} -w_i \log(1 + a_i s_i) \\ \text{subject to:} \quad & 0 \leq s_i, \quad \forall i; \\ & \sum_{i=1}^{lN_t} s_i \leq \sum_{i=1}^{lN_t} E_{in}(i) \text{ for } l = 1, \dots, N, \end{aligned} \quad (4.22)$$

is less than B , where $\text{RGWFM}(N)$ is the optimal solution to this problem, for $N = 1, \dots, N^* - 1$. Hence, the optimal value of problem (4.17) is not less than N^* . Stemming from the statement of RGWFM_n , $\text{RGWFM}(N^*)$ is a feasible solution to problem (4.17) and N^* is the evaluated objective value of problem (4.17) at $\text{RGWFM}(N^*)$. Thus, N^* is a feasible value. Together with the mentioned fact that the optimal value of problem (4.17) is not less than N^* , as a result, N^* is the optimal value and $\{\text{RGWFM}(N^*), N^*\}$ is the exact optimal solution to problem (4.17).

Therefore, Proposition 4.2 is proved.

Note that for the first constraint in (4.17), if the powers make the weighted sum-rate greater than the target B , we may reduce the powers to make the weighted sum-rate equal to B . The reduced powers is just a feasible solution and the sum power becomes less. This result comes from

$$\begin{aligned}
& \min_{\{\{s_i\}_{i=1}^{NN_t}, t\}} && t \\
& \text{subject to:} && 1 \leq N \leq K \text{ and } N \in \mathbb{Z}; \\
& && N_1(t) = \max\{N \mid \sum_{k=1}^N L_k \leq t\}; \\
& && \sum_{l=1}^{N_1} \sum_{j=1}^{N_t} w_{(l-1)N_t+j} \cdot \log(1 + a_{(l-1)N_t+j} \cdot s_{(l-1)N_t+j}) + \\
& && \quad \left(\frac{t}{2} - \sum_{k=1}^{N_1} w_k\right) \cdot \sum_{j=1}^{N_t} \log(1 + a_{(N_1-1)N_t+j} s_{(N_1-1)N_t+j}) = B; \\
& && 0 \leq s_i, \quad \forall i; 0 \leq t \leq T; \\
& && \sum_{i=1}^{lN_t} s_i \leq \sum_{i=1}^l E_{in}(i), l = 1 \dots, K.
\end{aligned} \tag{4.23}$$

continuity of the throughput or rate constraint function and rest of the constraints. Therefore, the problem that substitutes the inequality of “ \geq ” for the equality of “ $=$ ” in the first constraint, has the same optimal solution set.

4.4.2 Continuous Transmission Completion Time Minimization

The continuous transmission completion time minimization problem can be stated as follows: assume t to be a real number and N to denote the index variable of the sequence, consisting of the ends for the progressive epochs, \mathbb{Z} to denote the set of integers, and then the corresponding objective function is shown as in (4.23) given in next page. Note the mentioned $w_i = \frac{L_i}{2}, \forall i$.

If Lebesgue-Stieltjes integration [11] is used for problem (4.23), it can make the expression concise. The presented method is used to avoid introducing more abstract mathematical expressions. We use the proposed RGWFMn to design a recursive algorithm to solve the continuous transmission completion time minimization problem (4.23), referred to as RGWFMt. The steps of the RGWFMt are stated below: RGWFMt only replaces Line 18 and the $N^* = L$ of Line 20 of RGWFMn with the statement:

$$\Delta t^* = \frac{2B_1}{\sum_{j=1}^{N_t} \log(1 + a_{(N^*-1)N_t+j} s_{(N^*-1)N_t+j})}, \tag{4.24}$$

and the statements:

$$N^* = L, t^* = \Delta t^* + \sum_{k=1}^{N^*-1} L_k, \tag{4.25}$$

respectively. The optimality proof of RGWFMt can be proven, similarly to that of RGWFMn: Proposition 4.2. Therefore, its proof is ignored in this chapter.

4.5 Extension to a Hybrid System Coexisting with Grid Power

Since energy harvesting depends on natural condition and it is a random process, the energy from power grid is often added as a supplementary source. The corresponding maximum throughput problem of such a hybrid system can be stated as

$$\begin{aligned}
& \min_{\{\mathbf{S}_i, \mathbf{S}_{G,i}\}_{i=1}^K} && - \sum_{i=1}^K w_i \log |\mathbf{I} + \mathbf{H}_i^\dagger (\mathbf{S}_i + \mathbf{S}_{G,i}) \mathbf{H}_i| \\
& \text{subject to:} && \mathbf{S}_i \succeq 0, \forall i; \\
& && \mathbf{S}_{G,i} \succeq 0, \forall i; \\
& && \sum_{k=1}^l \text{Tr}(\mathbf{S}_k) \leq \sum_{k=1}^l E_{in}(k), \forall l; \\
& && \sum_{k=1}^K \text{Tr}(\mathbf{S}_{G,k}) \leq E_{(G,\text{total})},
\end{aligned} \tag{4.26}$$

where $\mathbf{S}_{G,k}, \forall k$, is the power from power grid, and $E_{(G,\text{total})}$ is the total energy from power grid. To easily understand the essence and the proof of this extension and avoid more subscripts being used, without loss of generality, let $N_t = 1$ in this section.

For problem (4.26), We add a statement

$$\{s_{G,k}^*\}_{k=1}^K = \text{GWF} \left(1, K, W, \frac{a_k}{1 + a_k s_{H,k}^*}, E_{(G,\text{total})} \right) |_I \tag{4.27}$$

at the end of RGWFM for grid power allocation. We refer to this as RGWFMH. RGWFMH implies that for this hybrid system, harvested energy is first allocated as in the algorithm RGWFM. The results from RGWFM defines the new water tank bottom. The total available grid power is then allocated again with water-filling algorithm.

Optimality of RGWFMH is stated and proven by the following proposition.

Proposition 4.3. RGWFMH can compute the optimal exact solution to the problem (4.26) within finite loops.

Proof of Proposition 4.3. Similar to the transformation from (4.4) to (4.5), problem (4.26) has its real representation or problem. For clarity, $\{s_{H,i}\}$ denotes the powers from energy harvesting and $\{s_{G,i}\}$ from power grid.

According to Proposition 4.1, for the real form of problem (4.26) under $E_{(G,\text{total})} = 0$, there

exist the optimal solution $\{s_{H,i}\}_{i=1}^K$ and the dual variables $\{\lambda_i, \mu_i\}_{i=1}^K$ satisfy the following KKT conditions:

$$\begin{cases} \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i}}{w_i}} = \sum_{k=i}^K \lambda_k - \mu_i, \forall i; \\ \mu_i s_{H,i} = 0, s_{H,i} \geq 0, \mu_i \geq 0, \forall i; \\ \lambda_l (\sum_{k=1}^l s_{H,k} - \sum_{k=1}^l E_{in}(k)) = 0, \\ \sum_{k=1}^l s_{H,k} \leq \sum_{k=1}^l E_{in}(k), \lambda_l \geq 0, \forall l. \end{cases} \quad (4.28)$$

We define the index set Λ_1 as follows:

$$\Lambda_1 = \{i | s_{H,i} > 0\} = \{i_t | 1 \leq i_1 < i_2 < \dots < i_{t_1} \leq K\}. \quad (4.29)$$

Expression (4.29) combined with RGWFM(K), implies that

$$\frac{1}{a_{i_k} w_{i_k}} + \frac{s_{H,i_k}}{w_{i_k}} \leq \frac{1}{a_{i_{k+1}} w_{i_{k+1}}} + \frac{s_{H,i_{k+1}}}{w_{i_{k+1}}}, \quad (4.30)$$

for any $\{i_k, i_{k+1}\} \subset \Lambda_1$. Note that optimal solution satisfies the mentioned inequalities (4.30), but only some variables which meet these inequalities cannot guarantee themselves to be (the part of) the optimal solution.

Let

$$\{\{s_{G,k}\}_{k=1}^K, k^*\} = \text{GWF}(1, K, W, \{\frac{a_k}{1+a_k S_{H,k}^*}\}, E_{(G, \text{total})}). \quad (4.31)$$

It is seen that

$$\frac{1}{a_{i_t} w_{i_t}} + \frac{s_{H,i_t}}{w_{i_t}} + \frac{s_{G,i_t}}{w_{i_t}} = \frac{1}{a_{k^*} w_{k^*}} + \frac{s_{G,k^*}}{w_{k^*}} + \frac{s_{G,k^*}}{w_{k^*}}, \quad (4.32)$$

for any $\{k^*, i_t\} \subset \Lambda_1$, where $1 \leq i_t \leq k^*, \forall t$.

Let $\bar{\mu}_{i_t} = 0$, as $i_t \in \Lambda_1$; $\bar{\lambda} = \frac{1}{\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{H,k^*}}{w_{k^*}} + \frac{s_{G,k^*}}{w_{k^*}}}$; and $\bar{\lambda}_{k^*} = \bar{\lambda}$ and $\bar{\lambda}_l = 0$, as $l \in \{1, 2, \dots, K\} \setminus \{k^*\}$. Also, $\bar{\mu}_j = \left(\bar{\lambda}_{k^*} - \frac{1}{\frac{1}{a_j w_j} + \frac{s_{H,j}}{w_j}} \right)_+$, as $j \in \{1, 2, \dots, K\} \setminus \Lambda_1$. In addition, $\bar{\nu}_{i_t} = 0$, as $i_t \in \Lambda_1$; and $\bar{\nu}_j = \left(\frac{1}{\frac{1}{a_j w_j} + \frac{s_{H,j}}{w_j} + \frac{s_{G,j}}{w_j}} - \bar{\lambda} \right)_+$, as $j \in \{1, 2, \dots, K\} \setminus \Lambda_1$.

Therefore, the $\{s_{H,i}, s_{G,i}\}_{i=1}^K$ mentioned above and the constructed dual variables $\{\bar{\lambda}_i, \bar{\mu}_i\}_{i=1}^K$ and $\{\bar{\lambda}, \bar{\nu}_1, \dots, \bar{\nu}_K\}$ satisfy the KKT conditions of the real form of problem (4.26), the Lagrange function

of which is:

$$\begin{aligned}
& L(\{s_{H,i}, s_{G,i}\}_{i=1}^K; \{\bar{\lambda}_i, \bar{\mu}_i\}_{i=1}^K, \{\bar{\lambda}, \bar{\nu}_1, \dots, \bar{\nu}_K\}) \\
&= -\sum_{i=1}^K w_i \log(1 + a_i(s_{H,i} + s_{G,i})) - \sum_{i=1}^K \mu_i s_{H,i} \\
&+ \sum_{l=1}^K \lambda_l (\sum_{k=1}^l s_{H,k} - \sum_{k=1}^l E_{in}(k)) \\
&- \sum_{i=1}^K \nu_i s_{G,i} + \lambda (\sum_{k=1}^K s_{G,k} - E_{(G,\text{total})}).
\end{aligned} \tag{4.33}$$

In addition, by observation, the general constraint qualification of problem (4.26) holds. Then $\{s_{H,i}, s_{G,i}\}_{i=1}^K$ computed by the proposed RGWFMH is the optimal solution to problem (4.26).

Therefore, Proposition 4.3 is proved.

As a remark, we should emphasize two points: (1) RGWFMH is actually a block coordinate ascent algorithm (BCAA, to find maximum) ([2]). BCAA is only guaranteed to be infinite iterations to find an optimal solution to the proposed problem. However, we further exploit the structure of the proposed problem and then the proposed RGWFMH is designed. Further, RGWFMH just uses one time iteration and rapidly obtains the optimal solution; (2) we may treat $E_{(G,\text{total})}$ to be the harvested energy at the starting of the first epoch and can obtain the equivalent solution, but we cannot obtain or distinguish the optimal power allocation from the two different energy sources. With the implementation of RGWFMH, this problem can be solved efficiently.

4.6 Numerical Examples and Computational Complexity

This section consists of two subsections: numerical examples and computational complexity analysis. The former subsection firstly uses one simple example to clearly illustrate the procedures of the proposed algorithms. Then two more complicated examples to compare with the primal-dual interior point method (PD-IPM) which is now regarded as an efficient optimization algorithm with great promise ([6] and references therein). The latter subsection discusses computational complexity of the proposed algorithms, and arrives at the conclusion of polynomial complexity ([38]). Due to exploiting the structure of the proposed problems, the proposed algorithms show significant efficiency.

4.6.1 Numerical Examples

The proposed algorithms scan the epochs sequentially to obtain the optimal power allocation.

Example 1. we assume that there are three epochs, each with unit length ($L_i = 1, i = 1, 2, 3$) and the same unit weight ($w_i = \frac{1}{2}, i = 1, 2, 3$). To clearly account for the procedures of the proposed algorithms, no power grid case is considered without loss of essence.

Let $N_t = N_r = 2$. At the beginning of each epoch, the energy is harvested as $E_{in}(i) = 2, i = 1, 2, 3$. The information required for transmission is $B = 3$ bits. Suppose the fading profile for the three epochs is

$$\begin{aligned} \mathbf{H}_1 &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}, \mathbf{H}_2 = \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} \text{ and} \\ \mathbf{H}_3 &= \sqrt{2} \begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix}. \end{aligned} \quad (4.34)$$

First, RGWFM(1) outputs $\{s_1 = 1, s_2 = 1\}$, *i.e.*, the allocated power sum is $\text{Tr}(\mathbf{S}_1) = 2$ to epoch 1, as shown in Fig. 4.1(a). The height of the blue (darker) stair bars denotes the reciprocals of the fading gains, *i.e.*, the height of the steps for water-filling [19]. The allocated powers are illustrated by the height of the green (grey) bars. Along the axis of “Index of epoch”, the index of epochs increases from left to right.

Then the process moves to the second epoch. By applying GWF(2,2), it gives $\{s_3 = 1, s_4 = 1\}$. Now check the water level of epoch 2 is $1+1/2=1.5$ and the water level for epoch 1 is $1+1=2$. Water level non-decreasing condition is violated. Power level adjustment procedure is triggered. By applying GWF to the first two epochs, we have $\text{GWF}(1,2)=\{s_1 = 0.75, s_2 = 0.75; s_3 = 1.25, s_4 = 1.25\}$. With this power adjustment, the output for RGWFM(2) is then $\{s_1 = 0.75, s_2 = 0.75; s_3 = 1.25, s_4 = 1.25\}$, *i.e.*, $\text{Tr}(\mathbf{S}_1) = 1.5$ and $\text{Tr}(\mathbf{S}_2) = 2.5$, as shown in Fig. 4.1(b).

Now the process moves to epoch 3, the output of $\text{GWF}(3,3)=\{s_5 = 1, s_6 = 1\}$. The corresponding water level for epoch 3 is $1 + \frac{1}{4}$, which is lower than that of the previous epoch ($=1.75$). Then the power adjustment is triggered. After twice power adjustment operations, the final output is $\{s_1 = 7/12, s_2 = 7/12; s_3 = 13/12, s_4 = 13/12; s_5 = 4/3, s_6 = 4/3\}$, which is the completed output for the optimal solution, *i.e.*, $\text{Tr}(\mathbf{S}_1) = 14/12$, $\text{Tr}(\mathbf{S}_2) = 26/12$ and $\text{Tr}(\mathbf{S}_3) = 8/3$, as shown in Fig. 4.1(c).

In this example, the channel states (fading gains) for the three epochs are continuously improving. Therefore, the harvested energy at the beginning of each epoch tries to flow to the later epochs, leading to the uniform water level of these three epochs. In addition, if $N_t = N_r = 1$,

the power gains are reduced into $h_i = 2^{i-1}, i = 1, 2, 3$, it can be calculated that the maximum transmission throughput is $3 \log(31/6)$. However, if $N_t = N_r = 2$, the maximum transmission throughput is $2 \log(8 \cdot (19/12)^3)$. Thus, the throughput ratio of the MIMO case to the SISO case is $\frac{2 \log(8 \cdot (19/12)^3)}{3 \log(31/6)} \doteq 1.40$ by adding one more antenna at both the transmitter and the receiver.

Let us compute the solutions to two types of the minimum transmission completion time problems. For the final result over the entire process, the optimal solution is $\{s_1 = 7/12, s_2 = 7/12; s_3 = 13/12, s_4 = 13/12; s_5^* = \log(4 \cdot (12/19)^2), s_6^* = \log(4 \cdot (12/19)^2)\}, N^* = 3\}$ and the optimal value is $N^* = 3$, *i.e.*, the allocated power is $\text{Tr}(\mathbf{S}_3) = 2 \log(4 \cdot (12/19)^2)$ to epoch 3, as shown in Fig. 4.1(d).

