



Université catholique de Louvain (UCL)

Ecole Polytechnique de Louvain

ICTEAM Institute

1348 Louvain-la-Neuve, Belgium

Resource Allocation in OFDM-based Wireless Communication Systems

Tao Wang

Thesis presented for the Ph.D. degree
in Applied Sciences

Jury members:

Luc VANDENDORPE (UCL, Belgium) - Supervisor

Christophe DE VLEESCHOUWER (UCL, Belgium) - Chair

François HORLIN (Université Libre de Bruxelles, Belgium)

Wei YU (University of Toronto, Canada)

Jérôme LOUVEAUX (UCL, Belgium)

François GLINEUR (UCL, Belgium)

February 2012

Acknowledgements

First of all, I would like to thank Prof. L. Vandendorpe for bringing me into the topics of resource allocation and supervising this thesis. His broad and insightful perspectives in this field, as well as constant inspiration and support greatly helped me along years to finish this thesis. I would also like to thank Prof. W. Yu, F. Glineur, J. Louveaux, F. Horlin and C. De Vleeschouwer for their precious effort to examine this thesis. Their insightful and specific comments led to significant improvement on the rigor and readability of this thesis.

I am grateful to Prof. Geoffrey Ye Li and Josep Vidal for coordinating the reviews of related papers that have been submitted, as well as anonymous reviewers who gave many precious comments for improving the quality of those papers. I would also thank Fonds de la Recherche Scientifique (FNRS) of Belgium, and the Walloon Region for funding the projects MIMOCOM, the ARC SCOOP and the FP7 Network NEWCOM++ under which this thesis has been carried out. TOMLAB Optimization and MOSEK ApS are also greatly appreciated for offering free academic licenses.

Moreover, I have been deeply indebted to my family Zihua ZHU, Baoxu WANG, Linlin WANG who are always there supporting me unselfishly. Without you, this thesis could have never been completed.

Last but not the least, I would like to thank Dr. F. Tao for inspiring discussion and encouragement. I would also thank J. Deschuyter, F. Hubin, X. Radu, S. N. Jha, F. Brah, M. H. Chaudhary, M. E. Soussi, T. E. Bogale, Z. Jin, R. Andreotti and all other members in the lab for the help in research and daily life.

Abstract

This thesis addresses resource allocation (RA) problems in orthogonal frequency division modulation or multiple access (OFDM or OFDMA) systems.

This thesis contains three parts. In the first part, general optimization theory and methods are presented. First, some fundamental optimization theory is reviewed. Next, the successive convex approximation (SCA) method, which iteratively solves convex approximations of an optimization problem, is generalized to the successive approximation (SA) method. Then, general methods are proposed to construct a convex upper bound approximation (CUBA) for a nonconvex function, which is the key to the SCA method. Finally, two interesting applications of the SA and CUBA construction methods, namely the expectation-maximization (EM) algorithm and the successive geometric program (GP) approximation algorithm are given.

In the second part, two RA problems for interference-corrupted OFDM(A) systems are investigated. The first one is a weighted sum rate (WSR) maximized dynamic spectrum management (DSM) problem for point-to-point OFDM systems subject to per transmitter sum power constraints. Specifically, SCA-based DSM algorithms are revisited to show how they can be derived from the SCA and CUBA construction methods. The second one is a RA problem to maximize weighted sum of per cell minimal user rates (WSMR) in a multi-cell OFDMA downlink system subject to per base station (BS) sum power constraints. An iterative algorithm is proposed to jointly optimize BSs' carrier and power allocation alternatively.

In the last part, two RA problems for multiple decode-and-forward (DF) relays aided OFDM(A) systems are investigated. The first one is a sum rate maximized RA problem for a point-to-point OFDM system aided by multiple DF relays subject to per device sum power constraints. Two RA algorithms

have been proposed to find globally optimum and suboptimum RAs, respectively. The second one is a WSR maximized RA problem for an OFDMA downlink system aided by multiple DF relays under a system sum power constraint. A two-step algorithm is proposed to find a globally optimum RA based on a divide-and-conquer strategy.

Contents

Notations and Acronyms	xii
List of Figures	xv
1 Introduction	1
1.1 Motivations	1
1.2 Outline and contributions	3
1.3 Related publications	4
2 Convex sets and functions	7
2.1 Convex sets	7
2.2 Convex or concave functions	8
2.3 Recognition of convex or concave functions	10
3 Optimality conditions and dual method	15
3.1 Overview of this chapter	15
3.2 Necessary conditions for local optimality	16
3.3 Optimality conditions in case of zero duality gap	16
3.4 Analysis of duality gap and sensitivity	19
3.5 Optimality conditions for a convex problem	21
3.6 The dual method	22
4 Successive approximation based optimization method	27
4.1 Related works and chapter overview	27
4.2 The SA method	28
4.3 The SCA and CUBA construction methods	32
4.4 Comparison of the SA method and the DCA	36

4.5	Application in parameter estimation	37
4.6	Application in GP	39
5	Successive convex approximation based DSM algorithms	41
5.1	Related works and chapter overview	41
5.2	System model and DSM problem	42
5.3	Revisit of the SCALE algorithm	44
5.4	Revisit of the CA-DSB algorithm	48
5.5	Revisit of the DCA-DSM algorithm	49
5.6	Numerical experiments	51
5.7	Appendix	53
6	WSMR maximized resource allocation in multi-cell OFDM-based systems	55
6.1	Related works and chapter overview	55
6.2	System model and problem formulation	56
6.3	Algorithm Development	58
6.4	Numerical Experiments	64
7	Sum rate maximized RA in multiple DF relays aided OFDM systems	69
7.1	Related works and chapter overview	69
7.2	System description and RA problem	71
7.3	The duality-based RA algorithm	74
7.4	The iterative RA algorithm	83
7.5	Numerical experiments	89
7.6	Appendix: justifying the zero duality gap of (7.9)	93
8	WSR maximized RA in multiple DF relays aided OFDMA systems	97
8.1	Related works and chapter overview	97
8.2	System description and RA problem	98
8.3	The two-step RA algorithm	101
8.4	The optimum RA in the second step for a special case	110
8.5	Numerical experiments	111
8.6	Appendix: maximization of $R_{u,n,1}$	120

Contents	vii
9 Summary and future work	125
9.1 A summary of the thesis	125
9.2 Perspectives for future work	126
Bibliography	137

Notations and Acronyms

Notations

$\mathcal{R}^N$	N -dimensional real vector domain
$\mathcal{R}_+^N$	domain of N -dimensional entrywise nonnegative vectors
$\mathcal{R}_{++}^N$	domain of N -dimensional entrywise real and positive vectors
x	a real number
$ x $	the absolute value of x
$\mathbf{x}$	a real vector
$\mathcal{C}_{\mathbf{x}}$	a set of vectors $\mathbf{x}$
$[\mathbf{x}]_i$	the i th entry of $\mathbf{x}$
$\nabla_{\mathbf{x}} f(\mathbf{x})$	the first-order gradient of $f(\mathbf{x})$ with respect to $\mathbf{x}$
$\nabla_{\mathbf{x}}^2 f(\mathbf{x})$	the Hessian of $f(\mathbf{x})$ with respect to $\mathbf{x}$
$\ \mathbf{x}\ _m$	the m th order vector norm of $\mathbf{x}$, i.e., $\ \mathbf{x}\ _m = (\sum_i x_i ^m)^{1/m}$
$\mathbf{X}$	a matrix
$[\mathbf{X}]_{i,j}$	the entry at the i th row and j th column of $\mathbf{X}$
$\mathbf{0}$	the zero matrix
$\mathbf{I}$	the identity matrix
$\arg \min_{\mathbf{x} \in \mathcal{C}_{\mathbf{x}}} f(\mathbf{x})$	the set of $\mathbf{x} \in \mathcal{C}_{\mathbf{x}}$ minimizing $f(\mathbf{x})$
$\arg \max_{\mathbf{x} \in \mathcal{C}_{\mathbf{x}}} f(\mathbf{x})$	the set of $\mathbf{x} \in \mathcal{C}_{\mathbf{x}}$ maximizing $f(\mathbf{x})$
$[x]_y^z$	$\max(y, \min(z, x))$
$[x]^+$	$\max(x, 0)$
$e^{\mathbf{x}}$ or $e^{\mathbf{X}}$	entrywise exponential of $\mathbf{x}$ or $\mathbf{X}$
$\mathbb{E}_{p(\mathbf{x})}(f(\mathbf{x}))$	expectation of $f(\mathbf{x})$ over $\mathbf{x}$ with the PDF $p(\mathbf{x})$, i.e., $\mathbb{E}_{p(\mathbf{x})}(f(\mathbf{x})) = \int_{\mathbf{x} \in \mathcal{R}^N} f(\mathbf{x}) p(\mathbf{x}) dx_1 \cdots dx_N$
$(\cdot)^T$	the transposition of a vector or a matrix
$\succ$	entrywise strictly "greater than"
$\succeq$	entrywise "greater than"
$\prec$	entrywise strictly "smaller than"
$\preceq$	entrywise "smaller than"

Acronyms

AF	Amplify and forward (AF)
BS	Base station
CA	coordinate ascent
CA-DSB	convex approximation distributed spectrum balancing
CCI	Cochannel interference
CUBA	Convex upper bound approximation
CSI	Channel state information
DCA	Difference of convex functions algorithm
DF	Decode and forward (DF)
DSL	Digital subscriber lines
DSM	Dynamic spectrum management
EM	Expectation and maximization
GP	Geometric program
Karush-Kuhn-Tucker	KKT
LRP	Lagrangian relaxation problem
MILP	Mixed integer linear programming
MIMO	Multiple input multiple output
MISO	Multiple input single output
ML	Maximum likelihood
OFDM	Orthogonal frequency division modulation
OFDMA	Orthogonal frequency division multiple access
PDF	Probability density function
RA	Resource allocation
SA	Successive approximation
SCA	Successive convex approximation
SCALE	SCA for low complexity
WSMR	Weighted sum of the minimal user rates (WSMR)
WSR	Weighted sum rate

List of Figures

2.1	Illustration of a convex set and a nonconvex set.	7
2.2	Illustration of convex and nonconvex functions over $\mathcal{C}_{\mathbf{x}}$	8
2.3	Illustration of the right-hand side of (2.5) as a global underestimator of $f(\mathbf{x})$ which is convex on $\mathcal{C}_{\mathbf{x}}$	9
2.4	Illustration of the pointwise maximum operator, where the lower border of the grey area represents $f(\mathbf{x}) = \max\{g(\mathbf{x}, \mathbf{y}_1), g(\mathbf{x}, \mathbf{y}_2)\}$	12
3.1	x_1, x_2 and x_3 illustrates the three claims on $\nabla_{\mathbf{x}}f(\mathbf{x}^\circ)$ when $\mathcal{C}_{\mathbf{x}}^f \subseteq \mathcal{R}$ due to the global feature (3.3), respectively.	17
3.2	Illustration of basic duality related concepts.	18
3.3	The visualization of duality gap in the hyperplane of $(\mathbf{p}, w)$, where the gray area represents the set $\mathcal{S}$	20
3.4	Illustration of the ellipsoid method as a generalization of the bisection method to the multi-dimensional case.	25
4.1	Illustration of the idea behind the SA method.	30
5.1	An example of interfering multicarrier systems when $K = 2$. . .	43
5.2	The average WSRs for all algorithms. In each bar group, the bars from the left to the right represent the performance of DCA-DSM, SCALE, CA-DSB, MIWF and ISB algorithms, respectively.	52
6.1	An example system setup.	56
6.2	A random realization of system setup.	65

6.3	The power allocation after optimization with the BSA/ESA initialization for the random realization of channel gains.	65
6.4	The carrier allocation for the first cell after optimization with the BSA/ESA initialization for the random realization of channel gains.	66
6.5	The computed results for the random realization of channel gains. In each bar group, the left and right bars represent the performance for the initial and final RAs, respectively.	66
6.6	Average WSMR over 100 random realizations of channel gains. In each bar group, the left and right bars represent the performance for the initial and final RAs, respectively.	67
7.1	Example patterns of selecting (a) one relay or (b) multiple relays assisting at each relay-aided carrier, where a black block at the i -th row and j -th column indicates the j -th relay r_j assists the transmission at the i -th carrier s_i	70
7.2	Illustration of a relay-aided transmission where r_2 and r_3 assist relaying at carrier n	72
7.3	A DF relays aided OFDM transmission system considered in numerical experiments.	89
7.4	A random realization of channel gains $ h_{s,d}(n) ^2$, $ h_{s,r_i}(n) ^2$, and $ h_{r_i,d}(n) ^2$, $\forall r_i \in \Psi$, $\forall k$	90
7.5	The sum rate for each algorithm when $P_s = P_{r_i}$, $\forall r_i \in \Psi$, and P_s varies from 10 to 50 dBm. In each bar group, the bars from the left to the right represent the performance for the heuristic algorithm, Algorithm 10 and Algorithm 7, respectively.	91
7.6	The RA evaluated by Algorithm 7 for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$	91
7.7	The RA evaluated by Algorithm 10 for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$	92
7.8	The RA evaluated by the heuristic algorithm for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$	92
7.9	The average sum rate for each algorithm when $P_s = P_{r_i}$, $\forall r_i \in \Psi$, and P_s varies from 10 to 50 dBm. In each bar group, the bars from the left to the right represent the performance for the heuristic algorithm, Algorithm 10 and Algorithm 7, respectively.	93

7.10	The visualization of the duality gap in the hyperplane of $(\mathbf{p}, w)$, where the grey area represents $\mathcal{S} = \{(\mathbf{p}, w) \mathbf{p} = g(\mathbf{x}), w = f(\mathbf{x}), \mathbf{x} \in \mathcal{D}_x\}$	94
8.1	The OFDMA system considered in numerical experiments. . . .	112
8.2	(a) $G_{s,u}(n)$ and (b) $G_{u,1}(n)$ computed by the proposed RA algorithm in the first step, at each carrier n for every destination.	113
8.3	When $P_t = 35$ dBm, (a) the visualization of $\mathcal{U}_D(n)$, $\forall k$, where a dot marked at the coordinate (n, u) indicates destination $u \in \mathcal{U}_D(n)$, otherwise $u \notin \mathcal{U}_D(n)$. (b) the visualization of $\mathcal{U}_R(n)$, $\forall k$, where a dot marked at the coordinate (n, u) indicates destination $u \in \mathcal{U}_R(n)$, otherwise $u \notin \mathcal{U}_R(n)$	114
8.4	For the reference and proposed protocols, the optimum RA found in the second step for each carrier n when $P_t = 35$ dBm. . .	115
8.5	The optimum source and relay power in the broadcasting slot for the two protocols when $P_t = 35$ dBm.	116
8.6	The visualization of $\mathcal{U}_R(n)$ for each carrier n when $P_t = 60$ dBm, where a dot marked at the coordinate (n, u) means that destination $u \in \mathcal{U}_R(n)$, otherwise $u \notin \mathcal{U}_R(n)$	116
8.7	For the reference and proposed protocols, the optimum RA found in the second step for each carrier n when $P_t = 60$ dBm and destinations have equal weights.	117
8.8	The optimum source and relay power in the broadcasting slot for the reference protocol when $P_t = 60$ dBm. Note that for the proposed protocol, the optimum sum power at each carrier n is equally allocated to $P_{s_1,u}(n)$ and $P_{s_2,u}(n)$, which is not shown here.	118
8.9	The optimum WSRs for the first 100 channel realizations for the reference and proposed protocols, and the average WSR for all channel realizations. In each bar group, the left and right bars represent the performance for the reference and proposed protocols, respectively.	119
8.10	Illustrations of $\gamma'_{u,c}(n)$, $\gamma_r(n)$, and $\gamma'_u(n)$ versus $P_{s,u}(n)$. $\gamma'_u(n)$ is highlighted by painting the area it bounds in light grey. . . .	123

Introduction

1

This chapter first explains motivations behind this thesis, then shows thesis outline and contributions, and finally lists related publications.

1.1 Motivations

Orthogonal frequency division modulation (OFDM) transforms a frequency-selective fading channel into parallel flat-fading channels, each at a different carrier. When used in cellular systems, orthogonal frequency division multiple access (OFDMA) can be adopted to eliminate intra-cell interference, i.e., each carrier of a base station (BS) is exclusively allocated to a user in the cell served by that BS. The OFDM(A) transmission naturally enables use of vast flexibility coming from multiple carriers and users for performance improvement. Therefore, OFDM(A) transmission has not only been widely accepted in many current wireless communication standards such as Wireless Fidelity (WiFi), but is also a strong candidate for future wireless communication systems [FK88, PH09].

When channel state information (CSI) is available at the OFDM(A) transmitters, the key to realizing performance improvement is dynamic resource allocation (RA) i.e., optimizing carrier and power allocations to improve certain performance metrics. In the past decade, plenty of RA problems have been proposed and solved for single-cell OFDMA systems. In general, those problems can be classified into two categories. One is margin-adaptive which minimizes the sum power consumption subject to prescribed rate requirements for users [WCLM99]. The other is rate-adaptive which maximizes the sum rate subject to a sum power constraint [JL03]. In addition, application-dependent

user-rate constraints may be prescribed [YL00,ZL04]. When there are no such constraints, proportional rate constraints can be imposed to guarantee proportional fairness among users [SAE05,SVL05,SLV08]. To achieve the maximum fairness among users, a max-min problem can be solved to provide similar rates to users [RC00,KPL06].

In this thesis, RA for the following two types of OFDM(A) systems is investigated:

■ Interference-corrupted OFDM systems

Specifically, two kinds of interference corrupted OFDM systems are considered. The first has multiple pairs of transmitter and receiver communicating simultaneously with OFDM. The other is a multi-cell OFDM system serving multiple users. The performance of these systems is impaired by cochannel interference (CCI), i.e., the transmission at a carrier to a user is interfered by that at the same carrier to another user.

A traditional way of mitigating CCI is to use frequency division multiplexing (FDM), i.e., adjacent users use nonoverlapping subsets of carriers. Pushed by increasing demand of higher data rate wireless services, full spectrum reuse is envisioned in next-generation systems.

Lately, two CCI mitigation methods have been attracting intensive research interest. The first is to use signal coordination, i.e., all users' data are first encoded and then transmitted simultaneously at each carrier [GKH⁺07]. This method in effect establishes multiple input and multiple output (MIMO) links in a distributed way to increase data rate. However, all users' data need to be collected at a central controller. Moreover, new signal processing hardware and algorithms have to be implemented. The second method is to use joint RA, also called dynamic spectrum management (DSM) for point-to-point OFDM systems, to increase data rate. This leads to an easier implementation in current systems [GKGb07]. Chapters 5-6 of this thesis propose joint RA algorithms for CCI mitigation in the above mentioned interference corrupted OFDM systems.

■ Relay-aided OFDM(A) systems

Relay-aided cooperative transmission finds plenty of promising applications when it is difficult to install multiple antennas at the same radio equipment, and therefore has been attracting intensive research interest in both

academia and industry lately [PWS⁺04]. From the information theory point of view, the capacity bounds for a general three-node relay channel were studied in [vdM71, CG79]. A full-duplex relaying based communication system has been studied in [SEA03].

Half-duplex relaying based transmission protocols, such as amplify and forward (AF) as well as decode and forward (DF), have been proposed and become the focus of recent research works, since they can simplify the implementation with practical half-duplex devices that can not transmit and receive simultaneously [LW03, LTW04]. Typically, both protocols propose to carry out a relay-aided transmission within two time slots, namely a broadcasting slot and a relaying slot. In [LTW04], the AF/DF which fixes every transmission in the relay-aided mode independently of source-relay channel conditions is referred to as fixed relaying AF/DF. In fact, improved protocols may be built for better performance. For instance, selection relaying AF/DF, which selects either the direct or relay-aided transmission mode depending on channel conditions, has been proposed to improve spectral efficiency [LTW04]. In particular, the direct mode, which refers to the direct source to destination transmission without any relay assisting, is used when the source-destination channel gain is higher than the source-relay channel gain. Most interestingly, not fixed but selection relaying DF can achieve full diversity [LTW04].

Compared to conventional OFDM(A) transmission, the relay-aided OFDMA transmission raises more complicated RA problems, since it introduces extra tasks such as deciding the transmission mode of each carrier, determining assisting relays and the power allocation to them for each relay-aided carrier, besides assigning source power (and destination for OFDMA) at each carrier. Therefore, novel efficient RA algorithms are solicited for relay-aided OFDM(A) transmission systems. Chapters 7-8 of this thesis propose novel RA algorithms for DF relays aided OFDM(A) systems.

1.2 Outline and contributions

In Chapter 2, definitions of convex sets and functions, as well as methods to recognize a convex function are briefly reviewed.

In Chapter 3, optimality conditions and the dual method are presented for an optimization problem in general form.

In Chapter 4, the successive convex approximation (SCA) method is generalized to successive approximation (SA) method for solving a general optimization problem. General methods are proposed to construct for a non-convex function a convex upper bound approximation (CUBA) that satisfies tightness and differentiation conditions, which is the key to the SCA method. To illustrate the effectiveness of the SA method, two interesting applications for parameter estimation and geometric programming are shown.

In Chapter 5, a weighted sum rate (WSR) maximized DSM problem is addressed to mitigate CCI in point-to-point OFDM systems subject to per transmitter sum power constraints. Specifically, SCA-based DSM algorithms are revisited to show how they can be derived from the SCA and CUBA construction methods introduced in Chapter 4.

In Chapter 6, a weighted sum of per cell minimal user rates (WSMR) maximized DSM problem is addressed to mitigate CCI in a multi-cell OFDM downlink system subject to per BS sum power constraints. A RA algorithm is proposed to jointly optimize BSs' carrier and power allocation alternatively.

In Chapter 7, a sum rate maximized RA problem is addressed for a point-to-point OFDM system aided by multiple DF relays subject to per device sum power constraints. Two RA algorithms are proposed to find the globally optimum RA.

In Chapter 8, a WSR maximized RA problem is addressed under a system sum power constraint for an OFDMA downlink transmission system aided by multiple DF relays. A two-step RA algorithm is proposed to find the globally optimum RA.

In Chapter 9, a summary and perspectives for future work are presented.

1.3 Related publications

The content of Chapter 4 and 5 has been published or submitted as

- [WV11c] T. Wang and L. Vandendorpe, *Successive convex approximation based methods for dynamic spectrum management*, to be presented in IEEE ICC 2012, Sep. 2011.
- [WV11b] T. Wang and L. Vandendorpe, *On the SCALE algorithm for multiuser multicarrier power spectrum management*, submitted to IEEE Trans. Signal Processing, Jun. 2011.

The content of Chapter 6 has been published in

- [WV11a] T. Wang and L. Vandendorpe, *Iterative resource allocation for maximizing weighted sum min-rate in downlink cellular OFDMA systems*, IEEE Trans. Sig. Proc., vol. 59, no. 1, p.p. 223-234, Jan. 2011.
- [WV10c] T. Wang and L. Vandendorpe, *Resource Allocation for Maximizing Weighted Sum Min-Rate in Downlink Cellular OFDMA Systems*, IEEE ICC 2010, p.p. 1-6, May 2010.
- [WV10b] T. Wang and L. Vandendorpe, *Iterative Power and Subcarrier Allocation for Maximizing WSMR in Cellular OFDMA Systems*, Proc. RAWNET, p.p. 324-329, Jan. 2010.
- [WV10a] T. Wang and L. Vandendorpe, *Coordinate ascent based resource allocation algorithm for maximizing WSMR in downlink cellular OFDMA systems*, Proc. PIMRC, p.p. 380-385, Sep. 2010.

The content of Chapter 7 has been published in

- [WV11d] T. Wang and L. Vandendorpe, *Sum rate maximized resource allocation in multiple DF relays aided OFDM transmission*, IEEE Journal on Selected Areas in Communications, vol. 29, no. 8, p.p. 1559-1571, Sep. 2011.

The content of Chapter 8 has been published in

- [WV11e] T. Wang and L. Vandendorpe, *WSR maximized resource allocation in multiple DF relays aided OFDMA transmission*, IEEE Trans. Sig. Proc., vol. 59, no. 8, p.p. 3964-3976, Aug. 2011.
- [WV11f] T. Wang and L. Vandendorpe, *A two-Step RA algorithm for multiple DF relays aided OFDMA downlink transmission*, Proc. Joint NEWCOM++/COST 2100 Workshop, Mar. 2011.

Convex sets and functions

2

This chapter briefly reviews definitions of convex sets and functions, and methods to recognize a convex function.

2.1 Convex sets

A set $\mathcal{C}_x \subseteq \mathcal{R}^N$ is convex if

$$\alpha \mathbf{x}_1 + \beta \mathbf{x}_2 \in \mathcal{C}_x \quad (2.1)$$

always holds $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{C}_x$ and $\forall \alpha, \beta \in [0, 1]$ satisfying $\alpha + \beta = 1$.

Geometrically speaking, $\mathcal{C}_x$ is convex if the line segment between any two points in $\mathcal{C}_x$ is still contained in $\mathcal{C}_x$. A convex set and a nonconvex set are illustrated in Figure 2.1.

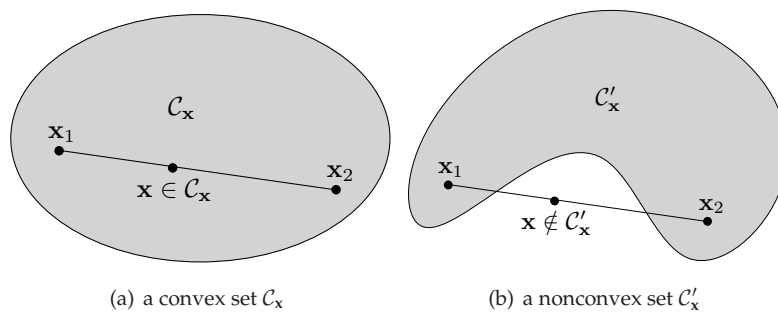


Figure 2.1 Illustration of a convex set and a nonconvex set.

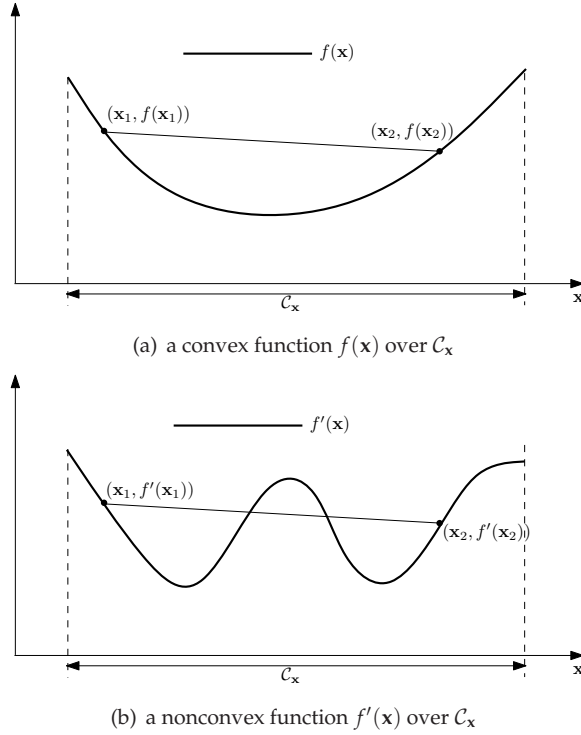


Figure 2.2 Illustration of convex and nonconvex functions over $\mathcal{C}_{\mathbf{x}}$.

2.2 Convex or concave functions

A function $f(\mathbf{x})$ is convex over a convex set $\mathcal{C}_{\mathbf{x}} \subseteq \mathcal{R}^N$ if

$$\alpha f(\mathbf{x}_1) + \beta f(\mathbf{x}_2) \geq f(\alpha \mathbf{x}_1 + \beta \mathbf{x}_2) \quad (2.2)$$

always holds $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{C}_{\mathbf{x}}$ and $\forall \alpha, \beta \in [0, 1]$ satisfying $\alpha + \beta = 1$. In particular, $f(\mathbf{x})$ is strictly convex over $\mathcal{C}_{\mathbf{x}}$ if strict inequality holds in (2.2), $\forall 0 < \alpha, \beta < 1$. Geometrically speaking, $f(\mathbf{x})$ is convex over $\mathcal{C}_{\mathbf{x}}$ if the line segment between any two points $(\mathbf{x}_1, f(\mathbf{x}_1))$ and $(\mathbf{x}_2, f(\mathbf{x}_2))$ is always above the curve $(\mathbf{x}, f(\mathbf{x}))$ when $\mathbf{x}$ runs from $\mathbf{x}_1$ to $\mathbf{x}_2$. A convex function and a nonconvex function are illustrated in Figure 2.2. $f(\mathbf{x})$ is (strictly) concave over $\mathcal{C}_{\mathbf{x}}$ if $-f(\mathbf{x})$ is (strictly) convex over $\mathcal{C}_{\mathbf{x}}$.

Properties of a convex function have been studied thoroughly in Chapter 3 of [BV04]. Some of them that will be used later are listed as below:

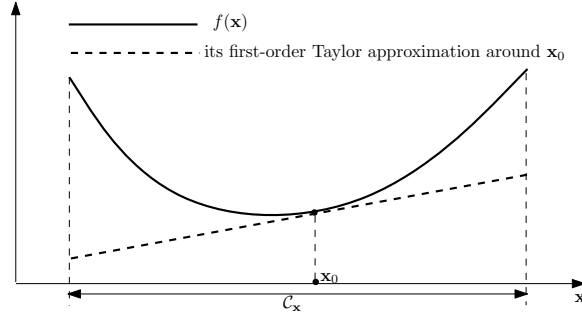


Figure 2.3 Illustration of the right-hand side of (2.5) as a global underestimator of $f(\mathbf{x})$ which is convex on $\mathcal{C}_x$.

1. **Jensen's inequality.** Specifically, $\forall \mathbf{x}_1, \dots, \mathbf{x}_M \in \mathcal{C}_x$,

$$\sum_{m=1}^M \theta_m f(\mathbf{x}_m) \geq f\left(\sum_{m=1}^M \theta_m \mathbf{x}_m\right) \quad (2.3)$$

holds $\forall \theta_1, \dots, \theta_M \in [0, 1]$ satisfying $\sum_{m=1}^M \theta_m = 1$. More generally, when $\mathbf{x}$ is a random vector,

$$\mathbb{E}_{p(\mathbf{x})}(f(\mathbf{x})) \geq f(\mathbb{E}_{p(\mathbf{x})}(\mathbf{x})) \quad (2.4)$$

holds for any probability density function (PDF) $p(\mathbf{x})$ over $\mathcal{C}_x$, where $\mathbb{E}_{p(\mathbf{x})}(\cdot)$ represents the expectation with $p(\mathbf{x})$. When $f(\mathbf{x})$ is strictly convex, the inequality in (2.3) holds strictly when $\forall m, 0 < \theta_m < 1$. This property is a generalization of (2.2) used to define a convex function. Obviously, the inequality is tightened when $f(\mathbf{x})$ is equal to a constant independently of $\mathbf{x}$.

2. **Global bound property.** Suppose $f(\mathbf{x})$ is differentiable over $\mathcal{C}_x$, then it is convex over $\mathcal{C}_x$ if and only if

$$\forall \mathbf{x}', \mathbf{x} \in \mathcal{C}_x, f(\mathbf{x}') \geq f(\mathbf{x}) + (\nabla_{\mathbf{x}} f(\mathbf{x}))^T (\mathbf{x}' - \mathbf{x}) \quad (2.5)$$

holds. Moreover, $f(\mathbf{x})$ is strictly convex over $\mathcal{C}_x$ if and only if the inequality in (2.5) holds strictly when $\mathbf{x} \neq \mathbf{x}'$.

In other words, $f(\mathbf{x})$ is convex over $\mathcal{C}_x$ if and only if its first-order Taylor approximation around any point in $\mathcal{C}_x$ is a *global* underestimator of itself, or

equivalently $\nabla_{\mathbf{x}}f(\mathbf{x})$ is a subgradient¹ of $f(\mathbf{x})$ with respect to $\mathbf{x} \in \mathcal{C}_{\mathbf{x}}$. This property is illustrated in Figure 2.3.

3. Hessian related property. Suppose $f(\mathbf{x})$ is twice differentiable over $\mathcal{C}_{\mathbf{x}}$, then it is convex over $\mathcal{C}_{\mathbf{x}}$ if and only if

$$\forall \mathbf{x} \in \mathcal{C}_{\mathbf{x}}, \nabla_{\mathbf{x}}^2 f(\mathbf{x}) \succeq \mathbf{0} \quad (2.6)$$

holds. Moreover, $f(\mathbf{x})$ is strictly convex if the inequality in (2.6) holds strictly. It is important to that the converse, however, is not true. For instance, the function $f(x) = x^4$ defined over $x \in \mathcal{R}$ is strictly convex of x , but $\nabla_x^2 f(x) = 0$ when $x = 0$.

The corresponding properties of a concave function can also be described as in the above after every word “convex”, “underestimator”, “ $\geq$ ” in (2.3), (2.4) and (2.5) and the “ $\succeq$ ” in (2.6) are replaced with “concave”, “overestimator”, “ $\leq$ ” and “ $\preceq$ ”, respectively.

2.3 Recognition of convex or concave functions

To examine if a given function is convex (respectively, concave), we can check if it satisfies the differentiation or Hessian related properties. Except for a few functions with simple structures, these methods often lead to very involved and demanding mathematical procedures, not to mention that they are only applicable to differentiable functions.

In practice, a more efficient way to recognize a convex (resp., concave) function is to check if that function can be transformed from certain convex or concave functions through operations that preserve the convexity (resp., concavity). Therefore, it would be very helpful to get familiar with typical convex or concave functions as well as convexity (resp., concavity) preserving operations, in order to recognize convex (resp., concave) functions efficiently.

2.3.1 Typical convex or concave functions

A number of typical convex or concave functions are listed in Chapter 3 of [BV04]. A few of them in their original or generalized forms that will be used in this thesis, are summarized as follows:

¹ For a convex (respectively, concave) function $g(\mathbf{x})$ defined over $\mathbf{x} \in \mathcal{C}_{\mathbf{x}}$, $\mathbf{s}(\mathbf{x})$ is a subgradient with respect to $\mathbf{x}$ if $\forall \mathbf{x}' \in \mathcal{C}_{\mathbf{x}}, g(\mathbf{x}') \geq g(\mathbf{x}) + (\mathbf{s}(\mathbf{x}))^T(\mathbf{x}' - \mathbf{x})$ (respectively, $g(\mathbf{x}') \leq g(\mathbf{x}) + (\mathbf{s}(\mathbf{x}))^T(\mathbf{x}' - \mathbf{x})$).

1. **Affine function**, i.e.,

$$f(\mathbf{x}) = \mathbf{a}^T \mathbf{x} + b \quad (2.7)$$

is convex over $\mathcal{R}^N$, $\forall \mathbf{a} \in \mathcal{R}^N, \forall b \in \mathcal{R}$. Most interestingly, this function is also concave over $\mathcal{R}^N$.

2. **Exponential function**, i.e.,

$$f(x) = e^x \quad (2.8)$$

is convex over $\mathcal{R}^1$.

3. **Logarithm function**, i.e.,

$$f(x) = \log(x) \quad (2.9)$$

is concave over $\mathcal{R}_{++}^1$.

4. **Vector norm function of m th order** ($m = 1, 2, \dots$), i.e.,

$$f(\mathbf{x}) = \|\mathbf{x}\|_m = \left(\sum_{n=1}^N |x_n|^m \right)^{1/m}, \quad (2.10)$$

is convex over $\mathcal{R}^N$.

5. **Log-weighted-sum-exponential function**, i.e.,

$$f(\mathbf{x}) = \log\left(z + \sum_{n=1}^N d_n e^{x_n}\right) \quad (2.11)$$

is convex over $\mathcal{R}^N$ if $\forall n, d_n \geq 0$ and $z \geq 0$.

When $\forall n, d_n = 1$ and $z = 0$, the convexity of $f(\mathbf{x})$ has been proved in Page 74 of [BV04]. For $f(\mathbf{x})$ expressed in the general form (2.11), its convexity was proved in [WV10c, WV11a] by verifying that $\nabla_{\mathbf{x}}^2 f(\mathbf{x}) \succeq \mathbf{0}$, $\forall \mathbf{x} \in \mathcal{R}^N$. Moreover, it was shown there $f(\mathbf{x})$ is strictly convex when $z > 0$ and $d_n > 0$, $\forall n$.

2.3.2 Typical convexity- or concavity-preserving operations

A number of typical convexity or concavity preserving operations are listed in Chapter 3 of [BV04]. A few of them in their original or generalized forms that will be used in this thesis consist of:

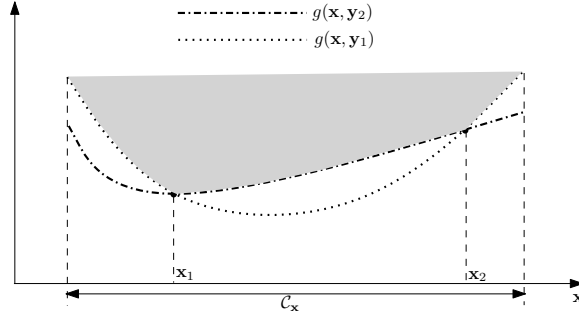


Figure 2.4 Illustration of the pointwise maximum operator, where the lower border of the grey area represents $f(\mathbf{x}) = \max\{g(\mathbf{x}, \mathbf{y}_1), g(\mathbf{x}, \mathbf{y}_2)\}$.

1. **Weighted summation.** Specifically,

$$f(\mathbf{x}) = \sum_i w_i f_i(\mathbf{x}) \quad (2.12)$$

is convex (resp., concave) over $\mathcal{C}_x$ if $\forall i$, one of the following two conditions is satisfied:

- $w_i \geq 0$ and $f_i(\mathbf{x})$ is convex (resp., concave) over $\mathcal{C}_x$,
- $w_i \leq 0$ and $f_i(\mathbf{x})$ is concave (resp., convex) over $\mathcal{C}_x$.

2. **Pointwise maximum (resp., minimum).** Specifically,

$$f(\mathbf{x}) = \max_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y}) \quad (\text{resp., } f(\mathbf{x}) = \min_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y})) \quad (2.13)$$

is convex (resp., concave) over $\mathcal{C}_x$ if $\forall \mathbf{y} \in \mathcal{C}_y$, $g(\mathbf{x}, \mathbf{y})$ is convex (resp., concave) over $\mathcal{C}_x$. Note that $\mathcal{C}_y$ can be either continuous or discrete, and $g(\mathbf{x}, \mathbf{y})$ can be nondifferentiable. The proof of this claim is given in Page 80 of [BV04]. A caveat deserving attention is that $f(\mathbf{x})$ might be nondifferentiable over $\mathcal{C}_x$ even if $\forall \mathbf{y} \in \mathcal{C}_y$, $g(\mathbf{x}, \mathbf{y})$ is differentiable over $\mathcal{C}_x$.

This operation is illustrated in Figure 2.4, where $f(\mathbf{x}) = \max\{g(\mathbf{x}, \mathbf{y}_1), g(\mathbf{x}, \mathbf{y}_2)\}$ is convex over $\mathcal{C}_x$, but nondifferentiable at $\mathbf{x}_1$ and $\mathbf{x}_2$ even though the two functions inside the max operator are both differentiable over $\mathcal{C}_x$. For this operation, an important property that will be used later is given as follows:

Corollary 2.1. Suppose $\forall \mathbf{y}$, $g(\mathbf{x}, \mathbf{y})$ is convex (resp., concave) and differentiable of $\mathbf{x} \in \mathcal{C}_x$. Then $\nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{y}_x)$ is a subgradient of the convex function $f(\mathbf{x}) = \max_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y})$ (resp., the concave function $f(\mathbf{x}) = \min_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y})$) with respect to $\mathbf{x} \in \mathcal{C}_x$, where $\mathbf{y}_x \in \arg \max_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y})$ (resp., $\mathbf{y}_x \in \arg \min_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}, \mathbf{y})$).

This corollary in the convex case is true because $\forall \mathbf{x}' \in \mathcal{C}_x$,

$$f(\mathbf{x}') = g(\mathbf{x}', \mathbf{y}_{\mathbf{x}'}) \geq g(\mathbf{x}', \mathbf{y}_{\mathbf{x}}) \geq f(\mathbf{x}) + (\mathbf{x}' - \mathbf{x})^T \nabla_{\mathbf{x}} g(\mathbf{x}, \mathbf{y}_{\mathbf{x}}) \quad (2.14)$$

holds, where the first inequality is because $\mathbf{y}_{\mathbf{x}'} = \arg \max_{\mathbf{y} \in \mathcal{C}_y} g(\mathbf{x}', \mathbf{y})$, and the second one is due to the global bound property of $g(\mathbf{x}, \mathbf{y}_{\mathbf{x}})$ with respect to $\mathbf{x}$. The concave case can easily be proved in a similar way.

Note that a generalization of this corollary is Danskin's Theorem (see Page 737 in [Ber03]), where a directional derivative that is more general than the subgradient is considered. Nevertheless, this corollary is simpler and thus more readily used for the analysis shown later.