For continuous solution t , we obtain a different solution using RGWFMt: $\{s_1 = 7/12, s_2 = 7/12; s_3 = 13/12, s_4 = 13/12; s_5^* = 4/3, s_6^* = 4/3\}$, and

$$\begin{aligned} t^* &= 2 + \frac{2B_1}{\sum_{j=1}^{N_t} \log(1 + a_{(N^*-1)N_t+j} s_{(N^*-1)N_t+j})} \\ &= 2 + \frac{\log(4 \cdot (12/19)^2)}{\log(1 + 4 \cdot 4/3)} \\ &\doteq 2.25. \end{aligned} \tag{4.35}$$

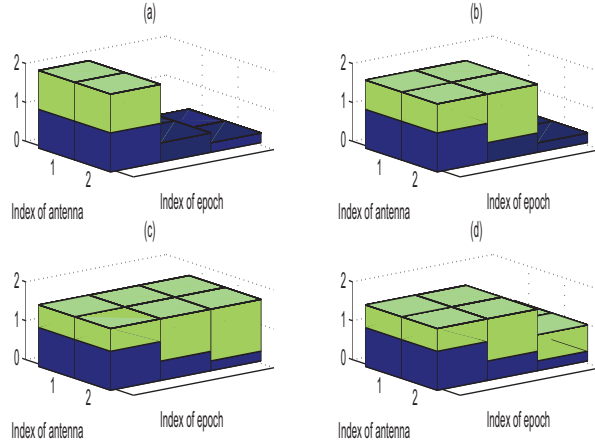


Figure 4.1: Procedures to solve Example 2: (a) $\text{Tr}(\mathbf{S}_1) = 2$; (b) $\text{Tr}(\mathbf{S}_1) = \frac{3}{2}, \text{Tr}(\mathbf{S}_2) = \frac{5}{2}$; (c) $\text{Tr}(\mathbf{S}_1) = \frac{14}{12}, \text{Tr}(\mathbf{S}_2) = \frac{26}{12}, \text{Tr}(\mathbf{S}_3) = \frac{8}{3}$; (d) $\text{Tr}(\mathbf{S}_1) = \frac{7}{6} \doteq 1.17, \text{Tr}(\mathbf{S}_2) = \frac{26}{12} \doteq 2.17, \text{Tr}(\mathbf{S}_3) = 2 \log(4 \cdot (\frac{12}{19})^2) \doteq 1.35$.

Through this simple example, the computation procedures and the effectiveness of the proposed algorithms are well demonstrated. The following general example further reveals the effectiveness of the proposed algorithms.

Example 2. The well known optimization algorithm, the primal-dual interior point method

(PD-IPM), is chosen for comparison purpose due to its competitiveness in computing the solutions to the convex optimization problems. The proposed minimum transmission completion time problems are non-convex mixed integer optimization problems. As far as the authors' knowledge, there is no algorithm reported in the open literature which can compute the exact solutions to the target problems. As a result, we only focus on the throughput maximization problem.

Figs. 4.2-4.4 are used to show the difference between PD-IPM and RGWFMH for the maximum throughput problems, through some choices of the number of antennas (N_t) at the user or the number of epochs (K). In calculation, the number of antennas at base station (N_r) is set to be 4. Channel gains are generated randomly using random $N_r \times N_t$ matrices with each entry drawn independently from the standard Gaussian distribution. $\{E_{in}(k)\}$ is the set of randomly chosen positive numbers. The sum power constraint of the power grid $E_{(G, \text{total})}$ is 5. A group of different weights are also generated randomly. The chosen parameters mentioned above are assigned to both algorithms with the identical values for comparability. In these figures, the circle markers and the cross markers represent the results of the proposed RGWFMH and PD-IPM respectively. For the proposed RGWFMH, since it uses recursion, no iteration is invoked. Therefore, the number of iterations of the circles maps to one iteration. The obtained throughput is summarized in the following table. The obtained throughput for PD-IPM is the result after 100 iterations.

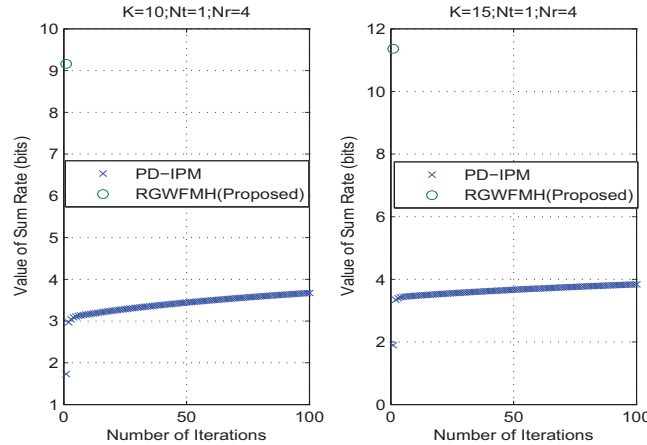


Figure 4.2: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as $K=10$ and 15

With different parameters, the achieved throughput ratio of the PD-IPM to that of RGWFMH is in the range of 0.33 to 0.49. These results show that the proposed RGWFMH exhibits much better performance. It also shows that as the number of the antennas increases, the throughput or

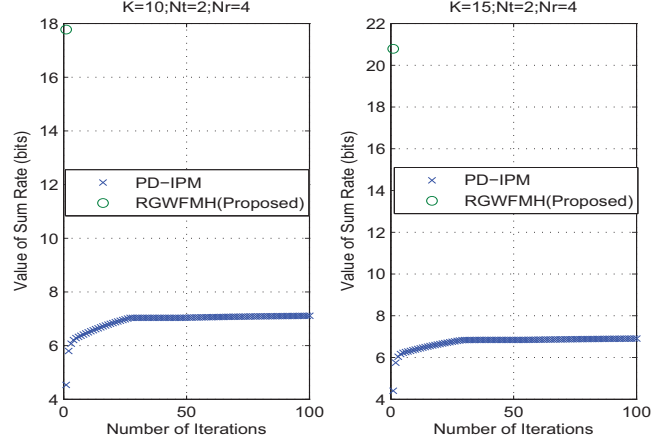


Figure 4.3: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as $K=10$ and 15

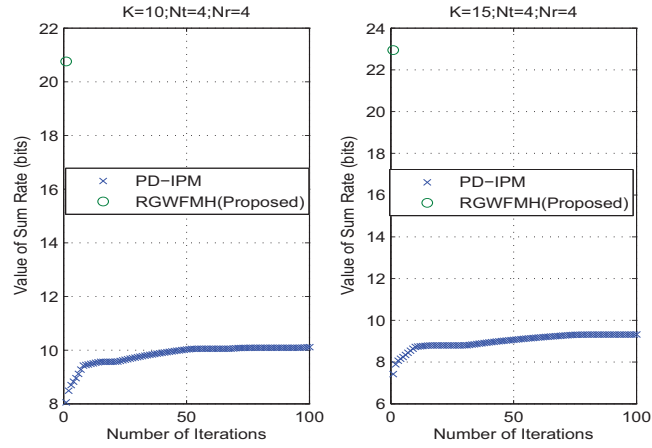


Figure 4.4: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as $K=10$ and 15

Table 4.1: Comparison of the achieved throughput ($N_r = 4$)

parameter	PD-IPM	RGWFMH	ratio
$K = 10, N_t = 1$	3.67	9.15	0.40
$K = 15, N_t = 1$	3.84	11.36	0.34
$K = 10, N_t = 2$	7.11	17.77	0.40
$K = 15, N_t = 2$	6.91	20.79	0.33
$K = 10, N_t = 4$	10.11	20.76	0.49
$K = 15, N_t = 4$	9.32	22.95	0.41

the weighted sum-rate increases.

Example 3. For ease to follow the simulation results, a deterministic example is given. The parameters are chosen as: assume there are five epochs with weight factor vector

$W = \{0.1633, 0.2132, 0.2282, 0.2035, 0.1918\}$, $E_{in}(i) = 6, \forall i$, $E_{(G, \text{total})} = 5$, the channel gain matrices of the two by two antenna array are randomly generated as

$$\begin{aligned}
 H_1^\dagger &= \begin{pmatrix} -0.2056 + 0.1700i & -0.3895 - 0.6354i \\ 0.2236 + 0.2518i & 1.5094 - 1.0604i \end{pmatrix}; \\
 H_2^\dagger &= \begin{pmatrix} 0.3851 - 0.2639i & 1.6777 + 0.3762i \\ -0.1068 - 0.1593i & -0.3660 - 0.9417i \end{pmatrix}; \\
 H_3^\dagger &= \begin{pmatrix} 0.2877 + 0.5690i & 0.5789 + 0.8900i \\ -0.2702 - 0.5321i & -0.2975 - 0.5033i \end{pmatrix}; \\
 H_4^\dagger &= \begin{pmatrix} -0.2851 - 0.5181i & 0.3035 - 0.1812i \\ 0.1038 - 0.4797i & 0.4999 - 0.4366i \end{pmatrix}; \\
 H_5^\dagger &= \begin{pmatrix} -0.7143 - 0.6832i & -0.1870 - 0.7028i \\ 0.2136 - 0.5346i & 0.2199 - 1.1445i \end{pmatrix}.
 \end{aligned} \tag{4.36}$$

The optimal power allocation of the harvested energy is:

$$\{4.6740, 6.1910, 6.4261, 5.1950, 7.5139\},$$

the i th member of which corresponds to the i th epoch. Similarly, the optimal power allocation of the grid power is:

$$\{0.6851, 0.8944, 0.9574, 0.8538, 1.6094\}.$$

The throughput is illustrated in Figure 4.5. It is seen that the achieved throughput of the proposed algorithm is 9.2, but the corresponding value of PD-IPM is 4.1, which reflects an almost 1.2 times gain in the achieved throughput.

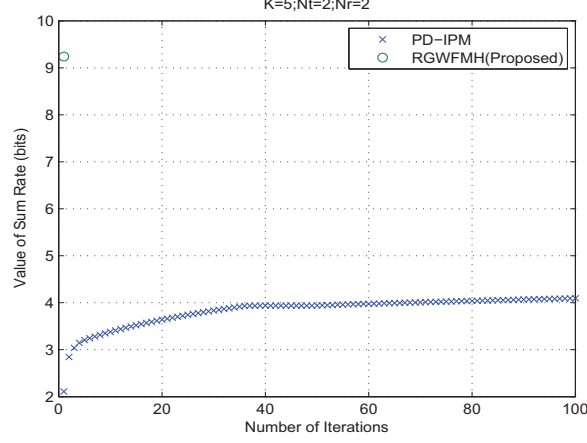


Figure 4.5: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as $K=5$

4.6.2 Computational Complexity Analysis

Since the proposed algorithms use the proposed real throughput mathematical representation by the unitary similarity matrix transformation, the proposed algorithms need the unitary similarity matrix transformation with computational complexity $K \times O(N_t^3)$ (refer to [38]) for (4.5). Second, to compute the optimal solution, RGWFMH, as a more general case than RGWFM, utilizes GWF $\sum_{L=1}^K (1+L)L/2 + 1$ times, so it needs $\sum_{L=1}^K \sum_{k=1}^L (8N_t k + 3) + 8N_t K + 3 = 4N_t K[K^2 + 3K + 8]/3 + 3K(K+1)/2 + 3$, *i.e.*, $N_t O(K^3)$ fundamental operations (refer to [19]). Because a valid algorithm needs to apply the proposed real throughput mathematical representation to avoid the differentiability problem from the several complex optimization variables, therefore, the comparison only focuses on the computational complexity led by the computation of the optimal solution. The complexity of RGWFMH is rather low $N_t O(K^3)$. For example, even let PD-IPM use the proposed transform to obtain an equivalent real problem, however, for the ϵ solution, *i.e.*, not the optimal solution, it still needs a polynomial computational complexity: $N_t^{3.5} O(K^{3.5}) \log(1/\epsilon)$ (refer to [55], [7]). Hence, PD-IPM cannot guarantee to output the optimal solution by finite computation. Our method eliminates any linear search but output the exact optimal solution with finite computation.

Note that the difference between iteration and recursion. The linear search is often computation demanding. This is one weakness of PD-IPM. When the feasible set has the sharp boundary where optimal point(s) is located and the objective function is non-linear, this weakness appears to be more remarkable.

Simply speaking, RGWFMH needs total $K \times O(N_t^3) + N_t O(K^3)$ basic operations to compute the exact solution, while PD-IPM needs total $K \times O(N_t^3) + N_t^{3.5} O(K^{3.5}) \log(1/\epsilon)$ basic operations to compute an ϵ solution.

Chapter 5

Exact Solution of Non-Commutative Composite Water-Fillings for EH and Smart Power Grid Coexisting System with Peak Power Constraints

Energy harvesting makes use of energy from the environment. However, since harvesting energy depends on natural conditions, it is not a stable energy source. As a result, the energy from power grid is often included to serve as a supplementary source to regulate the overall energy supply of the system. Further, the powers from the power grid are often subject to the constraints of peak powers and the energy budget. These constraints render solving the optimal power allocation problems more difficult. In this chapter, we extend our recently proposed geometric water-filling (GWF) and recursive geometric water-filling (RGWF) algorithms to solve the throughput maximization problem and transmission completion time minimization problems, *i.e.*, to make the power grid smart. Importantly, it can compute the exact (optimal) solutions to the problems via finite computation with lower computational complexity, compared with others. This point also lays down a solid fundamental basis for real-time and dynamical resource allocation. Beside more strict discussion of optimality for the proposed algorithms, numerical examples are presented to illustrate the detailed procedures to efficiently obtain the optimal power allocation by the proposed algorithms. The numerical example also shows the composite operation of the two water-fillings is non-commutative.

5.1 Introduction

Often, the powers from the power grid are subject to the constraints of peak powers and the energy budget. These constraints render solving the optimal power allocation problems more difficult. According to our best knowledge, there is no publication to handle them. Without more explanation, in fact, the constraint, which seems to be a trivial case, of an optimization problem just brings a significant watershed from classical to modern optimization. Although the presented throughput maximization problem, as one of three presented problems, is a convex optimization problem, our algorithm shows much more efficient computation than others, including the well-known efficient primal-dual interior point method (PD-IPM) ([7] and references therein). This advantage is especially leaning forwards to the dynamic optimal power resource allocation of wireless communications. For the other two non-convex optimization problems, this advantage is still kept.

In this chapter, GWFPP in [19] to solve the problems with peak power constraints, is utilized. In detail, GWFPP and RGWF [20] are applied to form the two step distribution and the one step adjustment with condition to obtain the optimal solution of the hybrid system for maximizing system throughput. GWFPP and RGWF are all derived from our GWF. Also, it is shown that these two water-filling steps are not commutative. Then the algorithm is developed to solve the transmission completion time minimization problem. Both discrete time case (to find the index n of the epoch to complete the transmission) and the continuous time case (to find time t to complete the transmission) are investigated. By checking if the target B bits transmission is completed, the algorithms are constructed and the optimality proof is provided. Note only for the simpler transmission completion time minimization problem without the consideration of power grid and its peak power constraints, [35] offered the scheme, not a detailed algorithm. Since the proposed optimal power allocation policy mainly results from the recursive computing epoch by epoch, it does not always need the information/solution of the entire process to solve the minimum transmission completion time problem. This is a distinct feature of the proposed algorithms compared to the algorithms reported in the open literature.

Note that, according to the definition of online algorithms ([1, p. 1430]), RGWF in the proposed algorithms possess some characteristics of the online algorithms. This is because the family of RGWF is defined by recursion, without extrapolation; and its input and computation choose the way of piece-by-piece information in a serial fashion. Furthermore, since the powers from the power

grid are, in practice, subject to the peak power constraints, these constraints are considered with a more general form in this chapter.

In the remaining of this chapter, the system model and the problem statement are presented in Section 5.2. Hybrid power allocation algorithm 1 (HPA1) for maximum throughput is investigated in Section 5.3. For minimum discrete or continuous transmission completion time problems, we can choose a similar approach, by the preceding two chapters, to handle them. Thus, solving such a class of the minimum time problems is not presented. Numerical examples and computational complexity discussion are presented in Section 5.4.

5.2 Problem Statement

In this section, the energy harvesting and the smart power grid coexisting system model in a fading channel is presented, followed by the optimization problem to maximize the throughput. For convenience and without loss of generality, the process is assumed to be a discrete time process, like the previous ones. Besides the harvested energy $E_{in}(i)$, the transmission is also connected with the smart power grid. Let $E_{(G,\text{total})}$ denote the energy budget of total energy supported by the power grid. Also, we assume that $E_{G,i} \geq 0$, as the peak power constraint from the power grid for the i th epoch, $\forall i$.

By interpreting the observed properties of the optimal harvested power allocation as a water-filling scheme, $E_{in}(i)$ units of water is filled into a rectangle container with bottom width $\frac{L_i}{2}, \forall i$. The last weighted power sum constraint from energy harvesting forms an equality. Furthermore, for unifying parameter notation, through a change of variables, we can obtain an equivalent target problem as follows:

$$\begin{aligned}
& \max_{\{s_{H,i}, s_{G,i}\}_{i=1}^K} \sum_{i=1}^K w_i \log(1 + a_i(s_{H,i} + s_{G,i})) \\
& \text{subject to:} \quad 0 \leq s_{H,i}, \forall i; \\
& \quad \quad \quad 0 \leq s_{G,i} \leq E_{G,i}, \forall i; \\
& \quad \quad \quad \sum_{i=1}^l s_{H,i} \leq \sum_{i=1}^l E_{in}(i), \forall l; \\
& \quad \quad \quad \sum_{i=1}^K s_{G,i} \leq E_{(G,\text{total})},
\end{aligned} \tag{5.1}$$

where $w_i \leftarrow \frac{L_i}{2}$, $a_i \leftarrow \frac{a_i}{L_i}$, $s_{H,i} \leftarrow L_i s_{H,i}$, $s_{G,i} \leftarrow L_i s_{G,i}$ and $E_{G,i} \leftarrow L_i E_{G,i}$, for any i . Note that the symbol \leftarrow is the assignment operator. Without consideration of trivial cases, $E_{(G,\text{total})} > 0$ can be

assumed. The second constraint remarkably increases difficulty to solve (5.1).