3. Composition of functions. Specifically,

$$f(\mathbf{x}) = h(z_1(\mathbf{x}), \dots, z_L(\mathbf{x})) \quad (2.15)$$

is convex (resp., concave) over $\mathcal{C}_x$, when $h(\mathbf{z})$ is convex (resp., concave) over $\mathcal{R}^L$ and $\forall l$, one of the following conditions is satisfied:

- $h(\mathbf{z})$ is increasing of z_l and $z_l(\mathbf{x})$ is convex (resp., concave) in $\mathcal{C}_x$,
- $h(\mathbf{z})$ is decreasing of z_l and $z_l(\mathbf{x})$ is concave (resp., convex) in $\mathcal{C}_x$.

The method in the convex case is valid because

$$\begin{aligned} & h(z_1(\alpha \mathbf{x}_1 + \beta \mathbf{x}_2), \dots, z_L(\alpha \mathbf{x}_1 + \beta \mathbf{x}_2)) \\ & \leq h(\alpha z_1(\mathbf{x}_1) + \beta z_1(\mathbf{x}_2), \dots, \alpha z_L(\mathbf{x}_1) + \beta z_L(\mathbf{x}_2)) \\ & \leq \alpha h(z_1(\mathbf{x}_1), \dots, z_L(\mathbf{x}_1)) + \beta h(z_1(\mathbf{x}_2), \dots, z_L(\mathbf{x}_2)), \end{aligned} \quad (2.16)$$

holds $\forall \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{C}_x$ and $\forall \alpha, \beta \in [0, 1]$ satisfying $\alpha + \beta = 1$, where the first inequality is due to the convexity or concavity of every $z_l(\mathbf{x})$ together with the monotonicity of $h(\mathbf{z})$, while the second one is because of the convexity of $h(\mathbf{z})$. The concave case can be proved in a similar way.

Note that this operation is a simple generalization of those introduced in [BV04], which only consider the cases where $h(\mathbf{z})$ is uniformly increasing or decreasing of all $z_1, \dots, z_L$, and meanwhile all $z_1(\mathbf{x}), \dots, z_L(\mathbf{x})$ are uniformly convex or concave of $\mathbf{x}$.

4. Perspective functions. Specifically, when $f(\mathbf{x})$ is convex (resp. concave) over $\mathcal{C}_x$,

$$g(\mathbf{y}, t) = t f\left(\frac{\mathbf{y}}{t}\right) \quad (2.17)$$

named as a perspective of $f(\mathbf{x})$ with respect to $t > 0$, is convex (resp. concave) over the set $\{(\mathbf{y}, t) \in \mathcal{R}^{N+1} | \frac{\mathbf{y}}{t} \in \mathcal{C}_x, t > 0\}$. Note that this property is the key to developing a RA algorithm in [WV11e, WV11f].

Optimality conditions and dual method

3

3.1 Overview of this chapter

This chapter briefly reviews optimality conditions and dual method to solve the problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{C}_{\mathbf{x}} \subseteq \mathcal{R}^N, \\ & f_j(\mathbf{x}) \leq 0, j = 1, \dots, J, \end{aligned} \tag{3.1}$$

which is feasible and has a finite optimum objective value. Note that $\mathcal{C}_{\mathbf{x}}$ is a subset of the definition domain of $f_j(\mathbf{x})$, $\forall j = 0, 1, \dots, J$. When $\mathcal{C}_{\mathbf{x}}$ is convex and $\forall j = 0, 1, \dots, J$, $f_j(\mathbf{x})$ is convex, (3.1) is a convex problem.

The Lagrangian of (3.1) is defined as

$$L(\boldsymbol{\mu}, \mathbf{x}) = f_0(\mathbf{x}) + \sum_{j=1}^J \mu_j f_j(\mathbf{x}) \tag{3.2}$$

where $\boldsymbol{\mu} = [\mu_1, \dots, \mu_J]^T \in \mathcal{R}_+^J$ with $\mu_j \geq 0$ being the dual variable, also named as the Lagrange multiplier for the j th constraint.

In this chapter, necessary conditions are first introduced for a local optimum of (3.1). Next, global optimality conditions that are both sufficient and necessary for (3.1) are derived when the problem has a zero duality gap. Then, the duality gap of (3.1) is studied. After that, sufficient and necessary global optimality conditions when (3.1) is convex are summarized. Finally, the dual method is introduced to find a global optimum for (3.1) when it has zero duality gap.

3.2 Necessary conditions for local optimality

First, the Karush-Kuhn-Tucker (KKT) conditions that are necessary for a local optimum of (3.1) are presented as follows:

Theorem 3.1 (Proposition 3.3.1 in [Ber03]). *Suppose $\mathcal{C}_x = \mathcal{R}^N$ and $\forall j, f_j(\mathbf{x})$ is continuously differentiable of $\mathbf{x} \in \mathcal{R}^N$ for (3.1). Let $\mathbf{x}^\circ$ be locally optimum and regular, i.e., all inequality constraints are satisfied strictly at $\mathbf{x}^\circ$. Then it satisfies the KKT conditions, i.e., there exists $\boldsymbol{\mu}' = [\mu'_1, \dots, \mu'_J]^T \succeq \mathbf{0}$, together with $\mathbf{x}^\circ$ satisfying simultaneously:*

- feasibility condition: $\mathbf{x}^\circ$ is feasible for (3.1),
- first-order differentiation condition: $\nabla_{\mathbf{x}} L(\boldsymbol{\mu}', \mathbf{x}^\circ) = \mathbf{0}$,
- complementary slackness condition: $\forall j \geq 1, \mu'_j f_j(\mathbf{x}^\circ) = 0$.

Let us now consider the case where $\mathcal{C}_x^f$, representing the feasible set of (3.1), is convex and $f_0(\mathbf{x})$ is differentiable over $\mathcal{C}_x^f$. In this case, a necessary condition for local optimality is given as follows:

Theorem 3.2 (Proposition 2.1.2 in [Ber03]). *Suppose $\mathcal{C}_x^f$ is convex and $f_0(\mathbf{x})$ is differentiable over $\mathcal{C}_x^f$. Then, a local optimum $\mathbf{x}^\circ$ for (3.1) must satisfy that*

$$\forall \mathbf{x} \in \mathcal{C}_x^f, (\nabla_{\mathbf{x}} f_0(\mathbf{x}^\circ))^T (\mathbf{x} - \mathbf{x}^\circ) \geq 0, \quad (3.3)$$

This theorem indicates a powerful global feature for a local optimum of (3.1) when it has a convex feasible set. Thanks to this feature, the following claims are true:

- Suppose $\mathbf{x}^\circ$ is in the interior of $\mathcal{C}_x^f$, then $\nabla_{\mathbf{x}} f(\mathbf{x}^\circ) = \mathbf{0}$ must hold.
- Suppose $\forall \mathbf{x} \in \mathcal{C}_x^f, \mathbf{x} \succeq \mathbf{x}^\circ$, then $\nabla_{\mathbf{x}} f(\mathbf{x}^\circ) \succeq \mathbf{0}$ must hold.
- Suppose $\forall \mathbf{x} \in \mathcal{C}_x^f, \mathbf{x} \preceq \mathbf{x}^\circ$, then $\nabla_{\mathbf{x}} f(\mathbf{x}^\circ) \preceq \mathbf{0}$ must hold.

These claims can be proved by contradiction method, i.e. suppose a given condition on $\nabla_{\mathbf{x}} f_0(\mathbf{x}^\circ)$ is not satisfied, then there always exists a $\mathbf{x} \in \mathcal{C}_x^f$ that does not satisfy (3.3). When $\mathcal{C}_x^f \subseteq \mathcal{R}$, these claims are illustrated in Figure 3.1. It is very important to note that the global feature in (3.3) might *not* remain once $\mathcal{C}_x^f$ becomes nonconvex.

3.3 Optimality conditions in case of zero duality gap

Most interestingly, a global optimum of (3.1) must satisfy conditions that are both necessary and sufficient when (3.1) has zero duality gap. In the following, the duality gap related basic concepts and analysis are first introduced. These concepts and analysis are very important since they lead to the

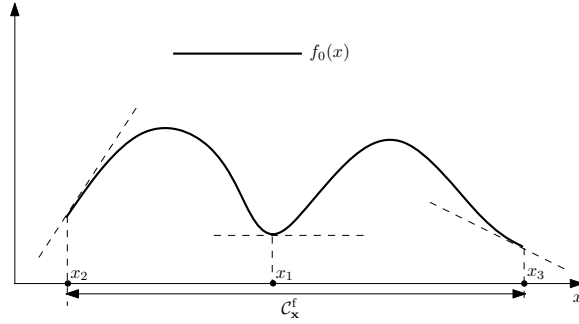


Figure 3.1 x_1, x_2 and x_3 illustrates the three claims on $\nabla_x f(x^o)$ when $C_x^f \subseteq \mathcal{R}$ due to the global feature (3.3), respectively.

dual method shown in Section 3.6. Then, the global optimality conditions are presented.

3.3.1 Basic concepts and analysis

Let's denote a global optimum for (3.1) as $\mathbf{x}^*$. The Lagrangian relaxation problem (LRP) of (3.1) is defined as the problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & L(\boldsymbol{\mu}, \mathbf{x}) = f_0(\mathbf{x}) + \sum_{j=1}^J \mu_j f_j(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in C_{\mathbf{x}}. \end{aligned} \quad (3.4)$$

The dual function $d(\boldsymbol{\mu})$ is defined as the optimum objective value of (3.4), i.e.,

$$d(\boldsymbol{\mu}) = \min_{\mathbf{x} \in C_{\mathbf{x}}} L(\boldsymbol{\mu}, \mathbf{x}) = L(\boldsymbol{\mu}, \mathbf{x}_{\boldsymbol{\mu}}), \quad (3.5)$$

where $\mathbf{x}_{\boldsymbol{\mu}}$ represents a global optimum for (3.4) when $\boldsymbol{\mu}$ is fixed. The dual problem is defined as

$$\begin{aligned} \max_{\boldsymbol{\mu}} \quad & d(\boldsymbol{\mu}), \\ \text{s.t.} \quad & \boldsymbol{\mu} \succeq \mathbf{0}. \end{aligned} \quad (3.6)$$

It can be easily seen that the dual problem is equivalent to a convex problem, since its feasible set is convex and $d(\boldsymbol{\mu})$ is concave of $\boldsymbol{\mu} \in \mathcal{R}^J$. This is because $d(\boldsymbol{\mu})$ is a pointwise minimum over $\mathbf{x} \in C_{\mathbf{x}}$ of $L(\boldsymbol{\mu}, \mathbf{x})$ that can be regarded as a concave function of $\boldsymbol{\mu} \in \mathcal{R}^J$ when $\mathbf{x}$ is fixed.

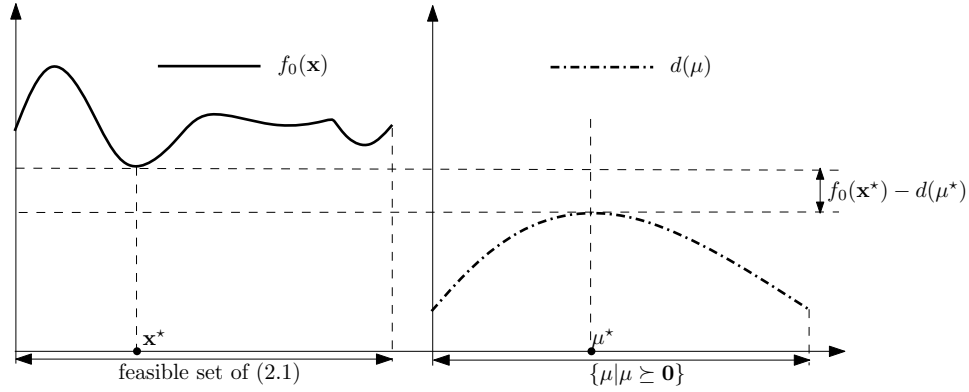


Figure 3.2 Illustration of basic duality related concepts.

The duality gap of (3.1) is defined as $f_0(\mathbf{x}^*) - d(\boldsymbol{\mu}^*)$, where $\boldsymbol{\mu}^* = [\mu_1^*, \dots, \mu_J^*]^T$ represents a global optimum for (3.6). Most importantly, the duality gap is always nonnegative because of

$$d(\boldsymbol{\mu}^*) = L(\boldsymbol{\mu}^*, \mathbf{x}_{\boldsymbol{\mu}^*}) \leq L(\boldsymbol{\mu}^*, \mathbf{x}^*) = f_0(\mathbf{x}^*) + \sum_{j=1}^J \mu_j^* f_j(\mathbf{x}^*) \leq f_0(\mathbf{x}^*), \quad (3.7)$$

where the first inequality is because $\mathbf{x}_{\boldsymbol{\mu}^*}$ and $\mathbf{x}^*$ are globally optimum and feasible for (3.4), respectively. The second inequality is because of $\forall j, \mu_j^* \geq 0$ and $f_j(\mathbf{x}^*) \leq 0$ since $\mathbf{x}^*$ is feasible for (3.1). Note that $\forall \mathbf{x} \in \mathcal{C}_x^f$ and $\forall \boldsymbol{\mu} \succeq \mathbf{0}$, $d(\boldsymbol{\mu}) \leq d(\boldsymbol{\mu}^*) \leq f_0(\mathbf{x}^*) \leq f_0(\mathbf{x})$ holds.

These concepts are illustrated in Figure 3.2 intuitively. As will be shown later, the inequality in (3.7) is the key for developing the dual method in Section 3.6.

3.3.2 Global optimality conditions

When (3.1) has zero duality gap, global optimality conditions that are both necessary and sufficient are given as follows:

Theorem 3.3 (Proposition 5.1.5 in [Ber03]). *Suppose (3.1) has zero duality gap. $\mathbf{x}'$ and $\boldsymbol{\mu}'$ are globally optimum for (3.1) and (3.6), respectively, if and only if they satisfy the following conditions:*

- *feasibility condition: $\mathbf{x}'$ is feasible for (3.1) and $\boldsymbol{\mu}' \succeq \mathbf{0}$,*
- *Lagrangian optimality condition: $\mathbf{x}' \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{C}_x} L(\boldsymbol{\mu}', \mathbf{x})$,*
- *complementary slackness condition: $\forall j \geq 1, \mu_j' f_j(\mathbf{x}') = 0$.*

3.4 Analysis of duality gap and sensitivity

This section introduces a visualization method to study the duality gap of (3.1), the Slater constraint qualification to justify zero duality gap of a convex problem, and the sensitivity analysis. The visualization method and the sensitivity analysis are the keys to the duality based RA algorithm proposed in [WV11d].

3.4.1 Duality gap analysis

For a general (not necessarily convex) problem (3.1), its duality gap can be studied with a visualization method proposed in [Ber03]. To introduce this method, all constraint functions are stacked into a vector function

$$\mathbf{f}(\mathbf{x}) = [f_1(\mathbf{x}), \dots, f_J(\mathbf{x})]^T. \quad (3.8)$$

Then, a cloud of points in the set $\mathcal{S} = \{(\mathbf{p}, w) | \mathbf{p} = \mathbf{f}(\mathbf{x}), w = f_0(\mathbf{x}), \mathbf{x} \in \mathcal{C}_\mathbf{x}\}$ can be plot in the hyperplane of $(\mathbf{p}, w)$ as shown in Figure 3.3. Most interestingly, $d(\boldsymbol{\mu})$ is equal to the w -coordinate of the lowest intersection between the line $\mathbf{p} = \mathbf{0}$ and a line passing through $\mathcal{S}$ and perpendicular to the vector $(\boldsymbol{\mu}, 1)$.

As shown in Fig. 3.3, the duality gap is equal to zero if the w -coordinate of the lower border of $\mathcal{S}$ is a convex function of $\mathbf{p}$. Mathematically, a point on the lower border of $\mathcal{S}$ has the coordinate $(\mathbf{p}, f_0(\mathbf{x}_\mathbf{p}))$, where

$$\mathbf{x}_\mathbf{p} \in \arg \min_{\mathbf{x} : \mathbf{x} \in \mathcal{C}_\mathbf{x}, \mathbf{f}(\mathbf{x}) = \mathbf{p}} f_0(\mathbf{x}). \quad (3.9)$$

In a word, (3.1) has zero duality gap provided that $f_0(\mathbf{x}_\mathbf{p})$ is convex of $\mathbf{p}$.

When $\mathcal{C}_\mathbf{x}$ is convex and $\forall j, f_j(\mathbf{x})$ is convex over $\mathcal{C}_\mathbf{x}$, (3.1) becomes a convex problem¹. A convex problem has zero duality gap, as long as it satisfies a simple condition known as Slater constraint qualification shown as follows:

Theorem 3.4 (Proposition 5.3.1 in [Ber03]). *Suppose $\mathcal{C}_\mathbf{x}$ is convex and $\forall j, f_j(\mathbf{x})$ is convex over $\mathcal{C}_\mathbf{x}$. The duality gap of (3.1) is zero, if there exists a feasible $\mathbf{x}$ that satisfies all inequality constraints strictly.*

It is important to note that a problem with zero duality gap might be non-convex. As an example, the RA problems considered in [WV11d, YL06] are nonconvex, but have zero duality gap as the number of carriers is sufficiently large.

¹When $\mathcal{C}_\mathbf{x}$ is convex and $\forall j = 0, 1, \dots, J, f_j(\mathbf{x})$ is concave, the problem of maximizing $f_0(\mathbf{x})$ subject to the constraints that $\forall j = 1, \dots, J, f_j(\mathbf{x}) \geq 0$, is equivalent to a convex problem.

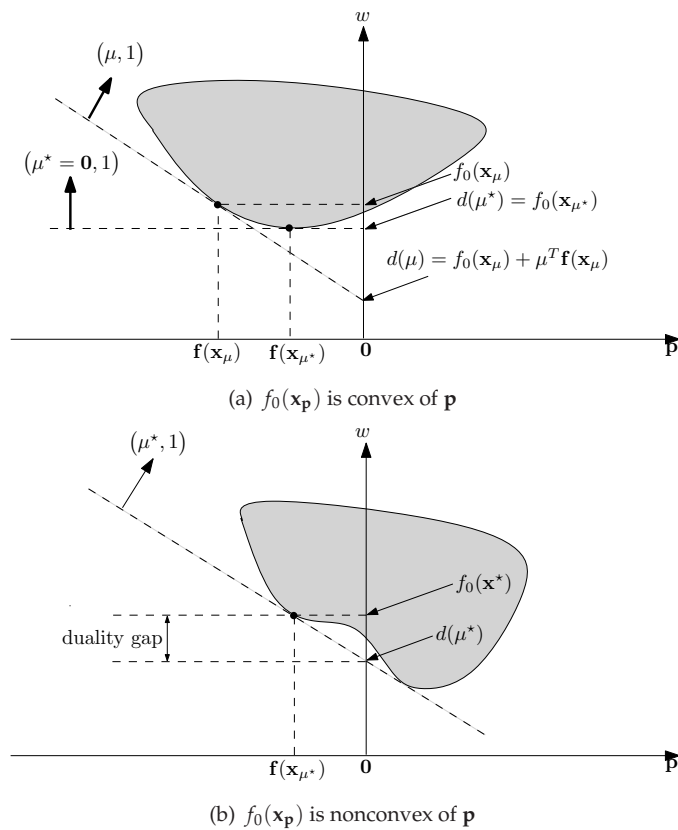


Figure 3.3 The visualization of duality gap in the hyperplane of $(\mathbf{p}, w)$, where the gray area represents the set $\mathcal{S}$.

3.4.2 Sensitivity analysis

To introduce the sensitivity analysis, the following problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{C}_{\mathbf{x}} \subseteq \mathcal{R}^N, \\ & f_j(\mathbf{x}) \leq u_j, j = 1, \dots, J, \end{aligned} \quad (3.10)$$

is considered. Note that (3.1) is a special case of (3.10) when $\forall j, u_j = 0$.

Let us stack all u_j into a vector $\mathbf{u} = [u_1, \dots, u_J]^T$. Assume (3.10) has zero duality gap $\forall \mathbf{u} \in \mathcal{R}^J$. The optimum $\mathbf{x}$ for (3.10) and the optimum dual variable for the dual problem of (3.10) are denoted as $\mathbf{x}^*(\mathbf{u})$ and $\boldsymbol{\mu}^*(\mathbf{u})$, respectively. Then, the following equality

$$\forall j, \frac{\partial f_0(\mathbf{x}^*(\mathbf{u}))}{\partial u_j} = -\mu_j^*(\mathbf{u}) \quad (3.11)$$

holds when $\boldsymbol{\mu}^*(\mathbf{u})$ is unique, where $\mu_j^*(\mathbf{u})$ is the j th entry of $\boldsymbol{\mu}^*(\mathbf{u})$ [BV04]. Note that $\mu_j^*(\mathbf{u}) \geq 0$ since $\boldsymbol{\mu}^*(\mathbf{u}) \succeq \mathbf{0}$.

In other words, the increment of u_j by Δ leads to a reduction of the optimum objective value of (3.10) by $\mu_j^*(\mathbf{u})\Delta$ when $\mu_j^*(\mathbf{u}) > 0$. This can be interpreted by the fact that when $\mu_j^*(\mathbf{u}) > 0$, $f_j(\mathbf{x}^*(\mathbf{u})) = u_j$ follows from the complementary slackness condition according to Theorem 3.3. This means that the j th constraint is tightened at $\mathbf{x}^*(\mathbf{u})$. In such case, increasing u_j relaxes this constraint and expands the feasible set of (3.10), thus results in a further reduction of the optimum objective value.

3.5 Optimality conditions for a convex problem

As shown earlier, a convex problem has many good features, e.g., the global bound property as explained in Section 2.2, and the zero duality gap property when the Slater constraint qualification is satisfied. These features lead to necessary and sufficient global optimality conditions. The first one is described as follows:

Corollary 3.5. *Suppose the feasible set $\mathcal{C}_{\mathbf{x}}^f$ of (3.1) is convex, and $f_0(\mathbf{x})$ is differentiable and convex over $\mathcal{C}_{\mathbf{x}}^f$. Then, $\mathbf{x}'$ is a global optimum for (3.1) if and only if*

$$\forall \mathbf{x} \in \mathcal{C}_{\mathbf{x}}^f, (\nabla_{\mathbf{x}} f_0(\mathbf{x}'))^T (\mathbf{x} - \mathbf{x}') \geq 0. \quad (3.12)$$

This theorem indicates that any local optimum of a convex problem is also a global optimum. The “only if” part of this theorem is due to Theorem 3.2. The “if” part is because

$$\forall \mathbf{x}, \mathbf{x}' \in \mathcal{C}_x^f, f(\mathbf{x}) \geq f(\mathbf{x}') + (\nabla_{\mathbf{x}} f(\mathbf{x}'))^T (\mathbf{x} - \mathbf{x}') \geq f(\mathbf{x}')$$

where the first inequality is due to the global bound feature (2.5).

The second one is given as follows:

Corollary 3.6. *Suppose $\mathcal{C}_x$ is convex, $\forall j, f_j(\mathbf{x})$ is convex over $\mathcal{C}_x$, and the Slater constraint qualification is satisfied. Then, $\mathbf{x}'$ and $\boldsymbol{\mu}'$ are globally optimum for (3.1) and its dual problem, respectively, if and only if they satisfy the same three conditions as in Theorem 3.3.*

This theorem is due to Theorem 3.3 because the duality gap of (3.1) is zero according to the given conditions. In particular, when $\mathcal{C}_x = \mathcal{R}^N$, the Lagrangian optimality condition is equivalent to $\nabla_{\mathbf{x}} L(\boldsymbol{\mu}', \mathbf{x}) = \mathbf{0}$, as can readily be derived from Corollary 3.5 and the fact that $L(\boldsymbol{\mu}', \mathbf{x})$ is convex of $\mathbf{x} \in \mathcal{R}^N$. In this case, the three conditions correspond to the KKT conditions for (3.1) as said earlier in Theorem 3.1. This means that the KKT conditions are both sufficient and necessary to justify global optimality for (3.1) when $\mathcal{C}_x = \mathcal{R}^N$.

Finally, it should be pointed out that there exists a unique global optimum for (3.1) when $\mathcal{C}_x^f$ is convex and $f_0(\mathbf{x})$ is strictly convex over $\mathcal{C}_x^f$. This can easily be proved as follows. Suppose there exist two different global optimum in $\mathcal{C}_x^f$, say $\mathbf{x}'$ and $\mathbf{x}'' \neq \mathbf{x}'$, i.e., $f_0(\mathbf{x}'') = f_0(\mathbf{x}') = \min_{\mathbf{x} \in \mathcal{C}_x^f} f_0(\mathbf{x})$. In this case, $\frac{\mathbf{x}'' + \mathbf{x}'}{2} \in \mathcal{C}_x^f$ (since $\mathcal{C}_x^f$ is convex), while $f(\frac{\mathbf{x}'' + \mathbf{x}'}{2}) < \frac{f(\mathbf{x}'') + f(\mathbf{x}')}{2} = \min_{\mathbf{x} \in \mathcal{C}_x^f} f(\mathbf{x})$ holds from Jensen’s inequality, leading to a contradiction. Therefore, the claim is true.

3.6 The dual method

The dual method is a powerful tool to find a global optimum $\mathbf{x}^*$ for (3.1) when it has zero duality gap. This means that it can be used to solve a convex problem if the problem satisfies the Slater constraint qualification. Note that interior-point method (IPM) usually runs faster than the dual method to solve a convex problem [BV04]. However, the development of IPM is highly complicated and problem specific, thus it is not introduced in this chapter. In practice, software packages are usually developed based on IPM to solve certain opti-

mization problems cast in standard forms, such as the MOSEK or gpcvx packages applicable to solving a geometric program (GP) [BKVH05, KKMB06].

The key to the dual method is to solve the dual problem (3.6) for a globally optimum solution μ^* , and then use $\mathbf{x}_{\mu^*}$ that is feasible for (3.1) and satisfies the complementary slackness condition as $\mathbf{x}^*$. The effectiveness of the dual method is justified by Theorem 3.3. Specifically, the inequality (3.7) always holds no matter the duality gap is zero or not. In case the duality gap is zero, meaning that $d(\mu^*) = f_0(\mathbf{x}^*)$, all inequalities in (3.7) must be tightened, therefore

$$L(\mu^*, \mathbf{x}_{\mu^*}) = f_0(\mathbf{x}_{\mu^*}) + \sum_{j=1}^J \mu_j^* f_j(\mathbf{x}_{\mu^*}) = f_0(\mathbf{x}^*) \quad (3.13)$$

must be satisfied. For a $\mathbf{x}_{\mu^*}$ that is feasible and satisfies the complementary slackness condition, $\sum_{j=1}^J \mu_j^* f_j(\mathbf{x}_{\mu^*}) = 0$ holds, and therefore $f_0(\mathbf{x}_{\mu^*}) = f_0(\mathbf{x}^*)$ follows, meaning that $\mathbf{x}_{\mu^*}$ is indeed a global optimum for (3.1).

The dual problem (3.6) is equivalent to a convex problem as said in Section 3.3.1. Note that $d(\mu)$ might be nondifferentiable and a subgradient of $d(\mu)$ with respect to μ can be computed as

$$\mathbf{s}(\mu) = \nabla_{\mu} L(\mu, \mathbf{x}_{\mu}) = \mathbf{f}(\mathbf{x}_{\mu}) \quad (3.14)$$

according to Corollary 2.1. Since $d(\mu)$ is concave and $\mathbf{s}(\mu)$ is a subgradient with respect to μ , the following inequality is satisfied:

$$\forall \mu', \mu \in \mathcal{R}^J, d(\mu') \leq d(\mu) + (\mathbf{s}(\mu))^T (\mu' - \mu). \quad (3.15)$$

Either subgradient or ellipsoid methods can be used to find the optimum dual variable μ^* iteratively. Both methods make use of $\mathbf{s}(\mu)$. This means that one of the tasks for the dual method is to solve the LRP (3.4) for $\mathbf{x}_{\mu}$. To facilitate description, a superscript $i + 1$ put to a variable indicates it is produced at the end of the i th iteration ($i \geq 0$). The subgradient and ellipsoid methods are introduced as follow:

1. The subgradient method is based on the idea that when μ moves along the subgradient direction iteratively with sufficiently small step size, the distance of μ and μ^* reduces gradually [Ber03].

Specifically, μ is iteratively updated with $\mu^{i+1} = [\mu^i + \delta^i \mathbf{s}^i]^+$ where $\mathbf{s}^i = \mathbf{s}(\mu^i)$ and δ^i is the step size at the i th iteration. When δ^i is sufficiently

small, μ^i converges to μ^* [Ber03]. A common criterion for choosing δ^i is that it must be square summable, but not absolute summable [Boy04]. In practice, $\delta^i = \alpha$, $\delta^i = \frac{\alpha}{i}$ or $\delta^i = \frac{\alpha}{\sqrt{i}}$ may be chosen where α must be sufficiently small [YL06]. In general, the subgradient method converges very slowly.

2. The ellipsoid method is a generalization of the bisection method to the multi-dimensional case [Boy04]. Specifically, it is known at the beginning of the i th iteration that μ^* is contained in the ellipsoid

$$E^i = \{\mu | (\mu - \mu^i)^T (A^i)^{-1} (\mu - \mu^i) \leq 1\}. \quad (3.16)$$

Obviously, μ^* must be confined within $S_i = E^i \cap \{\mu | (\mathbf{s}^i)^T (\mu - \mu^i) \geq 0\}$, otherwise $d(\mu^*) \leq d(\mu^i) + (\mathbf{s}^i)^T (\mu^* - \mu^i) < d(\mu^i)$ follows, contradicting the fact that $d(\mu^*) \geq d(\mu^i)$. Based on this idea, μ^{i+1} and $(A^{i+1})^{-1}$ can be computed as

$$\mu^{i+1} = \mu^i - \frac{1}{J+1} A^i \hat{\mathbf{s}}^i \quad (3.17)$$

$$A^{i+1} = \frac{J^2}{J^2 - 1} (A^i - \frac{2}{J+1} A^i \hat{\mathbf{s}}^i (\hat{\mathbf{s}}^i)^T A^i) \quad (3.18)$$

where $\hat{\mathbf{s}}^i = \frac{\mathbf{s}^i}{\sqrt{(\mathbf{s}^i)^T A^i \mathbf{s}^i}}$, to generate a new ellipsoid E^{i+1} containing S_i . It can be shown that the volume of E^{i+1} reduces as i increases, therefore the convergence to μ^* is guaranteed [Boy04]. The ellipsoid method is illustrated intuitively in Figure 3.4.

In practice, it is often difficult to find a $\mathbf{x}_\mu$ that satisfies simultaneously the feasibility condition and the complementary slackness condition exactly. To address this issue, we can terminate the dual method when $\mathbf{x}_\mu$ is feasible and $\sum_{j=1}^J \mu_j f_j(\mathbf{x}_\mu) \leq \epsilon$, where $\epsilon > 0$ is a very small value prescribed. Once the dual method is terminated according to this condition, it can be shown that

$$\begin{aligned} f_0(\mathbf{x}_\mu) - f_0(\mathbf{x}^*) &= f_0(\mathbf{x}_\mu) - d(\mathbf{x}^*) \\ &\leq f_0(\mathbf{x}_\mu) - d(\mu) \\ &= \sum_{j=1}^J \mu_j f_j(\mathbf{x}_\mu) \leq \epsilon, \end{aligned} \quad (3.19)$$

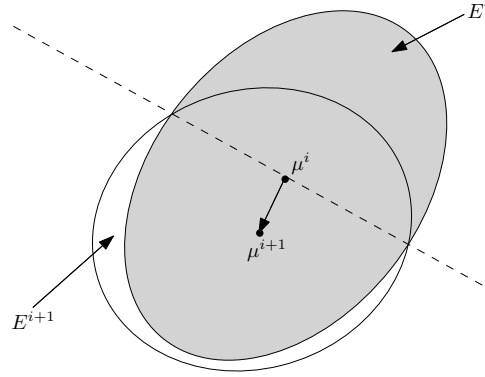


Figure 3.4 Illustration of the ellipsoid method as a generalization of the bisection method to the multi-dimensional case.

which means that $\mathbf{x}_\mu$ is approximately optimum, since $f_0(\mathbf{x}^*) \leq f_0(\mathbf{x}_\mu) \leq f_0(\mathbf{x}^*) + \epsilon$ is satisfied. In summary, the procedure of the dual method is summarized in Algorithm 1.

Algorithm 1 The dual method

Initialize μ ;

repeat

 solve the LRP (3.4) for a globally optimum $\mathbf{x}_\mu$;

 update μ with either subgradient or ellipsoid method;

until $\mathbf{x}_\mu$ is feasible and $\sum_{j=1}^J \mu_j f_j(\mathbf{x}_\mu) \leq \epsilon$

 output $\mathbf{x}_\mu$ as an approximately optimum solution for (3.1).

Successive approximation based optimization method

4

4.1 Related works and chapter overview

This chapter centers around solving the problem

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{R}^N, \\ & f_j(\mathbf{x}) \leq 0, j = 1, \dots, J, \end{aligned} \tag{4.1}$$

where $\forall j = 0, 1, \dots, J$, the definition domain of $f_j(\mathbf{x})$ is $\mathcal{R}^N$. Suppose (4.1) has a finite optimum objective value, and there exists a function $f_j(\mathbf{x})$ that is nonconvex of $\mathbf{x} \in \mathcal{R}^N$, which makes (4.1) be a nonconvex problem.

One of the possible ways to solve (4.1) is to use the successive convex approximation (SCA) method, first proposed and named as inner approximation method in [MW78]. In fact, the SCA method is a generalization of a few works reported in 1970s to solve a nongeometric program through successive GP approximations [AW70, DP72, RW74]. This SCA method solves a sequence of convex approximations of (4.1). As long as the objective/constraint functions of every approximate problem all satisfy tightness and differentiation conditions, convergence of the SCA method is guaranteed, and the solution at convergence satisfies the KKT conditions of (4.1). Note that this solution is not always a global optimum for (4.1).

In this chapter, the SCA method is generalized to a more general form, namely the successive approximation (SA) method. In particular, this general SA method does not require every approximation of (4.1) to be convex. More-

over, this SA method starts with change of variable (COV), which initially was not considered in [MW78].

Next, a general method is proposed to construct for a nonconvex function a convex upper bound approximation (CUBA) that satisfies tightness and differentiation conditions. Most interestingly, the CUBA construction method together with the COV often makes it easier to construct successive approximations of (4.1), as illustrated by an example shown later. Moreover, a generalized version of that CUBA can be constructed by incorporating a convex function. These ways provide plenty of degrees of freedom to improve the performance of the SA method, as illustrated by the design of novel dynamic power spectrum management algorithms in Chapter 5. Then, the SA method is compared with the Difference of Convex functions Algorithms (DCA), which is applicable to solve the problem of minimizing a difference of two convex functions.

Finally, two interesting applications of the SA method are shown. The first one is the expectation-maximization (EM) algorithm well known for parameter estimation. It will be shown that this algorithm can be derived from the SA method by using Jensen's inequality. Note that every approximate problem solved by the EM algorithm is not necessarily convex. This indicates that the SA method is indeed more general than the SCA method which requires every approximate problem to be convex. The second one is the successive GP approximation method proposed in [CTP⁺07] to approach a nongeometric problem, which was first derived from the arithmetic-geometric mean inequality. It will be shown that this method can readily be derived by making COV and using the proposed CUBA construction method.

4.2 The SA method

To solve (4.1), the SA method starts with making a COV from $\mathbf{x}$ to $\mathbf{y}$ (with $[\mathbf{y}]_i = y_i$), which initially was not considered in [MW78]. This COV needs to satisfy that every $\mathbf{y}$ is mapped to only one $\mathbf{x}$. $\mathbf{x}(\mathbf{y})$ represents the $\mathbf{x}$ corresponding to a given $\mathbf{y}$. The COV is motivated by the fact that the COV better generalizes the SA method, since it together with the CUBA construction method shown later often makes it easier to construct successive approximations of (4.1). Note that the COV can be regarded as a general procedure that

is always taken, since the case where the COV is unnecessary can be regarded as making the COV from $\mathbf{x}$ to itself.

After the COV and some mathematical arrangements, solving (4.1) is equivalent to finding a globally optimum $\mathbf{y}$ to

$$\begin{aligned} \min_{\mathbf{y}} \quad & g_0(\mathbf{y}) \\ \text{s.t.} \quad & \mathbf{y} \in \mathcal{R}^I, \\ & g_j(\mathbf{y}) \leq 0, j = 1, \dots, J, \end{aligned} \quad (4.2)$$

and then compute the corresponding $\mathbf{x}(\mathbf{y})$ as a global optimum for (4.1).

Initialized by a given $\mathbf{y}^0$ feasible for (4.2), the SA method solves approximations of (4.2) successively. To facilitate description, a superscript $(m+1)$ put to a variable indicates that variable is produced at the end of the m th iteration hereafter ($m \geq 0$). At the m th iteration, $\mathbf{y}^{m+1}$ is found as a global optimum for

$$\begin{aligned} \min_{\mathbf{y}} \quad & g_0^m(\mathbf{y}) \\ \text{s.t.} \quad & \mathbf{y} \in \mathcal{R}^J, \\ & g_j^m(\mathbf{y}) \leq 0, j = 1, \dots, J, \end{aligned} \quad (4.3)$$

where $\forall j$, $g_j^m(\mathbf{y})$ is an upper bound approximation of $g_j(\mathbf{y})$, i.e., $\forall \mathbf{y} \in \mathcal{R}^J$, $g_j(\mathbf{y}) \leq g_j^m(\mathbf{y})$. It is assumed that (4.3) can readily be solved by certain methods.

It can easily be seen that every $\mathbf{y}^{m+1}$ ($m = 0, 1, \dots$) is feasible for (4.2) because $\forall j$, $g_j(\mathbf{y}^{m+1}) \leq g_j^m(\mathbf{y}^{m+1}) \leq 0$. Suppose $\forall m, j$, $g_j^m(\mathbf{y})$ satisfies the tightness condition at $\mathbf{y}^m$ that

$$g_j^m(\mathbf{y}^m) = g_j(\mathbf{y}^m), \quad (4.4)$$

then

$$g_0(\mathbf{y}^{m+1}) \leq g_0^m(\mathbf{y}^{m+1}) \leq g_0^m(\mathbf{y}^m) = g_0(\mathbf{y}^m) \quad (4.5)$$

follows, where the second inequality is because for (4.3) $\mathbf{y}^{m+1}$ is globally optimum and $\mathbf{y}^m$ is feasible since $\forall j$, $g_j^m(\mathbf{y}^m) = g_j(\mathbf{y}^m) \leq 0$. This means that $\{g_0(\mathbf{y}^m) | m = 0, 1, \dots\}$ is a monotonically decreasing sequence. Since (4.1) has a finite optimum objective value, $g_0(\mathbf{y}^m)$ is bounded below. Therefore, $g_0(\mathbf{y}^m)$

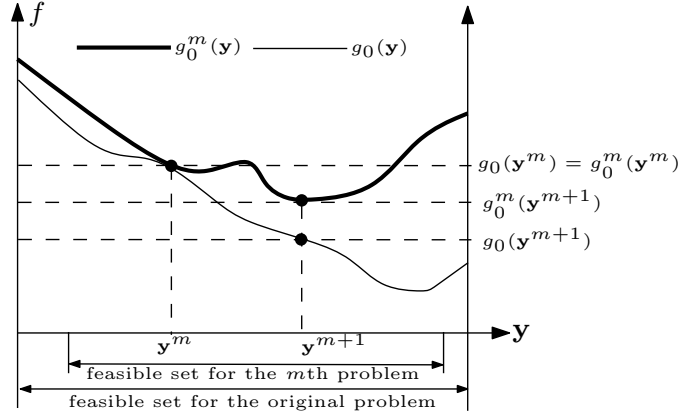


Figure 4.1 Illustration of the idea behind the SA method.

must converge as m increases (see Proposition A.3 in [Ber03]). The idea behind the SA method is illustrated in Figure 4.1.

Furthermore, when $g_0(\mathbf{y}^{m+1}) = g_0(\mathbf{y}^m)$, all inequalities in (4.5) is tightened, indicating that $g_0^m(\mathbf{y}^{m+1}) = g_0^m(\mathbf{y}^m)$ must hold. This means that $\mathbf{y}^m$ is a global optimum for (4.3), i.e., $\mathbf{y}^m$ is a fixed point for the SA method. An important property of this fixed point is shown as follows:

Theorem 4.1. Suppose $\forall m, j, g_j^m(\mathbf{y})$ satisfies the tightness condition and the differentiation condition at $\mathbf{y}^m$ that

$$\forall i, \frac{\partial g_j^m(\mathbf{y}^m)}{\partial y_i} = \frac{\partial g_j(\mathbf{y}^m)}{\partial y_i}, \quad (4.6)$$

then a fixed point $\mathbf{y}^m$ that is regular for (4.3) satisfies the KKT conditions for (4.2).