To find the solution to problem (5.1), the conventional water-filling approach starts to obtain the Karush-Kuhn-Tucker (KKT) conditions of problem (5.1) as a set of optimality conditions, and then it solves the conditions to determine the variables $\{s_{H,i}, s_{G,i}\}$ and their duals:

$$\left\{ \begin{array}{l} s_{H,i} + s_{G,i} = \left(\frac{w_i}{\sum_{k=i}^K \lambda_k} - \frac{1}{a_i} \right)^+ = \left(\frac{w_i}{\mu + \nu_i} - \frac{1}{a_i} \right)^+, \\ \text{for } i = 1, \dots, K; \\ 0 \leq s_{H,i}, \quad \forall i; \\ 0 \leq s_{G,i} \leq E_{G,i}, \\ \nu_i(s_{G,i} - E_{G,i}) = 0, \quad \nu_i \geq 0, \quad \forall i; \\ \lambda_l(\sum_{i=1}^l s_{H,i} - \sum_{i=1}^l E_{in}(i)) = 0, \quad \lambda_l \geq 0, \\ \sum_{i=1}^l s_{H,i} \leq \sum_{i=1}^l E_{in}(i), \quad 1 \leq l \leq K; \\ \mu(\sum_{i=1}^K s_{G,i} - E_{(G,\text{total})}) = 0, \quad \mu \geq 0, \\ \sum_{i=1}^K s_{G,i} \leq E_{(G,\text{total})}, \end{array} \right. \quad (5.2)$$

where the function $(x)^+$ means $(x)^+ = x$, for $x \geq 0$, and $(x)^+ = 0$, for $x < 0$. Furthermore, ν_i is the dual variable corresponding to the constraint: $s_{G,i} \leq E_{G,i}$, for any i ; λ_l is the dual variable corresponding to the l th harvested power sum constraint, for any l ; and μ is the dual variable corresponding to the total power sum constraint from the smart power grid. However, by only observing or using the monotonicity information $\frac{1}{\sum_{j=i}^K \lambda_j}$ with respect to i in the first KKT condition related to the sums of pairs $\{s_{H,i}, s_{G,i}\}, \forall i$, it is not sufficient to obtain a solution. The set of $\{\mu, \{\nu_i\}, \{\lambda_l\}\}$ or $\{\mu, \{\nu_i\}, \{\sum_{j=i}^K \lambda_j\}\}$ needs to further satisfy other KKT conditions in order to solve (5.2). The reciprocal of $\sum_{j=i}^K \lambda_j$ is called the water level at epoch i for the entire process from epoch 1 to K . Thus, it is important information that the water level at epoch i depends on the duration of the process (*e.g.*, the water level at epoch i is normally different for processes $[1, K1]$ and $[1, K2]$ where $K1$ and $K2$ are arbitrary ending epoch indexes). However, for system (5.2) in the original variables and the dual variables, there appears to be no existing method available in open literature to obtain an exact solution. The fact that the first equation in (5.2) is in the summation form $s_{H,i} + s_{G,i}$, renders determining an optimal allocation solution from the harvested energy $\{s_{H,i}\}$ and from the smart grid power $\{s_{G,i}\}$ separately, more difficult.

5.3 Hybrid Power Allocation Algorithm 1 (HPA1) for Maximum Throughput

Since the proposed algorithms are based on GWF and GWFPP, they are concisely introduced as follows.

In [19], we presented a geometric water-filling (GWF) approach for solving generalized radio resource allocation problems. As an extension, let L and K be two positive integers and $L \leq K$ denote the index of the starting channel and the ending channel, respectively. Then $K - L + 1$ is the total number of channels. Let P denote the total power for allocation. GWF can be regarded to as a mapping from the point of parameters $\{L, K, \{w_i\}_{i=L}^K, \{a_i\}_{i=L}^K, P\}$ to the solution $\{s_i\}_{i=L}^K$ and the important water level step index: k^* . That is to say, it can be written as a formal expression [19]:

$$\{\{s_i\}_{i=L}^K, k^*\} = \text{GWF}(L, K, \{w_i\}_{i=L}^K, \{a_i\}_{i=L}^K, P). \quad (5.3)$$

Since we often only use the first part, $\{s_i\}_{i=L}^K$ from GWF, we also write

$$\{s_i\}_{i=L}^K = \text{GWF}(L, K, \{w_i\}_{i=L}^K, \{a_i\}_{i=L}^K, P) |_I. \quad (5.4)$$

Note that, for conciseness and without confusion from context, we may write the right hand side of the expression as $\text{GWF}(L, K)$ to emphasize time stages from L to K .

Let \bar{P}_i denote the peak power constraint for the i th channel, GWFPP can be expressed as [19]

$$\begin{aligned} \{s_i\}_{i=1}^K &= \text{GWFPP}(1, K, \{w_i, a_i, \bar{P}_i\}_{i=1}^K, P) |_I, \\ E &= \text{GWFPP}(1, K, \{w_i, a_i, \bar{P}_i\}_{i=1}^K, P) |_E, \end{aligned} \quad (5.5)$$

where E is the final index set in which, there is no peak power constraint correspondingly, referring to [19]. Further, for convenience, E may be written as $\{i_t | 1 < i_1 < \dots < i_{|E|} \leq K\}$ where $|E|$ is the cardinality of the set E . Thus,

$$\{\{s_i\}_{i=1}^K, E\} = \text{GWFPP}(1, K, \{w_i, a_i, \bar{P}_i\}_{i=1}^K, P).$$

Without confusion,

$$\text{GWFPF} (1, K, \{w_i, a_i, \bar{P}_i\}_{i=1}^K, P) |_I$$

can be regarded to as $\text{GWFPF} (1, K, \{w_i, a_i, \bar{P}_i\}_{i=1}^K, P)$, due to a subordinate state of the final index set E .

In this section, we propose a novel algorithm to solve problem (5.1) using GWF approach. The last three constraints in (5.1) can be expanded into a matrix form and two scalar inequality forms, respectively, as follows:

$$\left\{ \begin{array}{l} \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} \begin{pmatrix} s_{H,1} \\ s_{H,2} \\ \vdots \\ s_{H,K} \end{pmatrix} \leq \begin{pmatrix} \sum_{i=1}^1 E_{in}(i) \\ \sum_{i=1}^2 E_{in}(i) \\ \vdots \\ \sum_{i=1}^K E_{in}(i) \end{pmatrix}, \\ 0 \leq s_{G,i} \leq E_{G,i}, \text{ for } i = 1, \dots, K, \\ s_{G,1} + \dots + s_{G,K} \leq E_{(G,\text{total})}, \end{array} \right.$$

where the empty elements inside the matrix denote zeros. The above coefficient matrix form a triangle matrix for the variable $\{s_{H,i}\}_{i=1}^K$. The proposed algorithm to solve (5.1) is referred to as the hybrid power allocation algorithm 1 (HPA1). The proposed HPA1(K) is stated by pseudocode, attached at the end of this chapter. In the remaining of this section, algorithm description and optimality analysis will be presented.

5.3.1 Two parts of HPA1(K) and Their Non-Commutativity

The proposed HPA1(K) consists of two parts: $\text{HPA1}(K)|_I$ for smart grid power allocation as

$$\{s_{G,k}^*\}_{k=1}^K = \text{HPA1}(K)|_I \quad (5.6)$$

and $\text{HPA1}(K)|_{II}$ for harvested energy power allocation portion as

$$\{s_{H,k}^*\}_{k=1}^K = \text{HPA1}(K)|_{II}. \quad (5.7)$$

From the definition of HPA1(K), it is seen that $\text{HPA1}(K)|_I$ is Algorithm GWFPF. GWFPF is used twice in the proposed algorithm: HPA1(K), and then the final index set E also appears

twice. To distinguish between the two E s, the first E is also denoted by $E(1)$ and, i_1 in $E(1)$ by $i_1(1)$. Similarly, we also have $E(2)$ and $i_1(2)$. In addition, $\text{HPA1(K)}|_{II}$ is Algorithm RGWF [20], in essence, with the only difference of the updated “step depths” or the updated channel gains. The implemented order for GWFPP and RGWF is: using GWFPP as $\text{HPA1(K)}|_I$ to compute the initial distribution of the powers from the power grid, then RGWF as $\text{HPA1(K)}|_{II}$ to compute the allocation of the powers from energy harvesting, and finally using GWFPP again, under the condition, to adjust the distributed powers from the power grid and then determine their allocation in the updated $\text{HPA1(K)}|_I$. In this way mentioned above, the entire optimal solution to the proposed problem is obtained. Simply speaking, GWFPP is used twice; and between them, RGWF is used once. The following Lemmas are proposed to study the optimality of HPA1.

Lemma 1. $\text{HPA1(K)}|_I$ can compute the optimal solution of problem (5.1) with finite loops, under $\sum_{i=1}^K E_{in}(i) = 0$.

Since $\text{HPA1(K)}|_I$ is Algorithm GWFPP, which has been detailed by [19], the proof of Lemma 1 can be referred to Proposition 3.1 in [19].

Lemma 2. $\text{HPA1(K)}|_{II}$ can compute the optimal solution of problem (5.1) with finite loops, under $E_{(G,\text{total})} = 0$.

Since $\text{HPA1(K)}|_{II}$ is Algorithm RGWF, which has been detailed by [20], the proof of Lemma 2 can be referred to Proposition 2 in [20].

It is seen that GWFPP and RGWF can be regarded as two functions in

$$\{1, K, \{w_i, a_i, E_{G,i}, E_{in}(i)\}_{i=1}^K, E_{G,\text{total}}\},$$

respectively. If

$$\{1, K, \{w_i, E_{G,i}, E_{in}(i)\}_{i=1}^K, E_{(G,\text{total})}\}$$

are kept unchanged except $\{a_i\}$, GWFPP can be written as $\text{GWFPP}(\{a_i\})$ to emphasize the relationship of the function in $\{a_i\}$, so can RGWF. Since these two functions are the set-valued ones, the k th evaluated value of the first function, GWFPP, is labelled as $[\text{GWFPP}(\{a_i\})]_k$, or simply $[\text{GWFPP}]_k$, so is that of the second function, RGWF. Two composite functions are defined as below.

$$(\text{RGWF} \circ \text{GWFPP}) \triangleq \text{RGWF}\left(\left\{\frac{1}{\frac{1}{a_i} + [\text{GWFPP}]_i}\right\}\right), \quad (5.8)$$

which refers the result by applying GWFPF to allocate the grid power first, followed by applying RGWF to allocate the harvested energy. Thus, using $(\text{RGWF} \circ \text{GWFPF})$ can output the powers, respectively from the energy harvesting and the power grid, earlier outputting the powers from the power grid and later the energy harvesting.

The second composite function, HPAF, is defined as:

$$\text{HPAF} = (\text{GWFPF} \circ \text{RGWF}) \triangleq \text{GWFPF}(\{\frac{1}{\frac{1}{a_i} + [\text{RGWF}]_i}\}). \quad (5.9)$$

Thus, $\text{HPAF}|_I = \text{RGWF}$ outputs the powers from the energy harvesting. $\text{HPAF}|_{II} = (\text{GWFPF} \circ \text{RGWF})$ outputs the powers from the power grid. Thus, using HPAF can also output the powers, respectively from the energy harvesting and the power grid. However, the composite operation mentioned above does not satisfy the commutative law, *i.e.*, $(\text{RGWF} \circ \text{GWFPF}) \neq \text{HPAF}$. This point is accounted for by the following example.

Example 5.3.1.

$$\begin{aligned} & \max_{\{(s_{H,i}, s_{G,i})\}_{i=1}^2} \quad \sum_{i=1}^2 \log(1 + (s_{H,i} + s_{G,i})) \\ & \text{subject to:} \quad s_{H,i} \geq 0, i = 1, 2; \\ & \quad \quad \quad 0 \leq s_{G,1} \leq 2; \\ & \quad \quad \quad 0 \leq s_{G,2} \leq 0.5; \\ & \quad \quad \quad \sum_{i=1}^l s_{H,i} \leq l, \quad l = 1, 2; \\ & \quad \quad \quad \sum_{i=1}^2 s_{G,i} \leq 2.2. \end{aligned} \quad (5.10)$$

According to the definitions of $(\text{RGWF} \circ \text{GWFPF})$ and HPAF, the output of $(\text{RGWF} \circ \text{GWFPF})$ is $\{s_{H,1} = 0.4, s_{H,2} = 1.6; s_{G,1} = 1.7, s_{G,2} = 0.5\}$, at which the objective function value is $\log 9.61$; while that of HPAF is $\{s_{H,1} = 1, s_{H,2} = 1; s_{G,1} = 1.7, s_{G,2} = 0.5\}$, at which the objective function value is $\log 9.25$. Thus, the performance of $(\text{RGWF} \circ \text{GWFPF})$ is truly better than that of HPAF due to $\log 9.61 > \log 9.25$. Then, the commutative law does not hold for this example. From this example, it is seen that the target problem cannot be decomposed into two decoupled sub-problems in the two classes of powers, respectively. Furthermore, HPAF, as the two water-fillings, RGWF and GWFPF successively being applied, cannot guarantee to find the optimal solution.

In HPA1, in Line 3, GWFPF is applied to update the new step heights, which are formed by the original fading gains and the power levels allocated in $\text{HPA1(K)}|_I$. From Line 4 to Line 22,

HPA1(K) computes the optimal solution for the harvested energy part, *i.e.*, $\text{HPA1(K)}|_{II}$ to complete the computation.

Note that based on GWFPP, the term $(1/a_i + s_{G,i})/w_i$ denotes the overall step depth after the power allocation of the power grid. Therefore, the reciprocal of $(1/a_i + s_{G,i})$ is $a_i/(1 + a_i S_{G,i})$, which is equivalent to the channel gain used by RGWF in [19].

The proposed algorithm eliminates the procedures to solve the non-linear system (5.2) in multiple variables and dual variables, provides exact solutions via finite computation steps, and offers helpful insights to the problems and the solutions.

5.3.2 Optimality of HPA1

This subsection discusses optimality of the proposed HPA1.

Remark 1. $\text{HPA1}|_{II}$ is an optimal dynamic power distribution process. The dynamics of this recursive process are shown by the generalized state equation:

$$\text{HPA1}(L+1)|_{II} = [\text{HPA1}(L)|_{II}, \text{GWF}(n+1, L+1)|_I], \text{ for } L = 1, \dots, K-1, \quad (5.11)$$

where n is the index of the starting epoch of the currently processing window. Note that the concept of dynamic processes is not identical to that of dynamic programming. The value of n is determined by $\text{HPA1}(L)|_{II}$. In this process, $\text{HPA1}(L)|_{II}$ can be regarded as the generalized system state at the time stage (or epoch) L ; $\text{GWF}(n+1, L+1)$ can be regarded as the generalized system control at the time stage (or epoch) L ; and then $\text{HPA1}(L+1)|_{II}$, as a state at the next time stage, can be derived or determined from its previous state and control. Due to optimality of $\text{HPA1}(L)|_{II}$ from Lemma 2, for any L , the proposed algorithm is indeed an optimal and efficient forwarding dynamic recursive water-filling algorithm.

Then, since Lemma 2 guarantees optimality of $\text{HPA1}(K)|_{II}$ under the special condition, and so does Lemma 1, we may obtain the following conclusion of $\text{HPA1}(K)$.

Proposition 5.1. HPA1 can compute the optimal exact solution to problem (5.1) within finite loops.

Proof of Proposition 5.1. First, $\text{HPA1}(K)|_I$ is implemented. Thus, it is equivalent to Lemma 1 being used. According to Lemma 1, for problem (5.1) under $\sum_{i=1}^K E_{in}(i) = 0$, there exist the

optimal solution $\{s_{G,i}\}_{i=1}^K$ and the dual variables $\{\lambda^{(1)}, \{\nu_i^{(1)}, \mu_i^{(1)}\}_{i=1}^K\}$ such that they satisfy the following KKT conditions:

$$\begin{cases} \frac{1}{\frac{1}{a_i w_i} + \frac{s_{G,i}}{w_i}} = \lambda^{(1)} + \nu_i^{(1)} - \mu_i^{(1)}, & \forall i; \\ \mu_i^{(1)} s_{G,i} = 0, s_{G,i} \geq 0, \mu_i^{(1)} \geq 0, & \forall i; \\ \nu_i^{(1)} (s_{G,i} - E_{G,i}) = 0, s_{G,i} \leq E_{G,i}, \nu_i^{(1)} \geq 0, & \forall i; \\ \lambda^{(1)} \left(\sum_{i=1}^K s_{G,i} - E_{(G,\text{total})} \right) = 0, \\ \sum_{i=1}^K s_{G,i} \leq E_{(G,\text{total})}, \lambda^{(1)} \geq 0. \end{cases}$$

Second, $\text{HPA1}(K)|_{II}$ is implemented. Thus, it is equivalent to Lemma 2 being used, however, with the updated “step depths” $\frac{1}{a_i w_i} + \frac{s_{G,i}}{w_i}$, or the updated channel gains $\{\frac{a_i}{1+a_i s_{G,i}}\}$.