Proof. For the fixed point $\mathbf{y}^m$, it is a global optimum for (4.3). Since $\mathbf{y}^m$ is regular for (4.3), the KKT conditions of (4.3) must be satisfied, i.e., $\mathbf{y}^m$ is feasible for (4.3) and there exist nonnegative real values $\mu_1, \dots, \mu_J$ satisfying

$$\forall i, \frac{\partial g_0^m(\mathbf{y}^m)}{\partial y_i} + \sum_{j=1}^J \mu_j \frac{\partial g_j^m(\mathbf{y}^m)}{\partial y_i} = 0, \quad (4.7)$$

$$\forall j > 0, \mu_j g_j^m(\mathbf{y}^m) = 0. \quad (4.8)$$

From the tightness and differentiation conditions, it can easily be seen that $\mathbf{y}^m$ is feasible for (4.2), and

$$\forall i, \frac{\partial g_0(\mathbf{y}^m)}{\partial y_i} + \sum_{j=1}^J \mu_j \frac{\partial g_j(\mathbf{y}^m)}{\partial y_i} = 0, \quad (4.9)$$

$$\forall j > 0, \mu_j g_j(\mathbf{y}^m) = 0, \quad (4.10)$$

meaning that $\mathbf{y}^m$ satisfies the KKT conditions for (4.2). $\square$

The above property for the SA method was first shown in [MW78]. Some new properties, which later prove useful for studying SA based RA algorithms, are presented as follows:

Theorem 4.2. *Suppose $\forall m, j, g_j^m(\mathbf{y})$ satisfies the tightness and differentiation conditions (4.4) and (4.6) at $\mathbf{y}^m$. The following claims are true:*

1. *Suppose $g_0^m(\mathbf{y})$ is a strict upper bound approximation of $g_0(\mathbf{y})$ except at $\mathbf{y}^m$, i.e., $g_0(\mathbf{y}) < g_0^m(\mathbf{y})$ in case $\mathbf{y} \neq \mathbf{y}^m$. Then if $\mathbf{y}^{m+1} \neq \mathbf{y}^m$, $g_0(\mathbf{y}^{m+1}) < g_0(\mathbf{y}^m)$ holds. Moreover, $g_0(\mathbf{y}^{m+1}) = g_0(\mathbf{y}^m)$ if and only if $\mathbf{y}^{m+1} = \mathbf{y}^m$.*
2. *Suppose $g_j(\mathbf{y}) = f_j(\mathbf{x}(\mathbf{y}))$ and every y_i is one-to-one mapped to x_i independently of other entries. Then $\mathbf{x}^m = \mathbf{x}(\mathbf{y}^m)$, where $\mathbf{y}^m$ is a fixed point satisfying the KKT conditions for (4.3), must satisfy the KKT conditions for (4.1).*
3. *Suppose*

$$g_0(\mathbf{y}^0) < \min_{\mathbf{y} \in C_{\mathbf{y}} - C_{\mathbf{y}}^*} g_0(\mathbf{y}) \quad (4.11)$$

where $C_{\mathbf{y}}$ and $C_{\mathbf{y}}^$ represent the solution set satisfying KKT conditions and the set of global optimum for (4.2), respectively. Then $\mathbf{y}^m \in C_{\mathbf{y}}^*$ must hold when $g_0(\mathbf{y}^{m+1}) = g_0(\mathbf{y}^m)$.*

Proof. Proof of claim 1): if $\mathbf{y}^{m+1} \neq \mathbf{y}^m$, $g_0(\mathbf{y}^{m+1}) < g_0^m(\mathbf{y}^{m+1}) \leq g_0^m(\mathbf{y}^m) = g_0(\mathbf{y}^m)$ follows, thus $g_0(\mathbf{y}^{m+1}) < g_0(\mathbf{y}^m)$ holds. Suppose when $g_0(\mathbf{y}^{m+1}) = g_0(\mathbf{y}^m)$, $\mathbf{y}^{m+1} \neq \mathbf{y}^m$. As said earlier, $g_0(\mathbf{y}^{m+1}) < g_0(\mathbf{y}^m)$ holds, leading to a contradiction. Therefore, $g_0(\mathbf{y}^{m+1}) = g_0(\mathbf{y}^m)$ if and only if $\mathbf{y}^{m+1} = \mathbf{y}^m$.

Proof of claim 2): since $g_j(\mathbf{y})$ is the composition of $f_j(\mathbf{x})$ and $\mathbf{x}(\mathbf{y})$ and every y_i is one-to-one mapped to x_i independently of other entries,

$$\forall j, i, \frac{\partial g_j^m(\mathbf{y}^m)}{\partial y_i} = \frac{\partial g_j(\mathbf{y}^m)}{\partial y_i} = \frac{\partial f_j(\mathbf{x}^m)}{\partial x_i} \frac{\partial x_i(y_i^m)}{\partial y_i} \quad (4.12)$$

follows from the differentiation condition at $\mathbf{y}^m$ and the chain rule of derivation, where $x_i(y_i)$ is the x_i corresponding to y_i . Since $\mathbf{y}^m$ is a fixed point and satisfies the KKT conditions for (4.2), (4.7) and (4.8) are also satisfied. It can easily be seen that $\mathbf{x}^m$ is feasible for (4.1) and there exist nonnegative real values $\nu_1, \dots, \nu_J$ satisfying

$$\forall i, \frac{\partial f_0(\mathbf{x}^m)}{\partial x_i} + \sum_{j=1}^J \nu_j \frac{\partial f_j(\mathbf{x}^m)}{\partial x_i} = 0 \quad (4.13)$$

$$\forall j > 0, \nu_j f_j(\mathbf{x}^m) = 0, \quad (4.14)$$

meaning that $\mathbf{x}^m$ satisfies the KKT conditions for (4.1).

Proof of claim 3): when $g_0(\mathbf{y}^m) = g_0(\mathbf{y}^{m+1})$, $\mathbf{y}^m \in \mathcal{C}_y$ follows. Suppose $\mathbf{y}^m \notin \mathcal{C}_y^*$, then $\mathbf{y}^m \in \mathcal{C}_y - \mathcal{C}_y^*$ holds, meaning that $g_0(\mathbf{y}^0) < \min_{\mathbf{y} \in \mathcal{C}_y - \mathcal{C}_y^*} g_0(\mathbf{y}) \leq g_0(\mathbf{y}^m)$. This is a contradiction with the fact that $g_0(\mathbf{y}^m)$ is decreasing with m . Therefore, $\mathbf{y}^m \in \mathcal{C}_y^*$ follows. $\square$

In practice, the right-hand side of (4.11) is rarely known a priori. Thus, it is desirable to find a $\mathbf{y}$ with as small as possible $g_0(\mathbf{y})$ as $\mathbf{y}^0$, to enhance the possibility to satisfy (4.11). Note that the idea behind the SA method is successive refinement, thus a solution produced by any other method, either rigorous or heuristic in nature, can be used as $\mathbf{y}^0$.

The key to the SA method is to build upper bound approximations of all objective/constraint functions in (4.2). To this end, mathematical inequalities, e.g., the Jensen's inequality, play important roles, as illustrated by the derivation of the EM algorithm from the SA method.

4.3 The SCA and CUBA construction methods

The SCA method is a special case of the SA method when the objective/constraint functions in (4.3) are all CUBAs of their counterparts in (4.2). In this case, the problem (4.3) is a convex approximation of (4.2) at the m th iteration, thus it can be solved by state-of-the-art optimization methods, e.g., the dual method as introduced in Chapter 3

For the trivial case where $g_j(\mathbf{y})$ is convex, its CUBA $g_j^m(\mathbf{y})$ is simply itself, i.e., $g_j^m(\mathbf{y}) = g_j(\mathbf{y})$. The key to applying the SCA method is to construct a CUBA for a nonconvex function.

A general method is presented to construct a CUBA for a nonconvex function $g_j(\mathbf{y})$ as follows. This method starts with writing $g_j(\mathbf{y})$ as either a whole

or the sum of multiple parts. Then, the CUBA of each part is constructed, and the sum of the CUBAs of all parts is taken as $g_j^m(\mathbf{y})$. For a part that is convex of $\mathbf{y}$, its CUBA is simply itself. For another nonconvex part of $\mathbf{y}$, denoted by $g(\mathbf{y})$, its CUBA $g^m(\mathbf{y})$ can be constructed with one of the following methods.

4.3.1 Concavity-based method

This method is applicable when $g(\mathbf{y})$ is concave of $\mathbf{y}$. As a result,

$$g(\mathbf{y}) \leq g(\mathbf{y}^m) + \sum_{i=1}^I \frac{\partial g(\mathbf{y}^m)}{\partial y_i} (y_i - y_i^m) \quad (4.15)$$

follows from the property that the first-order Taylor approximation of $g(\mathbf{y})$ around $\mathbf{y}^m$ is a global overestimator of $g(\mathbf{y})$. It can easily be seen that the right-hand side of (4.15) is convex of $\mathbf{y}$, therefore it can be taken as $g^m(\mathbf{y})$. Note that this $g^m(\mathbf{y})$ satisfies the tightness and differentiation conditions at $\mathbf{y}^m$.

4.3.2 Composition of a concave Function with concave/convex functions (CAVF) method

This method is applicable when $g(\mathbf{y})$ can be written as

$$g(\mathbf{y}) = h(z_1(\mathbf{y}), \dots, z_L(\mathbf{y})) \quad (4.16)$$

where $h(\mathbf{z})$ is concave of $\mathbf{z} = [z_1, \dots, z_L]^T$ and every $z_l(\mathbf{y})$ satisfies one of the following two conditions:

- $h(\mathbf{z})$ is *increasing* and *differentiable* of z_l , i.e., $\forall \mathbf{z}, \frac{\partial h(\mathbf{z})}{\partial z_l} \geq 0$, and $z_l(\mathbf{y})$ is *convex* of $\mathbf{y}$,
- $h(\mathbf{z})$ is *decreasing* and *differentiable* of z_l , i.e., $\forall \mathbf{z}, \frac{\partial h(\mathbf{z})}{\partial z_l} \leq 0$, and $z_l(\mathbf{y})$ is *concave* of $\mathbf{y}$.

Due to the concavity of $h(\mathbf{z})$ with respect to $\mathbf{z}$,

$$h(\mathbf{z}) \leq h(\mathbf{z}^m) + \sum_{l=1}^L \frac{\partial h(\mathbf{z}^m)}{\partial z_l} (z_l - z_l(\mathbf{y}^m)) \quad (4.17)$$

follows, where $\mathbf{z}^m = [z_1(\mathbf{y}^m), \dots, z_L(\mathbf{y}^m)]^T$. Replacing every z_l in both sides of (4.17) with $z_l(\mathbf{y})$,

$$g(\mathbf{y}) \leq h(\mathbf{z}^m) + \sum_{l=1}^L \frac{\partial h(\mathbf{z}^m)}{\partial z_l} (z_l(\mathbf{y}) - z_l(\mathbf{y}^m)). \quad (4.18)$$

holds. It can easily be seen that the right-hand side of (4.18) is convex of $\mathbf{y}$, therefore it can be taken as the CUBA $g^m(\mathbf{y})$. Note that this $g^m(\mathbf{y})$ satisfies the tightness and differentiation conditions at $\mathbf{y}^m$.

4.3.3 Composition of a convEX Function with concave/convex functions (CEXF) method

This method is applicable when $g(\mathbf{y})$ can still be written as (4.16), while $h(\mathbf{z})$ is a convex function of $\mathbf{z} = [z_1, \dots, z_L]^T$, and every $z_l(\mathbf{y})$ satisfies one of the following two conditions:

- $h(\mathbf{z})$ is *increasing* of z_l , and $z_l(\mathbf{y})$ is *concave* and *differentiable* of $\mathbf{y}$,
- $h(\mathbf{z})$ is *decreasing* of z_l , and $z_l(\mathbf{y})$ is *convex* and *differentiable* of $\mathbf{y}$.

When $z_l(\mathbf{y})$ is concave (resp. convex) of $\mathbf{y}$, $z_l(\mathbf{y})$ is smaller (resp. greater) than

$$z'_l(\mathbf{y}) = z_l(\mathbf{y}^m) + \sum_{i=1}^I \frac{\partial z_l(\mathbf{y}^m)}{\partial y_i} (y_i - y_i^m), \quad (4.19)$$

which is an affine function of $\mathbf{y}$. Based on this property and the monotonicity of $h(\mathbf{z})$ with respect to every z_l , it can easily be shown that

$$g(\mathbf{y}) \leq h(z'_1(\mathbf{y}), \dots, z'_L(\mathbf{y})). \quad (4.20)$$

Note that the right-hand side of (4.20) is convex of $\mathbf{y}$. To see this more clearly, it can be seen that $\forall \mathbf{y}_a, \mathbf{y}_b$ and $\forall \alpha, \beta \in [0, 1]$ satisfying $\alpha + \beta = 1$,

$$\begin{aligned} & \alpha \cdot h(z'_1(\mathbf{y}_a), \dots, z'_L(\mathbf{y}_a)) + \beta \cdot h(z'_1(\mathbf{y}_b), \dots, z'_L(\mathbf{y}_b)) \\ & \geq h(\alpha \cdot z'_1(\mathbf{y}_a) + \beta \cdot z'_1(\mathbf{y}_b), \dots, \alpha \cdot z'_L(\mathbf{y}_a) + \beta \cdot z'_L(\mathbf{y}_b)) \\ & = h(z'_1(\alpha \cdot \mathbf{y}_a + \beta \cdot \mathbf{y}_b), \dots, z'_L(\alpha \cdot \mathbf{y}_a + \beta \cdot \mathbf{y}_b)) \end{aligned} \quad (4.21)$$

follows from the convexity of $h(\mathbf{z})$ with respect to $\mathbf{z}$ and the linearity of every $z'_l(\mathbf{y})$ with respect to $\mathbf{y}$. Therefore, the right-hand side of (4.20) can be taken as the CUBA $g^m(\mathbf{y})$. Note that this $g^m(\mathbf{y})$ satisfies the tightness and differentiation conditions at $\mathbf{y}^m$.

4.3.4 Regularization by a convEX Function (REXF) method

This method is applicable in one of two following ways:

- Suppose a strongly convex function $\phi_1(\mathbf{y})$ can be chosen to guarantee that $\phi_1(\mathbf{y}) - g(\mathbf{y})$ is convex of $\mathbf{y}$. Here “strongly convex” means that

$\phi_1(\mathbf{y})$ must satisfy $\forall \mathbf{y} \in \mathcal{R}^J, \nabla_{\mathbf{y}}^2(\phi_1(\mathbf{y}) - g(\mathbf{y})) \succeq \mathbf{0}$. In such a case, $g(\mathbf{y})$ can be expressed as $g(\mathbf{y}) = \phi_1(\mathbf{y}) - (\phi_1(\mathbf{y}) - g(\mathbf{y}))$. By using the concavity of $-(\phi_1(\mathbf{y}) - g(\mathbf{y}))$ with respect to $\mathbf{y}$, a CUBA of $g(\mathbf{y})$ can be built as

$$g^m(\mathbf{y}) = \phi_1(\mathbf{y}) - (\phi_1(\mathbf{y}^m) - g(\mathbf{y}^m)) - \sum_{i=1}^I \left(\frac{\partial \phi_1(\mathbf{y}^m)}{\partial y_i} - \frac{\partial g(\mathbf{y}^m)}{\partial y_i} \right) (y_i - y_i^m). \quad (4.22)$$

- Suppose a strongly convex function $\phi_2(\mathbf{y})$ can be chosen to ensure that $\phi_2(\mathbf{y}) + g(\mathbf{y})$ is convex of $\mathbf{y} \in \mathcal{R}^J$, $g(\mathbf{y})$ can be expressed as $g(\mathbf{y}) = (\phi_2(\mathbf{y}) + g(\mathbf{y})) - \phi_2(\mathbf{y})$, leading to another CUBA of $g(\mathbf{y})$ as

$$g^m(\mathbf{y}) = \phi_2(\mathbf{y}) + g(\mathbf{y}) - \phi_2(\mathbf{y}^m) - \sum_{i=1}^I \frac{\partial \phi_2(\mathbf{y}^m)}{\partial y_i} (y_i - y_i^m). \quad (4.23)$$

The above ways of regularizing $g(\mathbf{y})$ to be the difference of two convex functions were first proposed in [Pha88]. Most interestingly, $g(\mathbf{y})$ is *kept* as a constituting part of $g^m(\mathbf{y})$ in (4.23), while $g^m(\mathbf{y})$ in (4.22) is *independent* of $g(\mathbf{y})$. Nevertheless, both forms of $g^m(\mathbf{y})$ satisfy the tightness and differentiation conditions at $\mathbf{y}^m$.

As said earlier, after constructing the CUBA of each part, the sum of the CUBAs of all parts is taken as $g_j^m(\mathbf{y})$. It can readily be shown that such a $g_j^m(\mathbf{y})$ satisfies the tightness and differentiation conditions at $\mathbf{y}^m$.

Most interestingly, by incorporating any convex function, say $\varphi(\mathbf{y})$, a generalized version for $g_j^m(\mathbf{y})$ can be constructed. Specifically, the single or multiple parts that make up $g_j(\mathbf{y})$ can well be added with $\varphi(\mathbf{y})$ and $-\varphi(\mathbf{y})$. Then, the above described method is applied to construct a generalized CUBA of $g_j(\mathbf{y})$ as

$$g_{jr}^m(\mathbf{y}) = g_j^m(\mathbf{y}) + \varphi(\mathbf{y}) - \varphi(\mathbf{y}^m) - \sum_{i=1}^I \frac{\partial \varphi(\mathbf{y}^m)}{\partial y_i} (y_i - y_i^m), \quad (4.24)$$

where $g_j^m(\mathbf{y})$ is the CUBA constructed when $\varphi(\mathbf{y})$ is a zero function. Obviously, $g_{jr}^m(\mathbf{y})$ is a more general CUBA than $g_j^m(\mathbf{y})$, and $g_{jr}^m(\mathbf{y})$ still satisfies the tightness and differentiation conditions at $\mathbf{y}^m$.

In practice, the above CUBA construction method in combination with different convex functions $\varphi(\mathbf{y})$ provide plenty of degrees of freedom to improve the performance of the SCA method. For instance, they can be exploited to

construct different sequences of convex approximations of (4.2), each leading to a solution satisfying the KKT conditions of (4.2). Then, the solution with the minimum objective value can be chosen.

4.4 Comparison of the SA method and the DCA

The DCA algorithm was first proposed in [Pha88] to solve

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) - r(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{C}_x, \end{aligned} \quad (4.25)$$

where $f(\mathbf{x})$ and $r(\mathbf{x})$ are both convex of $\mathbf{x}$ and $\mathcal{C}_x$ is a convex set. The procedure of the DCA algorithm is very simple. First, the conjugate function of $f(\mathbf{x})$ is defined as

$$f^*(\mathbf{s}) = \sup_{\mathbf{x} \in \mathcal{C}_x} (\mathbf{s}^T \mathbf{x} - f(\mathbf{x})), \quad (4.26)$$

and $\mathbf{x}_s$ as the maximizer to the right-hand side given a fixed $\mathbf{s}$. Note that $f^*(\mathbf{s})$ is convex of $\mathbf{s}$ since it is a pointwise supremum over $\mathbf{x}$ of $\mathbf{s}^T \mathbf{x} - f(\mathbf{x})$ that can be regarded as a convex function of $\mathbf{s}$.

Initialized by a given $\mathbf{x}^0 \in \mathcal{C}_x$, the DCA algorithm at the m th iteration first computes a subgradient of $r(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}^m$, denoted by $\mathbf{s}^m$. Then, a subgradient of $f^*(\mathbf{s})$ at $\mathbf{s} = \mathbf{s}^m$ is computed and assigned to $\mathbf{x}^{m+1}$. According to Corollary 2.1, $\mathbf{x}_{\mathbf{s}^m}$ is a subgradient of $f^*(\mathbf{s})$ at $\mathbf{s} = \mathbf{s}^m$. Therefore, the $\mathbf{x} \in \mathcal{C}_x$ minimizing $f(\mathbf{x}) - (\mathbf{s}^m)^T \mathbf{x}$ should be found as $\mathbf{x}^{m+1}$. It was shown that in general the DCA converges to a *locally* optimum solution.

When using the SA method to solve (4.25), the COV is unnecessary since a CUBA of $f(\mathbf{x}) - r(\mathbf{x})$ at the m th iteration can easily be constructed as $f(\mathbf{x}) - r(\mathbf{x}^m) - (\mathbf{s}^m)^T (\mathbf{x} - \mathbf{x}^m)$. Thus, the SA method also needs to find the $\mathbf{x} \in \mathcal{C}_x$ minimizing $f(\mathbf{x}) - (\mathbf{s}^m)^T \mathbf{x}$ as $\mathbf{x}^{m+1}$. Obviously, using the SA method leads to the same procedure as using the DCA to solve (4.25).

Note that the DCA algorithm can be used to solve (4.1) when all constraint functions are convex, since the objective function can always be written as a difference of two convex functions, e.g. with the REXF method. In contrast, the SA algorithm is applicable to solve (4.1) when after COV all objective/constraint functions in (4.2) take any forms (not necessarily as a difference of two convex functions) that allow their (not necessarily convex)

upper bounds satisfying the tightness and differentiation conditions to be constructed, using certain inequalities or the proposed CUBA construction method. From the theoretical point of view, the DCA and the SA method were derived in different ways. As elaborated in [PT97], the DCA algorithm was developed based on convex analysis of the duality and local optimality conditions for (4.25), while the SA algorithm was developed based on successive approximations as explained earlier.

4.5 Application in parameter estimation

The EM algorithm, initially proposed based on probability analysis, is a well known method for parameter estimation from incomplete measurements [MS00]. For instance, one of its successful applications is to design synchronization algorithms for coded communication systems [Wan08, Her06]. In fact, the EM algorithm is a special case of the SA method. To show this point explicitly, the EM algorithm is first described briefly, and then derived from the SA method as below.

Let's say $\mathbf{x}$ represents a set of unknown parameters to be estimated from the data $\mathbf{d}$ and $\mathbf{z}$. The joint PDF of $\mathbf{d}$ and $\mathbf{z}$ conditional on $\mathbf{x}$ is denoted as $p(\mathbf{d}, \mathbf{z}|\mathbf{x})$. Suppose the measurement of $\mathbf{d}$ is available while that of $\mathbf{z}$ is not. In this case, the maximum-likelihood (ML) estimator of $\mathbf{x}$ is the $\hat{\mathbf{x}}$ that maximizes the log-likelihood function of $\mathbf{d}$, i.e., a global optimum to

$$\begin{aligned} \max_{\mathbf{x}} \quad & L(\mathbf{x}) = \log p(\mathbf{d}|\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{R}^N, \end{aligned} \quad (4.27)$$

where $p(\mathbf{d}|\mathbf{x}) = \int_{\mathbf{z}} p(\mathbf{d}, \mathbf{z}|\mathbf{x}) d\mathbf{z}$, i.e., $p(\mathbf{d}|\mathbf{x})$ is computed from $p(\mathbf{d}, \mathbf{z}|\mathbf{x})$ by marginalization over $\mathbf{z}$.

The EM algorithm is an iterative method. At the m th iteration ($m \geq 0$ and $\mathbf{x}^0$ is a prescribed initial point in $\mathcal{R}^N$), $\mathbf{x}^{m+1}$ is computed as a global optimum for

$$\begin{aligned} \max_{\mathbf{x}} \quad & \mathbb{E}_{p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m)}(\log p(\mathbf{d}, \mathbf{z}|\mathbf{x})) = \int_{\mathbf{z}} (p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m) \log p(\mathbf{d}, \mathbf{z}|\mathbf{x})) d\mathbf{z}, \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{R}^N, \end{aligned} \quad (4.28)$$

where $\mathbb{E}_{p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m)}(\cdot)$ represents the expectation with respect to $\mathbf{z}$ with the PDF $p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m)$. It is shown that $\forall m \geq 0$, $L(\mathbf{x}^{m+1}) \geq L(\mathbf{x}^m)$, i.e., the EM algorithm

iteratively produces parameter values with increasing log-likelihood. Therefore, the EM algorithm is guaranteed to converge.

The EM algorithm can be derived from the SA method as follows. For any PDF $q(\mathbf{z})$ of $\mathbf{z}$,

$$\begin{aligned} -L(\mathbf{x}) &= -\log \int_{\mathbf{z}} p(\mathbf{d}, \mathbf{z}|\mathbf{x}) d\mathbf{z} \\ &= -\log \int_{\mathbf{z}} q(\mathbf{z}) \frac{p(\mathbf{d}, \mathbf{z}|\mathbf{x})}{q(\mathbf{z})} d\mathbf{z} \\ &\leq -\int_{\mathbf{z}} q(\mathbf{z}) \log \frac{p(\mathbf{d}, \mathbf{z}|\mathbf{x})}{q(\mathbf{z})} d\mathbf{z} \end{aligned} \quad (4.29)$$

follows from the Jensen's inequality due to the concavity of the logarithm function. This means that the right-hand side of (4.29) can naturally be regarded as an upper bound approximation of $-L(\mathbf{x})$. For a fixed $\mathbf{x}$, the inequality (4.29) is tightened when $q(\mathbf{z})$ satisfies that

$$\forall \mathbf{z}, \frac{p(\mathbf{d}, \mathbf{z}|\mathbf{x})}{q(\mathbf{z})} = \alpha \quad (4.30)$$

where α is a constant, or equivalently when

$$q(\mathbf{z}) = \frac{p(\mathbf{d}, \mathbf{z}|\mathbf{x})}{\int_{\mathbf{z}} p(\mathbf{d}, \mathbf{z}|\mathbf{x}) d\mathbf{z}} = p(\mathbf{z}|\mathbf{d}, \mathbf{x}). \quad (4.31)$$

According to the above analysis, an upper bound approximation $-L^m(\mathbf{x})$ of $-L(\mathbf{x})$ satisfying the tightness condition $L^m(\mathbf{x}^m) = L(\mathbf{x}^m)$ can be constructed as

$$-L^m(\mathbf{x}) = -\int_{\mathbf{z}} p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m) \log \frac{p(\mathbf{d}, \mathbf{z}|\mathbf{x})}{p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m)} d\mathbf{z},$$

and the SA method requires to compute the $\mathbf{x}$ minimizing $-L^m(\mathbf{x})$, or equivalently a global optimum for (4.28), as $\mathbf{x}^{m+1}$. This means that the EM algorithm can indeed be derived from the SA method. Moreover, it can be shown that $\forall m, i$,

$$\begin{aligned} \frac{\partial(L^m(\mathbf{x}^m))}{\partial x_i} &= \int_{\mathbf{z}} \frac{p(\mathbf{z}|\mathbf{d}, \mathbf{x}^m)}{p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m)} \frac{\partial p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m)}{\partial x_i} d\mathbf{z} \\ &= \int_{\mathbf{z}} \frac{1}{p(\mathbf{d}|\mathbf{x}^m)} \frac{\partial p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m)}{\partial x_i} d\mathbf{z} \end{aligned} \quad (4.32)$$

and

$$\begin{aligned}\frac{\partial(L(\mathbf{x}^m))}{\partial x_i} &= \frac{1}{\int_{\mathbf{z}} p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m) d\mathbf{z}} \int_{\mathbf{z}} \frac{\partial p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m)}{\partial x_i} d\mathbf{z} \\ &= \frac{1}{p(\mathbf{d}|\mathbf{x}^m)} \int_{\mathbf{z}} \frac{\partial p(\mathbf{d}, \mathbf{z}|\mathbf{x}^m)}{\partial x_i} d\mathbf{z},\end{aligned}\quad (4.33)$$

where $x_i = [\mathbf{x}]_i$, meaning that $-L^m(\mathbf{x})$ satisfies the differentiation condition at $\mathbf{x}^m$. Therefore, the $\mathbf{x}^m$ produced at convergence satisfies the KKT conditions of (4.27) according to Theorem 4.1.

Note that each approximate problem solved by the EM algorithm is not necessarily convex. This means that the SA method, which incorporates the EM algorithm as a special case, is indeed more general than the SCA method which solves convex approximate problems successively.

4.6 Application in GP

A GP in standard form is expressed as

$$\begin{aligned}\min_{\mathbf{x}} \quad & f_0(\mathbf{x}) \\ \text{s.t.} \quad & f_j(\mathbf{x}) \leq 1, j = 1, \dots, J, \\ & \mathbf{x} \in \mathcal{R}^N, x_i \geq 0, i = 1, \dots, N,\end{aligned}\quad (4.34)$$

where $\forall j, f_j(\mathbf{x})$ is a posynomial of $\mathbf{x}$.

Specifically, a posynomial of $\mathbf{x}$ can be expressed as

$$p(\mathbf{x}) = \sum_{a=1}^A u_a(\mathbf{x}) \text{ with } u_a(\mathbf{x}) = \beta_a \prod_{i=1}^N x_i^{\alpha_{ai}}$$

where $\beta_a > 0$ and α_{ai} ($i = 1, \dots, N$) are real constants. It is very important to note that

$$s(\mathbf{y}) = \log(p(e^{\mathbf{y}})) = \log\left(\sum_{a=1}^A e^{\mu_a(\mathbf{y})}\right)$$

where $\mu_a(\mathbf{y}) = \log(u_a(e^{\mathbf{y}})) = \log(\beta_a) + \sum_{i=1}^N \alpha_{ai} y_i$ is convex of $\mathbf{y} = [y_1, \dots, y_N]^T \in \mathcal{R}^N$ according to the analysis in Section 2.3.1. This means that although a GP in standard form is not convex, after making the COV from $\mathbf{x}$ to $\mathbf{y}$ satisfying $\mathbf{x} = e^{\mathbf{y}}$, it can be transformed to its equivalent convex form

$$\begin{aligned}\min_{\mathbf{y}} \quad & g_0(\mathbf{y}) = \log(f_0(e^{\mathbf{y}})) \\ \text{s.t.} \quad & g_j(\mathbf{y}) = \log(f_j(e^{\mathbf{y}})) \leq 0, j = 1, \dots, J,\end{aligned}\quad (4.35)$$

which can be solved by GP solvers (e.g. the MOSEK or gpcvx package). These solvers are able to solve large-scale problems based on the state-of-the-art IPM [BKVH05, KKMB06].

Suppose a constraint function $f_j(\mathbf{x})$ in (4.34) is not a posynomial but a ratio of two posynomials, say $f_j(\mathbf{x}) = \frac{q(\mathbf{x})}{p(\mathbf{x})}$ where $q(\mathbf{x})$ is also a posynomial of $\mathbf{x}$. This means that (4.34) is not a GP. It was shown in [CTP⁺07] that successive GP approximation could be used in this case, where the arithmetic-geometric mean inequality could be used to construct a GP approximation in standard form for (4.34). Specifically, the m th GP approximation is the same as (4.34) except that $f_j(\mathbf{x})$ is replaced by

$$f_j^m(\mathbf{x}) = \frac{q(\mathbf{x})}{\prod_{a=1}^A \left(\frac{u_a(\mathbf{x})}{\gamma_a^m}\right)^{\gamma_a^m}} = \frac{q(\mathbf{x})}{\prod_{a=1}^A (p(\mathbf{x}^m))^{\gamma_a^m} \left(\frac{u_a(\mathbf{x})}{u_a(\mathbf{x}^m)}\right)^{\gamma_a^m}} = \frac{q(\mathbf{x})}{p(\mathbf{x}^m) \prod_{a=1}^A \left(\frac{u_a(\mathbf{x})}{u_a(\mathbf{x}^m)}\right)^{\gamma_a^m}} \quad (4.36)$$

where $\gamma_a^m = \frac{u_a(\mathbf{x}^m)}{p(\mathbf{x}^m)}$ and $\mathbf{x}^m$ is the solution to the GP approximation at the $(m-1)$ th iteration. It can be shown by simple mathematical arrangements that $\forall m$, $f_j^m(\mathbf{x})$ satisfies the tightness and differentiation conditions at $\mathbf{x}^m$.

That GP approximation can also be derived from applying the CUBA construction method to build the m th approximation of (4.35), where $g_j(\mathbf{y})$ corresponding to that $f_j(\mathbf{x})$ is expressed as

$$g_j(\mathbf{y}) = \log(q(e^{\mathbf{y}})) - s(\mathbf{y}).$$

The m th convex approximation of (4.35) is the same as (4.35) except that $g_j(\mathbf{y})$ is replaced with its CUBA $g_j^m(\mathbf{y})$ built around $\mathbf{y}^m$. Obviously, the first part of $g_j(\mathbf{y})$ is convex of $\mathbf{y}$, and $-s(\mathbf{y})$ can be treated as a composition of $h(\mathbf{z}) = -\log(\sum_{a=1}^A e^{z_a})$ that is decreasing and concave of $\mathbf{z} = [z_1, \dots, z_A]^T$, and $z_a = \mu_a(\mathbf{y})$ that can be regarded as a concave function of $\mathbf{y}$. Therefore, the CAVF method can be used to construct a CUBA for $-s(\mathbf{y})$. Based on the above idea, $g_j^m(\mathbf{y})$ can be derived as

$$g_j^m(\mathbf{y}) = \log(q(e^{\mathbf{y}})) - s(\mathbf{y}^m) - \sum_{a=1}^A \gamma_a^m (\mu_a(\mathbf{y}) - \mu_a(\mathbf{y}^m)).$$

Obviously, $g_j^m(\mathbf{y}) = \log(f_j^m(e^{\mathbf{y}}))$. This means that the m th approximate problem with respect to $\mathbf{y}$ derived from the CUBA construction method is equivalent to the one with respect to $\mathbf{x}$ developed in [CTP⁺07].

Successive convex approximation based DSM algorithms

5

5.1 Related works and chapter overview

Dynamic power spectrum management (DSM) for multiuser multicarrier systems has lately been attracting much research interest, especially in the context of crosstalk-corrupted digital subcarrier line systems. The reported works can be categorized into two classes. The first one is to formulate the problem as a noncooperative Nash game with multiple conflicting objectives. Along this direction, significant effort has been made to study existence and uniqueness of a Nash equilibrium solution, derive algorithms and establish convergence conditions for them (see e.g. [YGC02, LP12, SPB08]). The second one is to optimize a certain system-wide performance metric subject to resource constraints, e.g., the WSR of all receivers subject to per-transmitter sum power constraints.

This chapter addresses the WSR maximized DSM problem subject to per transmitter sum power constraints. The DSM problem is NP-hard due to its nonconvexity [LZ08]. When there is a big number of carriers, the duality gap of this problem was proven to be zero [YL06]. In this case, duality based algorithms, e.g., Optimum Spectrum Balancing (OSB) and Iterative Spectrum Balancing (ISB) algorithms [CM05, LY05, YL06], were proposed.

Lately, a Successive Convex Approximations for Low-complexity (SCALE) algorithm, which is applicable independently of the problem's duality gap, was proposed in [PE06, PE09]. This algorithm solves convex approximations of the DSM problem successively, each derived from an inequality shown later. Based on the same idea, a Convex Approximation for Dis-

tributed Spectrum Balancing (CA-DSB) algorithm was proposed in [TDM08]. For both algorithms, low-complexity water-filling (WF) like implementation approaches were proposed according to the KKT conditions of each approximate problem. Based on the DCA proposed in [Pha88], a new approach was proposed in [HLLN] recently. This DCA based DSM (DCA-DSM) algorithm is similar to the CA-DSB algorithm, in that both were developed from writing the WSR as a difference of two convex or concave functions.

In the rest of this chapter, the system model and DSM problem are first presented. Next, the SCALE, CA-DSB and DCA-DSM algorithms are revisited to show how they can be derived from the SCA and CUBA construction methods introduced in Chapter 4. Finally, the performance and complexity of these algorithms are compared with those of the ISB algorithm and the modified iterative water-filling (MIWF) algorithm, which was developed in [Yu07] based on directly pursuing the power allocation satisfying the KKT conditions of the DSM problem.

5.2 System model and DSM problem

The system under consideration consists of K pairs of transmitter and receiver communicating simultaneously with OFDM using N carriers with cochannel interference. Each of the transmitter and receiver is equipped with a single antenna. Transmitter k ($k = 1, \dots, K$) emits data symbols to receiver k over all carriers, using the transmission power $p_{k,n}\bar{p}_k$ for carrier n ($n = 1, \dots, N$) where $\bar{p}_k$ and $p_{k,n}$ represent the sum power and fraction of $\bar{p}_k$ assigned to that carrier. To facilitate analysis, all power variables are stacked into the $K \times N$ matrix $\mathbf{P}$ with $[\mathbf{P}]_{k,n} = p_{k,n}$. The channel power gain at carrier n from transmitter l to receiver k is denoted by $g_{k,l,n} > 0$, and the additive white Gaussian noise (AWGN) power at carrier n of receiver k is $\sigma_{k,n}^2$. For illustration purposes, an example of two interfering systems is shown in Figure 5.1.

Assume every receiver decodes its own data by treating interference as noise. Every coherence period in which all channels remain unchanged is sufficiently long, so that a DSM algorithm can be implemented to optimize $\mathbf{P}$. The SNR at carrier n of receiver k can be expressed as

$$\gamma_{k,n}(\mathbf{P}) = \frac{G_{k,k,n}p_{k,n}}{I_{k,n}(\mathbf{P})} \quad (5.1)$$

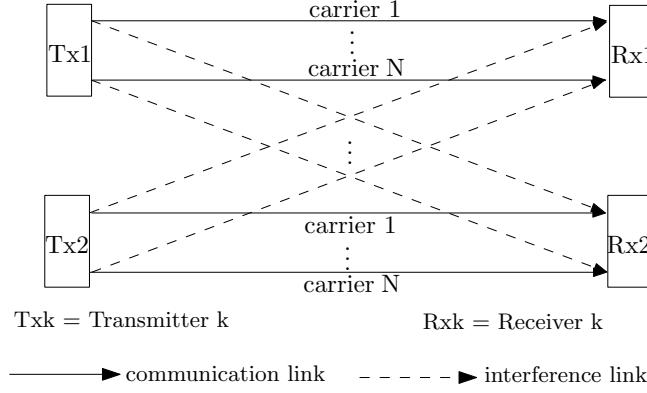


Figure 5.1 An example of interfering multicarrier systems when $K = 2$.

where $G_{k,l,n} = g_{k,l,n}\bar{p}_l$ and $I_{k,n}(\mathbf{P}) = \sigma_{k,n}^2 + \sum_{l \neq k} G_{k,l,n}p_{l,n}$. Note that (5.1) is equivalent to

$$p_{k,n} - \gamma_{k,n}(\mathbf{P}) \sum_{l \neq k} \frac{G_{k,l,n}}{G_{k,k,n}} p_{l,n} = \gamma_{k,n}(\mathbf{P}) \frac{\sigma_{k,n}^2}{G_{k,k,n}}. \quad (5.2)$$

Let's stack $\gamma_{k,n}(\mathbf{P}), \forall k, n$ into a $K \times N$ vector $\boldsymbol{\Gamma}(\mathbf{P})$ with $[\boldsymbol{\Gamma}(\mathbf{P})]_{k,n} = \gamma_{k,n}(\mathbf{P})$. Let's denote the n th column of $\mathbf{P}$ and $\boldsymbol{\Gamma}(\mathbf{P})$ by $\mathbf{p}_n$ and $\boldsymbol{\gamma}_n(\mathbf{P})$, respectively. From (5.2), it can easily be seen that $\boldsymbol{\gamma}_n(\mathbf{P})$ and $\mathbf{p}_n$ satisfy

$$(\mathbf{I} - \boldsymbol{\Lambda}(\boldsymbol{\gamma}_n(\mathbf{P}))\mathbf{G}_n)\mathbf{p}_n = \boldsymbol{\Lambda}(\boldsymbol{\gamma}_n(\mathbf{P}))\boldsymbol{\eta}_n \quad (5.3)$$

where $\boldsymbol{\Lambda}(\boldsymbol{\gamma}_n(\mathbf{P}))$ is a diagonal matrix with the k th diagonal entry equal to $\gamma_{k,n}(\mathbf{P})$. $\boldsymbol{\eta}_n$ is a vector with $[\boldsymbol{\eta}_n]_k = \frac{\sigma_{k,n}^2}{G_{k,k,n}}$. and $\mathbf{G}_n$ is a matrix with all diagonal entries equal to zero and $[\mathbf{G}_n]_{k,l} = \frac{G_{k,l,n}}{G_{k,k,n}}$ if $k \neq l$. Obviously, $\forall n$, $\mathbf{G}_n$ is nonnegative.

As said earlier, the DSM problem under consideration is to maximize the WSR subject to per-transmitter sum power constraints, i.e.,

$$\begin{aligned} \min_{\mathbf{P}} \quad & -R(\mathbf{P}) = -\sum_{k,n} w_k \log\left(1 + \frac{\gamma_{k,n}(\mathbf{P})}{\Gamma}\right) \\ \text{s.t.} \quad & \sum_n p_{k,n} - 1 \leq 0, \forall k, \\ & p_{k,n} \in [0, \bar{p}_{k,n}], \forall k, n, \end{aligned} \quad (5.4)$$

where w_k , Γ and $\bar{p}_{k,n}$ represent the prescribed positive weight for the rate of receiver k , the SNR gap between the adopted modulation and coding scheme

and the one achieving channel capacity, and the ratio of the power mask for carrier n of transmitter k to $\bar{p}_k$. It is assumed that $\sum_n \bar{p}_{k,n} \geq 1$ to ensure the sum power constraint for transmitter k can not be removed.