According to Lemma 2 with the updated channel gains, for problem (5.1) under $E_{(G,\text{total})} = 0$, there exist the optimal solution $\{s_{H,i}\}_{i=1}^K$ and the dual variables $\{\lambda_i^{(2)}, \mu_i^{(2)}\}_{i=1}^K$ such that they satisfy the following KKT conditions:

$$\begin{cases} \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} = \sum_{k=i}^K \lambda_k^{(2)} - \mu_i^{(2)}, & \forall i; \\ \mu_i^{(2)} s_{H,i} = 0, s_{H,i} \geq 0, \mu_i^{(2)} \geq 0, & \forall i; \\ \lambda_i^{(2)} (\sum_{k=1}^i s_{G,k} - \sum_{k=1}^i E_{in}(k)) = 0, \\ \sum_{k=1}^i s_{G,k} \leq \sum_{k=1}^i E_{in}(k), \lambda_i^{(2)} \geq 0, & \forall i. \end{cases}$$

On the one hand, if $E = \emptyset$ (the empty set), let $\lambda_G = 0$; $\nu_{G,i} = \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} \geq 0$ and $\mu_{G,i} = 0$, as $1 \leq i \leq K$. Also, let $\lambda_{H,i} = \lambda_i^{(2)}$, and $\mu_{H,i} = \mu_i^{(2)}$, for any i . Note that this set E mentioned above is obtained, when GWFPP is used at the first time. On the other hand, if $E \neq \emptyset$, and $\exists i_1 \in E$ such that $\frac{1}{a_{i_1} w_{i_1}} + \frac{s_{H,i_1} + s_{G,i_1}}{w_{i_1}} \geq \frac{1}{a_K w_K} + \frac{s_{H,K} + s_{G,K}}{w_K}$, let $\lambda_G = \frac{1}{\frac{1}{a_{i_1} w_{i_1}} + \frac{s_{H,i_1} + s_{G,i_1}}{w_{i_1}}}$; $\nu_{G,i} = \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} - \lambda_G \geq 0$; and $\mu_{G,i} = 0$, as $s_{G,i} > 0$ or the case of both $s_{G,i} = 0$ and $E_{G,i} = 0$, under $1 \leq i \leq K$. Note that this set E just mentioned above is obtained, when GWFPP is still used at the first time. Further, if $E \neq \emptyset$, then $i_1 \in E$ such that $\frac{1}{a_{i_1} w_{i_1}} + \frac{s_{H,i_1} + s_{G,i_1}}{w_{i_1}} \geq \frac{1}{a_k w_k} + \frac{s_{H,k} + s_{G,k}}{w_k}$, where $s_{G,k} > 0, 1 \leq k \leq K$. Note that this set E just mentioned above is obtained, when GWFPP is used at the second time. It has been emphasized that to distinguish between the two E s, the first E is also denoted by $E(1)$ and, i_1 in $E(1)$ by $i_1(1)$. Similarly, we also have $E(2)$ and $i_1(2)$. Let $\lambda_G = \frac{1}{\frac{1}{a_{i_1} w_{i_1}} + \frac{s_{H,i_1} + s_{G,i_1}}{w_{i_1}}}$; $\nu_{G,i} = \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} - \lambda_G \geq 0$; and $\mu_{G,i} = 0$, as $s_{G,i} > 0$ or the case of both $s_{G,i} = 0$ and $E_{G,i} = 0$,

under $1 \leq i \leq K$. Further, if $s_{G,i} = 0$ and $E_{G,i} > 0$, let $\nu_{G,i} = 0$; $\mu_{G,i} = \lambda_G - \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} \geq 0$ with λ_G defined above. Also, it has been noted that the water level for each of the epochs in the set $\{i | 1 \leq i < i_1(1)\}$ keeps unchanged, although the adjustment is done or GWFPP is used twice; difference between the water levels of epochs in the set $\{i | i_1(1) \leq i < i_1(2)\}$ is decreased; and difference between the water levels of epochs in the set $\{i | i_1(2) \leq i \leq K\}$ is leaning forward to the same. Thus, $\{\lambda_{H,i}, \mu_{H,i}\}_{i=1}^K$ can be easily constructed, similar to those in [20] together with the just mentioned distinguishing characteristics. That is to say, according to the two sets of KKT conditions, the definitions, or assigned and constructed values of the dual variables: $\{\{\lambda_{H,i}, \mu_{H,i}, \nu_{G,i}, \mu_{G,i}\}, \lambda_G\}$, and the solutions $\{s_{H,i}, s_{G,i}\}_{i=1}^K$ mentioned above, it is seen that these dual variables and solutions also satisfy the following KKT conditions:

$$\left\{ \begin{array}{l} \frac{1}{\frac{1}{a_i w_i} + \frac{s_{H,i} + s_{G,i}}{w_i}} = \lambda_G + \nu_{G,i} - \mu_{G,i} = \sum_{k=i}^K \lambda_{H,k} - \mu_{H,i}, \\ \forall i; \\ \mu_{H,i} s_{H,i} = 0, \quad 0 \leq s_{H,i}, \mu_{H,i} \geq 0, \quad \forall i; \\ \mu_{G,i} s_{G,i} = 0, \quad 0 \leq s_{G,i}, \mu_{G,i} \geq 0, \quad \forall i; \\ \nu_{G,i} (s_{G,i} - E_{G,i}) = 0, \quad s_{G,i} \leq E_{G,i}, \nu_{G,i} \geq 0, \quad \forall i; \\ \lambda_{H,l} (\sum_{i=1}^l s_{H,i} - \sum_{i=1}^l E_{in}(i)) = 0, \quad \lambda_{H,l} \geq 0, \\ \sum_{i=1}^l s_{H,i} \leq \sum_{i=1}^l E_{in}(i), \quad 1 \leq l \leq K; \\ \lambda_G (\sum_{i=1}^K s_{G,i} - E_{(G,\text{total})}) = 0, \quad \lambda_G \geq 0, \\ \sum_{i=1}^K s_{G,i} \leq E_{(G,\text{total})}, \end{array} \right.$$

where this set of KKT conditions is of problem (5.1), the Lagrange function of which is:

$$\begin{aligned} & L(\{s_{H,i}, s_{G,i}\}; \{\lambda_{H,i}, \mu_{H,i}\}, \{\lambda_G, \{\nu_{G,i}, \mu_{G,i}\}\}) \\ &= \sum_{i=1}^K w_i \log(1 + a_i(s_{H,i} + s_{G,i})) + \sum_{i=1}^K \mu_{H,i} s_{H,i} \\ &- \sum_{l=1}^K \lambda_{H,l} \left(\sum_{k=1}^l s_{H,k} - \sum_{k=1}^l E_{in}(k) \right) \\ &+ \sum_{i=1}^K \mu_{G,i} s_{G,i} - \sum_{i=1}^K \nu_{G,i} (s_{G,i} - E_{G,i}) \\ &- \lambda_G \left(\sum_{i=1}^K s_{G,i} - E_{(G,\text{total})} \right). \end{aligned}$$

In addition, we can observe that the general constraint qualification of problem (5.1) holds. Then $\{s_{H,i}, s_{G,i}\}_{i=1}^K$ computed by the proposed HPA1 is the optimal solution to problem (5.1).

Therefore, Proposition 5.1 is proved.

Therefore, we settled the points: (1) HPA1 can compute the optimal solution only from the causal information in finite steps. It does not need to solve any non-linear system, consisting of many equations and inequalities in multiple dual variables; (2) the relationship between $\text{HPA1}|_I$ and $\text{HPA1}|_{II}$ is determined; optimality of HPA1 stemming from $\text{HPA1}|_I$ and $\text{HPA1}|_{II}$ is revealed.

5.4 Numerical Example and Computational Complexity

HPA1 does not need to wait for the full information to be available but it can compute the exact optimal solution through finite computation for every sub-process that starts from epoch 1 and ends at epoch i , as $i = 1, \dots, K$. This point can also lean toward designing other efficient algorithms, such as algorithms to compute the minimum transmission completion time.

5.4.1 Numerical Example

Example 1. PD-IPM is chosen for the purpose of comparison due to its competitiveness in computing the solutions to the convex optimization problems. It has been known that the proposed minimum transmission completion time problems are non-convex and mixing continuous with integer variable optimization problems. As far as the authors' knowledge, there is no algorithm reported in the open literature which can compute the exact solutions to these problems. As a result, we only focus on the throughput maximization problem.

Figs. 5.1-5.3 are used to show the difference between PD-IPM and HPA1 for the maximum throughput problems, through some choices of the number of epochs ($K = 10, 15, 20$). Channel gains are generated randomly using random variables with the standard Gaussian distribution. For convenience, $\{E_{in}(k) = 6, \forall k\}$. The sum power constraint of the power grid $E_{(G, \text{total})} = K$, and the peak power constraints $\{E_{(G, k)} = k, \forall k\}$. A group of different weights are also generated randomly. The chosen parameters mentioned above are assigned to both algorithms with the identical values for comparability. In these figures, the circle markers and the cross markers represent the results of the proposed HPA1 and PD-IPM respectively. For the proposed HPA1, since it uses recursion, no iteration is invoked. Therefore, the number of iterations of the circles maps to one iteration. The obtained throughput is summarized in the following table. The obtained throughput for PD-IPM

is the result after 100 iterations.

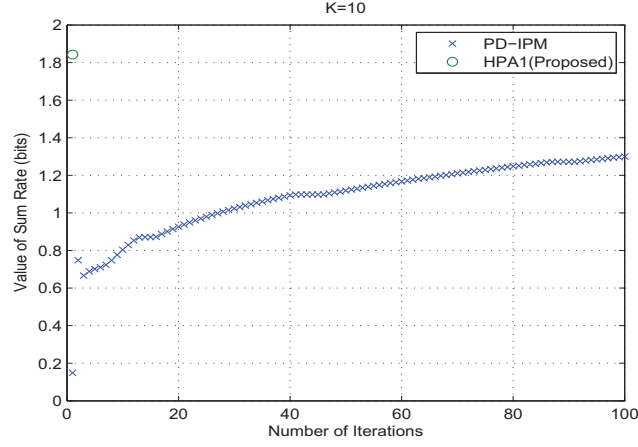


Figure 5.1: Weighted sum-rates (Unit: bits) of HPA1 and PD-IPM, as K=10

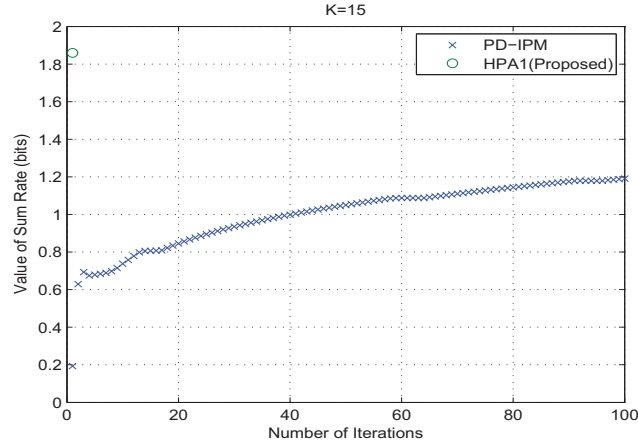


Figure 5.2: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as 15

With different parameters, the achieved throughput ratio of the PD-IPM to that of HPA1 is in the range of 0.55 to 0.70. These results show that the proposed HPA1 exhibits much better performance. It also shows that as the number of the users increases, the throughput or the weighted sum-rate increases.

As a supplement, for ease to follow the simulation results, a deterministic example is given. The parameters are chosen as: assume there are five epochs with weight factor vector $W = (0.1633, 0.2132, 0.2282, 0.2035, 0.1918)$. $E_{in}(k)$, $\{E_{(G,k)}, \forall k\}$, and $E_{(G,total)}$ are assigned by the same

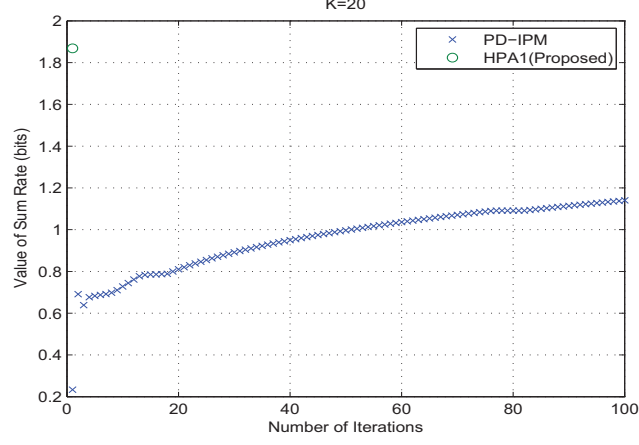


Figure 5.3: Weighted sum-rates (Unit: bits) of RGWFMH and PD-IPM, as K=20

Table 5.1: Comparison of the achieved throughput

parameter	PD-IPM	RGWFMH	ratio
$K = 10$	1.30	1.84	0.70
$K = 15$	1.19	1.86	0.63
$K = 20$	1.14	1.87	0.61
$K = 25$	1.09	1.87	0.58
$K = 30$	1.07	1.88	0.57
$K = 35$	1.04	1.88	0.55

values as the ones mentioned above. Further the channel gains are generated as

$$\begin{aligned}
 & (a_1, a_2, a_3, a_4, a_5) \\
 & = (0.2056^2, 0.3851^2, 0.2877^2, 0.2851^2, 0.7143^2).
 \end{aligned} \tag{5.12}$$

The optimal power allocation of the harvested energy is:

$$\{0, 10.1359, 6.4885, 4.2572, 9.1185\},$$

the i th member of which corresponds to the i th epoch. Similarly, the optimal power allocation of the grid power is:

$$\{0, 0.4705, 0, 0, 4.5295\}.$$

5.4.2 Computational Complexity Analysis

To compute the optimal solution, HPA1, utilizes GWF $\sum_{L=1}^K (1+L)L/2$ times for RGWF(K), so it needs $\sum_{L=1}^K \sum_{k=1}^L (8k+3) = K(K+1)(8K+25)/6$, *i.e.*, $O(K^3)$ fundamental operations for utilizing GWF (refer to [19]). Due to HPA1 also using GWFPP two time, this usage needs $8K^2 + 14K$ fundamental operations (refer to [19]). Therefore, HPA1 needs $K(K+1)(8K+25)/6 + 8K^2 + 14K$, *i.e.*, $O(K^3)$ fundamental operations. Therefore, the complexity of HPA1 is rather low $O(K^3)$. As a comparison, when PD-IPM being used can only obtain the ϵ solution, which is not the optimal solution, it still needs a polynomial computational complexity: $O(K^{3.5}) \log(1/\epsilon)$ (refer to [55, 7]). Hence, PD-IPM cannot guarantee to output the optimal solution by finite computation. Our method eliminates any linear search but output the exact optimal solution with finite computation.

One weakness of PD-IPM is the linear search being used. When the feasible set has the sharp boundary where optimal point(s) is located and the objective function is non-linear, this weakness appears to be more remarkable. In addition, since PD-IPM requires efficiently compute the solution, assuming the LU decomposition to be used, to the linear system which is determined by the fact that the barrier function gradient is the difference between the objective function gradient and the linear combination of the constraint function gradients, and the perturbed complementarity condition, applying Newton's method, it needs approximately $\frac{2}{3}(4K+1)^3$ fundamental operations. Together with evaluation of the coefficient matrix of the linear system and the computation burden of the linear search, PD-IPM requires an equivalent computational complexity, for *each* of iterations, to that used *totally* for HPA1.

Simply speaking, HPA1 needs total $O(K^3)$ basic operations to compute the exact solution, while PD-IPM needs total $O(K^{3.5}) \log(1/\epsilon)$ basic operations to compute an ϵ solution.

Algorithm 5 Pseudocode for HPA1

1: Initialize:

$$L = 1, K, E_{(G, \text{total})}, \{w_k, a_k, \bar{P}_k = E_{G,k}, E_{in}(k)\}_{k=1}^K;$$

2: Prepare:

$$\text{HPA1(K)}|_I = \{\{s_{G,k}^*\}_{k=1}^K, E\} = \text{GWFPP}(\{a_i\}_{i=1}^K);$$

3: Update:

$$\{a_k \leftarrow \frac{1}{\frac{1}{a_k} + s_{G,k}^*}\}_{k=1}^K;$$

4: Output the result for the epoch 1:

$$\text{RGWF}(L) = E_{in}(1);$$

5: **for** $L = 2 : 1 : K$ **do**

6: Input: $\{E_{in}(L), w_L, a_L\}$;

7: $\{s'_{H,k}\}_{k=1}^{L-1} = \text{RGWF}(L-1)$;

8: **for** $n = L : -1 : 1$ **do**

9: $W = \{w_j\}_{j=n}^L; A = \{a_j\}_{j=n}^L$;

10: $S_T = \sum_{j=n}^{L-1} s'_{H,j} + E_{in}(L)$;

11: $\{\{s_{H,k^*}\}_{k=n}^L, k^*\} = \text{GWF}(n, L, W, A, S_T)$;

12: $k_e^* = \max\{k | s'_{H,k} > 0, 1 \leq k \leq n-1\}$;

13: **if** $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{H,k^*}}{w_{k^*}} \geq \frac{1}{a_{k_e^*} w_{k_e^*}} + \frac{s_{H,k_e^*}}{w_{k_e^*}}$ **then**

14: output:

$$\text{RGWF}(L) = \{s'_{H,1}, \dots, s'_{H,n-1}, s_{H,n}^*, \dots, s_{H,L}^*\},$$

15: Move to the next epoch, *i.e.*, go to Line 18;

16: **end if**

17: **end for**

18: **end for**

19: **if** $E = \emptyset$ or $\frac{1}{a_{i_1} w_{i_1}} + \frac{s_{H,i_1} + s_{G,i_1}^*}{w_{i_1}} \geq \frac{1}{a_K w_K} + \frac{s_{H,K} + s_{G,K}^*}{w_K}$, where $s_{H,i}$ is the the i th member of $\text{RGWF}(K)$ **then**

20: output of HPA1(K) :

$$\{s_{G,k}^*\}_{k=1}^K = \text{HPA1(K)}|_I = \text{GWFPP from Line 2};$$

$$\{s_{H,k}^*\}_{k=1}^K = \text{HPA1(K)}|_{II} = \text{RGWF}(K);$$

21: **end if**

22: $\text{HPA1(K)}|_I = \{\{s_{G,k}^*\}_{k=1}^{i_1-1}, \text{GWFPP}(\{\frac{1}{\frac{1}{a_k} w_k} + \frac{s_{H,k}}{w_k}\}_{k=i_1}^K)\}$;

$$\text{HPA1(K)}|_{II} = \{\{s_{H,k}^*\}_{k=1}^{i_1-1}, \text{RGWF}(\{\frac{1}{\frac{1}{a_k} w_k} + \frac{(\text{HPA1(K)}|_I)_k}{w_k}\}_{k=i_1}^K)\}.$$

Chapter 6

Using GWF to Solve A Class of Sum Power Minimization Problems

In this chapter, we first propose a generalized water-filling approach to solve the power allocation problem of minimizing sum power while meeting the target sum rate constraint with weights. Based on this sum power objective function, we extend the proposed method to more complicated RRM problems with more stringent constraints. The proposed algorithms with this generalized approach possess several distinguished features. They provide exact optimal solutions based on non-derivative methods, as the implementation of the proposed algorithms invokes neither the derivative nor the gradient. With geometric interpretation, the proposed algorithms provide more insights to and intuitions of the problems and could be used to efficiently solve a family of the sum power minimization problems. Optimality of the proposed algorithms is strictly proved. Numerical results that illustrate the steps and demonstrate efficiency of the proposed algorithms are presented.

6.1 Introduction

Transmit power and data rates are two fundamental radio resources. The required Quality of Service (QoS) can be satisfied by adjusting either power or rate or both in a complementary way. The optimal allocation of these scarce radio resources for different users/channels directly affects system performance. Therefore, for radio resource management (RRM) with different target and/or priorities, the problems or mathematical models may be formulated through maximizing the objective function of transmission throughput (or sum rate), subject to the sum power constraint; or

through minimizing the objective function of sum power, subject to the throughput constraint. In this chapter, we simply refer to the former problem as the original RRM problem, and the latter one as the (sum) power problem to distinguish itself from the original RRM problem.

6.1.1 Our Work for Sum Power Minimization Problem

In this chapter, a set of new problems are formed by changing the throughput of objective function, in the original RRM problem, into the sum power; while by changing the sum power constraint, in the original RRM problem, into the throughput constraint. At the same time, we will use the operator of minimization for this set of new problems. Thus, this set of new problems are readily distinguishable from the original RRM problems mentioned above. Correspondingly, this set of problems, with the objective functions having (sum) power forms, are referred to as P-GWF, P-WFPP, P-WFGPP, and P-WFGBP respectively here. These problems have their practical applications. For example, when we consider different class of QoS services, the constraint of the different lower and upper bounds of the power allocation is reflected in our target problems. Without loss of clarity, we will use the same set of abbreviations to denote the problems and the algorithms that solve the corresponding problems.

In this chapter, we significantly extend beyond earlier proposed approach [19] to solve the sum power RRM problems, including the basic form (P-GWF) and its extended and generalized forms. The stated generalized approach has a distinguished feature that the proposed algorithms start from geometric interpretations of the target problems. These geometric interpretations and the relationships they formed provide more insights into the problems; and such insights assist us to efficiently solve the target problems with optimal solutions. The proposed algorithms possess simple procedures due to the fact that the proposed algorithms belong to the non-derivative methods (which have been defined in [2]) that use neither the derivative nor the gradient during their implementation. The proposed algorithm P-GWF for the basic sum power problem has two advantages: it provides the exact solution, and thus eliminates the iterative steps of finding the water level through solving the non-linear system. On the other hand, the machinery of the proposed approach enabled us to solve the more generalized RRM problems with more stringent constraints. In our numerical examples, it is shown that with optimal power allocation for the generalized RRM problems, the water levels are different for the different constraints of lower and upper bounds.

The conventional approach of determining the water level(s) might not be able to solve this kind of generalized problems. The difference between our approach and those of others is summarized below.

First, for the simple case of P-GWF problem that can be solved through the conventional approach, it is generalized into a weighted case in this chapter. Together, the corresponding algorithm with less computation is also proposed.

Second, for the more complicated P-WFPP, P-WFGPP, and P-WFGBP cases, their solutions cannot be computed exactly by the conventional approach, but these (optimal) solutions can be computed exactly by our approach in this chapter.

Third, for each of these problems, our approach only takes a low degree polynomial computational complexity for the exact solution, unlike the popular primal-dual interior point method (PD-IPM) that only computes an ϵ solution, which is not an optimal solution and requires more computations (refer to [55, 7], and references therein).

Fourth, for the mentioned problems of P-WFPP, P-WFGPP, and P-WFGBP, similar results have not been reported in the open literature, to the best of the authors' knowledge. For example, [19, 36] provided efficient algorithms for some RRM problems that have different structures, unlike the structure of these target problems in this chapter. The approaches discussed in [19, 36] cannot solve the target problems in this chapter. The proposed algorithms are novel and efficient.

In the remaining of the chapter, the problem statement, the conventional approach, and the preparation or the illustration for the proposed P-GWF are discussed in Section 6.2. The extended and generalized sum power RRM problems with additional stringent constraints are further investigated in Section 6.3. Numerical examples and complexity analysis are presented in Section 6.4. We provide where the strict optimality proofs appear for the extended and generalized algorithms to compute the minimum sum power problems with more constraints, to meet some readers' requirement.

6.2 Generalized Water-Filling for Sum Power Problem

6.2.1 Problem Statement and Conventional Approach

The *original basic RRM problem* can be described by the following: given $P > 0$, which is the total power or volume of the water; the allocated power and the propagation path gain of the i th channel, which are denoted by s_i and a_i respectively, $i = 1 \dots K$; and K which is the total number of channels, letting $\{a_i\}_{i=1}^K$ be a sorted sequence with monotonically decreasing (the indexes can be arbitrarily renumbered to satisfy this condition), in which $a_i > 0, \forall i$, find a group of the powers $\{s_i\}$ to satisfy:

$$\begin{aligned} \max_{\{s_i\}_{i=1}^K} \quad & \frac{1}{2} \sum_{i=1}^K \log_2(1 + a_i s_i) \\ \text{subject to:} \quad & \sum_{i=1}^K s_i = P; 0 \leq s_i, \forall i. \end{aligned} \quad (6.1)$$

Extensive investigation to solve problem (6.1) has been reported in the open literature. Using a geometrical approach to solve this problem has been discussed in our earlier chapter [19].

The *basic sum power RRM problem* can be stated as: given $B > 0$, which denotes the number of the target transmission bits (or sum rate of the system), find a group of the powers $\{s_i\}$ to satisfy:

$$\begin{aligned} \min_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K s_i \\ \text{subject to:} \quad & \sum_{i=1}^K \frac{1}{2} \log_2(1 + a_i s_i) = B; 0 \leq s_i, \forall i. \end{aligned} \quad (6.2)$$

Note that if only the first constraint is substituted with $\frac{1}{2} \sum_{i=1}^K \log_2(1 + a_i s_i) \geq B$, the new problem, as a convex optimization problem, is equivalent to (6.2). Solving problem (6.2) is important, especially when saving energy/power is indeed the first priority of the system design.

Problem (6.1) is to solve the throughput maximization problem; while problem (6.2) is to solve the sum power minimization problem. Generally, the solution to (6.1) can not be directly applied to (6.2). This chapter focuses on the investigation of the solution to (6.2) and its extended and more generalized forms.

To find the solution to problem (6.2), conventional approach usually starts from the equivalent form of the Karush-Kuhn-Tucker (KKT) conditions of problem (6.2). This equivalent form is:

$$\begin{cases} s_i = \left(\mu - \frac{1}{a_i} \right)^+, \text{ for } i = 1, \dots, K; \\ \sum_{i \in \{l | \mu - \frac{1}{a_l} \geq 0, 1 \leq l \leq K\}} \log_2(1 + a_i s_i) = 2B; \mu \geq 0, \end{cases} \quad (6.3)$$

where $(x)^+ = \max\{0, x\}$. μ is the water level chosen to satisfy the sum rate constraint with equality $(\frac{1}{2} \sum_{i=1}^K \log_2(1 + a_i s_i) = B)$.

Enumeration can be utilized to find the water level μ in (6.3). This statement means that solving the equation in μ :

$$\sum_{i \in \{l | \mu - \frac{1}{a_l} \geq 0, 1 \leq l \leq K\}} \log_2 \left(1 + a_i \left(\mu - \frac{1}{a_i} \right) \right) = 2B, \quad (6.4)$$

can find the water level μ in (6.3), where the index k runs up from 1 to K . Further, after the index k only runs some steps that are not greater than K , due to $\{\frac{1}{a_l}\}$ keeping monotonicity, the solution to (6.4) can be obtained. That is to say, we may use fewer steps to find μ . This algorithm or solution to (6.3) is referred to as the conventional sum power water-filling algorithm or solution, denoted by P-CWF. The detail can be furthermore referred to in [8]. Since P-CWF results from the motivation to solve the system directly, it is a non-geometric approach.

6.2.2 Illustration of the Proposed Generalized Water-Filling Algorithm (P-GWF)

In this section, we apply our proposed generalized water-filling methodology [19] to solve problem (6.2). Similar to GWF [19], Figs. 6.1(a)-(c) illustrate the proposed P-GWF algorithm for the sum power problems. Suppose there are 4 steps/stairs ($K = 4$) with unit width inside a water tank. In the conventional approach, the dashed horizontal line, which is the water level μ , needs to be determined first and then the powers (water volume above the step) are solved.

Let us use d_i to denote the “step depth” of the i th stair which is the height of the i th step to the bottom of the tank, and is given below:

$$d_i = \frac{1}{a_i}, \text{ for } i = 1, 2, \dots, K. \quad (6.5)$$

Since the sequence $\{a_i\}$ is sorted with monotonically decreasing, the step depth of the stairs indexed by $\{1, \dots, K\}$ is monotonically increasing.

Instead of trying to determine the water level μ , which is a real nonnegative number, we aim to determine the highest (shallowest) water level step under water, which is an integer number between 1 and K , and denoted it by k^* . Based on the result of k^* , we can write out the solutions for power

allocation in problem (6.2) instantly.

Fig. 6.1(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^* .

We define the achieved data rate using power below step k by $R(k)$ that can be expressed by

$$\begin{aligned}
 R(k) &= \frac{1}{2} \sum_{i=1}^{k-1} \log_2 (1 + a_i \bar{s}(k)_i) \\
 &= \frac{1}{2} \sum_{i=1}^{k-1} \log_2 \left[1 + a_i \left(\frac{1}{a_k} - \frac{1}{a_i} \right) \right] \\
 &= \frac{1}{2} \sum_{i=1}^{k-1} \log_2 \left[\frac{a_i}{a_k} \right] \\
 &= \frac{1}{2} \log_2 \left[\prod_{i=1}^{k-1} \left(\frac{a_i}{a_k} \right) \right],
 \end{aligned} \tag{6.6}$$

where $\bar{s}_i(k) = \frac{1}{a_k} - \frac{1}{a_i}$ in $R(k)$. To include the case of $k = 1$ in (6.6), we define the two special cases as follows: $\sum_{i=m}^n b_i = 0$, for $m > n$, and

$$\prod_{i=m}^n b_i = 1, \text{ for } m > n, \tag{6.7}$$

where $\{b_i\}$ is assumed to be a general number sequence.

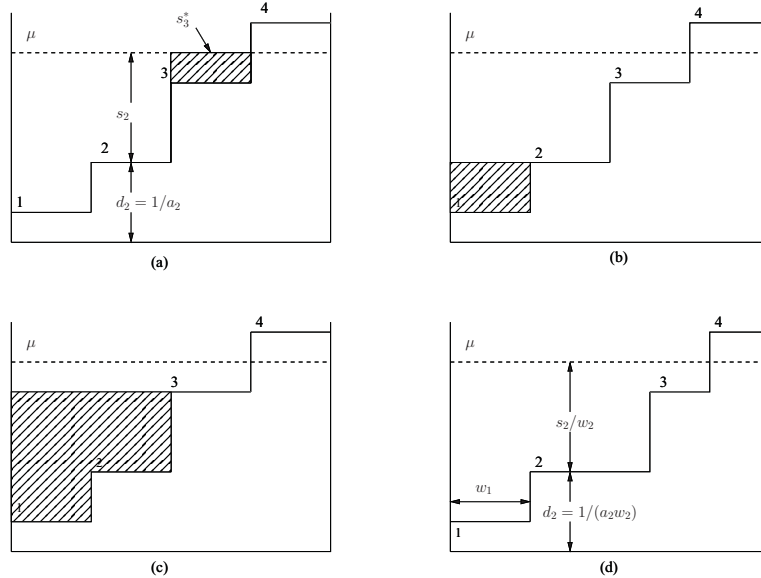


Figure 6.1: Illustration for the proposed generalized Water-Filling Algorithm for the sum power problems (P-GWF). (a) Water level step $k^* = 3$, allocated power for the third step s_3^* , and step/stair depth $d_i = 1/a_i$. (b) $ER(k)$ (which is determined by the shadowed area, representing the total water/power, up to, but excluding step k) when $k = 2$. (c) $ER(k)$ when $k = 3$. (d) The weighted case.

Let $ER(k)$ denote the Exponential Rate function achieved with the power below step k , which

can be written by

$$ER(k) = 2^{2R(k)} = \prod_{i=1}^{k-1} \left(\frac{a_i}{a_k} \right), \quad \text{for } k = 1, \dots, K, \quad (6.8)$$

where the factor “2” in the exponent comes from the fractional coefficient before the sum of logarithm functions in (6.6). Further the exponential rate target given is defined by η : $\eta = 2^{2B}$.

In the following, we explain how to find the water level step k^* without the knowledge of the water level μ . Fig. 6.1(b) and Fig. 6.1(c) illustrate the concepts of $ER(k)$ achieved by the power from the shadowed area for the cases of $k = 2$ and $k = 3$ respectively. As an example of Fig. 6.1(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, $(1/a_3 - 1/a_1)$, and (ii) the step depth difference between the 3rd and the 2nd step, $(1/a_3 - 1/a_2)$. Thus, the achieved data rate using power under the 3rd step can be written as

$$\begin{aligned} R(k=3) &= \frac{1}{2} \log_2 \left[1 + a_1 \left(\frac{1}{a_3} - \frac{1}{a_1} \right) \right] + \frac{1}{2} \log_2 \left[1 + a_2 \left(\frac{1}{a_3} - \frac{1}{a_2} \right) \right] \\ &= \frac{1}{2} \log_2 \left[\frac{a_1 a_2}{a_3^2} \right]. \end{aligned} \quad (6.9)$$

Therefore, the corresponding $ER(k=3)$ is given by

$$ER(k=3) = 2^{2 \cdot R(k=3)} = \frac{a_1 a_2}{a_3^2}, \quad (6.10)$$

which is an expansion of the composite form of (6.8).

To clearly understand the procedures of the proposed algorithm, the line of the methodology is briefly summarized by the following: using $\{\frac{1}{a_k} - \frac{1}{a_i}\}_{i=1}^{k-1}$, for $k = 1, \dots, K$, to define the achieved data rate or the exponential rate sequence in k ; using the exponential rate sequence to determine the highest step k^* ; and then this k^* is used to compute the optimal solution to the target problem. *Thus, $\{\frac{1}{a_k} - \frac{1}{a_i}\}_{i=1}^{k-1}, \forall k$, is not guaranteed to be the optimal solution. It is only utilized for computing the optimal solution in this proposed algorithm.*

The explicit solution, on the other hand, is optimal, and its optimality proof to (6.2) will be introduced, as an instance of the generalized case in following subsection.

6.2.3 Extend to Weighted Case

For the weighted or generalized case, an extended problem can be stated as: given the weighted coefficients $w_i > 0, \forall i$, associated with $\{a_i w_i\}_{i=1}^K$ which are assumed to be in decreasing order (similar to the case in Subsection 6.2.1, the indexes can be arbitrarily renumbered to satisfy this condition), find a group of the powers $\{s_i\}$ which are the solutions to the following problem,

$$\begin{aligned} \min_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K s_i \\ \text{subject to:} \quad & \frac{1}{2} \sum_{i=1}^K w_i \log_2(1 + a_i s_i) = B; 0 \leq s_i, \forall i. \end{aligned} \quad (6.11)$$

In Fig. 6.1(d), the width of the i th stair/step is denoted by w_i . The term s_i denotes the allocated power represented by the area above step i under water. The value of $1/a_i$ denotes the area, under the i th step to the bottom of the tank. Hence, the step depth of the i th step is given by

$$d_i = 1/(a_i w_i), \quad i = 1, \dots, K. \quad (6.12)$$

Then $R(k)$ can be expressed by $R(k) =$

$$\begin{aligned} & \frac{1}{2} \sum_{i=1}^{k-1} w_i \log_2 \left[1 + a_i w_i \left(\frac{1}{a_k w_k} - \frac{1}{a_i w_i} \right) \right] \\ = & \frac{1}{2} \sum_{i=1}^{k-1} w_i \log_2 \left[\frac{a_i w_i}{a_k w_k} \right] \\ = & \frac{1}{2} \sum_{i=1}^{k-1} \log_2 \left[\frac{a_i w_i}{a_k w_k} \right]^{w_i}. \end{aligned} \quad (6.13)$$

The corresponding exponential rate function is

$$ER(k) = 2^{2R(k)} = \prod_{i=1}^{k-1} \left(\frac{a_i w_i}{a_k w_k} \right)^{w_i}, \quad \text{for } k = 1, \dots, K. \quad (6.14)$$

Based on these extended definitions, we have the following proposition to compute the solution to (6.11).