The SCALE, CA-DSB and DCA-DSM algorithms are revisited in the following sections, to show how they can be derived from the SCA and CUBA construction methods. To facilitate the description of each algorithm hereafter, the superscript m put to a variable indicates that variable is produced at the end of the m th iteration ($m \geq 0$). Moreover, the following functions are defined:

$$A_{k,n}(\mathbf{P}) = w_k \log(G_{k,k,n} p_{k,n} + \Gamma I_{k,n}(\mathbf{P})), \quad (5.5)$$

$$B_{k,n}(\mathbf{P}) = w_k \log(\Gamma I_{k,n}(\mathbf{P})), \quad (5.6)$$

$$C_{k,n}(\mathbf{P}) = \frac{\partial(\sum_{l \neq k} A_{l,n}(\mathbf{P}))}{\partial p_{k,n}} = \sum_{l \neq k} \frac{w_l G_{l,k,n} \Gamma}{G_{l,l,n} p_{l,n} + \Gamma I_{l,n}(\mathbf{P})}, \quad (5.7)$$

$$D_{k,n}(\mathbf{P}) = \frac{\partial(\sum_{l \neq k} B_{l,n}(\mathbf{P}))}{\partial p_{k,n}} = \sum_{l \neq k} \frac{w_l G_{l,k,n}}{I_{l,n}(\mathbf{P})}, \quad (5.8)$$

$$E_{k,n}(\mathbf{P}) = \frac{\partial A_{k,n}(\mathbf{P})}{\partial p_{k,n}} = \frac{w_k}{p_{k,n} + \frac{\Gamma}{G_{k,k,n}} I_{k,n}(\mathbf{P})}. \quad (5.9)$$

5.3 Revisit of the SCALE algorithm

The SCALE algorithm was initially derived in [PE06] based on successively maximizing lower bounds of $R(\mathbf{P})$, each built from the inequality

$$\log(1+z) \geq \log(1+z_0) + \frac{z_0}{1+z_0} \log\left(\frac{z}{z_0}\right). \quad (5.10)$$

Specifically, the m th iteration of the algorithm is to find $\mathbf{P}^{m+1}$ that maximizes

$$R_s^m(\mathbf{P}) = R(\mathbf{P}^m) + \sum_{k,n} w_k \alpha_{k,n}^m \log \frac{\gamma_{k,n}(\mathbf{P})}{\gamma_{k,n}(\mathbf{P}^m)} \quad (5.11)$$

where $\alpha_{k,n}^m = \frac{\gamma_{k,n}(\mathbf{P}^m)}{\Gamma + \gamma_{k,n}(\mathbf{P}^m)}$ subject to per transmitter sum power constraints. According to the inequality, $R_s^m(\mathbf{P})$ is a lower bound of $R(\mathbf{P})$ satisfying the tightness condition at $\mathbf{P}^m$. Moreover, after making the COV from $\mathbf{P}$ to $\mathbf{Q}$ satisfying $\mathbf{P} = e^{\mathbf{Q}}$, i.e., $\mathbf{Q}$ is a $K \times N$ matrix with $[\mathbf{Q}]_{k,n} = q_{k,n} = \log(p_{k,n})$, the m th problem becomes convex since $\forall k, n, \alpha_{k,n}^m \geq 0$ and

$$\log(\gamma_{k,n}(e^{\mathbf{Q}})) = q_{k,n} + \log(G_{k,k,n}) - \log(I_{k,n}(e^{\mathbf{Q}}))$$

is concave of $\mathbf{Q}$ since $\log(I_{k,n}(e^{\mathbf{Q}}))$, as a log-weighted-sum-exponential function, is convex of $\mathbf{Q}$ as said in Section 2.3.1.

The SCALE algorithm can also be derived from the SCA and CUBA construction methods. To this end, the COV from $\mathbf{P}$ to $\mathbf{Q}$ is first made. Now, the objective function of (5.4) becomes $g_0(\mathbf{Q}) = -R(e^{\mathbf{Q}})$, which can be regarded as the composition of

$$h(\Phi) = -\sum_{k,n} w_k \log(1 + \frac{e^{\phi_{k,n}}}{\Gamma}) \quad (5.12)$$

which is strictly concave of Φ and decreasing with $\phi_{k,n}$, $\forall k, n$, (Φ is a $K \times N$ matrix with $[\Phi]_{k,n} = \phi_{k,n}$), and $\phi_{k,n} = \log(\gamma_{k,n}(e^{\mathbf{Q}}))$, $\forall k, n$, which is concave of $\mathbf{Q}$. Thus, a CUBA of $g_0(\mathbf{Q})$ can be built from the CAVF method as

$$g_s^m(\mathbf{Q}) = -R(e^{\mathbf{Q}^m}) - \sum_{k,n} w_k a_{k,n}^m \left(\log(\gamma_{k,n}(e^{\mathbf{Q}})) - \log(\gamma_{k,n}(e^{\mathbf{Q}^m})) \right). \quad (5.13)$$

Obviously, $g_s^m(\mathbf{Q}) = -R_s^m(e^{\mathbf{Q}})$, thus the m th iteration of the SCA method is to minimize $-R_s^m(e^{\mathbf{Q}})$ subject to per transmitter sum power constraints. This means that the SCALE algorithm indeed can be derived from the SCA and CUBA construction methods.

If $g_s^m(\mathbf{Q})$ is a strict CUBA of $g_0(\mathbf{Q})$ except at $\mathbf{Q} = \mathbf{Q}^m$, $g_0(\mathbf{Q}^m) < g_0(\mathbf{Q}^{m+1})$ if $\mathbf{Q}^m \neq \mathbf{Q}^{m+1}$, and $g_0(\mathbf{Q}^m) = g_0(\mathbf{Q}^{m+1})$ if and only if $\mathbf{Q}^m = \mathbf{Q}^{m+1}$ according to the first claim of Theorem 4.2. Since $h(\Phi)$ is strictly concave of Φ , $g_s^m(\mathbf{Q})$ is a strict CUBA of $g_0(\mathbf{Q})$ except at $\mathbf{Q} = \mathbf{Q}^m$, provided that $\Gamma(\mathbf{P})$ is a one-to-one mapping from $\mathbf{P} \in \mathcal{C}_{\mathbf{P}}$ to $\Gamma \in \mathcal{C}_{\Gamma}$, where $\mathcal{C}_{\mathbf{P}} = \{\mathbf{P} | \mathbf{P} \succ \mathbf{0}\}$ and $\mathcal{C}_{\Gamma} = \{\Gamma(\mathbf{P}) | \mathbf{P} \in \mathcal{C}_{\mathbf{P}}\}$. In Section 5.7, it is shown that $\Gamma(\mathbf{P})$ is a one-to-one mapping from $\mathbf{P} \in \mathcal{C}_{\mathbf{P}}$ to $\Gamma \in \mathcal{C}_{\Gamma}$, under the assumption that $\forall n$, $\mathbf{G}_n$ is *primitive*, i.e., there exists a positive integer i such that $(\mathbf{G}_n)^i \succ \mathbf{0}$ according to Theorem 8.5.2 of [HJ85]. Note that this assumption is mild since it can readily be satisfied in practice.

Theoretical analysis has been made in [PE06, PE09] to show that when $f(\mathbf{P}^{m+1}) = f(\mathbf{P}^m)$, $\mathbf{P}^m$ satisfies the KKT conditions for (5.4). In fact, this can also be proved according to the second claim of Theorem 4.2, because $g_s^m(\mathbf{Q})$ is a CUBA satisfying the tightness and differentiation conditions, and the COV from $\mathbf{P}$ to $\mathbf{Q}$ satisfies the conditions specified there.

To implement the SCALE algorithm, a low-complexity approach¹ was proposed in [PE06, PE09]. The key for this approach is to solve the m th ap-

¹This approach was also summarized as Algorithm 2 in [TDM08].

proximate problem based on its KKT conditions. To this end, let's define the Lagrange multipliers corresponding to the constraints $\sum_n e^{q_{k,n}} - 1 \leq 0$, $q_{k,n} \leq \log(\bar{p}_{k,n})$ and $q_{k,n} \geq \delta$ as λ_k^s , η_k^s and μ_k^s , respectively. Then $\mathbf{P} = e^{\mathbf{Q}^{m+1}}$ and the optimum Lagrange multipliers should satisfy:

$$\forall k, n, p_{k,n} \left(\lambda_k^s + \sum_{l \neq k} G_{l,k,n} \frac{\alpha_{l,n}^m}{I_{l,n}(\mathbf{P})} \right) = \alpha_{k,n}^m - \eta_k^s + \mu_k^s, \quad (5.14)$$

$$\forall k, n, \eta_k^s(p_{k,n} - \bar{p}_{k,n}) = 0, \mu_k^s(p_{k,n} - \delta) = 0, \quad (5.15)$$

$$\forall k, \lambda_k^s \left(\sum_n p_{k,n} - 1 \right) = 0. \quad (5.16)$$

It can readily be shown that those conditions are equivalent to

$$\forall k, n, p_{k,n} = \Omega_{kn}^m(\mathbf{P}, \lambda_k^s), \text{ and } \forall k, \lambda_k^s \left(\sum_n p_{k,n} - 1 \right) = 0, \quad (5.17)$$

where

$$\Omega_{kn}^m(\mathbf{P}, \lambda_k^s) = \left[\frac{\alpha_{k,n}^m}{\lambda_k^s + \sum_{l \neq k} w_k G_{l,k,n} \frac{\alpha_{l,n}^m}{I_{l,n}(\mathbf{P})}} \right]_{\delta}^{\bar{p}_{k,n}} \quad (5.18)$$

The low-complexity approach is summarized in Algorithm 2, where $\epsilon > 0$ is a very small tolerance prescribed. At the beginning, $p_{k,n}^0 = 0$ and $\alpha_{k,n}^0 = 1$, $\forall k, n$ are used. At the m th iteration, the $\lambda_k^s \geq 0$ satisfying

$$\lambda_k^s \left(\sum_n \Omega_{kn}^m(\mathbf{P}^m, \lambda_k^s) - 1 \right) = 0$$

is found, then $p_{k,n}^{m+1}$ is assigned with $\Omega_{kn}^m(\mathbf{P}^m, \lambda_k^s)$, $\forall k, n$, after which the $(m+1)$ th iteration starts. The above iteration proceeds until $\mathbf{P}^m$ converges or prescribed M iterations have been executed. It was shown in [HLLN] that the SCALE algorithm has a complexity of $O(K^2N)$.

Note that it was not proved if the low-complexity implementation approach for the SCALE algorithm always has guaranteed convergence as M approaches infinity [PE06, PE09], even though in practice convergence is usually observed. Nevertheless, every $\mathbf{P}^m$ computed satisfies the per-transmitter sum power constraints, meaning that this approach can be implemented in a distributed manner with message passing as elaborated in [PE06, PE09]. Specifically, every transmitter k collects the channel gain $\{G_{l,k,n} | \forall k\}$ at the

beginning by certain methods, e.g., by feedback from receivers or estimating uplink channel power gains if the channels are reciprocal. At each iteration, every receiver k can measure $\{I_{k,n}(\mathbf{P})|\forall n\}$ (or equivalently $\{\gamma_{k,n}(\mathbf{P})|\forall n\}$) and feed them back to all transmitters, then every transmitter k can locally update $\{p_{k,n}^m|\forall n\}$ according to Algorithm 2.

Algorithm 2 Implementation of the SCALE algorithm.

```

 $m = 0; \forall k, n, p_{k,n}^0 = 0; \alpha_{k,n}^0 = 1;$ 
repeat
  for  $k = 1 \rightarrow K$  do
    compute  $p_{k,n} = \Omega_{kn}^m(\mathbf{P}^m, 0), \forall n;$ 
    if  $\sum_n p_{k,n} \leq 1$  then
       $p_{k,n}^{m+1} = p_{k,n}, \forall n;$ 
    else
       $\lambda_{\min} = 0;$ 
      find  $\lambda_{\max} > 0$  satisfying  $\sum_n \Omega_{kn}^m(\mathbf{P}^m, \lambda_{\max}) < 1;$ 
      repeat
         $\lambda = \frac{\lambda_{\min} + \lambda_{\max}}{2};$ 
        compute  $p_{k,n} = \Omega_{kn}^m(\mathbf{P}^m, \lambda), \forall n;$ 
        if  $\sum_n p_{k,n} < 1 - \epsilon$  then
           $\lambda_{\max} = \lambda;$ 
        else if  $\sum_n p_{k,n} > 1$  then
           $\lambda_{\min} = \lambda;$ 
        end if
      until  $1 - \epsilon \leq \sum_n p_{k,n} \leq 1$ 
       $p_{k,n}^{m+1} = p_{k,n}, \forall n;$ 
    end if
  end for
   $\forall k, n, \text{ compute } \alpha_{k,n}^{m+1} = \frac{\gamma_{k,n}(\mathbf{P}^{m+1})}{\Gamma + \gamma_{k,n}(\mathbf{P}^{m+1})}; m = m + 1;$ 
until  $\mathbf{P}^m = \mathbf{P}^{m-1}$  or  $m = M$ 
output  $\mathbf{P}^m$  as a solution to (5.4).

```

5.4 Revisit of the CA-DSB algorithm

The CA-DSB algorithm was initially derived in [TDM08] based on the same idea as for the SCALE algorithm, i.e., maximizing lower bounds of $R(\mathbf{P})$, each built from writing $R(\mathbf{P})$ as

$$R(\mathbf{P}) = \sum_{k,n} A_{k,n}(\mathbf{P}) - \sum_{k,n} B_{k,n}(\mathbf{P}). \quad (5.19)$$

Based on the convexity of $-\sum_{k,n} B_{k,n}(\mathbf{P})$ with respect to $\mathbf{P}$, the lower bound of $R(\mathbf{P})$ maximized in the m th iteration is

$$R_c^m(\mathbf{P}) = \sum_{k,n} (A_{k,n}(\mathbf{P}) - B_{k,n}(\mathbf{P}^m) - D_{k,n}(\mathbf{P}^m)(p_{k,n} - p_{k,n}^m)). \quad (5.20)$$

Obviously, the CA-DSB algorithm is a special case of the SCA method introduced in Chapter 4, since $-R_c^m(\mathbf{P})$ corresponds to a CUBA of $-R(\mathbf{P})$ that can be built by the concavity-based method described in Section 4.3. It was shown in [TDM08] that when $R(\mathbf{P}^m) = R(\mathbf{P}^{m+1})$ holds, $\mathbf{P}^m$ satisfies the KKT conditions of (5.4). In fact, this can also be proved according to Theorem 4.1.

To implement the CA-DSB algorithm, a WF-like low-complexity approach² was developed in [TDM08]. The key for this approach is to solve (??) based on its KKT conditions. To this end, let's define the Lagrange multipliers corresponding to the constraints $\sum_n p_{k,n} - 1 \leq 0$, $p_{k,n} \leq \bar{p}_{k,n}$ and $p_{k,n} \geq 0$ as λ_k^c , $\eta_{k,n}^c$ and $\mu_{k,n}^c$, respectively. Then $\mathbf{P} = \mathbf{P}^{m+1}$ and the optimum Lagrange multipliers should satisfy:

$$\forall k, n, \lambda_k^c + D_{k,n}(\mathbf{P}^m) - C_{k,n}(\mathbf{P}) = E_{k,n}(\mathbf{P}) + \mu_{k,n}^c - \eta_{k,n}^c, \quad (5.21)$$

$$\forall k, n, \mu_{k,n}^c p_{k,n} = 0, \eta_{k,n}^c (p_{k,n} - \bar{p}_{k,n}) = 0, \quad (5.22)$$

$$\forall k, \lambda_k^c (\sum_n p_{k,n} - 1) = 0. \quad (5.23)$$

Moreover, it can readily be shown that those conditions are equivalent to

$$p_{k,n} = \Psi_{k,n}^m(\mathbf{P}, \lambda_k^c) \text{ and } \forall k, \lambda_k^c (\sum_n p_{k,n} - 1) = 0, \quad (5.24)$$

where

$$\Psi_{k,n}^m(\mathbf{P}, \lambda_k^c) = \left[\frac{w_k}{\lambda_k^c + D_{k,n}(\mathbf{P}^m) - C_{k,n}(\mathbf{P})} - \frac{\Gamma}{G_{k,k,n}} I_{k,n}(\mathbf{P}) \right]_0^{\bar{p}_{k,n}}. \quad (5.25)$$

²This approach was summarized as Algorithm 4 in [TDM08].

The WF-like implementation approach is summarized in Algorithm 3. At the beginning, $p_{k,n}^0 = 0, \forall k, n$ is used. At the m th iteration, $\forall k$, the $\lambda_k^c \geq 0$ satisfying

$$\lambda_k^c \left(\sum_n \Psi_{k,n}^m(\mathbf{P}^m, \lambda_k^c) - 1 \right) = 0 \quad (5.26)$$

is found in the same way as described for the SCALE algorithm, then $p_{k,n}^{m+1}$ is updated with $\Psi_{k,n}^m(\mathbf{P}^m, \lambda_k^c), \forall k, n$, after which the $(m+1)$ th iteration starts. The complexity of the CA-DSB algorithm scales up with respect to K and N in a similar way as that of the SCALE algorithm. Note that it was not proved if Algorithm 3 always has guaranteed convergence as M approaches infinity, even though in practice convergence is usually observed. Nevertheless, every $\mathbf{P}^m$ computed satisfies the per-transmitter sum power constraints, meaning that Algorithm 3 can also be implemented in a distributed manner as Algorithm 2.

Algorithm 3 Implementation of the CA-DSB algorithm.

```

 $m = 0; \forall k, n, p_{k,n}^0 = 0;$ 
repeat
  for  $k = 1 \rightarrow K$  do
    find the  $\lambda_k^c \geq 0$  satisfying (5.26) in the same way as described for the
    SCALE algorithm,
     $p_{k,n}^{m+1} = \Psi_{k,n}^m(\mathbf{P}^m, \lambda_k^c), \forall n,$ 
  end for
until  $\mathbf{P}^m = \mathbf{P}^{m-1}$  or  $m = M$ 
output  $\mathbf{P}^m$  as a solution to (5.4).

```

5.5 Revisit of the DCA-DSM algorithm

The DCA-DSM algorithm³ was initially derived in [HLLN] by writing $-R(\mathbf{P})$ as

$$-R(\mathbf{P}) = S_1(\mathbf{P}) - (S_1(\mathbf{P}) + R(\mathbf{P})), \quad (5.27)$$

³This algorithm was summarized as Algorithm 8 in [HLLN].

where $S_1(\mathbf{P}) = \frac{\xi_d}{2} \sum_{k,n} p_{k,n}^2$. It can easily be shown that when $\xi_d \geq \max_{\mathbf{P}} \rho(\nabla_{\mathbf{P}}^2 R(\mathbf{P}), \nabla_{\mathbf{P}}^2 (S_1(\mathbf{P}) + R(\mathbf{P}))) \succeq \mathbf{0}$ holds $\forall \mathbf{P}$, meaning that $S_1(\mathbf{P}) + R(\mathbf{P})$ is convex of $\mathbf{P}$. Then the DCA algorithm is used to minimize $-R(\mathbf{P})$.

Specifically, the m th iteration of the DCA-DSM algorithm is to solve the quadratic program

$$\begin{aligned} \min_{\mathbf{P}} \quad & S_1(\mathbf{P}) - \sum_{k,n} (\xi_d p_{k,n}^m + E_{k,n}(\mathbf{P}^m) + C_{k,n}(\mathbf{P}^m) - D_{k,n}(\mathbf{P}^m)) p_{k,n} \\ \text{s.t.} \quad & \sum_n p_{k,n} - 1 \leq 0, \forall k, \\ & p_{k,n} \in [0, \bar{p}_{k,n}], \forall k, n, \end{aligned} \quad (5.28)$$

and use its optimum solution as $\mathbf{P}^{m+1}$.

Specifically, minimizing the objective function of (5.28) is equivalent to minimizing a CUBA of $-R(\mathbf{P})$ constructed with the REXF method introduced in Section 4.3. This means that the DCA-DSM algorithm can also be derived by using the SCA method. It was shown in [HLLN] that $\mathbf{P}^m$ satisfies the KKT conditions of (5.4) when $R(\mathbf{P}^m) = R(\mathbf{P}^{m+1})$ holds. This can also be derived according to Theorem 4.1.

Note that ξ_d needs to be carefully chosen for the DCA-DSM algorithm. However, it is hard to determine analytically which value of ξ_d leads to the best performance. In practice, extensive off-line simulations have to be carried out to choose a ξ_d corresponding to a good average WSR.

Algorithm 4 The DCA-DSM algorithm.

```

 $m = 0; \forall k, n, p_{k,n}^0 = 0;$ 
repeat
  for  $k = 1 \rightarrow K$  do
    find the  $\lambda_k^d \geq 0$  satisfying (5.29) in the same way as described for the
    SCALE algorithm,
     $p_{k,n}^{m+1} = \Pi_{k,n}^m(\mathbf{P}^m, \lambda_k^d), \forall n,$ 
  end for
until  $\mathbf{P}^m = \mathbf{P}^{m-1}$  or  $m = M$ 
output  $\mathbf{P}^m$  as a solution to (5.4).

```

(5.28) can be easily solved based on its KKT conditions. To this end, let's define the Lagrange multiplier corresponding to the constraint $\sum_n p_{k,n} - 1 \leq 0$

as λ_k^d . It can readily be derived from the KKT conditions of (5.28) that $\forall k, n$, $p_{k,n}^{m+1} = \Pi_{k,n}^m(\mathbf{P}^m, \lambda_k^d)$ where

$$\Pi_{k,n}^m(\mathbf{P}^m, \lambda_k^d) = \left[p_{k,n}^m + \frac{1}{\xi_d} (E_{k,n}(\mathbf{P}^m) + C_{k,n}(\mathbf{P}^m) - D_{k,n}(\mathbf{P}^m) - \lambda_k^d) \right]_{0}^{\bar{p}_{k,n}},$$

and λ_k^d is the nonnegative value satisfying

$$\lambda_k^d \left(\sum_n \Pi_{k,n}^m(\mathbf{P}^m, \lambda_k^d) - 1 \right) = 0. \quad (5.29)$$

Based on these conditions, an implementation algorithm is given in Algorithm 4. This algorithm has guaranteed convergence if ξ_d is sufficiently large to ensure $S_1(\mathbf{P}) + R(\mathbf{P})$ is convex of $\mathbf{P}$. The complexity of the DCA-DSM algorithm scales up with respect to K and N in a similar way as that of the SCALE algorithm.

5.6 Numerical experiments

To illustrate the performance of the above mentioned DSM algorithms, numerical experiments have been carried out for a realistic scenario where $K = 15$ transmitter-receiver pairs use OFDM with $N = 64$ carriers to communicate simultaneously. Transmitter k and receiver k are located at coordinates $(x = 0, y = k)$ and $(x = 10, y = k)$, respectively. These coordinates are in the unit of meter. The parameters are set as $\Gamma = 0$ dB, $\epsilon = 10^{-5}$, $\forall k, n$, $w_k = 1$, $\sigma_{k,n}^2 = -50$ dBm and $\bar{p}_{k,n} = 0.8$. The following experiments are conducted with Matlab v7.1 on a laptop equipped with an Intel Duo CPU of 2.2 GHz and a memory of 2 GBytes.

Every channel is generated as follows. First, it is modeled as a 6-tap delay line, and the average received power at a distance of d is equal to d^{-3} . This means that the transmitted power is attenuated by 30 dB in average when received at a distance of 10 meter apart. Second, the amplitude of the i th tap is a circularly symmetric complex Gaussian random variable with zero mean and variance as $\sigma_i^2 = \sigma_0^2 e^{-3i}$, $i = 0, 1, \dots, 5$.

The performance of the SCALE, CA-DSB and DCA-DSM algorithms is compared with those of the MIWF algorithm and the ISB algorithm. Note that the ISB algorithm is based on finding the optimum dual variable $\kappa^* =$

$\arg \min_{\kappa \geq 0} d(\kappa)$ where

$$\begin{aligned} d(\kappa) = \max_{\mathbf{P}} \quad & L(\kappa, \mathbf{P}) = R(\mathbf{P}) + \sum_k \kappa_k (1 - \sum_n p_{k,n}) \\ \text{s.t.} \quad & p_{k,n} \in [0, \bar{p}_{k,n}], \forall k, n, \end{aligned} \quad (5.30)$$

and $\kappa = [\kappa_1, \dots, \kappa_K]$. The right-hand side of (5.30) is the LRP as said earlier in Chapter 3. When N is sufficiently large, a globally optimum $\mathbf{P}$ for the LRP when $\kappa = \kappa^*$ is also globally optimum for (5.4) according to the analysis in [YL06]. The OSB algorithm proposed in [CYM⁺06] uses nested bisections to find κ^* , and exhaustive grid search to find a globally optimum $\mathbf{P}$ for the LRP. To reduce complexity, the ISB algorithm in [YL06] uses subgradient based ellipsoid method to find κ^* , and computes the subgradient by solving the LRP through sequentially optimizing every $p_{k,n}$ with grid search over $[0, \bar{p}_{k,n}]$ (our implementation discretizes that range into 20 grid points for the search). This might lead to a locally optimum $\mathbf{P}$ for the LRP and thus imprecise subgradient. Therefore, the ISB algorithm is actually an approximate method that produces estimates of κ^* . However, it is more suitable for practical application, since it performs well as illustrated in [YL06] and runs much faster than the OSB algorithm which might take days for realistic scenarios.

When $\bar{p}_k = 70$ dBm, the average WSRs for the SCALE, CA-DSB and DCA-DSM algorithms over 200 random channel realizations have been computed,

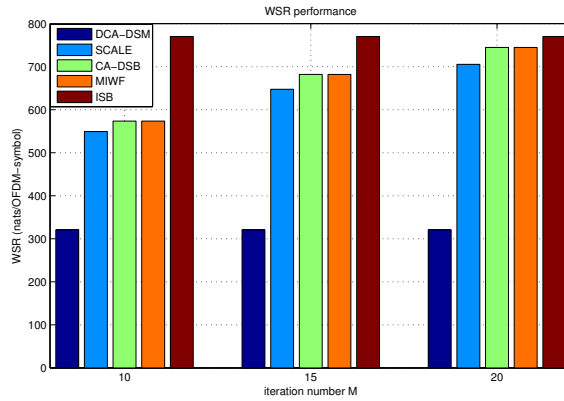


Figure 5.2 The average WSRs for all algorithms. In each bar group, the bars from the left to the right represent the performance of DCA-DSM, SCALE, CA-DSB, MIWF and ISB algorithms, respectively.

Table 5.1 Average execution time in seconds of each algorithm when $M = 20$

	SCALE	CA-DSB	DCA-DSM	MIWF	ISB
$K = 10, N = 64$	0.5	0.9	0.9	0.9	36.6
$K = 10, N = 128$	0.9	1.8	1.8	1.8	87.6
$K = 20, N = 64$	1.9	3.7	3.8	3.7	282.2
$K = 20, N = 128$	3.8	7.5	7.5	7.5	676.6

and are shown in Figure 5.2 when the iteration number M increases. Note that for the DCA-DSM algorithm, the average WSRs for different ξ_d between 0 and 2000 have been computed, and $\xi_d = 750$ corresponding to the best average WSR is chosen. The average WSR for the MIWF algorithm using the same iteration number, and that for the ISB algorithm is also shown in Figure 5.2. It can be seen that the SCALE, CA-DSB and MIWF algorithms lead to the similar average WSRs, which is higher than that for the DCA-DSM algorithm, when M is fixed. When M is small, the SCALE, CA-DSB and MIWF algorithms lead to smaller average WSRs than the ISB algorithm, while they lead to the similar average WSRs as the ISB algorithm when $M = 20$.

To compare the complexity of the studied algorithms, the average execution time of each algorithm is shown in Table 5.1 when different combinations of K and N are used. It can be seen that when K and N are fixed, the CA-DSB, DCA-DSM and MIWF algorithms have the similar execution time, which is higher than that for the SCALE algorithm, but lower than that for the ISB algorithm.

5.7 Appendix

Suppose $\forall n, \mathbf{G}_n$ is primitive. Proving $\Gamma(\mathbf{P})$ is a one-to-one mapping from $\mathbf{P} \in \mathcal{C}_{\mathbf{P}}$ to $\Gamma \in \mathcal{C}_{\Gamma}$, is equivalent to show $\mathbf{P}$ is the unique one mapping to the $\Gamma(\mathbf{P})$, or equivalently

$$\forall n, \mathbf{P} \in \mathcal{C}_{\mathbf{P}}, \mathbf{A}_n(\mathbf{P}) = \mathbf{I} - \Lambda(\gamma_n(\mathbf{P}))\mathbf{G}_n \quad (5.31)$$

is invertible. This is proved as follows:

Theorem 5.1. $\forall n, \mathbf{P} \in \mathcal{C}_{\mathbf{P}}, \mathbf{A}_n(\mathbf{P})$ is invertible. The unique $\mathbf{P}$ mapping to $\mathbf{\Gamma}$ can be computed as

$$\mathbf{p}_n = \sum_{i=0}^{+\infty} ((\mathbf{\Lambda}(\gamma_n)\mathbf{G}_n)^i \mathbf{\Lambda}(\gamma_n)\boldsymbol{\eta}_n), \forall n. \quad (5.32)$$

where γ_n is the n th column of $\mathbf{\Gamma}$.

Proof. It can readily be shown that $\forall n, \gamma_n(\mathbf{P}) \succ \mathbf{0}$ when $\mathbf{P} \in \mathcal{C}_{\mathbf{P}}$. Hence, $\mathbf{\Lambda}(\gamma_n)\mathbf{G}_n$ is primitive according to Theorem 8.5.2 of [HJ85]. Moreover, it can easily be seen that $\mathbf{p}_n \succ (\mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n)\mathbf{p}_n$. This means that $\rho(\mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n) < 1$ according to Corollary 8.1.29 in [HJ85], where $\rho(\mathbf{X})$ is the maximum magnitude of the eigenvalues, which are in general complex, of the matrix $\mathbf{X}$. Therefore,

$$\lim_{i \rightarrow \infty} (\mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n)^i = \lim_{i \rightarrow \infty} \rho^i \mathbf{L} = \mathbf{0} \quad (5.33)$$

follows from Theorem 8.5.1 in [HJ85], where $\mathbf{L}$ is a deterministic matrix depending on $\mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n$. Note that this theorem is a generalization of Peron's Theorem from the class of positive matrices to that of nonnegative primitive matrices. This means that $\forall n, \forall \mathbf{P} \in \mathcal{C}_{\mathbf{P}}, \mathbf{A}_n(\mathbf{P}) = \mathbf{I} - \mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n$ is nonsingular, since its inverse exists and can be computed from Neuman Series, i.e., $(\mathbf{A}_n(\mathbf{P}))^{-1} = \sum_{i=0}^{+\infty} (\mathbf{\Lambda}(\gamma_n(\mathbf{P}))\mathbf{G}_n)^i$ [Ste98]. Therefore, (5.32) follows. $\square$

WSMR maximized resource allocation in multi-cell OFDM-based systems

6

6.1 Related works and chapter overview

This chapter considers joint RA for BS coordination in the downlink of a multi-cell OFDM-based system. Recently, some margin-adaptive joint RA algorithms have been proposed for the downlink of multi-cell OFDMA systems [PJ04, AADM07]. In case that the joint RA is optimized for each carrier individually, the rate-adaptive algorithms in [KOG07, KG08, GGbK08] can be adopted. However, a joint RA over all carriers is preferred in order to better exploit the frequency and multiuser diversity inherent in OFDMA systems. To this end, two rate-adaptive joint RA algorithms have been proposed in [VPW07] and [VPW08]. Both algorithms optimize carrier allocation and power allocation iteratively, such that the WSR of all users keeps increasing until convergence. In particular, the weight of each user represents the priority of that user's rate. However, it is not clear how to determine the weights so as to fulfill a prescribed fairness criterion among users.

In this chapter, a novel RA problem is considered to maximize the WSMR of coordinated cells for the downlink of a multi-cell OFDMA system subject to per BS sum power constraints. Most interestingly, this problem leads to a RA that guarantees users in each cell have similar rates, which is of interest in practice. An coordinate-ascent (CA) based iterative algorithm is proposed to optimize the carrier allocation and the power allocation alternatively, so that the WSMR keeps increasing until convergence. In each iteration, the

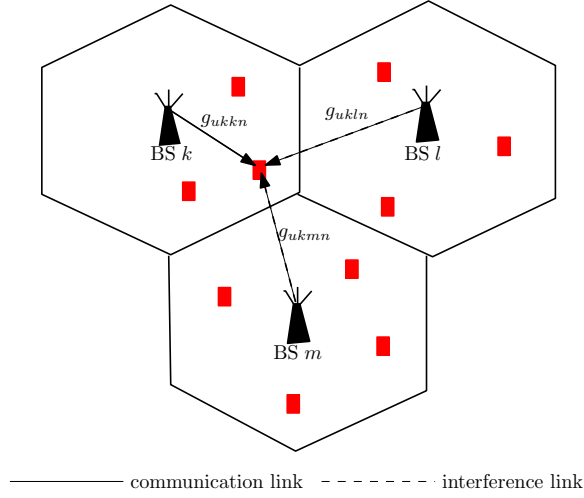


Figure 6.1 An example system setup.

power allocation is optimized by fixing the carrier allocation and using the SCA method introduced in Chapter 4. After that, the carrier allocation is updated by fixing the power allocation and solving a linear program for each cell.

The rest of this chapter is organized as follows. First, the system model and the RA problem are described. Then, the proposed algorithm is presented. Finally, the effectiveness of the proposed algorithm is illustrated by numerical experiments.

6.2 System model and problem formulation

Consider the downlink of a multi-cell OFDM system with K BSs coordinated by a central controller running the proposed RA algorithm. Using OFDM with N carriers, BS k serves a group of users belonging to the user set $\mathcal{U}_k$ for cell k . The channel power gain of carrier n from BS l to a user u in cell k is denoted as $g_{u,k,l,n}$, and assume $g_{u,k,l,n} > 0, \forall u \in \mathcal{U}_k, k, l, n$. $\sigma_{u,k,n}^2$ represents the total power of the AWGN and the interfering signals coming from uncoordinated BSs for carrier n at user u in cell k . Figure 6.1 shows an example system with 3 cells for illustration purposes.

The following assumptions are made for the considered system. First, the controller is assumed to know $\{g_{u,k,l,n} | \forall u, k, l, n\}$ and $\{\sigma_{u,k,n}^2 | \forall u, k, n\}$ pre-

cisely. Second, the channel between each BS-user pair is assumed to remain static over a sufficiently long duration, so that the RA can be carried out for that duration.

Power-allocation related notations are denoted as follow. $\bar{P}_k$ denotes the sum power available to BS k , and $p_{k,n}$ represents the proportion of $\bar{P}_k$ allocated to carrier n by BS k . $p_{k,n}, \forall k, n$ are stacked into a $K \times N$ matrix $\mathbf{P}$ with $p_{k,n} = [\mathbf{P}]_{kn}$.

The following notations related to the carrier allocation are defined. A continuous variable $A_{u,k,n}$ which is between 0 and 1 is defined to represent the proportion of the whole transmission time allocated to user u in cell k for data transmission. $\forall n$, $A_{u,k,n}$ are stacked into the vector $\mathbf{A}_{u,k} = [A_{u,k,1}, \dots, A_{u,k,N}]^T$, and $\mathbf{A}_{u,k}, \forall u \in \mathcal{U}_k$ are stacked into a matrix $\mathbf{A}_k$ column by column. Finally, all $\mathbf{A}_k, \forall k$ are stacked into a matrix $\mathbf{A} = [\mathbf{A}_1, \dots, \mathbf{A}_K]$ that indicates how carriers are allocated to all users.

The rate of user u in cell k , as a function of $\mathbf{A}_{u,k}$ and $\mathbf{P}$, is expressed as

$$R_{u,k}(\mathbf{A}_{u,k}, \mathbf{P}) = \sum_{n=1}^N A_{u,k,n} R_{u,k,n}(\mathbf{P}) = \sum_{n: A_{u,k,n}=1} R_{u,k,n}(\mathbf{P}) \quad (6.1)$$

where

$$R_{u,k,n}(\mathbf{P}) = \log \left(1 + \frac{\gamma_{u,k,n}(\mathbf{P})}{\Gamma} \right) \quad (6.2)$$

stands for the rate of carrier n when allocated to user u in cell k in the unit of nats/OFDM-symbol. $\gamma_{u,k,n}(\mathbf{P})$ is the SNR and Γ represents the SNR gap between the adopted modulation and coding scheme and the one achieving channel capacity [SCS99]. Particularly, $\gamma_{u,k,n}(\mathbf{P})$ is evaluated by

$$\gamma_{u,k,n}(\mathbf{P}) = \frac{\bar{P}_k p_{k,n} g_{u,k,k,n}}{\sigma_{u,k,n}^2 + \sum_{l=1, l \neq k}^K \bar{P}_l p_{l,n} g_{u,k,l,n}} = \frac{p_{k,n}}{I_{u,k,n}(\mathbf{P})} \quad (6.3)$$

where $I_{u,k,n}(\mathbf{P}) = x_{u,k,n} + \sum_{l=1, l \neq k}^K p_{l,n} y_{u,k,n}^l$, $x_{u,k,n} = \frac{\sigma_{u,k,n}^2}{\bar{P}_k g_{u,k,k,n}}$, and $y_{u,k,n}^l = \frac{\bar{P}_l g_{u,k,l,n}}{\bar{P}_k g_{u,k,k,n}}$.

The RA problem that maximizes the WSMR subject to per BS sum power constraints is considered. Specifically, the objective function is expressed as

$$f(\mathbf{A}, \mathbf{P}) = \sum_{k=1}^K w_k \cdot \min_{u \in \mathcal{U}_k} R_{u,k}(\mathbf{A}_{u,k}, \mathbf{P}) \quad (6.4)$$

where $w_k > 0$ represents the weight assigned to cell k 's minimal user rate. In particular, increasing w_k leads to a higher priority assigned to cell k .

Mathematically, the RA problem is formulated as

$$\begin{aligned}
 & \max_{\mathbf{A}, \mathbf{P}} f(\mathbf{A}, \mathbf{P}) \\
 & \text{s.t.} \quad \sum_{n=1}^N p_{k,n} \leq 1, \quad \forall k, \\
 & \quad \sum_{u \in \mathcal{U}_k} A_{u,k,n} \leq 1, \quad \forall k, n, \\
 & \quad A_{u,k,n} \in [0, 1], p_{k,n} \geq 0, \quad \forall u \in \mathcal{U}_k, k, n.
 \end{aligned} \tag{6.5}$$

Proposing (6.5) is motivated by the following consideration. As shown in [RC00, KPL06], the problem of maximizing the minimal user rate in a single cell was studied, in order to achieve the maximum fairness among the users in this cell. Since the considered system consists of multiple coordinated cells, the problem of maximizing the WSMR $f(\mathbf{A}, \mathbf{P})$ is considered, so that the maximum fairness is achieved among the users in each cell, and in the meanwhile different priorities can be assigned to those cells.

Note that (6.5) is a constrained nonlinear program incorporating both integer and continuous variables. In addition, the objective function, as the sum of the minimums of multiple rate functions, might not be differentiable with respect to the power allocation. As a result, simple algorithms like steepest descent method can not be directly applied [Ber03]. Furthermore, $f(\mathbf{A}, \mathbf{P})$ is not a concave function of $\mathbf{A}$ and $\mathbf{P}$, meaning that (6.5) might have multiple local optima. Therefore, it is difficult to find a global optimum for (6.5).

6.3 Algorithm Development

An iterative algorithm is proposed to optimize the carrier allocation and the power allocation alternatively. This algorithm is depicted in **Algorithm 5**, where the superscript m indicates the associated variable is produced at the end of the m th iteration.

The proposed algorithm starts with computing respectively initial power and carrier allocations $\mathbf{P}^0$ and $\mathbf{A}^0$ based on the methods proposed in Section 6.3.1. At the beginning of each iteration, it is desirable to compute the optimum $\mathbf{P}$ for (6.5) when $\mathbf{A} = \mathbf{A}^m$. However, it is hard to solve this problem due to its highly nonconvex structure. Alternatively, the SCA method

Algorithm 5 Proposed RA algorithm

```

 $m = 0$ , compute  $\mathbf{P}^0$  and  $\mathbf{A}^0$  with the initialization methods in Section 6.3.1;
repeat
    find a suboptimum solution for (6.5) when  $\mathbf{A} = \mathbf{A}^m$  using Algorithm 6 in
    Section 6.3.2, and assign it to  $\mathbf{P}^{m+1}$ ;
    solve (6.5) when  $\mathbf{P} = \mathbf{P}^{m+1}$  for the optimum  $\mathbf{A}$  using the method in Sec-
    tion 6.3.3, and assign it to  $\mathbf{A}^{m+1}$ ;
     $m = m + 1$ ;
until  $\mathbf{P}^m = \mathbf{P}^{m-1}$  or  $m = M$ 
 $\mathbf{P}^m$  and  $\mathbf{A}^m$  are adopted for the considered system.

```

introduced in Chapter 4 is used to compute a suboptimum $\mathbf{P}'$ satisfying $f(\mathbf{A}^m, \mathbf{P}') \geq f(\mathbf{A}^m, \mathbf{P}^m)$, then $\mathbf{P}'$ is assigned to $\mathbf{P}^{m+1}$. This step can be interpreted as finding a power allocation $\mathbf{P}^{m+1}$ better than $\mathbf{P}^m$ for the tentative carrier allocation $\mathbf{A}^m$.

Then the optimum $\mathbf{A}$ for (6.5) when $\mathbf{P} = \mathbf{P}^{m+1}$ is evaluated. This optimum carrier allocation is assigned to $\mathbf{A}^{m+1}$. It will be show in Section 6.3.3 that this optimum $\mathbf{A}$ can be found by solving a linear program for each cell. This step can be interpreted as finding the optimum carrier allocation for the tentative power allocation $\mathbf{P}^{m+1}$.

Note that **Algorithm 5** is CA-based, and it yields increasing WSMRs as the iterations continue, thus convergence is guaranteed. The iteration is terminated until $\mathbf{P}^m$ converges or when the number of iterations exceeds a prescribed value M that is sufficiently large while affordable in practice.

6.3.1 Initialization methods

The uniform power allocation (UPA) is used for initialization, i.e., $[\mathbf{P}^0]_{kn} = \frac{1}{N}$, $\forall n, k$. This is motivated by two considerations. One is that if the sum power at each BS is very small so that the CCI can be neglected, the multi-cell RA problem can be decomposed into K single-cell subproblems, each to find the optimum RA that maximizes the minimal user rate subject to a sum power constraint for the corresponding BS. This problem has been investigated in [RC00], and it was shown there through numerical experiments that the UPA and the associated optimum carrier allocation yield slightly degraded performance compared to the optimum RA. The other one is that the perfor-

mance of a RA adopting the UPA is often used by default to benchmark the performance of another RA. Initialized by a UPA-based RA, the proposed algorithm should be able to find a better RA, and the performance improvement can be used to evaluate the efficiency of the proposed algorithm.

Two different methods are considered for computing $\mathbf{A}^0$. One is to let users in each cell choose in turn the carrier that is not yet assigned and corresponds to the highest SNR. In this way, each user will be allocated almost the same number of carriers. This method is named as the even selection based carrier allocation (ESA). The other one is to compute $\mathbf{A}^0$ as the best carrier allocation that maximizes the WSMR when $\mathbf{P} = \mathbf{P}^0$. This method is called the best selection based carrier allocation (BSA). Note that this carrier allocation can be obtained by solving a linear program for each cell, which is elaborated in Section 6.3.3.

6.3.2 Power allocation optimization

When $\mathbf{A} = \mathbf{A}^m$, solving (6.5) is equivalent to minimizing $-f(\mathbf{A}^m, \mathbf{P})$ subject to per BS sum power constraints. Note that when $\mathbf{A} = \mathbf{A}^m$,

$$-f(\mathbf{A}^m, \mathbf{P}) = \sum_{k=1}^K w_k \left(-\min_{u \in \mathcal{U}_k} R_{u,k}(\mathbf{A}_{u,k}^m, \mathbf{P}) \right) = \sum_{k=1}^K w_k \left(\max_{u \in \mathcal{U}_k} -R_{u,k}(\mathbf{A}_{u,k}^m, \mathbf{P}) \right)$$

holds, where $\mathbf{A}_{u,k}^m$ is the $\mathbf{A}_{u,k}$ contained in $\mathbf{A}^m$. As said earlier, it is hard to find a global optimum for (6.5) when $\mathbf{A} = \mathbf{A}^m$ due to the nonconvexity of the problem.

The SCA method proposed in Chapter 4 is used to find a suboptimum $\mathbf{P}$ better than $\mathbf{P}^m$ when $\mathbf{A} = \mathbf{A}^m$. To this end, a COV from $\mathbf{P}$ to $\mathbf{Q}$ satisfying $\mathbf{P} = e^{\mathbf{Q}}$ (i.e., $q_{k,n} = [\mathbf{Q}]_{kn} = \log(p_{k,n})$) is first made. Now, (6.5) when $\mathbf{A} = \mathbf{A}^m$ is equivalent to

$$\begin{aligned} \min_{\mathbf{Q}} \quad & -f(\mathbf{A}^m, e^{\mathbf{Q}}) \\ \text{s.t.} \quad & \sum_n e^{q_{k,n}} \leq 1, \forall k. \end{aligned} \tag{6.6}$$

Then, an inner loop is executed to produce a set of $\mathbf{Q}$ with decreasing objective values for (6.6). In the following, a superscript (m, t) put to a variable indicates that variable is produced at the t th inner iteration ($t \geq 0$). At the beginning, $\mathbf{Q}^{m,0} = \log(\mathbf{P}^m)$. At the t th iteration, $\mathbf{Q}^{m,t+1}$ is found as a global

optimum for a convex approximation of (6.6). To build this approximation, we define

$$-f^{m,t}(\mathbf{A}^m, e^{\mathbf{Q}}) = \sum_{k=1}^K w_k \max_{u \in \mathcal{U}_k} -R_{u,k}^{m,t}(\mathbf{A}_{u,k}^m, e^{\mathbf{Q}})$$

where $-R_{u,k}^{m,t}(\mathbf{A}_{u,k}^m, e^{\mathbf{Q}})$ is a CUBA of $-R_{u,k}(\mathbf{A}_{u,k}^m, e^{\mathbf{Q}})$. Specifically, we can use the same method as for the SCALE algorithm to construct $-R_{u,k}^{m,t}(\mathbf{A}_{u,k}^m, e^{\mathbf{Q}})$ as

$$-R_{u,k}^{m,t}(\mathbf{A}_{u,k}^m, e^{\mathbf{Q}}) = \sum_n A_{u,k,n}^m (-R_{u,k,n}^{m,t}(e^{\mathbf{Q}})) \quad (6.7)$$

where

$$\begin{aligned} R_{u,k,n}^{m,t}(e^{\mathbf{Q}}) &= \alpha_{u,k,n}^{m,t} \log(\gamma_{u,k,n}(e^{\mathbf{Q}})) + \beta_{u,k,n}^{m,t} \\ &= \alpha_{u,k,n}^{m,t} (q_{k,n} - \log I_{u,k,n}(e^{\mathbf{Q}})) + \beta_{u,k,n}^{m,t} \end{aligned} \quad (6.8)$$

$$\alpha_{u,k,n}^{m,t} = \frac{\gamma_{u,k,n}(\mathbf{P}^{m,t})}{\Gamma + \gamma_{u,k,n}(\mathbf{P}^{m,t})} \quad (6.9)$$

$$\beta_{u,k,n}^{m,t} = R_{u,k,n}(\mathbf{P}^{m,t}) - \alpha_{u,k,n}^{m,t} \log(\gamma_{u,k,n}(\mathbf{P}^{m,t})). \quad (6.10)$$

It can readily be shown that $-f^{m,t}(\mathbf{A}^m, e^{\mathbf{Q}})$ is a CUBA of $-f(\mathbf{A}^m, e^{\mathbf{Q}})$ and satisfies the tightness condition at $\mathbf{Q}^{m,t}$, i.e., $\forall \mathbf{Q}, -f^{m,t}(\mathbf{A}^m, e^{\mathbf{Q}}) \geq -f(\mathbf{A}^m, e^{\mathbf{Q}})$ and this inequality is tightened at $\mathbf{Q} = \mathbf{Q}^{m,t}$. Therefore, the convex approximation of (6.6) at the t th iteration can be built by replacing $-f(\mathbf{A}^m, e^{\mathbf{Q}})$ with $-f^{m,t}(\mathbf{A}^m, e^{\mathbf{Q}})$. Note that the feasible set of $\mathbf{Q}$ for this approximate problem is unbounded below, thus its optimum solution might have negative entries of large magnitude beyond hardware's precision limit, leading to overflow. To address this problem, we add extra constraints $\forall k, n, q_{k,n} \geq \log(\delta)$ where δ is a prescribed very small positive value. Then, the approximation of (6.6) at the t th iteration with the added constraints can be formulated as the following equivalent problem

$$\begin{aligned} \min_{\mathbf{Q}, \{C_k, \forall k\}} \quad & \sum_{k=1}^K w_k C_k \\ \text{s.t.} \quad & -C_k - \sum_n A_{u,k,n}^m R_{u,k,n}^{m,t}(e^{\mathbf{Q}}) \leq 0, \forall u \in \mathcal{U}_k, k, \\ & \sum_n e^{q_{k,n}} - 1 \leq 0, \forall k, \\ & q_{k,n} \geq \log(\delta), \forall k, n. \end{aligned} \quad (6.11)$$

We use geometric programming to find a global optimum for (6.11), and then assign it to $\mathbf{Q}^{m,t+1}$. To this end, (6.11) is first reformulated into an equivalent GP in the convex form introduced in Section 4.6. Then, that GP can be solved by GP solvers such as MOSEK or gpcvx. Specifically, we first reformulate the first constraint in (6.11) as

$$-C_k - \sum_n a_{u,k,n}^{m,t} (q_{k,n} - \log I_{u,k,n}(e^{\mathbf{Q}})) - b_{u,k}^{m,t} \leq 0, \forall u \in \mathcal{U}_k, k, \quad (6.12)$$

where $a_{u,k,n}^{m,t} = A_{u,k,n}^m \alpha_{u,k,n}^{m,t}$ and $b_{u,k}^{m,t} = \sum_n A_{u,k,n}^m \beta_{u,k,n}^{m,t}$. It is equivalent to the constraints

$$-C_k - \sum_n a_{u,k,n}^{m,t} (q_{k,n} - i_{u,k,n}) - b_{u,k}^{m,t} \leq 0, \forall u \in \mathcal{U}_k, k, \quad (6.13)$$

jointly with

$$\log I_{u,k,n}(e^{\mathbf{Q}}) \leq i_{u,k,n}, \forall u \in \mathcal{U}_k, k, n, \quad (6.14)$$

because (6.12) corresponds to the same feasible set of $\{\mathbf{Q}, C_1, \dots, C_K | \forall k\}$ as (6.13) and (6.14). Furthermore, (6.14) and (6.14) are respectively equivalent to

$$\log \left(e^{-C_k - \sum_n a_{u,k,n}^{m,t} (q_{k,n} - i_{u,k,n}) - b_{u,k}^{m,t}} \right) \leq 0, \forall u \in \mathcal{U}_k, k, \quad (6.15)$$

and

$$\log(e^{\log(x_{u,k,n}) - i_{u,k,n}} + \sum_{l \neq k} e^{\log(y_{u,k,n}^l) + q_{l,n} - i_{u,k,n}}) \leq 0, \forall u \in \mathcal{U}_k, k, n. \quad (6.16)$$

Based on the the above analysis, we can finally reformulate (6.11) into an equivalent GP in the convex form as follows:

$$\begin{aligned} \min_{\mathbf{Q}, \{C_k | \forall k\}} \quad & \log(e^{-\sum_k w_k C_k}) \\ \text{s.t.} \quad & \log(\delta e^{-q_{k,n}}) \leq 0, \forall k, n, \\ & (6.15), \forall u \in \mathcal{U}_k, k, \\ & (6.16), \forall u \in \mathcal{U}_k, k, n, \\ & \log(\sum_n e^{q_{k,n}}) \leq 0, \forall k. \end{aligned} \quad (6.17)$$

In summary, the power allocation algorithm when $\mathbf{A} = \mathbf{A}^m$ is given in Algorithm 6, where T represents a prescribed value.

Algorithm 6 Power allocation optimization algorithm

```

 $t = 0, \mathbf{Q}^{m,0} = \log(\mathbf{P}^m);$ 
repeat
  solve (6.17) for its global optimum, and assign it to  $\mathbf{Q}^{m,t+1};$ 
   $t = t + 1;$ 
until  $\mathbf{Q}^{m,t} = \mathbf{Q}^{m,t+1}$  or  $t = T$ 
output  $\mathbf{P}^{m+1} = e^{\mathbf{Q}^{m,t}}.$ 

```

6.3.3 Carrier allocation optimization

The problem of finding the optimum carrier allocation $\mathbf{A}^{m+1}$ when $\mathbf{P} = \mathbf{P}^{m+1}$ can be decomposed into K subproblems, the k th of which is to maximize the minimal user rate in cell k by finding the associated optimum carrier allocation. More specifically, the k th subproblem is formulated as

$$\begin{aligned}
 \max_{\mathbf{A}_k} \quad & \min_{u \in \mathcal{U}_k} R_{u,k}(\mathbf{A}_{u,k}, \mathbf{P}^{m+1}) \\
 \text{s.t.} \quad & \sum_{u \in \mathcal{U}_k} A_{u,k,n} \leq 1, \quad \forall n, \\
 & A_{u,k,n} \in [0, 1], \quad \forall u \in \mathcal{U}_k, n.
 \end{aligned} \tag{6.18}$$

Let's denote the optimum $\mathbf{A}_k$ for (6.18) by $\mathbf{A}_k^{m+1}$. Then the optimum carrier allocation is constructed by $\mathbf{A}^{m+1} = [\mathbf{A}_1^{m+1}, \dots, \mathbf{A}_K^{m+1}]$. In fact, (6.18) can be reformulated into an equivalent form:

$$\begin{aligned}
 \max_{\mathbf{A}_k, S_k} \quad & S_k \\
 \text{s.t.} \quad & \sum_{n=1}^N A_{u,k,n} \cdot R_{u,k,n}(\mathbf{P}^{m+1}) \geq S_k, \quad \forall u \in \mathcal{U}_k, \\
 & \sum_{u \in \mathcal{U}_k} A_{u,k,n} \leq 1, \quad \forall n, \\
 & A_{u,k,n} \in [0, 1], \quad \forall u \in \mathcal{U}_k, n.
 \end{aligned} \tag{6.19}$$

where S_k is the extra optimization variable introduced to guarantee the equivalence between (6.18) and (6.19). Note that (6.19) is a linear problem, which can easily be solved by existing solvers, e.g., Lindo or linprog in Matlab.

It is interesting to examine the complexity of the proposed algorithm. For each optimization of carrier allocation, K linear programs need to be solved separately, and each is solved in polynomial time by state-of-art methods, e.g., simplex or interior-point methods [BV04]. For each optimization of power

allocation, T geometric programs need to be solved, and the state-of-the-art interior-point method to solve it has a polynomial complexity [BV04]. The complexity of the proposed algorithm depends on the specific methods implemented by the solvers to solve the linear program and GP. In the next section, the complexity of the proposed algorithm will be shown for certain solvers used.

6.4 Numerical Experiments

For illustration purposes, the downlink of a cellular OFDMA system with $K = 3$ coordinated cells and $N = 64$ carriers is considered. BS 1, 2 and 3 are located at the coordinates $(-100, -100)$, $(100, -100)$, and $(0, 100)$, respectively. Each BS serves 3 users, which are randomly and uniformly distributed in the regions $\{(x, y) | -70 \leq x \leq -10, -70 \leq y \leq -10\}$, $\{(x, y) | 10 \leq x \leq 70, -70 \leq y \leq -10\}$, and $\{(x, y) | -30 \leq x \leq 30, 10 \leq y \leq 70\}$ for the three cells, respectively. All the aforementioned coordinate related values have the unit of meter. The parameters are set as $w_1 = w_2 = w_3 = 1$, $\Gamma = 1$, $\sigma_{u,k,n}^2 = -50$ dBm, $\forall u \in \mathcal{U}_k, k, n$, $\bar{P}_k = P_M, \forall k$, $\delta = 10^{-5}$, $M = 4$ and $T = 3$. The channel between each BS-user pair is generated in the same way as explained in Section 5.6. The following experiments are conducted with Matlab v7.1 on a laptop equipped with an Intel Duo CPU of 2.2 GHz and a memory of 2 GBytes. In particular, the linprog in Matlab and gpcvx are used to solve (6.19) and (6.17), respectively.

A set of user positions was first randomly chosen, and the associated system setup is shown in Figure 6.2. For this setup, a random realization of the BS-user channel gains was generated. Algorithm 1 has been run when $P_M = 70$ dBm. After Algorithm 1 terminates, the power allocation and the carrier allocation for the first cell after optimization are shown in Figures 6.3 and 6.4, respectively. The computed WSMR for every iteration, as well as user rates at initialization and after optimization are shown in Figure 6.5. In particular, each line segment with the same mark shows the WSMRs when the carrier allocation is fixed.

It can be seen that both initialization methods lead to similar WSMRs after optimization, even though the BSA initialization results in a higher WSMR at the beginning. Besides, the power and carrier allocations after optimization lead to a significant WSMR increase than the initial allocations. Despite the

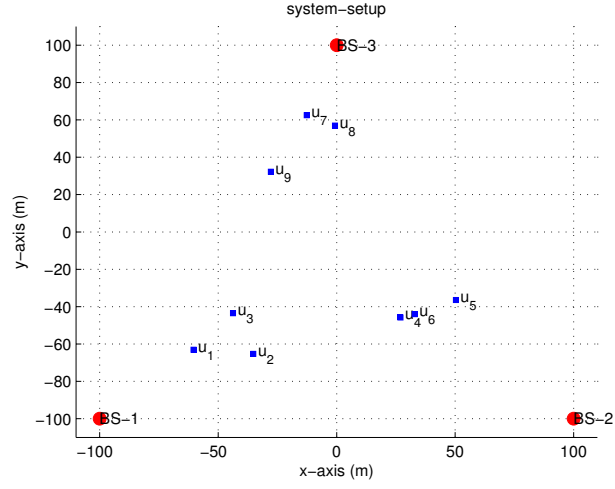


Figure 6.2 A random realization of system setup.

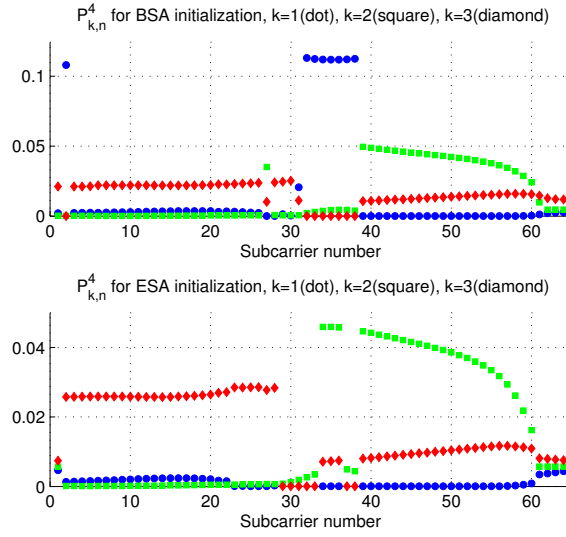


Figure 6.3 The power allocation after optimization with the BSA/ESA initialization for the random realization of channel gains.

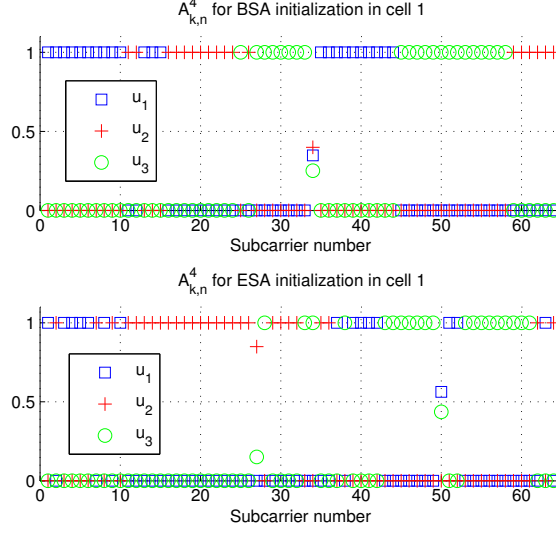


Figure 6.4 The carrier allocation for the first cell after optimization with the BSA/ESA initialization for the random realization of channel gains.

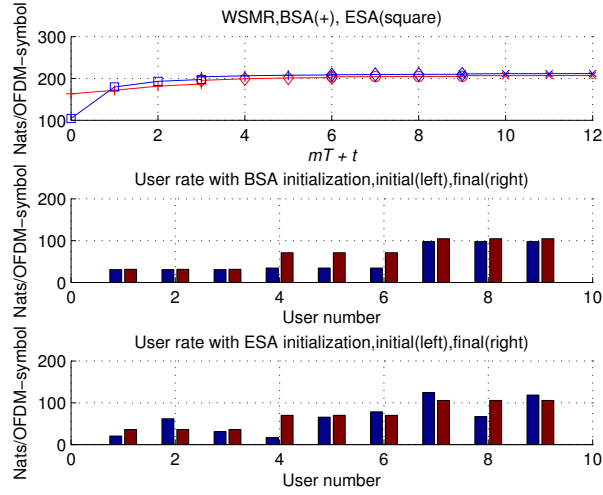


Figure 6.5 The computed results for the random realization of channel gains. In each bar group, the left and right bars represent the performance for the initial and final RAs, respectively.

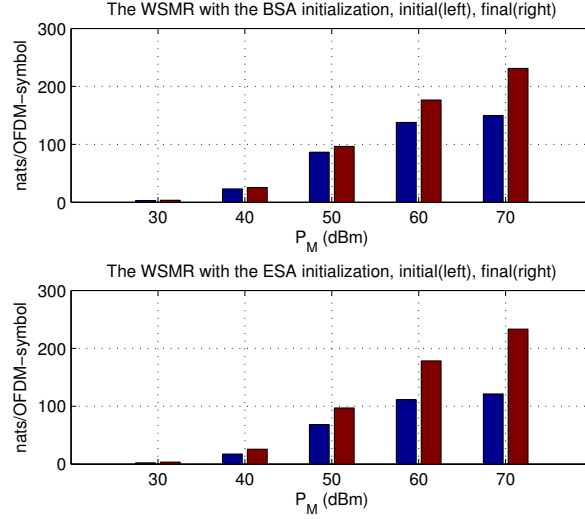


Figure 6.6 Average WSMR over 100 random realizations of channel gains. In each bar group, the left and right bars represent the performance for the initial and final RAs, respectively.

initialization method, the average time spent for solving each particular instance of (6.19) and (6.17) is around 0.2 second and 25.2 seconds, respectively. The power allocation after optimization shows that each BS allocates the dominant part of its power to a few carriers. Once a BS allocates a significant part of its power to a carrier, the other BSs usually allocate only a small part of its power to this carrier, so as to reduce CCI. After optimization, the minimal user rate in each cell is enhanced, and the users in each cell have similar rates after optimization. These observations illustrate the efficiency of the proposed algorithm.

The proposed algorithm was also run for 100 random realizations of the BS-user channel gains, when P_M varies from 30 to 70 dBm. The average value of the WSMR at initialization and after optimization are shown in Figure 6.6. Regardless of the initialization method, the average WSMR after optimization is slightly higher than the average initial WSMR when P_M is below 40 dBm. This is because the CCI is relatively low due to the relatively low sum power of each BS, and therefore the UPA is nearly optimal as explained earlier. However, the average initial WSMR increases slightly, while the average WSMR

Table 6.1 Average execution time in seconds for solving (6.19) with the linprog.

	$U = 3, N = 64$	$U = 6, N = 64$	$U = 3, N = 128$
ρ	0.2	0.56	0.60

Table 6.2 Average execution time in seconds for solving (6.17) with the GP.

	$K = 3, N = 64$	$K = 6, N = 64$	$K = 3, N = 128$
χ	7.8	65.0	62.0

after optimization increases significantly when P_M increases from 50 dBm to 70 dBm. Besides, the average WSMR after optimization is significantly higher than the average initial WSMR when P_M is greater than 50 dBm. These observations show that the proposed algorithm can effectively produce power and carrier allocations that lead to a significantly better WSMR performance than the UPA and the BSA/ESA, especially when the sum power of each BS is high.

It is interesting to examine the complexity of the proposed algorithm. When the power allocation is fixed, the carrier allocation is optimized on a per-cell basis. The total time spent for optimizing the carrier allocation is equal to $M * K * \rho$, where ρ represents the average time for solving (6.19) with the linprog. The ρ for different combinations of N and U which denotes the number of users in a cell is shown in Table 6.1. It can be seen that ρ scales up with increasing N at a similar rate as with increasing U .

When the carrier allocation is fixed, the power allocation is jointly optimized over all cells. The total time spent for optimizing the power allocation is equal to $M * T * \chi$, where χ represents the average time for solving (6.17) with the GP. The χ for different combinations of N and K is shown in Table 6.2. χ scales up with increasing N at a similar rate as with increasing K . Moreover, χ scales up at a much faster speed than ρ does when N increases. It can be seen that χ is much greater than ρ , indicating that to reduce the complexity of the proposed algorithm, the key is to solve (6.17) with a more efficient method than the GP.

Sum rate maximized RA in multiple DF relays aided OFDM systems

7

7.1 Related works and chapter overview

This chapter addresses RA for a point-to-point OFDM transmission system aided by multiple DF relays. To date, some related research works have been reported. To name a few, RA algorithms have been proposed in [JJL10, PNT10, DTM⁺10, DHH⁺10] for OFDM systems aided by AF relays. As for OFDM systems aided by DF relays, RA algorithms have been proposed in [GCJ08] to minimize sum power under rate constraints, and in [LWK⁺08, VDL⁺08] to maximize sum rate subject to power constraints, when a single relay exists. However, at a carrier in the direct mode the source is idle during the relaying slot, which wastes spectrum resource. To address this issue, rate-optimized RA algorithms, which allow for the source to destination transmission during the two slots at the carrier in the direct mode, have been proposed in [VLO⁺08, VLO⁺09a, VLO⁺, VLO⁺09b].

So far, the majority of proposed RA algorithms, as the aforementioned ones, restrict that at most one relay can assist the source at every relay-aided carrier. In fact, when there are multiple relays available, allowing not just one, but *each* of them to be eligible for assisting can exploit all degrees of freedom in the system to improve performance. For illustration purposes, example patterns of selecting single or multiple assisting relays are shown in Figure 7.1.

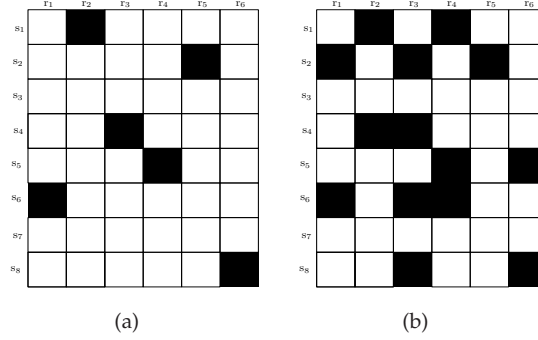


Figure 7.1 Example patterns of selecting (a) one relay or (b) multiple relays assisting at each relay-aided carrier, where a black block at the i -th row and j -th column indicates the j -th relay r_j assists the transmission at the i -th carrier s_i .

In this chapter, a sum rate maximized RA problem is addressed for an OFDM system aided by multiple DF relays subject to the individual (per device, i.e. a source or a relay) sum (over all carriers) power constraints¹ of the source and the relays. In particular, one or several relays may cooperate with the source to transmit at every relay-aided carrier. Specifically, the contributions of this chapter lie in the following aspects:

- when a big number of carriers is used, it is shown that the duality gap of the RA problem is approximately zero, based on the same idea first proposed in [YL06]. Assuming the number of carriers is sufficiently large, a duality-based algorithm is developed to find a globally optimum RA for the considered problem. Most interestingly, the *sensitivity analysis* introduced in Chapter 3 is used to derive a *closed-form* optimum solution to a related convex optimization problem, for which the method based on the KKT conditions is not applicable.
- a CA-based iterative algorithm, which is always applicable regardless of the duality gap of the RA problem, is developed to find a suboptimum RA. Specifically, this algorithm produces a successive set of RAs with increasing sum rate until convergence.

The rest of this chapter is organized as follows. First, the OFDM system and the RA problem are introduced. Then, the duality-based algorithm and

¹ In this chapter, the word “individual” will mean per device and “sum” will refer to a summation over all carriers, unless otherwise stated.

Table 7.1 Channel coefficient at carrier n between any two of the source, r_i , and the destination.

source to destination	source to r_i	r_i to destination
$h_{s,d}(n)$	$h_{s,r_i}(n)$	$h_{r_i,d}(n)$

the iterative algorithm are developed. Finally, the effectiveness of the proposed algorithms is illustrated by numerical experiments.

7.2 System description and RA problem

Consider an OFDM transmission system from a source to a destination aided by K DF relays collected in the set $\Psi = \{r_i | i = 1, \dots, K\}$. All links are assumed to be frequency selective, and OFDM is used to transform every link into N parallel channels, each at a different carrier facing flat fading. At every carrier, the transmission of a symbol is in either the direct mode, or the relay-aided mode spanning across two equal-duration time slots, namely the broadcasting slot and the relaying slot.

The following assumptions are made about the RA in the system. First, the RA is determined by an algorithm running at a central controller, which knows precisely the noise power at each node, as well as the channel coefficients at every carrier from the source to every r_i , from the source to the destination, and from every r_i to the destination. Second, all channels remain unchanged within a sufficiently long duration, over which RA can be carried out accordingly. Third, the RA information can be reliably disseminated to the source, every relay, and the destination.

Let's consider the transmission of a unit-variance symbol θ at carrier n . The coefficient of the channel between any two of the source, r_i , and the destination, are notated according to Table 8.1. The transmission in the relay-aided mode is first described. The source first emits in the broadcasting slot the symbol $\sqrt{P_s P_{s,n}} \theta$ as illustrated in Figure 7.2.a, where P_s and $P_{s,n}$ represent the source sum power and the fraction of that sum power allocated to the transmission at carrier n , respectively. At the end of this slot, both the destination and the relays receive the source signal. The signal samples received at the

destination and r_i can be expressed respectively by

$$y_n = \sqrt{P_s P_{s,n}} h_{s,d}(n) \theta + \eta_n \quad (7.1)$$

and

$$y_{r_i,n} = \sqrt{P_s P_{s,n}} h_{s,r_i}(n) \theta + \eta_{r_i,n}, \quad (7.2)$$

where η_n and $\eta_{r_i,n}$ represent the corruption of the AWGN at the destination and r_i , respectively. We assume $\forall i$, $\eta_{r_i,n}$ is a zero-mean circularly Gaussian random variable with the variance σ^2 . The SNR at r_i can be computed as $P_{s,n} G_{s,r_i}(n)$, where $G_{s,r_i}(n) = \frac{P_s |h_{s,r_i}(n)|^2}{\sigma^2}$ represents the normalized channel power gain from the source to r_i .

For carrier n , Ψ_n is defined as the set containing all relays sorted in the increasing order of $G_{s,r_i}(n)$, $r_i(n)$ as the i -th relay in Ψ_n , and $\Psi_n(i)$ as the set containing relays in Ψ_n with indices from i to N . Assume the r_i that has the minimum $G_{s,r_i}(n)$ among all assisting relays for carrier n is $r_{b_n}(n)$ where b_n represents the index of that r_i in Ψ_n , and all r_i in $\Psi_n(b_n)$ decode the received samples to recover θ .

After the recovery of θ , all relays in $\Psi_n(b_n)$ transmit simultaneously to the destination in the relaying slot, which in effect establishes a distributed multiple input and single output (MISO) transmission link as illustrated in Figure 7.2.b. Specifically, $r_i \in \Psi_n(b_n)$ transmits $w_{r_i} \theta$, where w_{r_i} is the complex weight for transmit beamforming and satisfies $|w_{r_i}|^2 = P_{r_i} P_{r_i,n}$. P_{r_i} and $P_{r_i,n}$ represent the sum power of r_i and the fraction of that sum power allocated to carrier n , respectively. To have the relays' signals add coherently when received at the destination, $w_{r_i} = \sqrt{P_{r_i} P_{r_i,n}} e^{-j \arg(h_{r_i,d}(n))}$ is used, where $\arg(h_{r_i,d}(n))$ stands

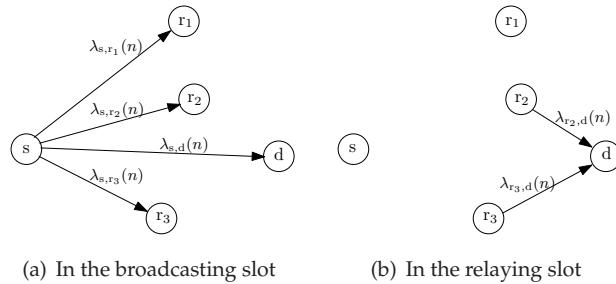


Figure 7.2 Illustration of a relay-aided transmission where r_2 and r_3 assist relaying at carrier n .

for the phase of $h_{r_i,d}(n)$. It should be noted that this transmission protocol enables a flexible use of all relays opportunistically through a general form of *adaptive* transmit beamforming, in that b_n and $\{P_{r_i,n} | r_i \in \Psi_n(b_n)\}$ are determined by the RA algorithm depending on channel state information, as will be developed later. At the end of the relaying slot, the signal sample received at the destination is denoted by

$$z_n = \sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i} P_{r_i,n}} |h_{r_i,d}(n)| \theta + v_n, \quad (7.3)$$

where v_n represents the AWGN corruption at the destination. We assume η_n and v_n are independent zero-mean circularly Gaussian random variables with the same variance σ^2 .

Finally, y_n and z_n are processed at the destination. From (7.1) and (7.3), it can be seen that the relay-aided transmission at carrier n is in effect over a channel with a single input and two outputs. To achieve the capacity of this channel, the maximum ratio combining (MRC) should be used, i.e., the destination combines y_n and z_n to construct a decision variable

$$c_k = \sqrt{P_s P_{s,n}} (h_{s,d}(n))^* y_n + \left(\sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i} P_{r_i,n}} |h_{r_i,d}(n)| \right)^* z_n, \quad (7.4)$$

which is then decoded. After mathematical arrangements, the SNR for decoding c_k is derived as

$$\zeta_n = P_{s,n} G_{s,d}(n) + \left(\sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i,n} G_{r_i,d}(n)} \right)^2 \quad (7.5)$$

where $G_{s,d}(n) = \frac{P_s |h_{s,d}(n)|^2}{\sigma^2}$ and $G_{r_i,d}(n) = \frac{P_{r_i} |h_{r_i,d}(n)|^2}{\sigma^2}$ represent the normalized channel power gains from the source to the destination and from r_i to the destination, respectively.

To ensure reliable recovery of θ at every relay in $\Psi_n(b_n)$, the source transmission rate at carrier n should not be higher than $\log(1 + P_{s,n} G_{s,r_{b_n}}(n))$ nats/two-slots. Moreover, the source transmission rate should not be higher than $\log(1 + \zeta_n)$ to ensure reliable decoding of c_k at the destination. Therefore, the source transmission rate at carrier n in the relay-aided mode should be [LTW04]

$$\begin{aligned} R_{n,1} &= \min \left(\log(1 + P_{s,n} G_{s,r_{b_n}}(n)), \log(1 + \zeta_n) \right) \\ &= \log \left(1 + \min(\zeta_n, P_{s,n} G_{s,r_{b_n}}(n)) \right). \end{aligned} \quad (7.6)$$

As for the direct transmission mode at carrier n , the source emits two independent symbols in the two slots, respectively, and only the destination decodes the corresponding two received signal samples. This protocol is more efficient than restricting the source to emit only in the broadcasting slot as in the related works [LTW04,GCJ08,LWK⁺08,VDL⁺08]. Assume the AWGN corruptions for the two received samples are independent zero-mean circularly Gaussian distributed with variance σ^2 , and the source uses the power $P_s P_{s,n}$ in total to transmit the two symbols. The maximum source transmission rate at carrier n in the direct mode can be derived as

$$R_{n,2} = 2 \log \left(1 + P_{s,n} \frac{G_{s,d}(n)}{2} \right) \quad (7.7)$$

and it is achieved when the source uses the power $\frac{P_s P_{s,n}}{2}$ to transmit each symbol.

To formulate the RA problem, a binary variable t_n is defined to indicate the transmission at carrier n is in the relay-aided mode (resp. the direct mode) if $t_n = 1$ (resp. $t_n = 0$). Mathematically, the RA problem is formulated as

$$\begin{aligned} \max \quad & \sum_{n=1}^N (t_n R_{n,1} + (1 - t_n) R_{n,2}) \\ \text{s.t.} \quad & \sum_{n=1}^N P_{s,n} \leq 1, \quad \sum_{n=1}^N P_{r_i,n} \leq 1, \forall r_i \in \Psi, \\ & P_{s,n} \geq 0, \forall n, \quad P_{r_i,n} \geq 0, \forall n, \forall r_i \in \Psi, \\ & t_n \in \{0, 1\}, \forall n, \quad b_n \in \{1, \dots, K\}, \forall n, \end{aligned} \quad (7.8)$$

where $\{t_n, b_n, P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi, \forall n\}$ are the optimization variable to be determined by the RA algorithm.

7.3 The duality-based RA algorithm

To facilitate analysis in the following sections, (7.8) can be formulated into the following equivalent form

$$\begin{aligned} \max_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} \in \mathcal{D}_{\mathbf{x}}, g(\mathbf{x}) \preceq \mathbf{1}, \end{aligned} \quad (7.9)$$

where $\mathbf{x}$ represents the vector stacking all optimization variables, $\mathcal{D}_{\mathbf{x}}$ stands for the definition domain of $\mathbf{x}$, $f(\mathbf{x})$ denotes the sum rate, $g(\mathbf{x}) =$

$[g_1(\mathbf{x}), \dots, g_{N+1}(\mathbf{x})]^T$ where $g_1(\mathbf{x}) = \sum_{k=1}^K P_{s,n}$ and $g_{i+1}(\mathbf{x}) = \sum_n P_{r_i,n}$ ($i = 1, \dots, N$) stacks the sum power of the source and those of the relays, and $\mathbf{1}$ represents an $(N+1) \times 1$ vector with every entry equal to 1.

The Lagrangian for (7.9) is defined as

$$L(\mathbf{x}, \boldsymbol{\mu}) = f(\mathbf{x}) + \boldsymbol{\mu}^T (\mathbf{1} - g(\mathbf{x})), \quad (7.10)$$

where $\boldsymbol{\mu} = [\mu_s, \mu_{r_1}, \dots, \mu_{r_N}]^T$, with μ_s and μ_{r_i} representing the dual variables related to the sum power constraints of the source and r_i , respectively. The dual function is defined as $d(\boldsymbol{\mu}) = \max_{\mathbf{x} \in \mathcal{D}_x} L(\mathbf{x}, \boldsymbol{\mu})$, and a $\mathbf{x}$ that maximizes $L(\mathbf{x}, \boldsymbol{\mu})$ is denoted as $\mathbf{x}_\mu$. A subgradient of $d(\boldsymbol{\mu})$ at $\boldsymbol{\mu}$ is equal to $\mathbf{1} - g(\mathbf{x}_\mu)$ according to Corollary 2.1 in Chapter 2.

Note that $\mathcal{D}_x$ is nonconvex, and therefore the duality gap of (7.9) might not be zero. Interestingly, it is shown in Section 7.6 based on the same idea proposed in [YL06] that, (7.9) has zero duality gap when N is sufficiently large.

Assuming sufficiently large N , a duality-based RA algorithm is developed in this section to find a global optimum for (7.9). The overall duality-based RA algorithm is summarized in Algorithm 7, where $\epsilon_1 > 0$ and $\delta_1 > 0$ are both very small values prescribed to guarantee convergence. This algorithm is a special instance of the dual method, namely Algorithm 1 introduced in Section 3.6. Specifically, $\boldsymbol{\mu}$ is initialized with $\mathbf{1}$ at the beginning, and $\mathbf{x}_\mu$ is found as the optimum solution to the LRP

$$\begin{aligned} \max_{\mathbf{x}} \quad & L(\mathbf{x}, \boldsymbol{\mu}) = \sum_n (t_n R_{n,1} + (1 - t_n) R_{n,2}) + \\ & \mu_s (1 - \sum_n P_{s,n}) + \sum_i \mu_{r_i} (1 - \sum_n P_{r_i,n}), \\ \text{s.t.} \quad & P_{s,n} \geq 0, \forall n, \quad P_{r_i,n} \geq 0, \forall n, \forall r_i \in \Psi, \\ & t_n \in \{0, 1\}, \forall n, \quad b_n \in \{1, \dots, K\}, \forall n, \end{aligned} \quad (7.11)$$

with Algorithm 8 developed in Section 7.3.1. Then, $\boldsymbol{\mu}$ is iteratively updated according to the subgradient method until the specified conditions are satisfied. Finally, the $\mathbf{x}_\mu$ is output as an approximately optimum solution. According to the analysis in Section 3.6, $f(\mathbf{x}_\mu)$ is above $f^* - \epsilon$ where f^* is the optimum objective for (7.9).