Proposition 6.1. The explicit solution, by finite amounts of computation, to (6.11) is:

$$s_i = \left[\frac{s_{k^*}}{w_{k^*}} + (d_{k^*} - d_i) \right] w_i, \quad \text{for } 1 \leq i \leq k^*; \text{ and } s_i = 0, \text{ for } k^* < i \leq K, \quad (6.15)$$

where

$$k^* = \max \left\{ k \mid ER(k) < \eta, \ 1 \leq k \leq K \right\} \quad (6.16)$$

with $ER(k)$ defined by (6.14), the power level for this step is

$$s_{k^*} = \frac{1}{a_{k^*}} \left[\left(\frac{\eta}{ER(k^*)} \right)^{\frac{1}{\sum_{i=1}^{k^*} w_i}} - 1 \right] \quad (6.17)$$

and the optimal sum power allocated in (6.11) is:

$$P^* = \sum_{i=1}^{k^*} \left[\frac{1}{a_{k^*} w_{k^*}} \cdot \left(\frac{\eta}{ER(k^*)} \right)^{\frac{1}{\sum_{i=1}^{k^*} w_i}} - \frac{1}{a_i w_i} \right] w_i. \quad (6.18)$$

Prior to the formal proof, we first show how s_{k^*} in (6.17) is obtained, and whether $\{s_i\}$ in (6.15), including s_{k^*} , is a feasible solution to (6.11). Using (6.15) and (6.16), the first constraint, *i.e.*, the rate constraint, of (6.11) leads to the following equation in s_{k^*} :

$$\begin{aligned} B &= \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2(1 + a_i s_i) \\ &= \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2 \left[1 + a_i w_i \left(\frac{s_{k^*}}{w_{k^*}} + \frac{1}{a_{k^*} w_{k^*}} - \frac{1}{a_i w_i} \right) \right] \\ &= \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2 \left[a_i w_i \left(\frac{s_{k^*}}{w_{k^*}} + \frac{1}{a_{k^*} w_{k^*}} \right) \right] \\ &= \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2 \left\{ \frac{a_i w_i}{a_{k^*} w_{k^*}} \left[a_{k^*} w_{k^*} \left(\frac{s_{k^*}}{w_{k^*}} + \frac{1}{a_{k^*} w_{k^*}} \right) \right] \right\} \\ &= \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2 \left(\frac{a_i w_i}{a_{k^*} w_{k^*}} \right) + \frac{1}{2} [\log_2(1 + a_{k^*} s_{k^*})] \sum_{i=1}^{k^*} w_i. \end{aligned} \quad (6.19)$$

So, this equation in s_{k^*} is further simplified as below:

$$B = \frac{1}{2} \sum_{i=1}^{k^*} w_i \log_2 \left(\frac{a_i w_i}{a_{k^*} w_{k^*}} \right) + \frac{1}{2} [\log_2(1 + a_{k^*} s_{k^*})] \sum_{i=1}^{k^*} w_i. \quad (6.20)$$

Finally, (6.17) is obtained by solving this equation: (6.20), and then $\{s_i\}$ in (6.15) is the feasible solution.

Proof of Proposition 6.1. The formal proof is stated as follows. System (6.15) implies that

$$\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} = \frac{w_i}{\frac{1}{a_i} + s_i}, \text{ for } 1 \leq i \leq k^*. \quad (6.21)$$

Let

$$\lambda = \left(\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} \right)^{-1}. \quad (6.22)$$

From geometric view, λ itself is the water level μ . According to the definitions of k^* and s_{k^*} , for $k^* < i \leq K$, $\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} \geq \frac{w_i}{\frac{1}{a_i} + s_i}$ and $s_i = 0$. This statement can be explained as follows: $s_i = 0$ first comes from (6.15). To show the inequality, assume to the contrary that, for $k^* < i \leq K$, $\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} < \frac{w_i}{\frac{1}{a_i} + s_i}$ holds. That is to say, $\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}}{w_{k^*}} > \frac{1}{a_{k^*+1} w_{k^*+1}}$. Then,

$$1 + a_i w_i \left(\frac{1}{a_{k^*} w_{k^*}} + \frac{s_{k^*}}{w_{k^*}} - \frac{1}{a_i w_i} \right) > 1 + a_i w_i \left(\frac{1}{a_{k^*+1} w_{k^*+1}} - \frac{1}{a_i w_i} \right), \forall i. \quad (6.23)$$

The logarithm operation is applied to both sides of the inequality; and then the summation operation with the weights is applied with the index running from 1 to k^* . Finally put the exponentiation on the last result of both sides. As a result, $\eta > ER(k^* + 1)$. However, a contradiction of the maximum k^* not being maximum is acquired. Therefore, for $k^* < i \leq K$, $\frac{w_{k^*}}{\frac{1}{a_{k^*}} + s_{k^*}} \geq \frac{w_i}{\frac{1}{a_i} + s_i}$.

Let $\sigma_i = 1 - \lambda \frac{w_i}{\frac{1}{a_i} + s_i}$. Then

$$\sigma_i \geq 0, \text{ for } k^* < i \leq K; \sigma_i = 0, \text{ for } 1 \leq i \leq k^*. \quad (6.24)$$

Therefore, the following system holds:

$$\begin{cases} 1 - \frac{\lambda w_i}{\frac{1}{a_i} + s_i} - \sigma_i = 0, \text{ for } 1 \leq i \leq K; s_i \geq 0, \sigma_i s_i = 0, \sigma_i \geq 0, \forall i; \\ \sum_{i=1}^K w_i \log_2(1 + a_i s_i) \geq 2B, \lambda \left[\sum_{i=1}^K w_i \log_2(1 + a_i s_i) - 2B \right] = 0, \lambda \geq 0. \end{cases} \quad (6.25)$$

By observation, the equation and inequality set above is a set of the KKT conditions of the problem in (6.11) and the water level μ is equal to the Lagrange multiplier λ mentioned above. Note that the Lagrange function of the problem (6.11) is

$$L(\{s_i\}, \lambda, \{\sigma_i\}) = \sum_{i=1}^K s_i - \lambda \left[\sum_{i=1}^K w_i \log_2(1 + a_i s_i) - 2B \right] - \sum_{i=1}^K \sigma_i s_i. \quad (6.26)$$

Since problem (6.11) is, in essence, a differentiable convex optimization problem, not only are the KKT conditions mentioned above sufficient, but they are also necessary for optimality. We observe that the General Constraint Qualification of the problem holds. This (3.71) is often abbreviated as

GCQ, for which, it is seen that Slater's condition [7] is a special case and implies GCQ. Proposition 6.1 hence is proved.

Note that Proposition 6.1 is the first proposition in section 6.2 of this chapter.

Remark 1. Proposition 6.1, at the first line in the formal proof, stated that $\{s_k\}_{k=1}^K$ in (6.15) implies (6.21), then the Lagrange multipliers are constructed by (6.22) and (6.24), and it is seen that for $\{s_k\}_{k=1}^K$ in (6.15), there exists the group of Lagrange multipliers constructed above to satisfy the KKT conditions of the problem (6.11). Therefore, according to optimization theory, $\{s_k\}_{k=1}^K$ in (6.15) is the optimal solution to problem (6.11). Within the statement of Proposition 6.1, it is worth mentioning that (6.14) determines $ER(k)$, then obtains k^* by (6.16), and further s_{k^*} by (6.17).

Thus, the first step is to calculate $ER(k)$, then find the water level step, k^* from (6.16), which is the maximal index of $ER(k) < \eta$. The corresponding power level for this step, s_{k^*} , can be obtained by applying (6.17). Then for those steps with index higher than k^* , the power level is assigned with zero. For those steps below k^* , the power level is assigned by the first expression in (6.15). The first term (s_{k^*}/w_{k^*}) inside the square bracket denotes the depth of the k^* th step to the water level. The second term inside the square bracket denotes the step depth difference between the k^* th step and the i th step. Therefore, the sum inside the square bracket means the depth of the i th step to the water level. When this quantity is multiplied by the width of this step, the volume of the water above this step (allocated power) can be obtained.

When the weighting factors are set to ones, a corollary of Proposition 6.1 is stated as follows.

Corollary 6.1. The explicit solution to (6.2) is:

$$s_i = \begin{cases} s_{k^*} + (d_{k^*} - d_i) & 1 \leq i \leq k^* \\ 0, & k^* < i \leq K, \end{cases} \quad (6.27)$$

where the water level step k^* is given by

$$k^* = \max \left\{ k \mid ER(k) < \eta, \ 1 \leq k \leq K \right\} \quad (6.28)$$

with $ER(k)$ defined by (6.8), the power level for this step is

$$s_{k^*} = \frac{1}{a_{k^*}} \left[\left(\frac{\eta}{ER(k^*)} \right)^{\frac{1}{k^*}} - 1 \right] \quad (6.29)$$

and the optimal sum power allocated in (6.2) is:

$$P^* = \sum_{i=1}^K s_i = \sum_{i=1}^{k^*} \left[\frac{1}{a_{k^*}} \cdot \left(\frac{\eta}{ER(k^*)} \right)^{\frac{1}{k^*}} - \frac{1}{a_i} \right]. \quad (6.30)$$

Note that the solution $\{s_i\}$ expressed in (6.27) has an identical geometric form to that in [19]. However it is solved differently, since it is stemming from (6.28) and (6.29).

Similar to the weighed case, the first step of the proposed approach is to find the water level step k^* based on (6.28). Then the power s_{k^*} at this step can be determined based on (6.29). For those steps with index higher than k^* , no power is assigned. For those steps with index lower than k^* , their power levels are obtained by adding s_{k^*} to the corresponding level depth difference, between the k^* th step and the i th step, which are shown in (6.27).

In the following descriptions of algorithmic implementation, only weighted case is provided.

From Proposition 6.1, when k^* is obtained, $ER(k^*)$ is known. Then it is memorized to compute s_{k^*} . Thus, how to search k^* is a key point for the proposed P-GWF. The procedure of P-GWF approach is stated below:

Algorithm P-GWF:

- 1) Let $\eta = 2^{2B}$. Initialize $W_s = 0$; $ER_M = ER^* = 1$; $i = 1$. If $K = 1$, output the optimal solution $s_1^* = \frac{1}{a_1}[\eta^{\frac{1}{w_1}} - 1]$; else go to 2).
- 2) Compute $W_s \leftarrow W_s + w_i$; $ER^* \leftarrow ER^* \cdot (\frac{d_{i+1}}{d_i})^{W_s}$. Then $i \leftarrow i + 1$, where the symbol " \leftarrow " represents the assignment operation.
- 3) If $ER^* < \eta$ and $i < K$, $ER_M = ER^*$, and repeat the step 2); else, for $ER^* \geq \eta$, output $k^* = i - 1$, $W_s = W_s - w_i$; for $i = K$ and $ER^* < \eta$, $k^* = i$, $W_s = W_s + w_i$ and $ER_M = ER^*$. Finally, let $s_{k^*} = \frac{1}{a_{k^*}} \left[\left(\frac{\eta}{ER_M} \right)^{\frac{1}{W_s}} - 1 \right]$.

We can observe that $\frac{s_{k^*}}{w_{k^*}} + d_{k^*}$ is the water level due to $\frac{s_{k^*}}{w_{k^*}} + d_{k^*} = \frac{s_i}{w_i} + d_i$, for $1 \leq i \leq k^*$. In addition, W_s and ER_M are used, in each of the iteration, for a factor of less computation.

6.3 Solving Generalized RRM Problem Using P-GWF

In this section, we generalize the basic sum power RRM problem (6.2) to the sum power problems of WFPP, WFGPP, and WFGBP. The last case, P-WFGBP, is the most generalized RRM problem which will strip down to the other forms when applied to special values of lower and upper bounds and the number of its groups. To the best of the authors' knowledge, there is no existing algorithm reported in the open literatures to compute the exact solution for the generalized sum power problem P-WFGBP.

6.3.1 Weighted Water-Filling with Individual Peak Power Constraints (P-WFPP) for Sum Power Problems

Let P_i denote the peak power restriction of the i th channel. The weighted P-WFPP problem is stated by

$$\begin{aligned} \min_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K s_i \\ \text{subject to:} \quad & \sum_{i=1}^K w_i \log_2(1 + a_i s_i) \geq 2B; 0 \leq s_i \leq P_i, \forall i. \end{aligned} \quad (6.31)$$

Comparing the problem (6.31) with (6.11), the constraint of $0 \leq s_i$ is replaced with $0 \leq s_i \leq P_i$, *i.e.*, adding additional individual peak power constraint, and $\sum_{i=1}^K w_i \log_2(1 + a_i s_i) = 2B$ is replaced with $\sum_{i=1}^K w_i \log_2(1 + a_i s_i) \geq 2B$. In fact, by properly further reducing some allocated power(s), we can reach the equality constraint of the transmitted bits. Thus, problem (6.31) is reasonably assumed here.

Proposition 6.1 in subsection 6.2.3 provides an explicit exact solution using the proposed approach. P-WFPP problem can be obtained with some modifications to P-GWF. For convenience, the expression (6.14) can be extended into the expression:

$$ER(i_k) = \prod_{t=1}^{k-1} \left(\frac{d_{i_k}}{d_{i_t}} \right)^{w_{i_t}}, \text{ for } k = 1, \dots, |E|,$$

where E is a subsequence of the sequence $\{1, 2, \dots, K\}$, $|E|$ is the cardinality of the set E , so E can be expressed through $\{i_1, i_2, \dots, i_{|E|}\}$. Especially, if E is taken as the sequence $\{1, 2, \dots, K\}$, then the extended expression is regressed into the original expression (6.14). Similarly, some corresponding changes in (6.15)-(6.17) are also made (*i.e.*, the subscripts of sequence are replaced with

those of the subsequence). For avoiding notation-wise tediousness, these extended expressions are still labelled by (6.15)-(6.17) in the following algorithm descriptions.

Algorithm P-WFPP:

Input: arrays $\{a_i, w_i, P_i\}$ for $i = 1, 2, \dots, K$, the set $E = \{1, 2, \dots, K\}$, and $\eta = 2^{2B}$.

- 1) Utilize (6.15)-(6.17) to compute $\{s_i\}$.
- 2) The set Λ is defined by the set $\{i | s_i > P_i, i \in E\}$. If Λ is the empty set, output $\{s_i\}_{i=1}^K$; else, $s_i = P_i$, for $i \in \Lambda$.
- 3) Update E with $E \setminus \Lambda$ and η with $\eta / [\prod_{t \in \Lambda} (1 + a_t P_t)^{w_t}]$. Then return to 1) of the P-WFPP.

Remark 2. 3) in P-WFPP is a dynamic power distribution process. The state of this process is the difference between the individual peak power sequence and the current power distribution sequence obtained by P-GWF. The control of this process is to use (6.15)-(6.17) of Algorithm P-GWF based on the state mentioned above. And, a new state appears for next time stage. Therefore, a dynamic power distribution process, P-WFPP, with the state feedback is formed. Since the finite set E is getting smaller and smaller until the set Λ is empty, P-WFPP carries out K loops to compute the optimal solution, at most. In detail, updating E with $E \setminus \Lambda$ is to remove the set $\{i | s_i > P_i\}$, when $s_i \leftarrow P_i, \forall i$ in the set. Then over the updated set E , the exponent rate η is updated with $\eta / [\prod_{t \in \Lambda} (1 + a_t P_t)^{w_t}]$ correspondingly. Further, the process of updating E and η is a middle process, from the current state to form the current control, based on system theory.

For Algorithm P-WFPP, we can obtain the following results:

Proposition 6.2: P-WFPP can provide the exact optimal solution to the problem (6.31) by finite amounts of computation.

Its proof can be referred to in [21].

6.3.2 Weighted Water-Filling with Group Peak Power Constraints (P-WFGPP) for Sum Power Problems

Let $\{\chi_i\}_{i=1}^T$ be a partition of the index set: $\{1, \dots, K\}$. For convenience, the elements of χ_i can be listed, monotonically increasing, *i.e.*, $i_1 < i_2 < \dots < i_{|\chi_i|}$. Let $\overline{P}_i (> 0)$ denote the upper or

peak power bound of the power constraint for the i th group channels, $\forall i$. The weighted P-WFGPP problem can be written by

$$\begin{aligned} \min_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K s_i \\ \text{subject to:} \quad & \sum_{i=1}^K w_i \log_2(1 + a_i s_i) \geq 2B; 0 \leq s_k, \forall k; \sum_{k \in \chi_i} s_k \leq \bar{P}_i, \quad i = 1, \dots, T. \end{aligned} \quad (6.32)$$

Comparing the problem (6.32) with (6.31), we know that the constraints of $0 \leq s_i \leq P_i, \forall i$, are extended to $0 \leq \sum_{k \in \chi_i} s_k \leq \bar{P}_i$, *i.e.*, if every χ_i is taken as a singleton, the problem (6.32) is regressed into the problem (6.31).