7.3.1 Algorithm to solve the LRP

It can be seen that the optimum solution to (7.11) can be found on a per subcarrier basis, i.e., $\{t_n(\boldsymbol{\mu}), b_n(\boldsymbol{\mu}), P_{s,n}(\boldsymbol{\mu}), P_{r_i,n}(\boldsymbol{\mu}) | \forall r_i \in \Psi\}$ representing the

Algorithm 7 The duality-based RA algorithm

```

 $q = 1, \mu = \mathbf{1};$ 
repeat
   $\mu = [\mu - \frac{\delta_1}{q}(\mathbf{1} - g(\mathbf{x}_\mu))]^+$ 
   $q = q + 1;$ 
  Find  $\mathbf{x}_\mu$  with Algorithm 8;
until  $g(\mathbf{x}_\mu) \preceq \mathbf{1}$  and  $\mu^T(\mathbf{1} - g(\mathbf{x}_\mu)) < \epsilon_1$ 
output  $\mathbf{x}_\mu$  as an approximately optimum solution.

```

$\{t_n, b_n, P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi\}$ contained in $\mathbf{x}_\mu$ for carrier n , can be found as an optimum solution to

$$\begin{aligned}
 \max \quad & L_n = t_n R_{n,1} + (1 - t_n) R_{n,2} - \mu_s P_{s,n} - \sum_i \mu_{r_i} P_{r_i,n} \\
 \text{s.t.} \quad & P_{s,n} \geq 0, \quad P_{r_i,n} \geq 0, \forall r_i \in \Psi, \\
 & b_n \in \{1, \dots, K\}, \quad t_n \in \{0, 1\}.
 \end{aligned} \tag{7.12}$$

When t_n and b_n are fixed, (7.12) is reduced to a convex optimization problem. When $t_n = 0$, the maximum L_n is not influenced by b_n , while this not the case when $t_n = 1$ and b_n is fixed. Based on the above analysis, an exhaustive-search based algorithm is designed to solve (7.12). To facilitate the algorithm design, let's denote the maximum L_n when $t_n = 0$ by $L_{n,0}$, the maximum L_n and the optimum $\{P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi\}$ when $t_n = 1$ and b_n is fixed by $L_{n,1}(b_n)$ and $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$, respectively. Note that $L_{n,1}(b_n)$ and $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$ can be evaluated with Algorithm 9 developed in Section 7.3.2.

It is important to note that $L_{n,1}(b_n) \leq L_{n,0}$ if $G_{s,r_{b_n}(n)}(n) \leq G_{s,d}(n)$, because of

$$\begin{aligned}
 L_{n,1}(b_n) &= \log(1 + \min(\zeta'_n, P_{s,n}(b_n) G_{s,r_{b_n}(n)}(n))) + X_{b_n} \\
 &\leq \log(1 + P_{s,n}(b_n) G_{s,d}(n)) + X_{b_n} \\
 &\leq 2 \log \left(1 + P_{s,n}(b_n) \frac{G_{s,d}(n)}{2} \right) + X_{b_n} \\
 &\leq L_{n,0},
 \end{aligned} \tag{7.13}$$

where $X_{b_n} = -\mu_s P_{s,n}(b_n) - \sum_i \mu_{r_i} P_{r_i,n}(b_n)$, and ζ'_n represents the ζ_n corresponding to $\Psi_n(b_n)$ and $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$.

Algorithm 8 Algorithm to solve (7.11) when μ is fixed

```

for  $n = 1$  to  $N$  do
  if  $G_{s,d}(n) \geq \max_{r_i \in \Psi} G_{r_i,d}(n)$  then
     $t_n(\mu) = 0$ ;
     $P_{s,n}(\mu)$  and  $P_{r_i,n}(\mu), \forall r_i \in \Psi$  are evaluated by (7.14) and (7.15), respectively;
  else
    Compute  $L_{n,0}$  with (7.16);
    Find  $i_n$  as the minimum  $i$  satisfying  $G_{s,r_i(n)}(n) > G_{s,d}(n)$ ;
    for  $b_n = i_n$  to  $K$  do
      Find  $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$  with Algorithm 9 when  $t_n = 1$  and  $b_n$  is
      fixed, then compute the  $L_{n,1}(b_n)$ ;
    end for
    if  $L_{n,0} \geq \max_{b_n \in \mathcal{B}_n} L_{n,1}(b_n)$  then
       $t_n(\mu) = 0$ ;
       $P_{s,n}(\mu)$  and  $P_{r_i,n}(\mu), \forall r_i \in \Psi$  are evaluated by (7.14) and (7.15), respectively;
    else
       $t_n(\mu) = 1$ ;
       $b_n(\mu) = \arg \max_{b_n \in \mathcal{B}_n} L_{n,1}(b_n)$ ;
       $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$  when  $b_n = b_n(\mu)$  is assigned to
       $\{P_{s,n}(\mu), P_{r_i,n}(\mu) | \forall r_i \in \Psi\}$ .
    end if
  end if
end for

```

Based on the above analysis, $t_n(\boldsymbol{\mu})$, $b_n(\boldsymbol{\mu})$, $P_{s,n}(\boldsymbol{\mu})$, and $\{P_{r_i,n}(\boldsymbol{\mu})|\forall r_i \in \Psi\}$ can be found with one of the following procedures:

1. when $\max_{r_i} G_{s,r_i}(n) \leq G_{s,d}(n)$, $t_n(\boldsymbol{\mu}) = 0$, because $L_{n,0} \geq L_{n,1}(b_n)$ holds for every feasible value of b_n according to (7.13). In this case, it can be derived according to the KKT conditions that

$$P_{s,n}(\boldsymbol{\mu}) = 2 \left[\frac{1}{\mu_s} - \frac{1}{G_{s,d}(n)} \right]^+, \quad (7.14)$$

$$P_{r_i,n}(\boldsymbol{\mu}) = 0, \forall r_i \in \Psi, \quad (7.15)$$

$$L_{n,0} = 2 \log \left(1 + G_{s,d}(n) \left[\frac{1}{\mu_s} - \frac{1}{G_{s,d}(n)} \right]^+ \right) \quad (7.16)$$

2. when $\max_{r_i} G_{s,r_i}(n) > G_{s,d}(n)$, $t_n(\boldsymbol{\mu})$ can be determined by an exhaustive-search based method, i.e., if $\max_{b_n \in \{1, \dots, K\}} L_{n,1}(b_n) > L_{n,0}$, $t_n(\boldsymbol{\mu}) = 1$, otherwise $t_n(\boldsymbol{\mu}) = 0$. Let's denote i_n as the minimum i satisfying $G_{s,r_i}(n) > G_{s,d}(n)$. Note that when $1 \leq b_n \leq i_n - 1$, $G_{s,r_{b_n}}(n) \leq G_{s,d}(n)$, hence $L_{n,1}(b_n) \leq L_{n,0}$ holds for sure according to (7.13). This means that the comparison of $L_{n,0}$ with $\max_{b_n \in \{1, \dots, K\}} L_{n,1}(b_n)$ is equivalent to comparing $L_{n,0}$ with $\max_{b_n \in \mathcal{B}_n} L_{n,1}(b_n)$ where $\mathcal{B}_n = \{i_n, \dots, K\}$. Based on this idea, $L_{n,0}$ is evaluated with (7.16), and $L_{n,1}(b_n)$ is computed for all values of $b_n \in \mathcal{B}_n$ with Algorithm 9. If $L_{n,0} > \max_{b_n \in \mathcal{B}_n} L_{n,1}(b_n)$, $t_n(\boldsymbol{\mu}) = 0$, and $P_{s,n}(\boldsymbol{\mu})$ and $\{P_{r_i,n}(\boldsymbol{\mu})|\forall r_i \in \Psi\}$ are computed with (7.14) and (7.15), respectively. Otherwise, $t_n(\boldsymbol{\mu}) = 1$, $b_n(\boldsymbol{\mu}) = \arg \max_{b_n \in \mathcal{B}_n} L_{n,1}(b_n)$, and the $\{P_{s,n}(b_n), P_{r_i,n}(b_n)|\forall r_i \in \Psi\}$ when $b_n = b_n(\boldsymbol{\mu})$ is taken as $\{P_{s,n}(\boldsymbol{\mu}), P_{r_i,n}(\boldsymbol{\mu})|\forall r_i \in \Psi\}$.

In summary, the overall algorithm of finding $\mathbf{x}_\mu$ is summarized in Algorithm 8. The next section will proceed with developing Algorithm 9 to find $L_{n,1}(b_n)$ and $\{P_{s,n}(b_n), P_{r_i,n}(b_n)|\forall r_i \in \Psi\}$.

7.3.2 Algorithm to solve (7.12) when $t_n = 1$ and b_n is fixed

When $t_n = 1$ and b_n has a fixed value in $\mathcal{B}_n$, (7.12) is equivalent to

$$\begin{aligned} \max L_n &= R_{n,1} - \mu_s P_{s,n} - \sum_i \mu_{r_i} P_{r_i,n} \\ &= \log(1 + \gamma_n) - \mu_s P_{s,n} - \sum_i \mu_{r_i} P_{r_i,n} \end{aligned}$$

$$\begin{aligned}
\text{s.t. } \gamma_n &\leq P_{s,n} G_{s,r_{b_n}}(n), \\
\gamma_n &\leq P_{s,n} G_{s,d}(n) + \left(\sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i,n} G_{r_i,d}(n)} \right)^2, \\
\gamma_n &\geq 0, \\
P_{r_i,n} &\geq 0, \forall r_i \in \Psi,
\end{aligned} \tag{7.17}$$

where γ_n is an intermediate variable to guarantee the equivalence.

It can readily be shown that (7.17) is a convex optimization problem. To solve it, one may formulate a set of equations based on the KKT conditions and then solve them for $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$. *It is very important to note that this method is effective only when the objective function and all the constraint functions are differentiable at the optimum solution.* However, the second term in the right hand side of the second constraint in (7.17) is not differentiable at $P_{r_i,n} = 0, \forall r_i \in \Psi_n(b_n)$. This means that if $\exists r_i \in \Psi_n(b_n), P_{r_i,n}(b_n) = 0$, which might happen as shown later, the KKT conditions based method is not capable of finding that optimum solution.

To address this issue, we solve (7.17) based on the idea that $P_{s,n}(b_n)$ is the optimum solution to

$$\begin{aligned}
\max \quad & \log(1 + \gamma_n) - \mu_s P_{s,n} \\
\text{s.t. } \quad & \gamma_n \leq P_{s,n} G_{s,r_{b_n}}(n), \\
& \gamma_n \leq P_{s,n} G_{s,d}(n) + x, \\
& \gamma_n \geq 0,
\end{aligned} \tag{7.18}$$

and $\{P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$ is the optimum solution to

$$\begin{aligned}
\max \quad & \sum_i (-\mu_{r_i} P_{r_i,n}) \\
\text{s.t. } \quad & \sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i,n} G_{r_i,d}(n)} = \sqrt{x}, \\
& P_{r_i,n} \geq 0, \forall r_i \in \Psi,
\end{aligned} \tag{7.19}$$

when $x = x_0$ with

$$x_0 = \left(\sum_{r_i \in \Psi_n(b_n)} \sqrt{P_{r_i,n}(b_n) G_{r_i,d}(n)} \right)^2.$$

Specifically, x_o is first determined, and then (7.18) and (7.19) with $x = x_o$ are solved to compute $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$. At first glance, this method seems confronted with a chicken-and-egg dilemma: though $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$ can be computed by solving (7.18) and (7.19) once x_o is known, it seems that $\{P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$ needs to be known first in order to compute x_o . In fact, this dilemma can be elegantly circumvented by using the *sensitivity analysis* introduced in Chapter 3 to first determine x_o without knowing $\{P_{s,n}(b_n), P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$, as elaborated in the following.

7.3.2.1 Solutions to (7.18) and (7.19) given x

Let's denote the optimum objective values of (7.18) and (7.19) by $f_1(x)$ and $f_2(x)$, respectively. Obviously, (7.18) is a convex optimization problem. Let's denote the optimum γ_n and the optimum dual variables associated with the first and second constraints of (7.18) by $\gamma_n(x)$, $\alpha_n(x)$, and $\beta_n(x)$, respectively. According to the KKT conditions of (7.18),

$$\mu_s = G_{s,r_{b_n}}(n)\alpha_n(x) + G_{s,d}(n)\beta_n(x) \quad (7.20)$$

and

$$\gamma_n(x) = \left[\frac{1}{\alpha_n(x) + \beta_n(x)} - 1 \right]^+ \quad (7.21)$$

should be satisfied.

As for (7.19), it can readily be derived that

- when $\forall r_i \in \Psi_n(b_n)$, $\mu_{r_i} > 0$, $f_2(x) = -\frac{x}{\theta_{b_n}}$, and the optimum $P_{r_i,n}$ to (7.19) is

$$P_{r_i,n} = \begin{cases} 0 & \text{if } r_i \notin \Psi_n(b_n), \\ \frac{G_{r_i,d}(n)}{(\mu_{r_i}\theta_{b_n})^2}x & \text{if } r_i \in \Psi_n(b_n), \end{cases} \quad (7.22)$$

where $\theta_{b_n} = \sum_{r_i \in \Psi_n(b_n)} G_{r_i,d}(n) / \mu_{r_i}$.

- when $\exists r_i \in \Psi_n(b_n)$, $\mu_{r_i} = 0$, $f_2(x) = 0$, and the optimum $P_{r_i,n}$ to (7.19) is equal to 0 if $r_i \notin \Psi_n(b_n)$ or if $r_i \in \Psi_n(b_n)$ with $\mu_{r_i} > 0$. The optimum $\{P_{r_i,n} | \forall r_i \in \Psi_n(b_n), \mu_{r_i} = 0\}$ is any set of nonnegative values satisfying

$$\sum_{r_i: r_i \in \Psi_n(b_n), \mu_{r_i} = 0} \sqrt{P_{r_i,n} G_{r_i,d}(n)} = \sqrt{x}. \quad (7.23)$$

Moreover, it can readily be shown based on the Cauchy-Schwartz inequality that, $\forall \mathbf{r}_i : \mathbf{r}_i \in \Psi_n(b_n)$ and $\mu_{\mathbf{r}_i} = 0$, the optimum $P_{\mathbf{r}_i,n}$ satisfying (7.23) and minimizing the sum power of relays is $P_{\mathbf{r}_i,n} = \frac{G_{\mathbf{r}_i,d}(n)x^2}{(\sum_{\mathbf{r}_i: \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i}=0} G_{\mathbf{r}_i,d}(n))^2}$.

7.3.2.2 Finding x_o based on the sensitivity analysis

Let's denote the L_n in (7.17) computed with the optimum $\{\gamma_n, P_{s,n}\}$ to (7.18) and the optimum $\{P_{\mathbf{r}_i,n} | \forall \mathbf{r}_i \in \Psi\}$ to (7.19) by $L_{n,1}(x, b_n)$ when $x \geq 0$. Obviously, $L_{n,1}(x, b_n) = f_1(x) + f_2(x)$ and $L_{n,1}(x, b_n) \leq L_{n,1}(b_n)$, since $L_{n,1}(b_n)$ is defined in Section 7.3.1 as the maximum L_n for (7.17), while $L_{n,1}(x, b_n)$ is the L_n computed with the above mentioned $\{\gamma_n, P_{s,n}\}$ and $\{P_{\mathbf{r}_i,n} | \forall \mathbf{r}_i \in \Psi\}$ which are feasible for (7.17). When $x = x_o$, the $P_{s,n}$ and $\{P_{\mathbf{r}_i,n} | \forall \mathbf{r}_i \in \Psi\}$ used for computing $L_{n,1}(x, b_n)$ are equal to $P_{s,n}(b_n)$ and $\{P_{\mathbf{r}_i,n}(b_n) | \forall \mathbf{r}_i \in \Psi\}$, respectively, and therefore $L_{n,1}(x_o, b_n) = L_{n,1}(b_n)$. This means that x_o is the $x \geq 0$ that maximizes $L_{n,1}(x, b_n)$.

To determine x_o , let's consider $L'_{n,1}(x, b_n)$ which represents the first order derivative of $L_{n,1}(x, b_n)$ with respect to x . According to the sensitivity analysis, $f'_1(x) = \beta_n(x)$. Therefore,

$$L'_{n,1}(x, b_n) = \begin{cases} \beta_n(x) - 1/\theta_{b_n} & \text{if } \forall \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} > 0, \\ \beta_n(x) & \text{if } \exists \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} = 0. \end{cases} \quad (7.24)$$

Based on the above analysis, the determination of x_o , $\beta_n(x_o)$, $\alpha_n(x_o)$ and $P_{s,n}(b_n)$ falls into one of the following cases:

- when $\forall \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} > 0$ and $\frac{\mu_s}{G_{s,d}(n)} \leq \frac{1}{\theta_{b_n}}$, $x_o = 0$. This is because $L_{n,1}(x, b_n)$ is a nonincreasing function of x since $\beta_n(x) \in [0, \mu_s/G_{s,d}(n)]$ and $L'_{n,1}(x, b_n) \leq 0$ for any $x \geq 0$. In this case, the first constraint in (7.18) when $x = x_o$ is relaxed, whereas the second one is tightened, since $G_{s,d}(n) < G_{s,\mathbf{r}_{b_n}}(n)$. Therefore, $\alpha_n(x_o) = 0$, $\beta_n(x_o) = \frac{\mu_s}{G_{s,d}(n)}$, $\gamma_n(x_o) = \left[\frac{G_{s,d}(n)}{\mu_s} - 1 \right]^+$, and

$$P_{s,n}(b_n) = \frac{\gamma_n(x_o)}{G_{s,d}(n)} = \left[\frac{1}{\mu_s} - \frac{1}{G_{s,d}(n)} \right]^+. \quad (7.25)$$

- when $\forall \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} > 0$ and $\frac{\mu_s}{G_{s,d}(n)} > \frac{1}{\theta_{b_n}}$, $\beta_n(x_o) = \frac{1}{\theta_{b_n}}$ since $L'_{n,1}(x_o, b_n) = 0$ should be satisfied. In this case, $\alpha_n(x_o) =$

$\frac{\mu_s - G_{s,d}(n)/\theta_{b_n}}{G_{s,r_{b_n}}(n)} > 0$ because (7.20) is satisfied, and $\gamma_n(x_o)$ can be computed with (7.21). This means that both constraints in (7.18) when $x = x_o$ are saturated, and therefore

$$\begin{aligned} P_{s,n}(b_n) &= \frac{\gamma_n(x_o)}{G_{s,r_{b_n}}(n)} \\ &= \left[\frac{1}{\mu_s + \Delta_{b_n}/\theta_{b_n}} - \frac{1}{G_{s,r_{b_n}}(n)} \right]^+, \end{aligned} \quad (7.26)$$

and

$$\begin{aligned} x_o &= P_{s,n}(b_n) \Delta_{b_n} \\ &= \left[\frac{1}{\mu_s/\Delta_{b_n} + 1/\theta_{b_n}} - \frac{\Delta_{b_n}}{G_{s,r_{b_n}}(n)} \right]^+, \end{aligned} \quad (7.27)$$

where $\Delta_{b_n} = G_{s,r_{b_n}}(n) - G_{s,d}(n)$.

- when $\exists r_i \in \Psi_n(b_n)$, $\mu_{r_i} = 0$, $\beta_n(x_o) = 0$ since $L'_{n,1}(x_o, b_n) = 0$ should be satisfied. In this case, $\alpha_n(x_o) = \frac{\mu_s}{G_{s,r_{b_n}}(n)}$, and $\gamma_n(x_o) = \left[\frac{G_{s,r_{b_n}}(n)}{\mu_s} - 1 \right]^+$. This means that the first constraint in (7.18) when $x = x_o$ is saturated, whereas the second one is relaxed. Therefore,

$$P_{s,n}(b_n) = \frac{\gamma_n(x_o)}{G_{s,r_{b_n}}(n)} = \left[\frac{1}{\mu_s} - \frac{1}{G_{s,r_{b_n}}(n)} \right]^+, \quad (7.28)$$

and x_o can be any value satisfying $x_o \geq \gamma_n(x_o) - P_{s,n}(b_n)G_{s,d}(n) = x_{th}$ where

$$x_{th} = \Delta_{b_n} \left[\frac{1}{\mu_s} - \frac{1}{G_{s,r_{b_n}}(n)} \right]^+. \quad (7.29)$$

After knowing x_o , the optimum $\{P_{r_i,n} | \forall r_i \in \Psi\}$ for (7.19) when $x = x_o$ can be found and assigned to $\{P_{r_i,n}(b_n) | \forall r_i \in \Psi\}$. Note that in the third case x_o can be any value no smaller than x_{th} , and $\{P_{r_i,n}(b_n) | \forall r_i \in \Psi_n(b_n), \mu_{r_i} = 0\}$ can be any set of nonnegative values satisfying (7.23) with $x = x_o$. To improve the system energy efficiency, the following values are chosen: $x_o = x_{th}$, and $\forall r_i : r_i \in \Psi_n(b_n)$ and $\mu_{r_i} = 0$, $P_{r_i,n}(b_n) = \frac{G_{r_i,d}(n)x_{th}^2}{(\sum_{r_i:r_i \in \Psi_n(b_n), \mu_{r_i}=0} G_{r_i,d}(n))^2}$ to minimize the sum power of the relays.

Algorithm 9 Algorithm to solve (7.17) when μ is fixed

```

if  $\forall \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} > 0$  and  $\frac{\mu_s}{G_{s,d}(n)} \leq \frac{1}{\theta_{b_n}}$  then
     $P_{s,n}(b_n)$  is evaluated by (7.25);
     $P_{\mathbf{r}_i,n}(b_n) = 0, \forall \mathbf{r}_i \in \Psi$ ;
else if  $\forall \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} > 0$  and  $\frac{\mu_s}{G_{s,d}(n)} > \frac{1}{\theta_{b_n}}$  then
     $P_{s,n}(b_n)$  is evaluated by (7.26);
     $\forall \mathbf{r}_i \notin \Psi_n(b_n), P_{\mathbf{r}_i,n}(b_n) = 0$ ;
     $\forall \mathbf{r}_i \in \Psi_n(b_n), P_{\mathbf{r}_i,n}(b_n)$  is evaluated by (7.22) with  $x$  equal to  $x_o$  computed with
    (7.27);
else if  $\exists \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} = 0$  then
     $P_{s,n}(b_n)$  is evaluated by (7.28),
     $\forall \mathbf{r}_i \notin \Psi_n(b_n), P_{\mathbf{r}_i,n}(b_n) = 0$ ;
     $\forall \mathbf{r}_i \in \Psi_n(b_n)$  with  $\mu_{\mathbf{r}_i} > 0, P_{\mathbf{r}_i,n}(b_n) = 0$ ;
     $\forall \mathbf{r}_i \in \Psi_n(b_n)$  with  $\mu_{\mathbf{r}_i} = 0, P_{\mathbf{r}_i,n}(b_n) = \frac{G_{\mathbf{r}_i,d}(n)x_{th}^2}{(\sum_{\mathbf{r}_i: \mathbf{r}_i \in \Psi_n(b_n), \mu_{\mathbf{r}_i} = 0} G_{\mathbf{r}_i,d}(n))^2}$ .
end if

```

The overall algorithm of finding $\{P_{s,n}(b_n), P_{\mathbf{r}_i,n}(b_n) | \forall \mathbf{r}_i \in \Psi\}$ is summarized in Algorithm 9. Based on the above analysis, it can be seen that the ς_n corresponding to $\{P_{s,n}(b_n), P_{\mathbf{r}_i,n}(b_n) | \forall \mathbf{r}_i \in \Psi\}$ and b_n must be equal to or smaller than $P_{s,n}(b_n)G_{s,r_{b_n}}(n)$.

It is interesting to examine the complexity of the duality-based algorithm. Suppose T is the average number of iterations to update the Lagrange multipliers. In each iteration, the number of operations is proportional to N and K . Therefore, the complexity of the duality-based algorithm is $O(TKN)$.

7.4 The iterative RA algorithm

In case (7.8) has a nonzero duality gap, Algorithm 7 fails to find a global optimum. To address this issue, a CA-based iterative algorithm, which is sub-optimum but always applicable regardless of the duality gap of (7.8), is developed in this section.

7.4.1 The CA-based iterative RA algorithm

First of all, it should be noted that $R_{n,1} \leq R_{n,2}$ always holds independently of $\{P_{s,n}, P_{\mathbf{r}_i,n} | \forall \mathbf{r}_i \in \Psi\}$ when b_n satisfies $G_{s,r_{b_n}}(n) \leq G_{s,d}(n)$. This means that when $\max_{\mathbf{r}_i \in \Psi} G_{s,\mathbf{r}_i}(n) \leq G_{s,d}(n)$, $t_n = 0$ is the optimum independently of

b_n and $\{P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi\}$, i.e., the direct transmission mode should always be used for carrier n .

To simplify algorithm design, assume t_n is fixed as 0 for every $n \in \mathcal{D}_a = \{n | \max_{r_i} G_{s,r_i}(n) \leq G_{s,d}(n)\}$. If $n \notin \mathcal{D}_a$, the optimum t_n and b_n must lie in the set $\{0, 1\} \times \mathcal{B}_n$, where $\times$ represents the Cartesian product. Furthermore, when $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$ is fixed, (7.8) is reduced to a convex optimization problem, which has zero duality gap since the Slater constraint qualification is satisfied. Thus a globally optimum $\{P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi, \forall n\}$ and the maximum sum rate when $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$ is fixed can be found with a duality based algorithm as shown in Section 7.4.2.

Now the difficulty of solving (7.8) lies in finding the optimum $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$. To this end, one may use an exhaustive-search based algorithm. Specifically, this algorithm finds the maximum sum rate for every possible $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$ in $\prod_{n \notin \mathcal{D}_a} (\{0, 1\} \times \mathcal{B}_n)$, then chooses the best one as the optimum $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$. However, the complexity of exhaustive search might be unaffordable for practical systems using a big number of carriers.

To address this issue, a coordinate-ascent based iterative algorithm is developed to produce a successive set of RAs with increasing sum rate until convergence. In the following, a superscript m added to a variable indicates that variable is produced at the m -th iteration to facilitate description. At the beginning, $\forall n \notin \mathcal{D}_a$, t_n^1 and b_n^1 are initialized as 1 and N , respectively, i.e. every carrier not in $\mathcal{D}_a$ is set in the relay-aided mode with only the relay having the highest source-relay channel gain enabled for assisting. In the m -th iteration, $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall n\}$, which is the optimum solution to (7.8) when $t_n = t_n^m$ and $b_n = b_n^m, \forall n \notin \mathcal{D}_a$, is first found with a duality based algorithm, namely Algorithm 11 developed in Section 7.4.2. This step can be interpreted as finding the optimum source/relay power allocation when $\{t_n, b_n | \forall n \notin \mathcal{D}_a\}$ is fixed as $\{t_n^m, b_n^m | \forall n \notin \mathcal{D}_a\}$.

Then, $\{t_n^{m+1}, b_n^{m+1} | \forall n \notin \mathcal{D}_a\}$ which maximizes the sum rate is found when the power allocation is fixed as $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall n\}$. This step can be interpreted as finding the optimum mode and assisting relays when the source/relay power allocation is prescribed by $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall n\}$. Note that this can be accomplished on a per carrier basis, i.e., for every carrier $n \notin \mathcal{D}_a$, t_n^{m+1} and b_n^{m+1} are found to maximize the rate when the power allocation is fixed as $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi\}$. In this case, the rate is

Algorithm 10 The iterative RA algorithm

$\forall n \in \mathcal{D}_a, t_n = 0; \forall n \notin \mathcal{D}_a, t_n^1 = 1$ and $b_n^1 = N$;
 $m = 1$;
repeat
 Find $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall n\}$ as the optimum solution to (7.8) when $t_n = t_n^m$ and $b_n = b_n^m, \forall n \notin \mathcal{D}_a$ with Algorithm 11;
 for every $n \notin \mathcal{D}_a$ **do**
 if $t_n^m = 0$ **then**
 $t_n^{m+1} = 0$;
 else
 Compute $\zeta_n^m(b_n^m)$, $R_{n,1}^m(b_n^m)$, and $R_{n,2}^m$;
 if $R_{n,1}^m(b_n^m) < R_{n,2}^m$ **then**
 $t_n^{m+1} = 0$;
 else
 $t_n^{m+1} = 1$;
 b_n^{m+1} is set as the minimum i satisfying $\zeta_n^m(b_n^m) \leq P_{s,n}^m G_{s,r_i(n)}(n)$;
 end if
 end if
 end for
 $m = m + 1$;
until $\forall n \notin \mathcal{D}_a, \{t_n^{m-1}, b_n^{m-1}\}$ is equal to $\{t_n^m, b_n^m\}$
 $\{t_n = 0 | \forall k \in \mathcal{D}_a\} \cup \{t_n^m, b_n^m | \forall k \notin \mathcal{D}_a\}$ and $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall n\}$ produced in the last iteration are a suboptimum solution.

denoted by $R_{n,2}^m = 2\log(1 + \frac{P_{s,n}^m}{2} G_{s,d}(n))$ if the direct mode is used. If the relay-aided mode with a given b_n is used, the rate is denoted by $R_{n,1}^m(b_n) = \log(1 + \min(\zeta_n^m(b_n), P_{s,n}^m G_{s,r_{b_n}(n)}(n)))$ where $\zeta_n^m(b_n)$ represents the ζ_n corresponding to $\Psi_n(b_n)$ and $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi\}$.

Let's first consider the evaluation of t_n^{m+1} and b_n^{m+1} when $t_n^m = 0$, meaning that carrier n was set in the direct mode when $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall k\}$ was evaluated. Suppose the relay-aided mode with any b_n^{m+1} is now used instead, the rate is reduced independently of b_n^{m+1} because of

$$\begin{aligned} R_{n,1}^m(b_n^{m+1}) &= \log(1 + \min(\zeta_n^m(b_n^{m+1}), P_{s,n}^m G_{s,r_{b_n^{m+1}}(n)}(n))) \\ &\leq \log(1 + \zeta_n^m(b_n^{m+1})) \\ &= \log(1 + P_{s,n}^m G_{s,d}(n)) \leq R_{n,2}^m, \end{aligned}$$

where the equality in the third line is because of $\forall r_i \in \Psi, P_{r_i,n}^m = 0$ as will be shown in Section 7.4.2. In order to maximize the rate, $t_n^{m+1} = 0$, i.e. carrier n remains in the direct mode.

Next, let's consider the evaluation of t_n^{m+1} and b_n^{m+1} when $t_n^m = 1$, meaning that carrier n was set in the relay-aided mode with $b_n = b_n^m$ when $\{P_{s,n}^m, P_{r_i,n}^m | \forall r_i \in \Psi, \forall k\}$ was evaluated. As will be shown in Section 7.4.2, $\zeta_n^m(b_n^m) \leq P_{s,n}^m G_{s,r_{b_n^m}(n)}(n)$ always holds and thus $R_{n,1}^m(b_n^m) = \log(1 + \zeta_n^m(b_n^m))$. It can readily be seen from (7.5) that

$$\zeta_n^m(b_n) \begin{cases} \leq \zeta_n^m(b_n^m) & \text{if } b_n > b_n^m; \\ = \zeta_n^m(b_n^m) & \text{if } b_n \leq b_n^m, \end{cases}$$

because $\forall r_i \notin \Psi_n(b_n^m), P_{r_i,n}^m = 0$ as will be shown in Section 7.4.2. This means that $\forall b_n \in \{1, \dots, N\}, R_{n,1}^m(b_n) \leq \log(1 + \zeta_n^m(b_n)) \leq R_{n,1}^m(b_n^m)$ always follows, i.e., if carrier n remains in the relay-aided mode, the rate can not be increased no matter which value is assigned to b_n . In this case, t_n^{m+1} and b_n^{m+1} are determined with one of the following procedures:

- if $R_{n,2}^m > R_{n,1}^m(b_n^m)$, t_n^{m+1} is set as 0 since the rate is increased if the direct mode is used;
- if $R_{n,2}^m \leq R_{n,1}^m(b_n^m)$, t_n^{m+1} is still set as 1. b_n^{m+1} is assigned as the smallest b_n satisfying $R_{n,1}^m(b_n) = R_{n,1}^m(b_n^m)$. Obviously, $b_n^{m+1} \leq b_n^m$, and $\Psi_n(b_n^{m+1})$ is the biggest set of assisting relays that leads to the same rate as $\Psi_n(b_n^m)$. The motivation behind this assignment is twofold. One is to guarantee the rate when $b_n = b_n^{m+1}$ is not smaller

than when $b_n = b_n^m$. The other is to increase the degrees of freedom for optimizing the power allocation in the next iteration, since $\forall r_i : r_i \in \Psi_n(b_n^{m+1})$ and $r_i \notin \Psi_n(b_n^m)$, $P_{r_i,n}^m = 0$ always holds but $P_{r_i,n}$ is free to be optimized in the $(m+1)$ -th iteration. It can easily be shown that b_n^{m+1} is actually equal to the minimum i satisfying $\zeta_n^m(b_n^m) \leq P_{s,n}^m G_{s,r_i(n)}(n)$. Note that $G_{s,r_{b_n^{m+1}}(n)}(n) \geq \frac{\zeta_n^m(b_n^m)}{P_{s,n}^m} \geq G_{s,d}(n)$, which means that $b_n^{m+1} \in \mathcal{B}_n$.

It can easily be seen that a successive set of RAs with nondecreasing sum rate are produced as the iteration proceeds. After the algorithm converges, the RA produced by the last iteration is output as a suboptimum solution. In summary, the iterative algorithm is described in Algorithm 10.

7.4.2 The algorithm to solve (7.8) when t_n and b_n are fixed

When $t_n = t_n^m$ and $b_n = b_n^m$, $\forall n \notin \mathcal{D}_a$, and $t_n = 0$, $\forall n \in \mathcal{D}_a$, (7.8) is equivalent to

$$\begin{aligned} \max \quad & \sum_{n \notin \mathcal{D}_a} (t_n^m R_{n,1} + (1 - t_n^m) R_{n,2}) + \sum_{n \in \mathcal{D}_a} R_{n,2} \\ \text{s.t.} \quad & \sum_n P_{s,n} \leq 1, \quad \sum_n P_{r_i,n} \leq 1, \forall r_i \in \Psi, \\ & P_{s,n} \geq 0, \forall n, \quad P_{r_i,n} \geq 0, \forall n, \forall r_i \in \Psi, \end{aligned} \quad (7.30)$$

which has zero duality gap as shown earlier, and therefore can be solved with a duality based algorithm. To this end, let's define the dual variables ν_s and ν_{r_i} related to the sum power constraints of the source and r_i , respectively. Let's stack ν_s and $\{\nu_{r_i} | \forall r_i \in \Psi\}$ into a vector ν , and $\{P_{s,n}, P_{r_i,n} | \forall r_i \in \Psi, \forall n\}$ into a vector $\mathbf{P}$. Then, we define the Lagrangian as

$$\begin{aligned} W(\mathbf{P}, \nu) = & \sum_{n \notin \mathcal{D}_a} (t_n^m R_{n,1} + (1 - t_n^m) R_{n,2}) + \sum_{n \in \mathcal{D}_a} R_{n,2} \\ & + \nu_s (1 - \sum_n P_{s,n}) + \sum_i \nu_{r_i} (1 - \sum_n P_{r_i,n}). \end{aligned}$$

The duality based algorithm looks for the optimum ν with the subgradient based method. In each iteration, the optimum $\mathbf{P}$ that maximizes $W(\mathbf{P}, \nu)$ subject to the constraints $P_{s,n} \geq 0$, $P_{r_i,n} \geq 0$, $\forall n, \forall r_i \in \Psi$, denoted by $\mathbf{P}_\nu$, is found when ν is fixed. Let's denote $P_{s,n}$ and $P_{r_i,n}$ contained in $\mathbf{P}_\nu$ as $P_{s,n}(\nu)$ and $P_{r_i,n}(\nu)$, respectively. Specifically, for carrier n with $t_n^m = 1$ and $b_n = b_n^m$, the problem of finding $\{P_{s,n}(\nu), P_{r_i,n}(\nu) | \forall r_i \in \Psi\}$ is equivalent to (7.17) with

Algorithm 11 The duality based RA algorithm to solve (7.8) when $t_n = t_n^m$ and $b_n = b_n^m, \forall n \notin \mathcal{D}_a$, and $t_n = 0, \forall n \in \mathcal{D}_a$

```

 $l = 1, v = 1;$ 
repeat
   $v_s = [v_s - \frac{\delta_2}{T}(1 - \sum_n P_{s,n}(v))]^+;$ 
   $v_{r_i} = [v_{r_i} - \frac{\delta_2}{T}(1 - \sum_n P_{r_i,n}(v))]^+, \forall r_i \in \Psi;$ 
   $l = l + 1;$ 
  for  $n = 1$  to  $N$  do
    if  $n \in \mathcal{D}_a$  or  $n \notin \mathcal{D}_a$  with  $t_n^m = 0$  then
       $P_{s,n}(v)$  and  $P_{r_i,n}(v), \forall r_i$  are found with (7.14) and (7.15) after replacing  $\mu_s$ 
      with  $v_s$ ;
    else if  $k \notin \mathcal{D}_a$  with  $t_n^m = 1$  then
       $P_{s,n}(v)$  and  $P_{r_i,n}(v), \forall r_i$  are found with Algorithm 9 after replacing
       $\{b_n, \mu_s, \mu_{r_i} | \forall r_i \in \Psi\}$  with  $\{b_n^m, v_s, v_{r_i} | \forall r_i \in \Psi\};$ 
    end if
  end for
until  $P_v$  is feasible for (7.30) and  $v_s(1 - \sum_n P_{s,n}(v)) + \sum_{r_i \in \Psi} v_{r_i}(1 - \sum_n P_{r_i,n}(v)) < \epsilon_2$ 
 $P_{s,n}(v)$ , and  $P_{r_i,n}(v), \forall r_i \in \Psi$  produced in the last iteration are the optimum solution
to (7.8) when  $t_n = t_n^m$  and  $b_n = b_n^m, \forall n \notin \mathcal{D}_a$ , and  $t_n = 0, \forall n \in \mathcal{D}_a$ .

```

$b_n = b_n^m, \mu_s = v_s$, and $\mu_{r_i} = v_{r_i}, \forall r_i \in \Psi$, and therefore can be solved with Algorithm 9 after replacing $\{b_n, \mu_s, \mu_{r_i} | \forall r_i \in \Psi\}$ with $\{b_n^m, v_s, v_{r_i} | \forall r_i \in \Psi\}$. It is important to note that the ς_n corresponding to $\{P_{s,n}(v), P_{r_i,n}(v) | \forall r_i \in \Psi\}$ and $b_n = b_n^m$ must be equal to or smaller than $P_{s,n}(v)G_{s,r_{b_n^m}}(n)$, according to the analysis at the end of Section 7.3.2. For carrier n with $t_n^m = 0$ or in $\mathcal{D}_a$, the problem of finding $\{P_{s,n}(v), P_{r_i,n}(v) | \forall r_i \in \Psi\}$ is equivalent to (7.12) with $t_n = 0, \mu_s = v_s$, and $\mu_{r_i} = v_{r_i}, \forall r_i \in \Psi$, and therefore can be solved with (7.14) and (7.15) after replacing μ_s with v_s . Note that if $k \in \mathcal{D}_a$ or $t_n^m = 0, P_{r_i,n}(v) = 0$ follows $\forall r_i \in \Psi$, and if $t_n^m = 1, P_{r_i,n}(v) = 0$ follows $\forall r_i \notin \Psi_n(b_n^m)$.

The overall algorithm is summarized as Algorithm 11, where $\epsilon_2 > 0$ and $\delta_2 > 0$ are both very small values prescribed to guarantee convergence. The step size and the termination condition are chosen in the same way as in Algorithm 7.

It is interesting to examine the complexity of Algorithm 10. Suppose T_1 and T_2 are the average number of outer iterations for Algorithm 10 and that of

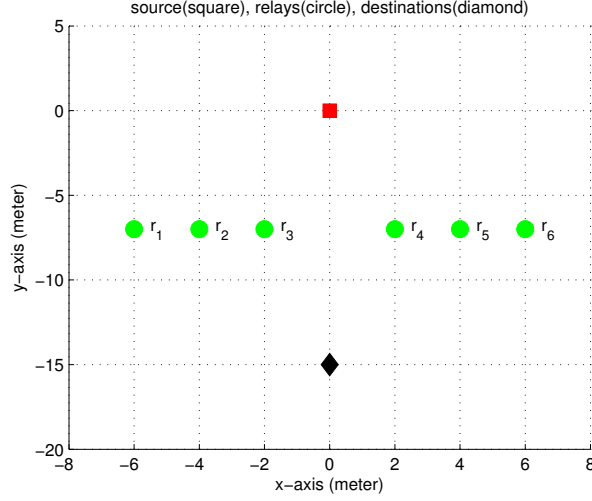


Figure 7.3 A DF relays aided OFDM transmission system considered in numerical experiments.

iterations for updating Lagrange multipliers by Algorithm 11. The complexity of the duality-based algorithm is $O(T_1 T_2 KN)$.

7.5 Numerical experiments

For illustration purposes, consider an OFDM transmission system aided by 6 DF relays shown in Figure 8.1. The source is located at the origin, the destination is located at the coordinate $(0, -15)$, and $r_i, i = 1, \dots, 6$, is located at the coordinates $(-6, -7), (-4, -7), (-2, -7), (2, -7), (4, -7)$, and $(6, -7)$, respectively. Note that all the aforementioned coordinate-related values have the unit of meter. The parameters are set as $\sigma^2 = -50$ dBm, $\epsilon = 0.1$, $\delta_1 = \delta_2 = 0.01$ and $N = 256$ for which the convergence of Algorithm 7 was always observed. Every channel is generated in the same way as explained in Section 5.6. The following experiments are conducted with Matlab v7.1 on a laptop equipped with an Intel Duo CPU of 2.2 GHz and a memory of 2 GBytes.

In order to illustrate the effectiveness of the proposed algorithms, a heuristic RA algorithm was implemented and compared with the proposed RA algorithms. This heuristic algorithm allocates the source sum power uniformly to all carriers, i.e., $P_{s,n} = \frac{1}{N}$. Then, for every r_i , $\Omega_{r_i} = \{n | n \notin \mathcal{D}_a, G_{s,r_i}(n) = \max_{r_j \in \Psi} G_{s,r_j}(n)\}$ which contains every carrier not belonging to $\mathcal{D}_a$ and at

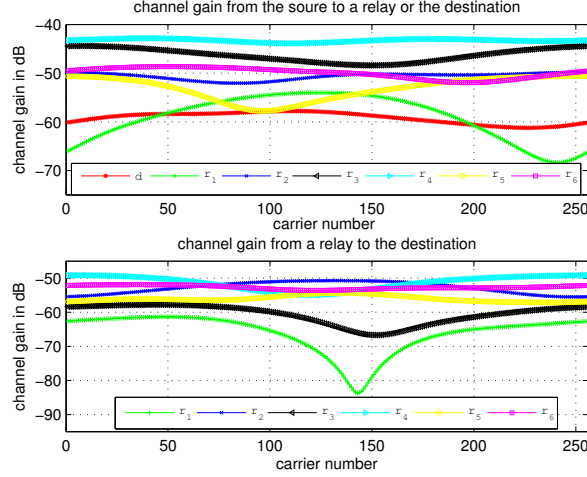


Figure 7.4 A random realization of channel gains $|h_{s,d}(n)|^2$, $|h_{s,r_i}(n)|^2$, and $|h_{r_i,d}(n)|^2$, $\forall r_i \in \Psi, \forall k$.

which r_i has the maximum source to relay channel gain, is first found. The sum power of r_i is uniformly allocated to all carriers in Ω_{r_i} (i.e., if $n \notin \Omega_{r_i}$, $P_{r_i,n} = 0$, otherwise $P_{r_i,n} = \frac{1}{|\Omega_{r_i}|}$ where $|\Omega_{r_i}|$ denotes the number of carriers in Ω_{r_i}). For every carrier $n \in \mathcal{D}_a$, the direct transmission mode is used. For every carrier $n \notin \mathcal{D}_a$, $R_{n,2}$ is first computed, and then the $R_{n,1}$ when a single r_i with the maximum $G_{s,r_i}(n)$ assists relaying is computed. If $R_{n,2}$ is equal to or greater than the computed $R_{n,1}$, the direct mode is used. Otherwise, the relay mode is used, and only the r_i used for computing $R_{n,1}$ assists relaying.

The three RA algorithms were first tested for a single random channel realization shown in Figure 7.4 when $P_s = P_{r_i}, \forall r_i \in \Psi$, and P_s varies from 10 to 50 dBm. For the RA evaluated by each algorithm, the sum rate is shown in Figure 7.5. It can be seen that for each P_s , both Algorithm 7 and Algorithm 10 produce better RAs with a higher sum rate than the heuristic algorithm, and Algorithm 7 produces a better RA than Algorithm 10. The RAs evaluated by the three algorithms for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$, are shown in Figure 7.6, 7.7, and 7.8, respectively. When the RA evaluated by Algorithm 7 is used, r_3 and r_4 are enabled to assist relaying simultaneously at a few carriers, while only r_4 does when the RA computed by Algorithm 10 or the heuristic algorithm is used.

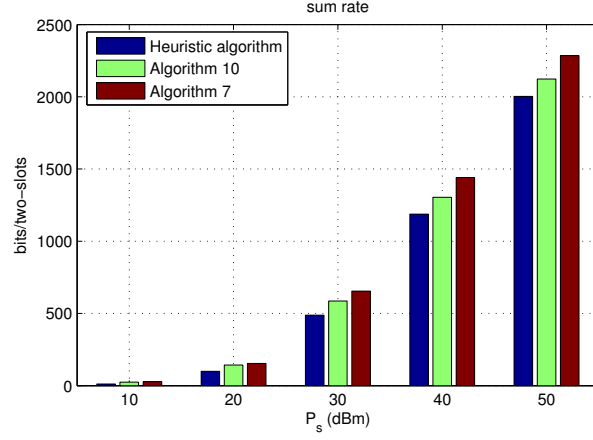


Figure 7.5 The sum rate for each algorithm when $P_s = P_{r_i}, \forall r_i \in \Psi$, and P_s varies from 10 to 50 dBm. In each bar group, the bars from the left to the right represent the performance for the heuristic algorithm, Algorithm 10 and Algorithm 7, respectively.

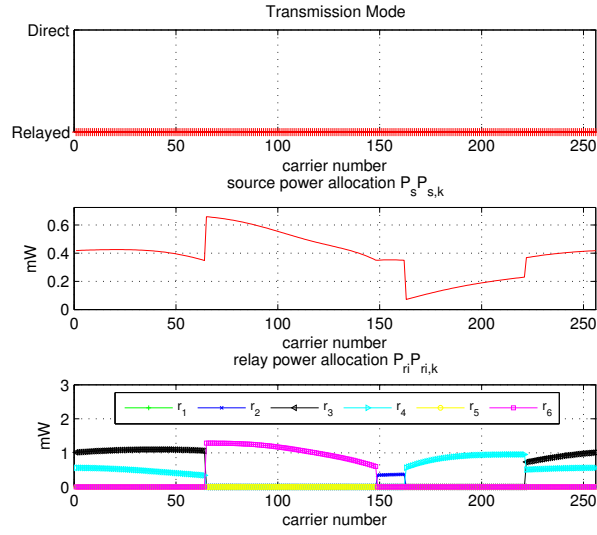


Figure 7.6 The RA evaluated by Algorithm 7 for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$.

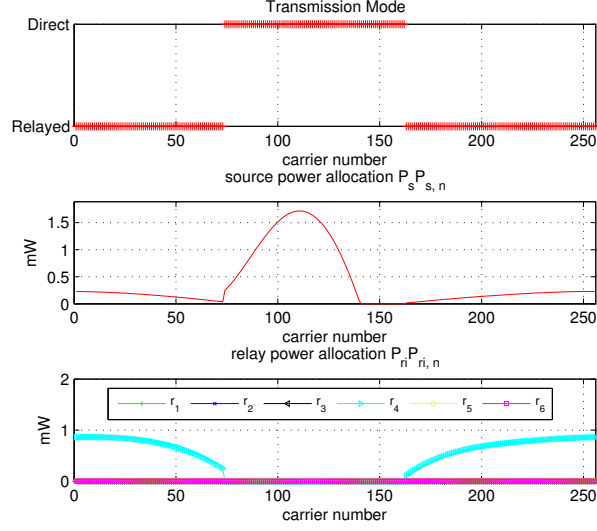


Figure 7.7 The RA evaluated by Algorithm 10 for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$.

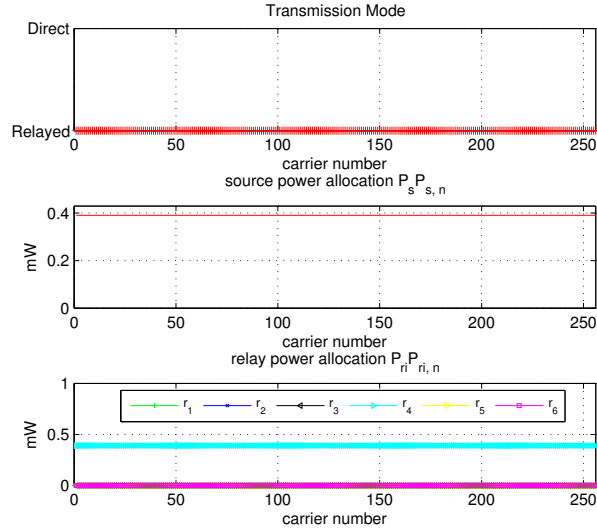


Figure 7.8 The RA evaluated by the heuristic algorithm for the random channel realization when $P_s = P_{r_i} = 20$ dBm, $\forall r_i \in \Psi$.

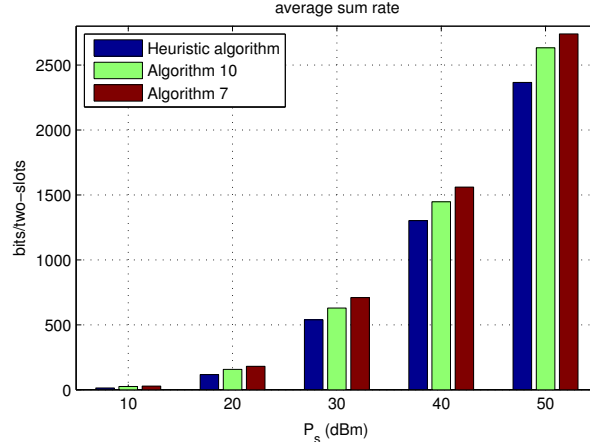


Figure 7.9 The average sum rate for each algorithm when $P_s = P_{r_i}, \forall r_i \in \Psi$, and P_s varies from 10 to 50 dBm. In each bar group, the bars from the left to the right represent the performance for the heuristic algorithm, Algorithm 10 and Algorithm 7, respectively.

Table 7.2 Average execution time in seconds for the two algorithms.

K, N	3, 256	6, 256	6, 512
Algorithm 7/10	6.2/28.4	10.8/54.3	23.5/112.9

The three RA algorithms have also been tested over 100 random channel realizations, when $P_s = P_{r_i}, \forall r_i \in \Psi$ and P_s varies from 10 to 50 dBm. For the RAs evaluated by each algorithm, the average sum rates are shown in Figure 7.9. It can be seen that for each P_s , both Algorithm 10 and Algorithm 7 produce better RAs with a higher average sum rate than the heuristic algorithm, and Algorithm 7 produces better RAs than Algorithm 10.

To show the complexity of the proposed two-step algorithm, the average execution time is shown in Table 7.2. It can be seen that Algorithm 7 takes much longer time than Algorithm 10, and the execution times of the two algorithms scales up at a similar rate as N and K increase.

7.6 Appendix: justifying the zero duality gap of (7.9)

The duality gap of (7.9) can be studied with the visualization method earlier introduced in Section 3.4.1. Specifically, a cloud of points in the set

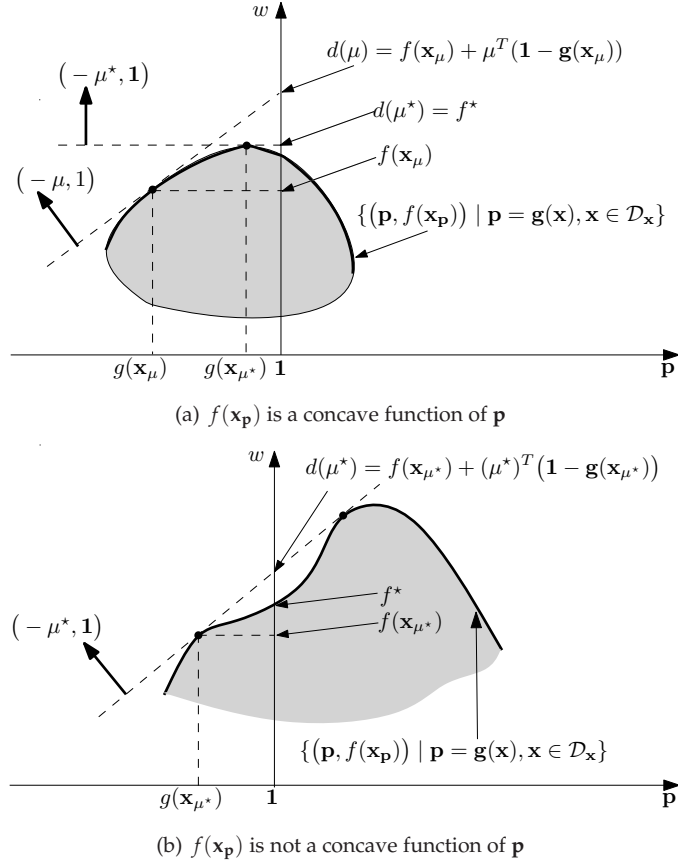


Figure 7.10 The visualization of the duality gap in the hyperplane of $(\mathbf{p}, w)$, where the grey area represents $\mathcal{S} = \{(\mathbf{p}, w) \mid \mathbf{p} = \mathbf{g}(\mathbf{x}), w = f(\mathbf{x}), \mathbf{x} \in \mathcal{D}_{\mathbf{x}}\}$.

$\mathcal{S} = \{(\mathbf{p}, w) \mid \mathbf{p} = \mathbf{g}(\mathbf{x}), w = f(\mathbf{x}), \mathbf{x} \in \mathcal{D}_{\mathbf{x}}\}$ can be plot in the hyperplane of $(\mathbf{p}, w)$ shown in Figure 7.10. Most interestingly, $d(\mu)$ is equal to the w -coordinate of the highest intersection between the line $\mathbf{p} = 1$ and a line passing through $\mathcal{S}$ and perpendicular to the vector $(-\mu, 1)$. As shown in Fig. 7.10, the duality gap is equal to zero if the w -coordinate of the upper border of $\mathcal{S}$ is a concave function of $\mathbf{p}$. Mathematically, a point on the upper border of $\mathcal{S}$ has the coordinate $(\mathbf{p}, f(\mathbf{x}_{\mathbf{p}}))$, where

$$\mathbf{x}_{\mathbf{p}} = \arg \max_{\mathbf{x} \in \mathcal{D}_{\mathbf{x}}, \mathbf{g}(\mathbf{x}) = \mathbf{p}} f(\mathbf{x}). \quad (7.31)$$

Based on the same idea first proposed in [YL06], it can be shown that $f(\mathbf{x}_{\mathbf{p}})$ is a concave function of $\mathbf{p}$ when N is very large. This is equivalent to show that for any $\theta \in [0, 1]$,

$$f(\mathbf{x}_{\theta\mathbf{p}_1+(1-\theta)\mathbf{p}_2}) \geq \theta f(\mathbf{x}_{\mathbf{p}_1}) + (1-\theta)f(\mathbf{x}_{\mathbf{p}_2}) \quad (7.32)$$

holds for any $\mathbf{p}_1$ and $\mathbf{p}_2$ stacking the sum power of the source and relays. Note that the above condition can be interpreted in a very interesting way as follows. Let's adopt the RAs $\mathbf{x}_{\mathbf{p}_1}$ and $\mathbf{x}_{\mathbf{p}_2}$ in the θ and $1-\theta$ portions of the whole transmission duration, respectively, which is called θ time sharing of $\mathbf{x}_{\mathbf{p}_1}$ and $\mathbf{x}_{\mathbf{p}_2}$ hereafter. In this way, an average sum rate $\theta f(\mathbf{x}_{\mathbf{p}_1}) + (1-\theta)f(\mathbf{x}_{\mathbf{p}_2})$ can be achieved with an average sum power $\theta\mathbf{p}_1 + (1-\theta)\mathbf{p}_2$. *This means that showing the validity of (7.32) for any $\theta \in [0, 1]$, is equivalent to show that the optimum RA for the sum power $\theta\mathbf{p}_1 + (1-\theta)\mathbf{p}_2$ provides a higher sum rate than θ time sharing of $\mathbf{x}_{\mathbf{p}_1}$ and $\mathbf{x}_{\mathbf{p}_2}$.*

When N is very large, a RA $\mathbf{x}' \in \mathcal{D}_{\mathbf{x}}$, which is of the sum power $\theta\mathbf{p}_1 + (1-\theta)\mathbf{p}_2$ and yields a sum rate equal to θ time sharing of $\mathbf{x}_{\mathbf{p}_1}$ and $\mathbf{x}_{\mathbf{p}_2}$, can be constructed by θ spectrum sharing of $\mathbf{x}_{\mathbf{p}_1}$ and $\mathbf{x}_{\mathbf{p}_2}$, i.e., taking the entries of $\mathbf{x}_{\mathbf{p}_1}$ at θ portion of all carriers, and the entries of $\mathbf{x}_{\mathbf{p}_2}$ at all the other carriers to construct $\mathbf{x}'$. Specifically, all carriers are first divided into S subbands, each consisting of N_s adjacent carriers experiencing the same channel conditions. Suppose N is sufficiently large so that N_s is also very large, and $\forall \theta \in [0, 1]$ there exists an integer N_θ in $\{1, \dots, N_s\}$ satisfying $\theta \approx \frac{N_\theta}{N_s}$. Since the carriers in each subband experience the same channel conditions, the entries of $\mathbf{x}_{\mathbf{p}_1}$ (or $\mathbf{x}_{\mathbf{p}_2}$) at the carriers in the same subband should be the same. Let's construct $\mathbf{x}'$, which in every subband adopts for the first N_θ carriers the same entries as in $\mathbf{x}_{\mathbf{p}_1}$, and for the remaining $N_s - N_\theta$ carriers the same entries as in $\mathbf{x}_{\mathbf{p}_2}$. It can easily be seen that the conditions $\mathbf{x}' \in \mathcal{D}_{\mathbf{x}}$ and

$$\begin{aligned} g(\mathbf{x}') &\approx \theta\mathbf{p}_1 + (1-\theta)\mathbf{p}_2, \\ f(\mathbf{x}') &\approx \theta f(\mathbf{p}_1) + (1-\theta)f(\mathbf{p}_2) \end{aligned}$$

are all satisfied. Thus $f(\mathbf{x}_{\theta\mathbf{p}_1+(1-\theta)\mathbf{p}_2}) \geq f(\mathbf{x}') \approx \theta f(\mathbf{p}_1) + (1-\theta)f(\mathbf{p}_2)$ follows. Therefore, the duality gap of (7.9) is equal to zero when N is very large.

WSR maximized RA in multiple DF relays aided OFDMA systems

8

8.1 Related works and chapter overview

This chapter addresses RA for an OFDMA downlink transmission system from a single source to multiple destinations aided by multiple DF relays. For the point-to-point OFDM transmission aided by DF relays, some RA algorithms, e.g., those earlier said in Chapter 7, have been proposed lately. For the OFDMA downlink transmission aided by DF relays, some RA algorithms have been proposed as well. For example, power constrained RA algorithms proposed in [NCC⁺07, KP07, KC07] and [CLW09] consider respectively the maximization of the sum rate and the weighted sum goodput, whereas the one in [SAR⁺10] aims at maximizing a metric depending on the rates and queue lengths of the source and relays. Using those algorithms, each destination may decode the source transmitted signals, each at a distinct subcarrier in the broadcasting and relaying (except for [CLW09]) slots, as well as the relays transmitted signals, each at a distinct subcarrier from a single relay in the relaying slot. Note that when the source transmits signals to relays in the broadcasting slot, every destination discards the received signal replicas and thus spatial diversity is not exploited. In [NY07], a RA algorithm is proposed to maximize the sum utility of multiple uplink/downlink data streams aided by a single destination adopting selection relaying AF or DF.

So far, the majority of the proposed RA algorithms, as the aforementioned ones, restrict that at most one relay can assist the source for every relay-aided transmission. When there are multiple relays available, allowing not just one but *each* of them to be eligible for assisting at every subcarrier, can better ex-

exploit the degrees of freedom in the system for performance improvement as said earlier in Chapter 7.

In this chapter, the WSR maximized RA problem is addressed under a system sum power constraint for an OFDMA downlink transmission system aided by multiple DF relays. In particular, multiple relays may cooperate with the source for every relay-aided transmission. A two-step algorithm is proposed to find the globally optimum RA based on a divide-and-conquer strategy. In particular, it is shown that the optimum RA in the second step can be easily derived when the system sum power is very high.

The rest of this chapter is organized as follows. First, the considered system and the RA problem are introduced. Next, the two-step RA algorithm is proposed, and the optimum RA for the second step is derived when the system sum power is very high. Finally, the effectiveness of the proposed algorithm is illustrated by numerical experiments.

8.2 System description and RA problem

Consider an OFDMA downlink transmission system from a source to U destinations aided by K DF relays collected in the set $\Psi = \{r_i | i = 1, \dots, K\}$. All links are assumed to be frequency selective, and OFDM with sufficiently long cyclic prefix is used to transform every link into N parallel channels, each at a different carrier facing flat fading. At each carrier, the transmission of a symbol is in either the direct mode, or the relay-aided mode spanning across two equal-duration time slots, namely the broadcasting slot and the relaying slot. Due to the OFDMA, each carrier is allocated to one destination exclusively.

The following assumptions are made for the considered system. First, assume the carrier frequency and symbol timing of the source are perfectly synchronized with those of the relays, e.g. with the techniques in [MKP07]. Second, every channel in the system remains unchanged within a sufficiently long duration, over which the RA algorithm can be implemented at a central controller knowing precisely the CSI of the system. Furthermore, the RA information can be reliably disseminated to the source, every relay, and every destination.

The transmission of a unit-variance symbol s at a carrier to a destination, say at carrier n to destination u , is exactly the same as that considered in Chap-

ter 7. Let's first describe the transmission in the relay-aided mode. The coefficient of the channel between any two of the source, r_i , and destination u , are notated according to Table 8.1. Suppose a set of relays, collected in the set $\mathcal{R}_n(u)$, are selected by the RA algorithm to assist relaying. Using the transmit power $P_{s,u}(n)$, the source first emits the symbol $\sqrt{P_{s,u}(n)}s$, while each relay does not transmit anything in the broadcasting slot. At the end of this slot, both destination u and every relay receive the source signal. The signal samples at destination u and r_i can be expressed by

$$y_{u,b}(n) = \sqrt{P_{s,u}(n)}\lambda_{s,u}(n)s + \eta_{u,b}(n) \quad (8.1)$$

and

$$y_{r_i}(n) = \sqrt{P_{s,u}(n)}h_{s,r_i}(n)s + n_{r_i}(n), \quad (8.2)$$

respectively, where $\eta_{u,b}(n)$ and $\eta_{r_i,n}$ represent the corruption of the additive white Gaussian noise (AWGN) at destination u and r_i , respectively.

After decoding and then reencoding s , all relays in $\mathcal{R}_n(u)$ transmit simultaneously to destination u in the relaying slot, which in effect establishes a distributed MISO transmission link. Specifically, $r_i \in \mathcal{R}_n(u)$ transmits $\alpha_{r_i}s$, where α_{r_i} is the complex weight for transmit beamforming and satisfies $|\alpha_{r_i}|^2 = P_{r_i,u}(n)$, with $P_{r_i,u}(n)$ denoting the transmit power allocated to r_i . To have the relays' signals add coherently when received at destination u , $\alpha_{r_i} = \sqrt{P_{r_i,u}(n)}e^{-j\arg(\lambda_{r_i,u}(n))}$ is used, where $\arg(\lambda_{r_i,u}(n))$ stands for the phase of $\lambda_{r_i,u}(n)$. Note that the above described relay-aided transmission enables a flexible use of all relays opportunistically through a general form of adaptive transmit beamforming, in that $\mathcal{R}_n(u)$ and $\{P_{r_i,u}(n)|r_i \in \mathcal{R}_n(u)\}$ are determined dynamically by the RA algorithm based on the CSI, as will be developed later. At the end of the relaying slot, the signal sample at destination u is denoted by

$$y_{u,r}(n) = \sum_{r_i \in \mathcal{R}_n(u)} \sqrt{P_{r_i,u}(n)}|\lambda_{r_i,u}(n)|s + \eta_{u,r}(n), \quad (8.3)$$

Table 8.1 Channel coefficients at carrier n .

source to destination u	source to r_i	r_i to destination u
$\lambda_{s,u}(n)$	$h_{s,r_i}(n)$	$\lambda_{r_i,u}(n)$

where $\eta_{u,r}(n)$ represents the corruption of AWGN at destination u .

Finally, destination u combines $y_{u,b}(n)$ and $y_{u,r}(n)$ based on the maximum ratio combining (MRC), and the output is denoted by

$$y_{u,c}(n) = \sqrt{P_{s,u}(n)}(\lambda_{s,u}(n))^* y_{u,b}(n) + \left(\sum_{r_i \in \mathcal{R}_n(u)} \sqrt{P_{r_i,u}(n)} |\lambda_{r_i,u}(n)| \right)^* y_{u,r}(n). \quad (8.4)$$

Assume $\{\eta_{r_i,n} | r_i \in \mathcal{R}_n(u)\}$, $\eta_{u,b}(n)$, and $\eta_{u,r}(n)$ are independent zero-mean circular Gaussian random variables with the same variance σ^2 . By some mathematical arrangements, the SNR associated with decoding s from $y_{r_i,n}$ at $r_i \in \mathcal{R}_n(u)$ in the broadcasting slot is

$$\gamma_{r_i,n} = P_{s,u}(n) G_{s,r_i}(n), \quad (8.5)$$

and the SNR associated with decoding s from $y_{u,c}(n)$ at destination u is

$$\gamma_{u,c}(n) = P_{s,u}(n) G_{s,u}(n) + \left(\sum_{r_i \in \mathcal{R}_n(u)} \sqrt{P_{r_i,u}(n) G_{r_i,u}(n)} \right)^2, \quad (8.6)$$

where $G_{s,r_i}(n) = \frac{|h_{s,r_i}(n)|^2}{\sigma^2}$, $G_{s,u}(n) = \frac{|\lambda_{s,u}(n)|^2}{\sigma^2}$, and $G_{r_i,u}(n) = \frac{|\lambda_{r_i,u}(n)|^2}{\sigma^2}$ represent the noise power normalized channel gains from the source to r_i , from the source to destination u , and from r_i to destination u , respectively.

To guarantee reliable decoding at destination u and every relay in $\mathcal{R}_n(u)$, the maximum achievable rate at carrier n is

$$R_{u,n,1} = \min \left\{ \min_{r_i \in \mathcal{R}_n(u)} \{\log(1 + \gamma_{r_i,n})\}, \log(1 + \gamma_{u,c}(n)) \right\} = \log(1 + \gamma_u(n)) \quad (8.7)$$

in nats/two-slots, where

$$\gamma_u(n) = \min\{\gamma_{u,c}(n), \gamma_r(n)\} \quad (8.8)$$

and

$$\gamma_r(n) = P_{s,u}(n) \cdot \min_{r_i \in \mathcal{R}_n(u)} G_{s,r_i}(n). \quad (8.9)$$

As for the direct transmission, the source emits two independent symbols with transmit powers $P_{s_1,u}(n)$ and $P_{s_2,u}(n)$ in the two slots, respectively, and only destination u decodes the transmitted symbols from the two received signal samples. Assume the AWGN corruption associated with the two received

samples are independent zero-mean circular Gaussian distributed with variance σ^2 . By simple mathematical arrangements, the sum achievable rate to destination u in nats/two-slots can be evaluated as

$$R_{u,n,2} = \log(1 + P_{s_1,u}(n)G_{s,u}(n)) + \ln(1 + P_{s_2,u}(n)G_{s,u}(n)). \quad (8.10)$$

The RA problem under consideration is to maximize the WSR of all destinations by optimizing the optimum assignment of destination, transmission mode, and source power for each carrier, as well as the optimum assisting relays and their transmit powers for every relay-aided carrier, when the considered system consumes a sum power no greater than P_t .

8.3 The two-step RA algorithm

A two-step algorithm is proposed to solve the RA problem based on a divide-and-conquer strategy. An overview of this algorithm is first given. Then, the algorithm is developed.

8.3.1 An overview of the two-step algorithm

In the first step, the optimum $\mathcal{R}_n(u)$, $P_{s,u}(n)$, and $\{P_{r_i,u}(n) | \forall r_i \in \mathcal{R}_n(u)\}$ that maximize $R_{u,n,1}$, and the optimum $P_{s_1,u}(n)$ and $P_{s_2,u}(n)$ that maximize $R_{u,n,2}$, are found for every combination of carrier n and destination u , assuming that a sum power P is used by the transmission at carrier n to destination u in the relay-aided mode and the direct mode, respectively. Such maximized $R_{u,n,1}$ and $R_{u,n,2}$ are denoted by $R_{u,n,1}(P)$ and $R_{u,n,2}(P)$, respectively. The associated algorithm will be developed in Section 8.3.2.

To simplify the RA in the second step, two user sets, namely $\mathcal{U}_D(n) = \{u | \forall P \in [0, P_t], R_{u,n,2}(P) \geq R_{u,n,1}(P)\}$ and $\mathcal{U}_R(n) = \{u | \forall P \in [0, P_t], R_{u,n,1}(P) > R_{u,n,2}(P)\}$, are found for every carrier n . Once a destination $u \in \mathcal{U}_D(n)$ (resp. $u \in \mathcal{U}_R(n)$) is allocated with carrier n , the direct mode (resp. the relay-aided mode) should always be used to maximize the WSR, since it results in a rate no smaller than the relay-aided mode (resp. the direct mode), independently of the sum power allocated to this carrier.

To formulate the maximum WSR with the $R_{u,n,1}(P)$ and $R_{u,n,2}(P)$ derived in the first step, define for every destination u in $\mathcal{U}_R(n)$ (resp. $\mathcal{U}_D(n)$) a binary variable $t_{u,n,1}$ (resp. $t_{u,n,2}$) and a nonnegative variable $P_{u,n,1}$ (resp. $P_{u,n,2}$), where $t_{u,n,1} = 1$ (resp. $t_{u,n,2} = 1$) indicates that carrier n in the relay-aided

(resp. direct) mode is allocated to destination u , and $P_{u,n,1}$ (resp. $P_{u,n,2}$) represents the corresponding sum power allocated to carrier n . For every destination $u \notin \mathcal{U}_D(n) \cup \mathcal{U}_R(n)$, $t_{u,n,1}$, $t_{u,n,2}$, $P_{u,n,1}$, and $P_{u,n,2}$ are defined with the same interpretations as explained above. Note that $t_{u,n,1}$ and $P_{u,n,1}$ are defined for every destination $u \notin \mathcal{U}_D(n)$, and $t_{u,n,2}$ and $P_{u,n,2}$ are defined for every destination $u \notin \mathcal{U}_R(n)$. Now, the maximum WSR can be expressed by

$$f = \sum_n \left(\sum_{u \notin \mathcal{U}_D(n)} w_u t_{u,n,1} R_{u,n,1}(P_{u,n,1}) + \sum_{u \notin \mathcal{U}_R(n)} w_u t_{u,n,2} R_{u,n,2}(P_{u,n,2}) \right) \quad (8.11)$$

where $w_u > 0$ satisfying $\sum_{u=1}^U w_u = 1$ is the weight assigned by system designers to the rate of destination u . In particular, increasing w_u leads to a higher priority given to destination u .

In the second step, for every carrier n the optimum $\{t_{u,n,1}, P_{u,n,1} | \forall u \notin \mathcal{U}_D(n)\}$ and $\{t_{u,n,2}, P_{u,n,2} | \forall u \notin \mathcal{U}_R(n)\}$ that maximize f are found subject to the system sum power constraint

$$\sum_n \left(\sum_{u \notin \mathcal{U}_D(n)} t_{u,n,1} P_{u,n,1} + \sum_{u \notin \mathcal{U}_R(n)} t_{u,n,2} P_{u,n,2} \right) \leq P_t, \quad (8.12)$$

as well as the constraint

$$\sum_{u \notin \mathcal{U}_D(n)} t_{u,n,1} + \sum_{u \notin \mathcal{U}_R(n)} t_{u,n,2} \leq 1, \forall n, \quad (8.13)$$

due to the OFDMA. It is important to note that there exist default constraints that $\forall n, u$, $\mathcal{R}_n(u)$ should be a subset of Ψ , and every power (resp. indicator) variable should be non-negative (resp. binary). This problem, consisting of both binary and continuous optimization variables, is not convex since the feasible set for the binary variables is not convex. In Section 8.3.3, an algorithm is developed to find the globally optimum solution. Combining the RAs found in the two steps, the globally optimum RA for maximizing the WSR can be found.

8.3.2 RA algorithm in the first step

First, let's consider the maximization of $R_{u,n,2}$ under the constraint $P_{s1,u}(n) + P_{s2,u}(n) = P$. According to the Jensen's inequality, the maximum $R_{u,n,2}$ in this case can be easily found as

$$R_{u,n,2}(P) = 2 \log \left(1 + G_{s,u}(n) \frac{P}{2} \right), \quad (8.14)$$

and the optimum RA is to allocate the sum power P equally to $P_{s_1,u}(n)$ and $P_{s_2,u}(n)$.

Next, consider the maximization of $R_{u,n,1}$ with respect to $\mathcal{R}_n(u)$, $P_{s,u}(n)$, and $\{P_{r_i,u}(n) | \forall r_i \in \mathcal{R}_n(u)\}$ constrained by

$$P_{s,u}(n) + \sum_{r_i \in \mathcal{R}_n(u)} P_{r_i,u}(n) = P. \quad (8.15)$$

To facilitate derivation, define $G_{\mathcal{R}_n(u),u}(n) = \sum_{r_i \in \mathcal{R}_n(u)} G_{r_i,u}(n)$, which represents the sum of the channel gains from all assisting relays to destination u . Ψ_n is defined as the set incorporating all relays sorted in the increasing order of $G_{s,r_i}(n)$, and the i -th relay in Ψ_n is denoted by $r_n(i)$. In particular, $G_{s,r_n(N)}(n) = \max_{r_i \in \Psi} G_{s,r_i}(n)$. Let's define $\Psi_n(i)$ as the set containing all relays in Ψ_n with indices from i up to N . When $G_{s,u}(n) < G_{s,r_n(N)}(n)$, $x_{u,n}$ is defined as the smallest i satisfying $G_{s,u}(n) < G_{s,r_n(i)}(n)$.

By means of intuitive figure illustrations, the procedure of maximizing $R_{u,n,1}$ is derived and put in Section 8.6 for clarity. In particular, it is found that

$$R_{u,n,1}(P) = \log(1 + G_{u,1}(n)P), \quad (8.16)$$

where $G_{u,1}(n)$ can be evaluated by one of the following formulas:

1. when $G_{s,u}(n) \geq G_{s,r_n(N)}(n)$, $G_{u,1}(n) = G_{s,r_n(N)}(n)$. In this case, $\forall P \in [0, P_t]$, $R_{u,n,2}(P) \geq R_{u,n,1}(P)$ holds.
2. when $G_{\Psi_n(x_{u,n}),u}(n) \leq G_{s,u}(n) < G_{s,r_n(N)}(n)$, $G_{u,1}(n) = G_{s,u}(n)$. In this case, $\forall P \in [0, P_t]$, $R_{u,n,2}(P) \geq R_{u,n,1}(P)$ holds.
3. when $G_{s,u}(n) < G_{s,r_n(N)}(n)$ and $G_{s,u}(n) < G_{\Psi_n(x_{u,n}),u}(n)$,

$$G_{u,1}(n) = \max_{x_{u,n} \leq b \leq y_{u,n}} \frac{G_{s,r_n(b)}(n)G_{\Psi_n(b),u}(n)}{G_{\Psi_n(b),u}(n) + G_{s,r_n(b)}(n) - G_{s,u}(n)}, \quad (8.17)$$

where $y_{u,n}$ is the greatest i satisfying $G_{\Psi_n(i),u}(n) > G_{s,u}(n)$. In this case, $G_{u,1}(n) > G_{s,u}(n)$ always holds as shown in Appendix A. Suppose $b = z_{u,n}$ is the maximizer for the right-hand side of (8.17), the optimum $\mathcal{R}_n(u)$ is $\Psi_n(z_{u,n})$. The optimum $P_{s,u}(n)$ is computed by (8.46) with $b = z_{u,n}$, and the optimum $P_{r_i,u}(n)$, $\forall r_i \in \Psi_n(z_{u,n})$ is

$$P_{r_i,u}(n) = \frac{(P - P_{s,u}(n))G_{r_i,u}(n)}{G_{\Psi_n(z_{u,n}),u}(n)}. \quad (8.18)$$

The above analysis reveals the fact that, the relay-aided mode with the optimum RA in effect transforms all channels at carrier n into a source to destination u channel with normalized gain $G_{u,1}(n)$ and half of the system bandwidth. As a matter of fact, only one symbol can be sent during the two slots, while the system bandwidth can actually support sending two independent symbols per two slots. When $G_{u,1}(n) \leq G_{s,u}(n)$, the direct mode should always be used. When $G_{u,1}(n) > G_{s,u}(n)$, using the relay-aided mode leads to an increased channel gain but sacrificing half of the bandwidth, compared to using the direct mode. In this case, $R_{u,n,1}(P) \geq R_{u,n,2}(P)$ holds if $P \leq \frac{4(G_{u,1}(k) - G_{s,u}(n))}{(G_{s,u}(n))^2}$, otherwise $R_{u,n,2}(P) > R_{u,n,1}(P)$, which means that the relay-aided and direct modes should be used for the low and high power regimes, respectively. The interpretation is that, in the low power regime, it is more beneficial to increase the received power, while in the high power regime, it is better to increase the number of channel uses per time unit [TV05]. Based on the above analysis, $\mathcal{U}_D(n)$ and $\mathcal{U}_R(n)$ for every carrier n can be expressed as

$$\mathcal{U}_D(n) = \{u | G_{u,1}(n) \leq G_{s,u}(n)\} \quad (8.19)$$

and

$$\mathcal{U}_R(n) = \{u | G_{u,1}(n) > G_{s,u}(n), P_t \leq \frac{4(G_{u,1}(k) - G_{s,u}(n))}{(G_{s,u}(n))^2}\}. \quad (8.20)$$

8.3.3 RA algorithm in the second step

An algorithm is proposed to solve the RA problem in the second step based on the following strategy. First, for every carrier n , $t_{u,n,1}$ and $t_{u,n,2}$ are relaxed to be real variables between 0 and 1. Now, $t_{u,n,1}$ (resp. $t_{u,n,2}$) can be interpreted as the fraction of the whole duration allocated to transmitting at carrier n to destination u in the relay-aided (resp. direct) mode, and $\eta_{u,n,1} = t_{u,n,1}R_{u,n,1}(P_{u,n,1})$ (resp. $\eta_{u,n,2} = t_{u,n,2}R_{u,n,2}(P_{u,n,2})$) can be interpreted as the rate averaged over the whole duration for transmitting to destination u at carrier n in the relay-aided (resp. direct) mode. Then, the relaxed RA problem can be transformed with change of variables into a convex optimization problem, and its globally optimum RA can be found as will be shown later. Most interestingly, it will be shown that every optimum $t_{u,n,1}$ and $t_{u,n,2}$ for the relaxed problem are still equal to either 0 or 1, which means that the globally optimum RA to the relaxed problem is also globally optimum to the original RA problem.

To make the change of variables, $P_{u,n,1}$ is substituted with $\frac{\rho_{u,n,1}}{t_{u,n,1}}$ for every $u \notin \mathcal{U}_D(n)$, and $P_{u,n,2}$ with $\frac{\rho_{u,n,2}}{t_{u,n,2}}$ for every $u \notin \mathcal{U}_R(n)$. Now, the sum power constraint, $\eta_{u,n,1}$, and $\eta_{u,n,2}$ are respectively expressed as

$$\sum_n \left(\sum_{u \notin \mathcal{U}_D(n)} \rho_{u,n,1} + \sum_{u \notin \mathcal{U}_R(n)} \rho_{u,n,2} \right) \leq P_t, \quad (8.21)$$

$$\eta_{u,n,1} = t_{u,n,1} \ln \left(1 + G_{u,1}(k) \frac{\rho_{u,n,1}}{t_{u,n,1}} \right), \quad (8.22)$$

$$\eta_{u,n,2} = 2 t_{u,n,2} \ln \left(1 + \frac{G_{s,u}(n)}{2} \frac{\rho_{u,n,2}}{t_{u,n,2}} \right). \quad (8.23)$$

Note that $\eta_{u,n,1}$ (resp. $\eta_{u,n,2}$) is a concave function of $t_{u,n,1} > 0$ and $\rho_{u,n,1}$ (resp. $t_{u,n,2} > 0$ and $\rho_{u,n,2}$), since it is a perspective of the concave function $R_{u,n,1}(P_{u,n,1})$ (resp. $R_{u,n,2}(P_{u,n,2})$). One delicate issue is that after the change of variables, $\eta_{u,n,1}$ (resp. $\eta_{u,n,2}$) can not be evaluated at $t_{u,n,1} = 0$ (resp. $t_{u,n,2} = 0$). To address this issue, the domain of $\eta_{u,n,1}$ is expanded to incorporate $t_{u,n,1} = 0$, and define $\eta_{u,n,1} = 0$ when $t_{u,n,1} = 0$. After the expansion and definition, $\eta_{u,n,1}$ is still a concave and continuous function of $t_{u,n,1} \geq 0$, because $\lim_{t_{u,n,1} \rightarrow 0} \eta_{u,n,1} = 0$. Motivated by the same consideration, the domain of $\eta_{u,n,2}$ is expanded to incorporate $t_{u,n,2} = 0$, and define $\eta_{u,n,2} = 0$ when $t_{u,n,2} = 0$, so as to ensure $\eta_{u,n,2}$ is a concave and continuous function of $t_{u,n,2} \geq 0$.