To solve the problem (6.32), let us recall the original preliminary RRM problem and its solution (GWF) reported in our earlier work [19], for preparation:

$$\begin{aligned} \max_{\{s_k\}_{k=1}^{K'}} \quad & \sum_{k=1}^{K'} w_k \log_2(1 + a_k s_k) \\ \text{subject to:} \quad & \sum_{k=1}^{K'} s_k \leq P_{\text{total}}; 0 \leq s_k, \forall k. \end{aligned} \quad (6.33)$$

GWF gives the following solution to (6.33) by [19]:

$$s_m = \left[\frac{s_{k'^*}}{w_{k'^*}} + (d_{k'^*} - d_m) \right] w_m, \text{ for } 1 \leq m \leq k'^*; s_m = 0, \text{ for } k'^* < m \leq K', \quad (6.34)$$

where

$$k'^* = \max \left\{ k \mid P_2(k) > 0, \quad 1 \leq k \leq K' \right\} \quad (6.35)$$

and the power level for this step is

$$s_{k'^*} = \frac{w_{k'^*}}{\sum_{m=1}^{k'^*} w_m} P_2(k'^*), \quad (6.36)$$

where

$$P_2(k) = \left[P_{\text{total}} - \sum_{m=1}^{k-1} (d_k - d_m) w_m \right]^+, \text{ for } k = 1, \dots, K', \quad (6.37)$$

and

$$d_m = 1/(a_m w_m), \quad m = 1, \dots, K'. \quad (6.38)$$

The following statement presents an algorithm which is a combination and modification of the GWF and P-GWF. This algorithm is termed as the P-WFGPP.

Similarly, for convenience, the expression (6.14) can be extended into the expression:

$$ER(i_k) = \prod_{t=1}^{k-1} \left(\frac{d_{i_k}}{d_{i_t}} \right)^{w_{i_t}}, \text{ for } k = 1, \dots, |E|,$$

where E is a subsequence of the sequence $\{1, 2, \dots, K\}$.

Algorithm P-WFGPP:

Input: Sets $\{a_k, w_k\}_{k=1}^K$, $\{\bar{P}_i\}_{i=1}^T$, $E = \{1, 2, \dots, T\}$, and $\eta = 2^{2B}$.

- 1) Let $n = 1$ and $\Lambda = \emptyset$ (empty set). Utilize (6.15)-(6.17) to compute $\{s_k\}_{k=1}^K$.
- 2) The set Λ_n is assigned by the set $\{i | \sum_{k \in \chi_i} s_k > \bar{P}_i, i \in E\}$. If Λ_n is the empty set, output $\{s_i\}_{i=1}^K$; else, $P_{\text{total}} \leftarrow \bar{P}_i$, $K' \leftarrow |\chi_i|$, χ_i is renamed into the set $\{i_1, \dots, i_{K'}\}$, and then utilize (6.34)-(6.36) from GWF, for $i \in \Lambda_n$.
- 3) Update E with $E \setminus \Lambda_n$ and η with $\eta / [\prod_{i \in \Lambda_n} \prod_{t \in \chi_i} (1 + a_t s_t)^{w_t}]$. Then $n \leftarrow n + 1$, $K \leftarrow K - \sum_{i \in \Lambda_n} |\chi_i|$ and return to the second statement in 1) of the P-WFGPP.

Proposition 6.3: P-WFGPP can provide the exact optimal solution to the problem (6.32) by finite amounts of computation.

Its proof can be referred to in [21].

6.3.3 Weighted Water-Filling with Group Bounded Power Constraints for Sum Power Problem (P-WFGBP)

Let $\{\chi_i\}_{i=1}^T$ be a partition of the index set: $\{1, \dots, K\}$. Assume that $0 \leq \underline{P}_i \leq \bar{P}_i$, and \underline{P}_i and \bar{P}_i denote the lower bound and the upper bound of the power constraint for the i th group channels, $\forall i$. The weighted P-WFGBP problem is stated by

$$\begin{aligned} \min_{\{s_i\}_{i=1}^K} \quad & \sum_{i=1}^K s_i \\ \text{subject to:} \quad & \sum_{i=1}^K w_i \log_2(1 + a_i s_i) \geq 2B; 0 \leq s_k, \forall k; \\ & \underline{P}_i \leq \sum_{k \in \chi_i} s_k \leq \bar{P}_i, i = 1, \dots, T. \end{aligned} \tag{6.39}$$

Comparing the problem (6.39) with (6.32), it is seen that the constraints of $0 \leq \sum_{k \in \chi_i} s_k \leq \bar{P}_i, \forall i$, are generalized to $\underline{P}_i \leq \sum_{k \in \chi_i} s_k \leq \bar{P}_i$, *i.e.*, adding additional group lower bound power constraints.

The lower bound of the additional constraint can be used to guarantee the fair transmitted rate from the i th group transmission, whereas the upper bound of the additional constraint can be used to limit the total interference from the i th group. The problem (6.39) is thus referred to as power (weighted) water-filling with group bounded power constraints (P-WFGBP).

Similarly, due to the explicit solution using generalized view approach that is provided in Proposition 6.1, the proposed GWF and P-WFGPP can be applied to the P-WFGBP problem with some modifications. The following statement presents a generalized algorithm, which is based on a meaningful combination and modification of the GWF and P-WFGPP.

Note, for the problem:

$$\begin{aligned} \max_{\{s_k\}_{k \in \chi_i}} \quad & \sum_{k \in \chi_i} w_k \log_2(1 + a_k s_k) \\ \text{subject to:} \quad & \sum_{k \in \chi_i} s_k = \bar{P}_i; 0 \leq s_k, \forall k, \end{aligned} \quad (6.40)$$

its optimal value is denoted by \bar{V}_i , for $i = 1, \dots, T$. It can be observed that there does not exist any solution to problem (6.39), if $\sum_{i=1}^T \bar{V}_i < 2B$. Further, if $\sum_{i=1}^T \bar{V}_i = 2B$, the optimal solution to (6.40) denoted by $\{s_k^*\}_{k \in \chi_i}$, for $i = 1, \dots, T$, can determine the optimal solution, $\{\{s_k^*\}_{k \in \chi_i}\}_{i=1}^T$, to (6.39). Hence, $\sum_{i=1}^T \bar{V}_i > 2B$ is assumed in the following. Further, \bar{P}_i , in the problem mentioned above, is replaced with $\underline{P}_i, \forall i$, and the corresponding optimal value is denoted by $\underline{V}_i, \forall i$. If $\sum_{i=1}^T \underline{V}_i \geq 2B$, the optimal solutions to the problems undergo a similar process to that mentioned above. For $i = 1, \dots, T$, this constitutes an optimal solution to problem (6.39) and the optimal value is $\sum_{i=1}^T \underline{P}_i$ with practical meaning. Therefore, we only consider the cases under $\sum_{i=1}^T \bar{V}_i > 2B$, which has been assumed before, together with $\sum_{i=1}^T \underline{V}_i < 2B$.

It is seen that if $\underline{P}_i = 0, \forall i$, then problem P-WFGBP (6.39) is reduced into problem P-WFGPP (6.32); and if χ_i is regressed to a singleton and $\underline{P}_i = 0, \forall i$, then problem P-WFGBP (6.39) is reduced into problem P-WFPP (6.31). Thus, problem P-WFGBP (6.39) is the most general form of the RRA problems. It is called the generalized problem in this chapter. The corresponding algorithm is described below.

Algorithm P-WFGBP:

Input: the channel gains $\{a_k\}_{k=1}^K$, the weights $\{w_k\}_{k=1}^K$, the group lower and upper power bounds $\{\underline{P}_i, \bar{P}_i\}_{i=1}^T$, the partition $\{\chi_i\}_{i=1}^T$ and the (weighted) sum-rate constraint B .

- 1) $P_{\text{total}} \leftarrow \underline{P}_i, K' \leftarrow |\chi_i|, \chi_i$ is written into the set $\{i_1, \dots, i_{K'}\}$, and then utilize (6.34)-(6.36) from GWF, for $i = 1, \dots, T$. Hence, the solutions $\{\{s'_{i_t}\}_{t \in \chi_i}\}_{i=1}^T$ are obtained.
- 2) Update B with $B - \frac{1}{2} \sum_{k=1}^K w_k \log_2(1 + a_k s'_k), \frac{1}{a_k}$ with $\frac{1}{a_k} + s'_k, \forall k$, and \overline{P}_i with $\overline{P}_i - \underline{P}_i, \forall i$.
- 3) Utilize P-WFGPP to compute $\{s_k\}_{k=1}^K$ as the optimal solution to (6.32) under the updated parameters.
- 4) Output the optimal solution $\{s_k\} \leftarrow \{s_k + s'_k\}$ to the problem (6.39).

Remark 3. Due to its definition mentioned above, P-WFGBP carries out T loops to compute the exact optimal solution, at most.

For optimality of the proposed P-WFGBP, we have the following conclusion:

Proposition 6.4: P-WFGBP can provide the exact optimal solution to the problem (6.39) via finite amounts of computation.

Its proof can be referred to in [21].

Remark 4. If we chose an approach, similar to P-CWF to directly solve the class of problems (6.39), a non-linear system with non-linear equations and inequalities in multiple dual variables (as below) would have had to be solved in a difficult manner:

$$\left\{ \begin{array}{l} \sum_{i=1}^T \sum_{j \in \chi_i} w_j \log_2[1 + a_j \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+] \geq 2B; \\ \lambda \{ \sum_{i=1}^T \sum_{j \in \chi_i} w_j \log_2[1 + a_j \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+] - 2B \} = 0; \\ \underline{P}_i \leq \sum_{j \in \chi_i} \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+ \leq \overline{P}_i, \text{ for } i = 1, 2, \dots, T; \\ \underline{\sigma}_i [\sum_{j \in \chi_i} \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+ - \underline{P}_i] = 0, \text{ for } i = 1, 2, \dots, T; \\ \overline{\sigma}_i [\sum_{j \in \chi_i} \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+ - \overline{P}_i] = 0, \text{ for } i = 1, 2, \dots, T; \\ \lambda \geq 0; \underline{\sigma}_i \geq 0, \overline{\sigma}_i \geq 0, \text{ for } i = 1, 2, \dots, T. \end{array} \right. \quad (6.41)$$

For example, the system of (3) and (4) in [36] is defined, which was claimed to find a very general multiple water level multiple constrained water filling result. However, it cannot be used for the exact solution to the mentioned problem (6.39). The reason is stated as follows. It is seen that $s_j = \left(\frac{w_j \lambda}{1 + \overline{\sigma}_i - \underline{\sigma}_i} - \frac{1}{a_j} \right)^+$, for $j \in \chi_i, i = 1, 2, \dots, T$, where $\{s_j\}$ is the solution to problem (6.39). According to (3) in [36], the water levels should be taken as $\mu_k = \frac{\lambda}{1 + \overline{\sigma}_k - \underline{\sigma}_k}$, for $k = 1, 2, \dots, T$. Since $\{s_j\}$ is the solution, it should also satisfy the second, and the fourth to the sixth constraints in the

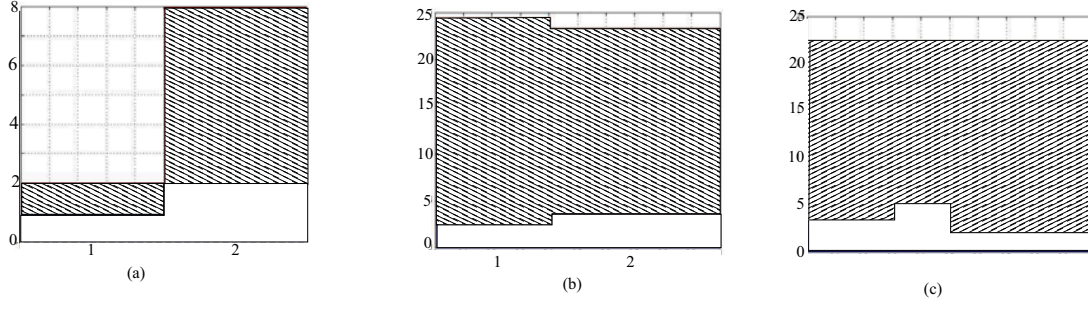


Figure 6.2: Illustration for Examples 1, 3, 4 respectively.

system (6.41). Thus, a further developed version from (4) in [36] cannot include only the water-levels $\{\mu_k\}$. It also should include other dual variables, as a prerequisite for solving the problem, although (4) in [36] does not need the dual variables which correspond to non-negativeness constraints of the solution $\{s_j\}$ due to using the function $(\cdot)^+$. This point for the assumed form (4) in [36] results in [36] not being able to be used to solve the target problem (6.39).

The algorithms proposed in [19] are to compute the solutions to the maximum throughput problems. The minimum sum power problems discussed in this chapter are different. In addition, KKT conditions for these two cases are different. As a result, algorithms of [19] cannot directly be used for the problems discussed here.

From duality between the throughput maximization and the sum power minimization, we can obtain uniqueness of the solution, discussed in this section.

6.4 Numerical Results and Complexity Analysis

A few numerical examples are presented in this section to illustrate the steps of the proposed algorithms.

Example 1. Instantiate a case of P-WFPP problem by

$$\begin{aligned} \min_{\{s_i\}_{i=1}^2} \quad & \sum_{i=1}^2 s_i \\ \text{subject to:} \quad & \sum_{i=1}^2 \log(1 + a_i s_i) \geq 3; 0 \leq s_1 \leq 1; 0 \leq s_2 \leq 8, \end{aligned} \tag{6.42}$$

where $a_1 = 1$ and $a_2 = 0.5$. Utilizing the proposed P-WFPP, the optimal solution is $\{s_1 = 1, s_2 = 6\}$ that is shown in Fig. 6.2(a).

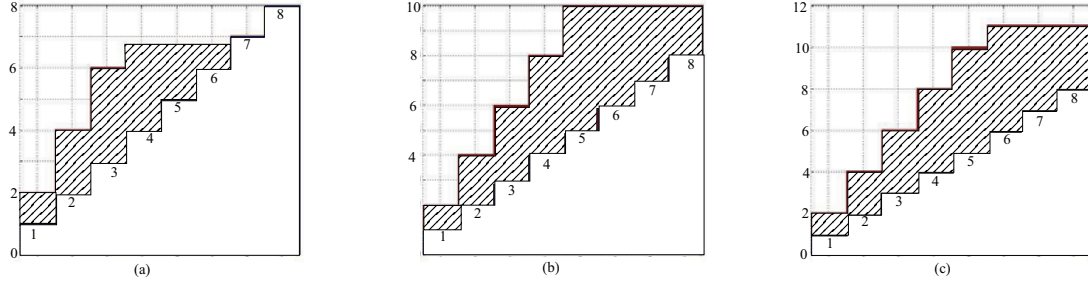


Figure 6.3: Illustration for Example 2, results for the first, second and third iterations respectively.

Example 2. Instantiate another case of P-WFPP problem with multiple channels:

$$\begin{aligned} \min_{\{s_i\}_{i=1}^8} \quad & \sum_{i=1}^8 s_i \\ \text{subject to:} \quad & \sum_{i=1}^8 \log(1 + a_i s_i) \geq 7; 0 \leq s_i \leq i, \forall i, \end{aligned} \quad (6.43)$$

where $a_i = 1/i, \forall i$. That is to say, the step depth monotonically increases from 1 to 8, as shown in Fig. 6.3 with the solution.

Using the proposed P-WFPP, the result of the first iteration is $s_6 = s_{k^*} = 6[(\frac{128}{64.8})^{\frac{1}{6}} - 1](\doteq 0.72)$ and then $\{s_i = i\}_{i=1}^3 \cup \{s_i = s_6 + 6 - i\}_{i=4}^6 \cup \{s_i = 0\}_{i=7}^8 \doteq \{s_i = i\}_{i=1}^3 \cup \{s_i = 6.72 - i\}_{i=4}^6 \cup \{s_i = 0\}_{i=7}^8$. The result of the second iteration is $s_8 = s_{k^*} = 8[(\frac{105}{32})^{\frac{1}{5}} - 1](\doteq 2.146)$ and then $\{s_i = i\}_{i=1}^5 \cup \{s_i = s_8 + 8 - i\}_{i=6}^8 (\doteq \{s_i = i\}_{i=1}^5 \cup \{s_i = 10.146 - i\}_{i=6}^8)$. The result of the third iteration is $s_8 = s_{k^*} = 8[(\frac{21}{8})^{\frac{1}{3}} - 1](\doteq 3.036)$ and then $\{s_i = i\}_{i=1}^5 \cup \{s_i = s_8 + 8 - i\}_{i=6}^8 (\doteq \{1, 2, 3, 4, 5, 5.036, 4.036, 3.036\})$.

According to the algorithm and Proposition 6.2, the result of the third iteration is indeed the optimal solution. These results are illustrated in Figs. 6.3(a)-(c).

Example 3. Instantiate a case of weighted P-WFPP problem by

$$\begin{aligned} \min_{\{s_i\}_{i=1}^2} \quad & \sum_{i=1}^2 s_i \\ \text{subject to:} \quad & \sum_{i=1}^2 w_i \log(1 + a_i s_i) \geq 3; 0 \leq s_i \leq 12, \forall i, \end{aligned} \quad (6.44)$$

where $a_1 = 1, a_2 = 0.5, w_1 = 0.4$ and $w_2 = 0.6$. Utilizing the proposed P-WFPP, the optimal result is $\{s_1 = \frac{64}{49}\sqrt{56} - 1, s_2 = 12\}$, which is shown in Fig. 6.2(b).

Example 4. As the last example, we instantiate a case of the sum power weighted water-filling

with group bounded power constraints (P-WFGBP) problem by:

$$\begin{aligned}
& \min_{\{s_i\}_{i=1}^3} \quad \sum_{i=1}^3 s_i \\
& \text{subject to: } \sum_{i=1}^3 w_i \log(1 + a_i s_i) \geq 3; 0 \leq s_i, \forall i; \\
& \quad 1 \leq s_1 + s_2 \leq 12; s_3 \leq 12,
\end{aligned} \tag{6.45}$$

where $a_1 = a_2 = a_3 = 1$, $w_1 = 0.3$, $w_2 = 0.2$ and $w_3 = 0.5$.

Utilizing the proposed Algorithm: P-WFGBP with inputting: $\chi_1 = \{1, 2\}$, $\chi_2 = \{3\}$ and $B \leftarrow 1.5$. The optimal solution is $\{s_1 = \frac{2^{2.8} \times 3^{0.7}}{5^{0.5}} - 1, s_2 = \frac{2^{3.8}}{5^{0.5} \times 3^{0.3}} - 1, s_3 = \frac{2^{2.8} \times 5^{0.5}}{3^{0.3}} - 1\}$ which is shown in Fig. 6.2(c), where the stair width for the three channels are 0.3, 0.2, 0.5 respectively specified by their weighting factors. The step depth is calculated using $1/(a_i w_i)$, leading to the step depth values of 10/3, 5, and 2 respectively for the three channels.

6.4.1 Complexity Analysis

For the non-weighted basic sum power water-filling problem (6.2), according to the expressions (9) and (10) in [36, Section 3], the conventional (sum power) water-filling algorithm had an exponential worst-case complexity [36, Section 3] of 2^K , where K is the number of the channels, even though the channel gains had been sorted in decreasing order. Pointing to this case, [36] and [8] proposed an improved algorithm with worst-case complexity of K iterations. Since each iteration consists of the multiple basic elementary function evaluations, the arithmetic operations, and the logical operations, the proposed P-GWF is measured on these operations more accurately than the iterations. P-GWF uses K iterations, each of which includes 8 operations: 1 basic elementary function evaluation (BE), 5 arithmetic operations (AOs), and 2 logical operations (LOs). For an algorithm, a total of these numbers right down to such operation(s) can measure the complexity level of the algorithm [38, Chapter 8].

The conventional approach [8, p. 310] requires a total of $O(K^2)$ operations which consist of $\frac{(K+1)K}{2} + 1$ BEs, $\frac{(K+1)K}{2} + 4$ AOs and K LOs, under the $K + 4$ memory unit requirement with a worst-case complexity of K iterations.

As mentioned above, the proposed P-GWF uses $8K$ operations, which consist of K BEs, $5K$ AOs, and $2K$ LOs under the $K + 4$ memory unit requirement with a worst-case complexity of K iterations.

For P-WFPP, it needs K loops to compute the optimal solution, at most. The required number of operations is, at worst, $\sum_{i=1}^K 8i = 4K^2 + 4K$ fundamental basic elementary function evaluations, arithmetical and logical operations.

For P-WFGBP, it needs T loops to compute the optimal solution, at most, where $T \leq K$. The required number of operations, at worst, is $T \times O(K^2)$ fundamental basic elementary function evaluations, arithmetical and logical operations. However, it is known from the prior works mentioned above that PD-IPM needs the computational complexity of $O(K^{3.5} \log(1/\epsilon))$, to compute an ϵ solution.

In this complexity analysis, we didn't take sorting procedure into consideration. It is stated in [36] that the channel gain sequences come from the eigenvalues of a matrix. There are many algorithms to compute the eigenvalues and eigenvectors, with the eigenvalues sorted.

Chapter 7

Conclusions

A few new algorithms are presented for the MIMO and EH maximum throughput problems, in this thesis. Optimality proofs for the algorithms are also provided. *Compared with existing methods by others, our algorithms have the distinct characteristics: the proposed theory is strict and the proposed computation is fast.*

In detail, this thesis in Chapter 2 presents the efficient methods for computation of exact optimal solutions to the problems with more and more constraints. It also constructs solutions to the optimization problems based on the water-filling geometrical principle, and offers a formal proof of optimality for the constructed solutions. Thus, the proposed methods can solve the problems the conventional methods cannot. Further, they own the polynomial complexity, and the polynomial has a lower degree.

This thesis also presents a new mathematical model up to the hybrid systems of the EH, MIMO and power grid. The first, it designs the recursive geometric water-filling algorithm (RGWF) in Chapter 3 for the EH system, stemming from GWF. Then it extends RGWF for finding the optimal solutions of the new mathematical models with both the EH capability and the MIMO technology; and those with hybrid power sources of EH and the grid power, in Chapters 4-5, respectively. In addition, using GWF to compute the exact solutions to the sum power minimization problems, has been presented in Chapter 6.

On the theoretical level, this thesis sets up a solid theoretical framework for the discussion of the investigated algorithms for the target problems. Under such a framework, optimality proof for each of our algorithms is presented. *Note the difference between the recursive definition and the explicit*

definition (of an algorithm). This point can be referred to in any standard textbook on mathematical logic. Then, it is omitted in this thesis. If the word of recursive has been mentioned for an algorithm in this thesis, the algorithm is guaranteed to be recursive; else, it is not. In addition, this thesis does not consider other objective functions, such as the one to look for the fairness purpose. Maybe this fairness topic will be of our future research.

In addition, not only may the proposed approach keep up optimality of all the proposed water-filling algorithms, but it also may lift the performance of computing objective values much more. Numerical examples also show that utilization of the proposed algorithms can significantly improve efficiency of the computation.

For the future, we can extend our approach to designing optimal on-line power allocation algorithms for wireless communication systems, together with EH, MIMO and cognitive radio (CR) techniques.

Bibliography

- [1] S. Albers, *Online algorithms*, In Interactive Computation: The New Paradigm edited by D.Q. Goldin, S.A. Smolka and P. Wegner, pp. 143-164, 2006.
- [2] D. P. Bertsekas, *Nonlinear Programming*, 2nd Edition, Athena Scientific, Belmont, MA, 1999.
- [3] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 4th Edition, Nashua, NH, USA: Athena Scientific, 2007.
- [4] D. P. Bertsekas, and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, MA, 1997.
- [5] E. Biglieri, R. Calderbank, A. Constantinides, A. Goldsmith, A. Paulraj, and H. V. Poor, *MIMO Wireless Communications*, Cambridge University Press, 2007.
- [6] J. F. Bonnans, J. C. Gilbert, C. Lemaréchal, and C. A. Sagastizábal, *Numerical Optimization: Theoretical and Practical Aspects*, 2nd Edition, Springer, 2006.
- [7] S. Boyd, and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- [8] J. Cioffi, *Ee379c course reader, chapter 4*, <http://www.stanford.edu/group/cioffi/ee379c>, frequently updated.
- [9] N. Devroye, M. Vu and V. Tarokh, *Cognitive radio networks: Information theory limits, models and design*, IEEE Signal Process. Mag., Vol. 25, pp. 1223, 2008.
- [10] U. Erez, S. Shamai, and R. Zamir, *Capacity and lattice strategies for cancelling known interference*, IEEE Trans. Inf. Theory, Vol. 51, pp. 3820-3833, 2005.
- [11] G. B. Folland, *Real Analysis: Modern Techniques and Their Applications*, John Wiley And Sons, North York, 1999.

- [12] M. Gatzianas, L. Georgiadis, and L. Tassiulas, *Control of wireless networks with rechargeable batteries*, IEEE Trans. on Wireless Communications, Vol. 9, pp. 581-593, 2010.
- [13] M. Gorlatova, A. Wallwater, and G. Zussman, *Networking lowpower energy harvesting devices: Measurements and algorithms*, IEEE Transactions on Mobile Computing, Vol. 12, pp. 1853-1865, 2013.
- [14] J. Gong, S. Zhou, and Z. Niu, *Optimal power allocation for energy harvesting and power grid coexisting wireless communication systems*, IEEE Trans. Communications, Vol. 61, pp. 3040-3049, 2013.
- [15] M. Gregori, and M. Payaro, *Deploying multi-antenna energy-harvesting cooperative jammers in the mimo wiretap channel*, Asilomar conference on signals, systems and computers, pp. 1886-1890, 2012.
- [16] S. Haykin, *Cognitive radio: brain-empowered wireless communications*, IEEE J. Sel. Areas Commun., Vol. 23, pp. 201-220, 2005.
- [17] P. He, and L. Zhao, *Correction of convergence proof for iterative water-filling in Gaussian MIMO broadcast channels*, IEEE Trans. Inf. Theory, Vol. 57, pp. 2539-2543, 2011.
- [18] P. He, L. Zhao, and J. H. Lu, *Weighted sum-rate maximization for multi-user SIMO multiple access channels in cognitive radio networks*, EURASIP Journal on Advances in Signal Processing, 15 pages, doi: 10.1186/1687-6180-2013-80, 2013.
- [19] P. He, L. Zhao, S. Zhou, and Z. S. Niu, *Water-filling: a geometric approach and its application to solve generalized radio resource allocation problems*, IEEE Trans. on Wireless Communications, Vol. 12, pp. 3637-3647, 2013.
- [20] P. He, L. Zhao, S. Zhou, and Z. S. Niu, *Recursive water-filling for wireless links with energy harvesting transmitters*, IEEE Trans. on Vehicular Technology, Vol. 63, pp. 1232-1241, 2014.
- [21] P. He, and L. Zhao, *Solving A Class of Sum Power Minimization Problems Using Generalized Water-Filling*, IEEE Trans. on Wireless Communications, dio: 10.1109/TWC.2015.2459714, 2015.

- [22] P. He, L. Zhao, S. Zhou, and Z. S. Niu, *Optimal recursive power allocation for energy harvesting nodes with multiple antennas*, IEEE Trans. on Vehicular Technology, published on-line, DOI: 10.1109/TVT.2014.2366560, 2014.
- [23] P. He, and L. Zhao, *Non-commutative composite water-fillings for energy harvesting and smart power grid hybrid system with peak power constraints*, IEEE Trans. on Vehicular Technology, published on-line, DOI: 10.1109/TVT.2015.2418318, 2015.
- [24] P. He, L. Zhao, S. Zhou, and Z. S. Niu, *Radio Resource Management Using Geometric Water-filling*, Springer Brief in Computer Science, Springer, 2014.
- [25] C. Ho, and R. Zhang, *Optimal energy allocation for wireless communications with energy harvesting constraints*, IEEE Trans. on Signal Process., Vol. 60, pp. 4808-4818, 2012.
- [26] C. Hsu, H. Su, and P. Lin, *Joint subcarrier pairing and power allocation for OFDM transmission with decode-and-forward relaying*, IEEE Trans. on Signal Process., Vol. 59, pp. 399-414, 2010.
- [27] W. Jia, H. Zhu, Z. Cao, X. Dong, and C. Xiao, *Human-factor-aware privacy-preserving aggregation in smart grid*, IEEE Systems Journal, Vol. 8, pp. 598-607, 2014
- [28] H. Jiang, L. Lai, R. Fan, and H.V. Poor, *Optimal selection of channel sensing order in cognitive radio*, IEEE Trans. on Wireless Communications, Vol. 8, pp. 297-307, 2009.
- [29] 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, pp. 1570-1580, 2005.
- [30] N. Jindal, S. Vishwanath, A. Goldsmith, *On the duality of Gaussian multiple-access and broadcast channels*, IEEE Trans. Inf. Theory, Vol. 50, pp.768-783, 2004.
- [31] T. Kailath, A. Sayed, and B.Hassibi, *Linear Estimation*, Prentice Hall, 2000.
- [32] M. Kobayashi, and G. Caire, *An Iterative water-filling algorithm for maximum weighted sum-rate of Gaussian MIMO-BC*, IEEE J. Sel. Areas Commun., Vol. 24, pp. 1640-1646, 2006.

- [33] J. Lei, R. Yates, and L. Greenstein, *A generic model for optimizing single-hop transmission policy of replenishable sensors*, IEEE Trans. on Wireless Communications, Vol. 8, pp. 547-551, 2009.
- [34] J. Mitola, and G. Q. Maguire, *Cognitive radios: making software radios more personal*, IEEE Personal Commun. Vol. 6, 13-18, 1999.
- [35] O. Ozel, K. Tutuncuoglu, J. Yang, S. Ulukus, and A. Yener, *Transmission with energy harvesting nodes in fading wireless channels: Optimal policies*, IEEE J. Sel. Areas Commun., Vol. 29, pp. 1732-1743, 2011.
- [36] D. Palomar, *Practical algorithms for a family of waterfilling solutions*, IEEE Trans. Signal Process., Vol. 53, pp. 686-695, 2005.
- [37] D. Palomar, *A unified framework for communications through MIMO channels*, Ph.D. thesis, Universitat Politecnica De Catalunya, 2003.
- [38] C. H. Papadimitriou, and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Unabridged Edition, Dover Publications, 1998.
- [39] R. V. Prasad, P. Pawelczak, J. A. Hoffmeyer, and H. S. Berger, *Cognitive functionality in next generation wireless networks: standardization efforts*, IEEE Commun. Mag. Vol. 46, pp. 72-78, 2008.
- [40] Q. Qi, A. Minturn, and Y. Yang, *An efficient water-filling algorithm for power allocation in OFDM-based cognitive radio systems*, in Proc. International Conference on Systems and Informatics (ICSAI), pp. 2069-2073, Yantai, 2012.
- [41] Y. Rong, X. Tang, and Y. Hua, *A unified framework for optimizing linear non-regenerative multicarrier MIMO relay communication systems*, IEEE Trans. on Signal Process., Vol. 57, pp. 4837-4852, 2009.
- [42] C. E. Shannon, *A mathematical theory of communication*, Bell Sys. Tech. J., Vol. 27, pp. 379-423, 1948.

- [43] V. Sharma, U. Mukherji, V. Joseph, and S. Gupta, *Optimal energy management policies for energy harvesting sensor nodes*, IEEE Trans. on Wireless Communications, Vol. 9, pp. 1326-1336, 2010.
- [44] E. Telatar, *Capacity of multi-antenna Gaussian channels*, Europ. Trans. on Telecomm., Vol. 10, pp. 585-596, 1999.
- [45] D. Tse, and S. Hanly, *Multiaccess fading channels. Part I: Polymatroid structure, optimal resource allocation and throughput capacities*, IEEE Trans. Inf. Theory, Vol. 44, 2796-2815, 1998.
- [46] K. Tutuncuoglu, and A. Yener, *Optimum transmission policies for battery limited energy harvesting nodes*, IEEE Trans. on Wireless Communications, Vol. 11, pp. 1180-1189, 2012.
- [47] S. Vishwanath, S. Jafar, and A. Goldsmith, *Optimum power and rate allocation strategies for multiple access fading channels*, in Proc. IEEE Vehicular Technology Conf., pp. 2888-2892, Rhodes, 2001.
- [48] P. Viswanath, and D. Tse, *Sum capacity of the multiple antenna Gaussian broadcast channel and uplink-downlink duality*, IEEE Trans. Inf. Theory, Vol. 49, pp. 1912-1921, 2003.
- [49] H. Weingarten, Y. Steinberg, and S. Shamai, *The capacity region of the Gaussian multiple-input multiple-output broadcast channel*, IEEE Trans. Inf. Theory, Vol. 52, pp. 3936-3964, 2006.
- [50] J. Yang, and S. Ulukus, *Optimal packet scheduling in an energy harvesting communication system*, IEEE Trans. Commun., Vol. 60, pp. 220-230, 2012.
- [51] W. Yu, W. Rhee, Boyd, S., and J. M. Cioffi, *Iterative water-filling for Gaussian vector multi-access channels*, IEEE Trans. Inf. Theory, Vol. 50, pp. 145-152, 2004.
- [52] W. Yu, *Sum-capacity computation for the Gaussian vector broadcast channel via dual decomposition*, IEEE Trans. Inf. Theory, Vol. 52, pp. 754-759, 2006.
- [53] W. I. Zangwill, *Nonlinear Programming: A Unified Approach*, Prentice-Hall, Englewood Cliffs, N.J., 1969.

- [54] L. Zhang, Y. C. Liang, and Y. Xin, *Joint beamforming and power allocation for multiple access channels in cognitive radio networks*, IEEE J. Sel. Areas Commun., Vol. 26, pp. 38-51, 2008.
- [55] L. Zhang, Y. Xin, Y. C. Liang, and H. V. Poor, *Cognitive multiple access channels: optimal power allocation for weighted sum rate maximization*, IEEE Trans. Commun., Vol. 57, pp. 2754-2762, 2009.
- [56] R. Zhang, S. Cui, and Y. C. Liang, *On ergodic sum capacity of fading cognitive multiple-access and broadcast channels*, IEEE Trans. Inf. Theory, Vol. 55, pp. 5161-5178, 2009.