Based on the above analysis, the relaxed RA problem with new variables, i.e., maximizing f subject to the constraints (8.13) and (8.21), is a convex optimization problem. Obviously, this problem satisfies the Slater constraints qualification, i.e. there exists at least one feasible solution satisfying all inequality constraints strictly. This justifies the zero duality gap for the problem [Ber03]. Therefore, the dual method introduced in Chapter 3 can be used to solve this problem for a globally optimum RA. Specifically, dual variables are defined for certain constraints, then an iterative algorithm consisting of an inner loop and an outer loop is implemented. In the inner loop, the Lagrangian maximization problem (LMP), which is to find the optimum RA variables that maximize the Lagrangian, is solved with the dual variables fixed. In the outer loop, the optimum dual variables are found by iteratively updating the dual variables with the subgradient method, until the KKT conditions are satisfied. The optimum RA variables for the LMP given the optimum dual variables, are the globally optimum solution for the relaxed RA problem.

To use the dual method, a nonnegative dual variable μ is introduced for the system sum power constraint (8.21), and the associated LMP is

$$\begin{aligned} \max \quad & L(\mu) = f + \mu (P_t - P_x) \\ \text{s.t.} \quad & (8.13), \\ & t_{u,n,1} \geq 0, \rho_{u,n,1} \geq 0, \forall n, \forall u \notin \mathcal{U}_D(n), \\ & t_{u,n,2} \geq 0, \rho_{u,n,2} \geq 0, \forall n, \forall u \notin \mathcal{U}_R(n), \end{aligned} \quad (8.24)$$

where $L(\mu)$ represents the Lagrangian given μ , and P_x is the left-hand side of (8.21). The optimum dual variable denoted by μ^* , must satisfy the KKT conditions consisting of (a) $\mu^* \geq 0$, (b) $\mu^*(P_t - P_x(\mu^*)) = 0$, and (c) $P_x(\mu^*) \leq P_t$, where $P_x(\mu)$ represents the P_x evaluated with the optimum power variables to (8.24) given μ [BV04, Ber03]. It is important to note that μ^* must be strictly positive, which can be justified as follows. Suppose $\mu^* = 0$. In this case, the maximum $L(\mu^*)$ subject to the constraints of (8.24) is infinity, since f , as a function of $\eta_{u,n,1}$ and $\eta_{u,n,2}$, is monotonically increasing with $\rho_{u,n,1}$ and $\rho_{u,n,2}$, which can be increased unboundedly while remaining feasible to (8.24). This means that $P_x(\mu^*)$ is infinity, which contradicts (c) of the KKT conditions. Therefore, $\mu^* \neq 0$, and the KKT conditions are reduced to $\mu > 0$ and $P_t = P_x(\mu^*)$.

Let's now find the optimum RA variables to (8.24) given $\mu > 0$. To this end, let's find for every carrier n , $\forall u \notin \mathcal{U}_D(n)$, the optimum $\rho_{u,n,1}$ given $t_{u,n,1}$, and $\forall u \notin \mathcal{U}_R(n)$, the optimum $\rho_{u,n,2}$ given $t_{u,n,2}$, to maximize $L(\mu)$. After mathematical arrangements, it can be shown that these optimum values are

$$\rho_{u,n,i} = \begin{cases} t_{u,n,1} \left[\frac{w_u}{\mu} - \frac{1}{G_{u,1}(n)} \right]^+ & \text{if } i = 1, \\ 2t_{u,n,2} \left[\frac{w_u}{\mu} - \frac{1}{G_{s,u}(n)} \right]^+ & \text{if } i = 2, \end{cases} \quad (8.25)$$

where $[x]_0 = \max\{x, 0\}$. Using (8.25), $L(\mu)$ is reduced to

$$L(\mu) = \sum_n L_n(\mu) + \mu P_t, \quad (8.26)$$

where

$$L_n(\mu) = \sum_{u \notin \mathcal{U}_D(n)} t_{u,n,1} X_n(u) + \sum_{u \notin \mathcal{U}_R(n)} t_{u,n,2} Y_n(u), \quad (8.27)$$

and

$$X_n(u) = w_u \log \left(1 + G_{u,1}(n) \left[\frac{w_u}{\mu} - \frac{1}{G_{u,1}(n)} \right]^+ \right) - \mu \left[\frac{w_u}{\mu} - \frac{1}{G_{u,1}(n)} \right]^+, \quad (8.28)$$

$$Y_n(u) = 2w_u \log \left(1 + G_{s,u}(n) \left[\frac{w_u}{\mu} - \frac{1}{G_{s,u}(n)} \right]^+ \right) - \mu \left[\frac{2w_u}{\mu} - \frac{2}{G_{s,u}(n)} \right]^+. \quad (8.29)$$

Now, the problem of finding the optimum $\{t_{u,n,1} | \forall u \notin \mathcal{U}_D(n), \forall n\}$ and $\{t_{u,n,2} | \forall u \notin \mathcal{U}_R(n), \forall n\}$ for (8.24) can be decomposed into K subproblems, the n -th of which is to find the optimum $\{t_{u,n,1} | \forall u \notin \mathcal{U}_D(n)\}$ and $\{t_{u,n,2} | \forall u \notin \mathcal{U}_R(n)\}$ for maximizing $L_n(\mu)$ subject to the constraints in (8.24). It can be readily shown that the optimum $\{t_{u,n,1} | \forall u \notin \mathcal{U}_D(n)\} \cup \{t_{u,n,2} | \forall u \notin \mathcal{U}_R(n)\}$ has all entries equal to 0, except for one entry t_{u_n, n, i_n} equal to 1. Obviously, (u_n, i_n) is the (u, i) with the maximum metric $m_{u,i}(n)$, expressed as

$$m_{u,i}(n) = \begin{cases} X_n(u) & \text{if } i = 1, \\ Y_n(u) & \text{if } i = 2. \end{cases} \quad (8.30)$$

Note that when computing $m_{u,i}(n)$, i can only be 1 (resp. 2) if $u \in \mathcal{U}_R(n)$ (resp. $u \in \mathcal{U}_D(n)$), while i may be either 1 or 2 if $u \notin \mathcal{U}_R(n) \cup \mathcal{U}_D(n)$. There might exist multiple combinations of (u, i) corresponding to the maximum $m_{u,i}(n)$. In this case, any combination of them can be chosen to be (u_n, i_n) . According to (8.25), the optimum power variables related to carrier n are all equal to zero, except for P_{u_n, n, i_n} computed with

$$P_{u_n, n, i_n} = \begin{cases} \left[\frac{w_{u_n}}{\mu} - \frac{1}{G_{u_n,1}(k)} \right]^+ & \text{if } i_n = 1, \\ 2 \left[\frac{w_{u_n}}{\mu} - \frac{1}{G_{s,u_n}(k)} \right]^+ & \text{if } i_n = 2. \end{cases} \quad (8.31)$$

The procedure of finding the optimum RA variables to (8.24) is summarized in **Algorithm 12**. Next, let's address the problem of finding μ^* in the outer loop iteratively. To speed up the convergence rate, μ is first initialized by a value close to μ^* . To this end, define

$$P_L(\mu) = \sum_n \left[\frac{\min_u \{w_u\}}{\mu} - M_U(n) \right]^+ \quad (8.32)$$

$$P_U(\mu) = \sum_n \left[\frac{2 \max_u \{w_u\}}{\mu} - M_L(n) \right]^+, \quad (8.33)$$

Algorithm 12 Procedure to solve (8.24)

```

for  $n = 1$  to  $N$  do
   $\forall u$  and the associated  $i$ , compute  $m_{u,i}(n)$  defined in (8.30);
   $\{u_n, i_n\} = \arg \max_{u,i} m_{u,i}(n)$ ;
   $t_{u_n, n, i_n} = 1$  and compute  $P_{u_n, n, i_n}$  by (8.31);
   $P_{u, n, i} = 0$  and  $t_{u, n, i} = 0$  if either  $u \neq u_n$  or  $i \neq i_n$ .
end for

```

Algorithm 13 The RA algorithm in the second step

```

Find  $\mu_U$  satisfying  $P_U(\mu_U) = P_t$ ;
Find  $\mu_L$  satisfying  $P_L(\mu_L) = P_t$ ;
For each  $\mu$  in  $S = \{\mu_L + \frac{n}{N_s}(\mu_U - \mu_L), n = 0, 1, \dots, N_s - 1\}$ , use Algorithm 12 to
find the optimum RA variables to (8.24), and compute  $P_x(\mu)$ ;
Initialize  $\mu$  with the  $\mu \in S$  satisfying  $P_x(\mu) \leq P_t$  and having the minimum  $P_t - P_x(\mu)$ ;

repeat
  Update  $\mu$  with  $[\mu - \delta(P_t - P_x(\mu))]_0$ ;
  Use Algorithm 12 to find the optimum solution for (8.24);
  Compute  $P_x(\mu)$ ;
until  $\mu > 0$  and  $0 \leq P_t - P_x(\mu) < \epsilon$ 

The optimum solution for (8.24) found in the last iteration corresponds to the optimum sum power and mode for each carrier in the second step.

```

where

$$M_U(n) = \max \left\{ \max_{u \notin \mathcal{U}_D(n)} \frac{1}{G_{u,1}(n)}, \max_{u \notin \mathcal{U}_R(n)} \frac{2}{G_{s,u}(n)} \right\} \quad (8.34)$$

$$M_L(n) = \min \left\{ \min_{u \notin \mathcal{U}_D(n)} \frac{1}{G_{u,1}(n)}, \min_{u \notin \mathcal{U}_R(n)} \frac{2}{G_{s,u}(n)} \right\}. \quad (8.35)$$

It can be readily shown that $P_L(\mu^*) \leq P_x(\mu^*) \leq P_U(\mu^*)$ always holds. Suppose μ_U and μ_L satisfy $P_U(\mu_U) = P_t$ and $P_L(\mu_L) = P_t$, respectively. Since $P_U(\mu_U) = P_t = P_x(\mu^*) \leq P_U(\mu^*)$ and $P_U(\mu)$ is a non-increasing function of μ , $\mu_U \geq \mu^*$. Similarly, it can be shown that $\mu^* \geq \mu_L$, thus μ^* is confined in the interval $[\mu_L, \mu_U]$. To initialize μ with a value close to μ^* , this interval is sampled uniformly at N_s points to build the set $S = \{\mu_L + \frac{n}{N_s}(\mu_U - \mu_L), n = 0, 1, \dots, N_s - 1\}$. Given each μ in S , (8.24) is solved and $P_x(\mu)$ computed, then μ is initialized with the $\mu \in S$ satisfying $P_x(\mu) \leq P_t$ and having the minimum $P_t - P_x(\mu)$.

After initialization, μ is iteratively updated based on the subgradient method, i.e., μ is updated with $[\mu - \delta(P_t - P_x(\mu))]_0$ where δ is a sufficiently small value to guarantee convergence, until μ^* is found when $P_t = P_x(\mu)$ and $\mu > 0$. The RA algorithm in the second step is summarized in **Algorithm 13**. Due to numerical issues, let's regard the KKT conditions to be satisfied when $\mu > 0$ and $0 \leq P_t - P_x(\mu) \leq \epsilon$, where ϵ is a very small positive value.

It is interesting to examine the complexity of the proposed two-step algorithm. For the first-step algorithm, the average number of operations for computing the $G_{u,1}(n)$ for a given combination of u and n is proportional to the number of relays, i.e., K . Since there are U destinations and N carriers, the complexity of the first step algorithm is $O(KUN)$. For the second-step algorithm, suppose T represents the average number of iterations for updating μ . For each iteration, the number of operations is proportional to N and U . thus the complexity of the second step algorithm is $O(TUN)$. From the above analysis, the complexity of the proposed algorithm is $O((K + T)UN)$.

8.4 The optimum RA in the second step for a special case

We will show the optimum RA in the second step can be easily derived when P_t is very high so that the associated μ_U satisfies

$$\forall n, \frac{\min_u w_u}{\mu_U} \gg \max_u \left\{ \max \left\{ \frac{1}{G_{u,1}(n)}, \frac{1}{G_{s,u}(n)} \right\} \right\}, \quad (8.36)$$

$$\forall n, \frac{\min_u w_u}{\mu_U} \gg \max_u \{ \max \{ G_{u,1}(n), G_{s,u}(n) \} \}. \quad (8.37)$$

Since (8.36) holds and $\mu^* \leq \mu_U$, the optimum sum power P_n for carrier n satisfies

$$\forall n, P_n \approx \begin{cases} \frac{w_{u_n}}{\mu^*} & \text{if } i_n = 1 \\ \frac{2w_{u_n}}{\mu^*} & \text{if } i_n = 2, \end{cases} \quad (8.38)$$

according to (8.31), where (u_n, i_n) corresponds to the optimum destination and mode for carrier n found by the RA algorithm in the second step. From (8.36), (8.37), and (8.38), the conditions

$$\forall n, u, P_n G_{s,u}(n) \gg 1, P_n \gg G_{s,u}(n), \quad (8.39)$$

$$\forall n, u, P_n G_{u,1}(n) \gg 1, P_n \gg G_{u,1}(n), \quad (8.40)$$

hold. This means that in the high power regime, even though the optimum sum power for each subcarrier is not precisely known, it is for sure that $\forall n, u$, the following approximations hold:

$$w_u R_{u,n,2}(P_n) \approx 2w_u \left(\log(P_n) + \ln \frac{G_{s,u}(n)}{2} \right) \approx 2w_u \log(P_n), \quad (8.41)$$

$$w_u R_{u,n,1}(P_n) \approx w_u \left(\log(P_n) + \ln G_{u,1}(n) \right) \approx w_u \log(P_n). \quad (8.42)$$

By comparing (8.41) and (8.42), it can easily be seen that the direct mode for each carrier n is the optimum for maximizing the WSR, independently of the destination assigned. All this confirms the former interpretation that, for the high power regime, increasing the number of channel uses per time unit is more beneficial than increasing the received power. If equal weights are used for all destinations, the optimum destination for carrier n is the one with the maximum $G_{s,u}(n)$. Otherwise, the optimum destination for all subcarriers is the same and given by the one with the maximum weight w_u . In either case, the optimum sum power to carrier n is $P_n \approx \frac{P_t}{N}$ in order to satisfy the KKT conditions according to (8.38).

8.5 Numerical experiments

In numerical experiments, consider a multiple DF relays aided OFDMA downlink transmission system with $N = 4$ relays. The source is allocated at the origin, and the relays r_1, r_2, r_3 , and r_4 are located at coordinates $(-15, -5)$, $(-5, -5)$, $(5, -5)$, and $(15, -5)$, respectively. Note that all the aforementioned coordinate-related values have the unit of meter. The random channel between every two of the source, the relays, and the destinations is generated in the same way as explained in Section 5.6. The following experiments are conducted with Matlab v7.1 on a laptop equipped with an Intel Duo CPU of 2.2 GHz and a memory of 2 GBytes.

To illustrate the effectiveness of the optimum RA found for the proposed protocol, it is compared to the optimum RA for a reference protocol. Both protocols are the same except that with the reference protocol, at every subcarrier in the direct mode the source only transmits in the broadcasting slot (and remains idle during the relaying slot). Interestingly, when the destinations use *equal* weights, the optimum RA for the reference protocol can easily be derived on a per subcarrier basis. For carrier n , the optimum destination u_n is the one with the maximum $G_u(k)$ where $G_u(k) = \max\{G_{u,1}(n), G_{s,u}(n)\}$, i.e., the relay-aided mode is optimum only if $G_{u_n,1}(k) > G_{s,u_n}(k)$. The optimum per subcarrier sum power can be calculated according to water filling of the system sum power P_t to K parallel channels with normalized gain $G_{u_n}(k)$ for the n -th channel. It is important to note that for the reference protocol, the above RA might *not* be optimum when destinations use *unequal* weights. In this case, the RA needs to be optimized jointly over all subcarriers with the same two-step optimization strategy as proposed in Section 8.3, which is not developed here. On the contrary, the RA for the proposed protocol *always* needs to be jointly optimized over all subcarriers.

8.5.1 Results for a random realization of channels

For clarity of presentation, assume $N = 32$ subcarriers are used, $\sigma^2 = -30$ dBm, $U = 4$ destinations exist with equal weights, $N_s = 100$, $\delta = 10^{-3}(\mu_U - \mu_L)$, and $\epsilon = 0.1$. For the randomly generated destination coordinates shown in Figure 8.1, a random realization of all channels has been generated, for which the optimum RAs for the proposed and reference protocols are evaluated. For every destination u , the $G_{u,1}(n)$ computed by the proposed RA

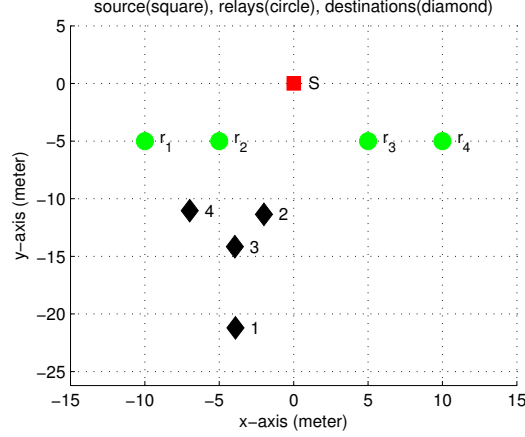


Figure 8.1 The OFDMA system considered in numerical experiments.

algorithm in the first step, and $G_{s,u}(n)$ are shown in Figure 8.2 for every carrier n and destination.

First, P_t is set as 35 dBm which corresponds to the low power regime, since if P_t is uniformly allocated to the subcarriers of the source, the average SNR is 2.17, 5.80, -2.38 , and 5.23 dB at four destinations, respectively. The corresponding $\mathcal{U}_D(n)$ and $\mathcal{U}_R(n)$ are found and visualized in Figure 8.3 for every subcarrier. The optimum RA and the corresponding WSR have been evaluated for the proposed protocol with the RA algorithm developed in Section 8.3, and for the reference protocol with the aforementioned method. The results are shown in Figure 8.4. It can be seen that for the reference protocol, the optimum destination and mode at every carrier n are $u_n = 4$ with the maximum $G_u(k) = \max\{G_{u,1}(n), G_{s,u}(n)\}$ and the relay-aided mode, respectively. The uniform sum power allocation to all subcarriers is optimum. The reason is that $G_{u,1}(n)$ for destination 4 varies slightly over all subcarriers. It is very interesting to note that, compared to the reference protocol, assigning a few subcarriers to destination 2 or 4 in the direct mode with a higher per subcarrier sum power, leads to a higher optimum WSR for the proposed protocol, even though at those subcarriers $G_{s,u}(n)$ for destination 2 or 4 is smaller than $G_{u,1}(n)$ for destination 4. For both protocols, the optimum $P_{s1,u}(n)$ and $\{P_{r_i,u}(n) | \forall r_i\}$ computed by the RA algorithm in the first step are shown in Figure 8.5. It is shown that r_3 and r_4 assist relaying simultaneously at a few subcarriers for both protocols.

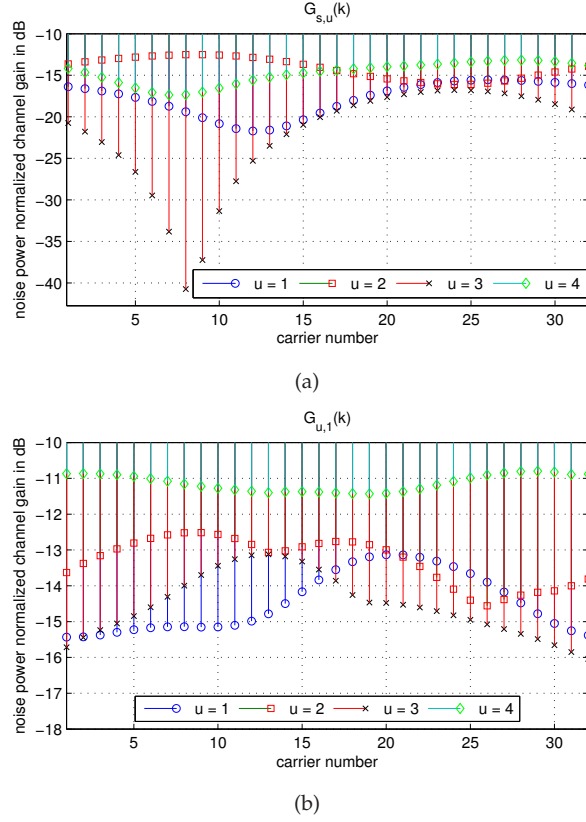


Figure 8.2 (a) $G_{s,u}(n)$ and (b) $G_{u,1}(n)$ computed by the proposed RA algorithm in the first step, at each carrier n for every destination.

Next, P_t is set as 60 dBm, which satisfies the conditions (8.36) and (8.37). For each carrier n , the $\mathcal{U}_D(n)$ is the same as when $P_t = 35$ dBm, since it is determined independently of P_t according to the conditions in (8.19). The $\mathcal{U}_R(n)$ is visualized in Figure 8.6 for every subcarrier. Now, $\mathcal{U}_R(n)$ contains none of the destinations at most of the subcarriers, since P_t is so high that the conditions in (8.20) are not satisfied by any destination at many subcarriers. The optimum RAs and WSRs have been evaluated for the two protocols and shown in Figure 8.7. It is shown that the optimum RA for the proposed protocol leads to a much higher WSR compared to that for the reference protocol. For the reference protocol, the optimum mode and destination at each subcarrier are the same as when $P_t = 35$ dBm. This is because when the destinations have

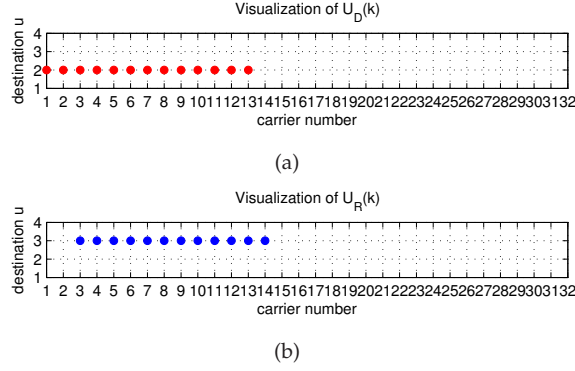


Figure 8.3 When $P_t = 35$ dBm, (a) the visualization of $\mathcal{U}_D(n)$, $\forall k$, where a dot marked at the coordinate (n, u) indicates destination $u \in \mathcal{U}_D(n)$, otherwise $u \notin \mathcal{U}_D(n)$. (b) the visualization of $\mathcal{U}_R(n)$, $\forall k$, where a dot marked at the coordinate (n, u) indicates destination $u \in \mathcal{U}_R(n)$, otherwise $u \notin \mathcal{U}_R(n)$.

equal weights, the optimum mode and destination for the reference protocol only depends on the $G_{s,u}(n)$ and $G_{u,1}(n)$ of the destinations. The uniform sum power allocation to all subcarriers is still optimum, as explained for the case when $P_t = 35$ dBm. For the proposed protocol, the optimum destination and mode for every carrier n are the destination with the maximum $G_{s,u}(n)$ and the direct mode, respectively, and the uniform sum power allocation is also optimum. For the reference protocol, the optimum $P_{s1,u}(n)$ and $\{P_{r_i,u}(n) | \forall r_i\}$ computed by the RA algorithm in the first step are shown in Figure 8.8. It is shown that r_3 and r_4 assist relaying simultaneously at a few subcarriers. Note that for the proposed protocol, the optimum sum power at carrier n is equally allocated to $P_{s1,u}(n)$ and $P_{s2,u}(n)$, which is not shown here.

For the considered channel realization, the above observations illustrate the effectiveness of the proposed protocol and algorithm to adapt the RA dynamically for WSR maximization. They also support the analysis of Section 8.4 for the high power regime. When $P_t = 60$ dBm and the weights for destination 1, 2, 3, and 4 are 0.4, 0.2, 0.2, 0.2, respectively, the optimum RA for the proposed protocol has also been evaluated, and it was found that destination 1 with the maximum weight and the direct mode are optimum for each subcarrier. The uniform sum power allocation is optimum. These observations illustrate the effectiveness of the analysis of Section 8.4 for unequal destination weights in the high power regime.

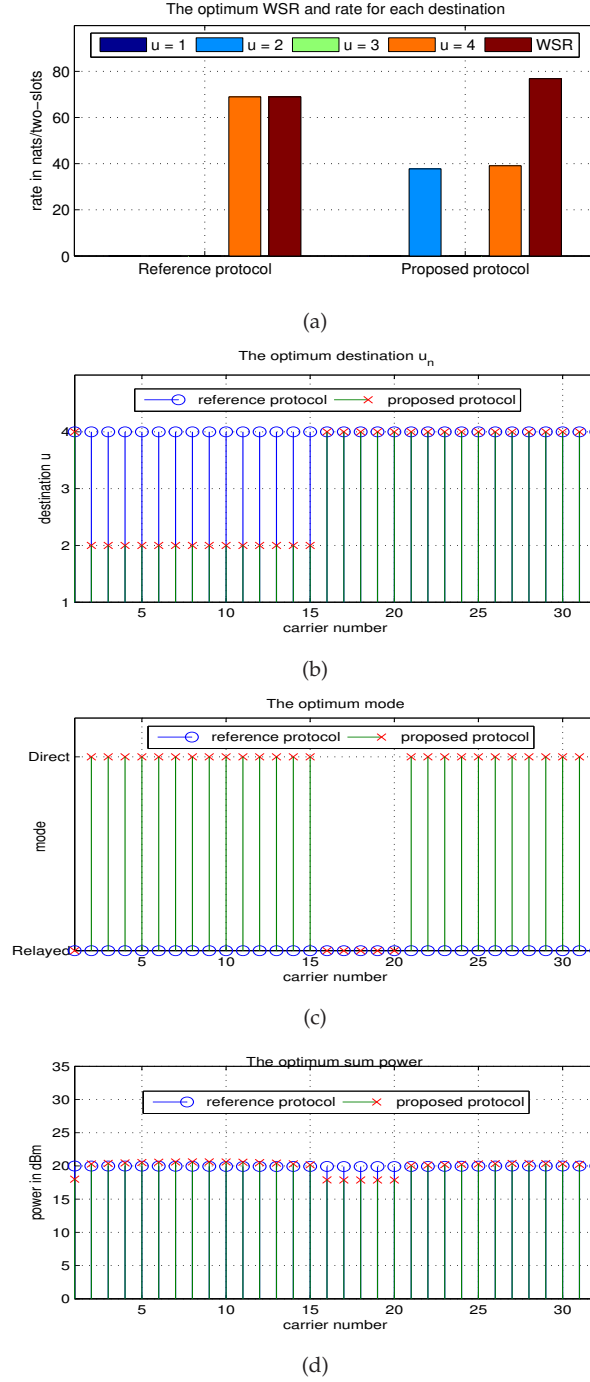
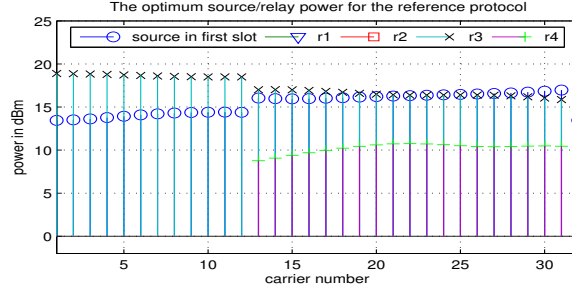
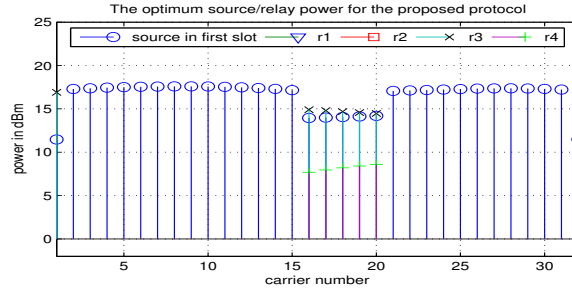


Figure 8.4 For the reference and proposed protocols, the optimum RA found in the second step for each carrier n when $P_t = 35$ dBm.



(a)



(b)

Figure 8.5 The optimum source and relay power in the broadcasting slot for the two protocols when $P_t = 35$ dBm.

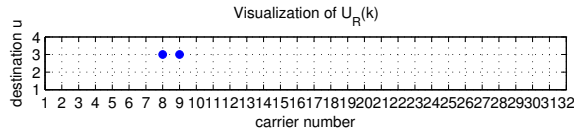
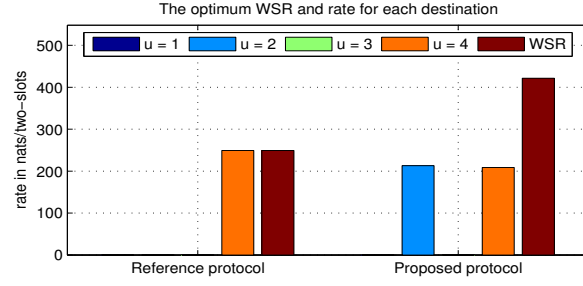
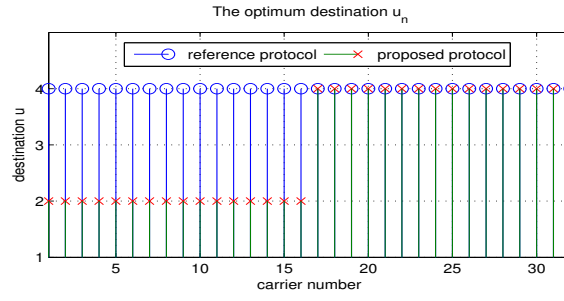


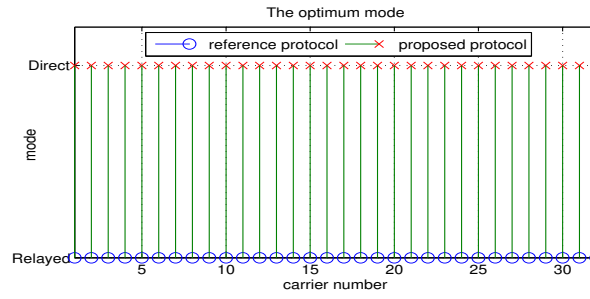
Figure 8.6 The visualization of $\mathcal{U}_R(n)$ for each carrier n when $P_t = 60$ dBm, where a dot marked at the coordinate (n, u) means that destination $u \in \mathcal{U}_R(n)$, otherwise $u \notin \mathcal{U}_R(n)$.



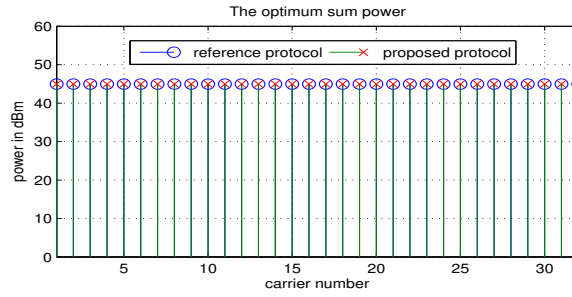
(a)



(b)



(c)



(d)

Figure 8.7 For the reference and proposed protocols, the optimum RA found in the second step for each carrier n when $P_t = 60$ dBm and destinations have equal weights.

8.5.2 Results for a set of random realizations of channels

For comprehensive illustration of scenarios close to reality, 1000 random channel realizations have been generated. For each realization, the optimum RAs and the corresponding WSRs for the reference and proposed protocols are evaluated. Assume $N = 64$ subcarriers are used, $\sigma^2 = -30$ dBm, $U = 8$ destinations exist with equal weights, $N_s = 100$, $\delta = 10^{-3}(\mu_U - \mu_L)$, and $\epsilon = 0.1$. To produce each channel realization, the destination coordinates are first generated in the region $\{(x, y) | -10 \leq x \leq 10, -30 \leq y \leq -10\}$ uniformly and independently of each other, then the random channels are produced.

For better clarity, the optimum WSRs for only the first 100 channel realizations, and the average WSR for all channel realizations are shown in Figure 8.9 for both protocols when P_t is 35 and 60 dBm, respectively. It is found that for every channel realization, the optimum WSR of the proposed protocol is equal to or greater than that for the reference protocol. The reason is that the set of feasible RAs searched for the optimum RA of the reference protocol, is a subset of that of the proposed protocol, since extra constraints are imposed on the RA optimization for the reference protocol, i.e. $P_{s_2,u}(n)$ must be zero for every subcarrier in the direct mode. Compared to the reference protocol, the optimum RA for the proposed protocol leads to either the same or a slightly greater optimum WSR for each channel realization when $P_t = 35$ dBm. However, a much higher optimum WSR results from the optimum RA and the proposed protocol for the majority of channel realizations when $P_t = 60$ dBm. As a result, the average optimum WSRs for the two protocols are very close

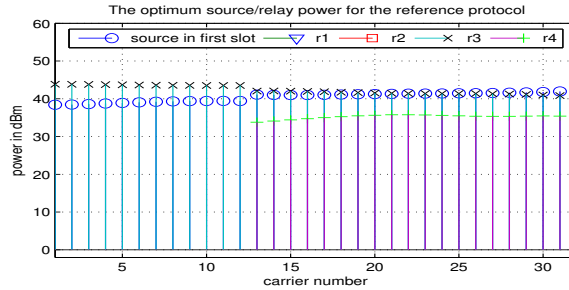


Figure 8.8 The optimum source and relay power in the broadcasting slot for the reference protocol when $P_t = 60$ dBm. Note that for the proposed protocol, the optimum sum power at each carrier n is equally allocated to $P_{s_1,u}(n)$ and $P_{s_2,u}(n)$, which is not shown here.

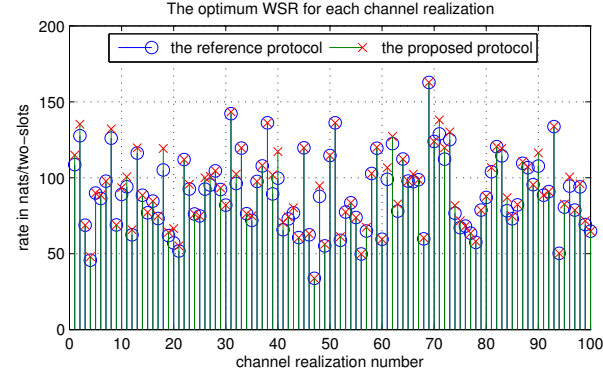
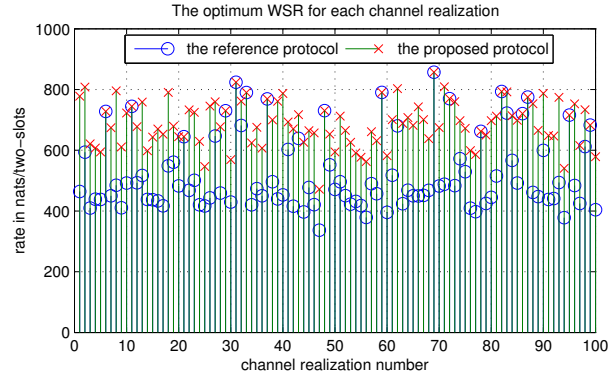
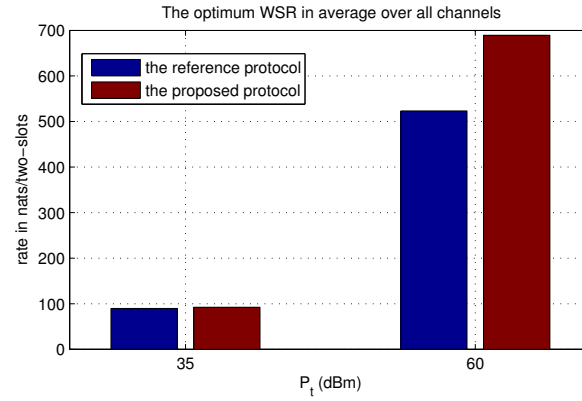
(a) when $P_t = 35$ dBm(b) when $P_t = 60$ dBm(c) when $P_t = 35$ or 60 dBm

Figure 8.9 The optimum WSRs for the first 100 channel realizations for the reference and proposed protocols, and the average WSR for all channel realizations. In each bar group, the left and right bars represent the performance for the reference and proposed protocols, respectively.

Table 8.2 Average execution time in seconds for the two-step algorithm.

K, U, N	4, 8, 64	4, 8, 128	4, 16, 64
average time	56.3	115.6	110.3
K, U, N	8, 8, 64	8, 8, 128	8, 16, 64
average time	110.1	210.2	212.6

when $P_t = 35$ dBm, while the proposed protocol and RA result in a much higher average optimum WSR than the reference protocol when $P_t = 60$ dBm. As already said, this is due to the better exploitation of channel uses by the direct mode of the proposed protocol in the high power regime.

To show the complexity of the proposed two-step algorithm, the average execution time is shown in Table 8.2 when different combinations of U and N are used. It can be seen that the increase of any one of N , U and K leads to a similar up-scaling of the execution time.

8.6 Appendix: maximization of $R_{u,n,1}$

When $G_{s,u}(n) \geq G_{s,r_n(N)}(n)$, $\gamma_r(n) \leq \gamma_{u,c}(n)$ always holds, thus $R_{u,n,1} = \ln(1 + P_{s,u}(n) \cdot \min_{r_i \in \mathcal{R}_n(u)} G_{s,r_i}(n))$, which means that $R_{u,n,1}$ is equal to the maximum rate allowed for reliable transmission from the source to all relays in $\mathcal{R}_n(u)$. In this case, the maximum $R_{u,n,1}$ is $R_{u,n,1}(P) = \ln(1 + G_{s,r_n(N)}(n)P)$. To achieve it, $\mathcal{R}_n(u)$ and $P_{s,u}(n)$ should be equal to $\{r_n(N)\}$ and P , respectively.

Now consider the case when $G_{s,r_n(N)}(n) > G_{s,u}(n)$. The maximization of $R_{u,n,1}$ is equivalent to the maximization of $\gamma_u(n)$, as the minimum of $\gamma_{u,c}(n)$ and $\gamma_r(n)$. Note that $\gamma_r(n)$ depends on $P_{s,u}(n)$ and $\mathcal{R}_n(u)$, while besides them $\gamma_{u,c}(n)$ depends on $\{P_{r_i,u}(n) | r_i \in \mathcal{R}_n(u)\}$ as well. To maximize $\gamma_u(n)$ given P , $\gamma_{u,c}(n)$ can be first maximized with $P_{s,u}(n)$ and $\mathcal{R}_n(u)$ fixed, by finding the optimum allocation of the sum relay power $P - P_{s,u}(n)$ to $\{P_{r_i,u}(n) | r_i \in \mathcal{R}_n(u)\}$. Denoting such maximized $\gamma_{u,c}(n)$ by $\gamma'_{u,c}(n)$, the optimum $\{P_{s,u}(n), \mathcal{R}_n(u)\}$ that maximizes $\min\{\gamma'_{u,c}(n), \gamma_r(n)\}$ is then found. Finally, that $\{P_{s,u}(n), \mathcal{R}_n(u)\}$, as well as the associated optimum allocation of $P - P_{s,u}(n)$ to $\{P_{r_i,u}(n) | r_i \in \mathcal{R}_n(u)\}$, constitute the optimum solution to maximizing $\gamma_u(n)$ given P .

To facilitate derivation, assume the relays in $\mathcal{R}_n(u)$ are sorted in the increasing order of $G_{s,r_i}(n)$, and the first relay in $\mathcal{R}_n(u)$ is $r_n(b)$, namely the b -th relay in Ψ_n . This means that $\mathcal{R}_n(u)$ is a subset of $\Psi_n(b)$, and $G_{\mathcal{R}_n(u),u}(n) \leq G_{\Psi_n(b),u}(n)$. It is important to note that $G_{s,r_n(i)}(n)$ is an increasing function of i , while $G_{\Psi_n(i),u}(n)$ is a decreasing function of i .

Now, $\gamma_r(n) = P_{s,u}(n)G_{s,r_n(b)}(n)$, and according to the Cauchy-Schwartz inequality,

$$\gamma_{u,c}(n) \leq P_{s,u}(n)G_{s,u}(n) + (P - P_{s,u}(n))G_{\mathcal{R}_n(u),u}(n), \quad (8.43)$$

where the equality holds when

$$\forall r_i \in \mathcal{R}_n(u), P_{r_i,u}(n) = \frac{(P - P_{s,u}(n))G_{r_i,u}(n)}{\sum_{r_i \in \mathcal{R}_n(u)} G_{r_i,u}(n)}. \quad (8.44)$$

Obviously, the right hand side of (8.43) is equal to $\gamma'_{u,c}(n)$. As $P_{s,u}(n)$ increases from 0 to P , $(P_{s,u}(n), \gamma_r(n))$ moves along the line from the origin to A, whereas $(P_{s,u}(n), \gamma'_{u,c}(n))$ along the line from B to C in the coordinate space shown in Figure 8.10, where the coordinates of A, B, and C are given in Table 8.3. Note that C is fixed, while both A and B depend on $\mathcal{R}_n(u)$. In particular, with $\mathcal{R}_n(u)$ and P fixed, the maximum of $\min\{\gamma'_{u,c}(n), \gamma_r(n)\}$ with respect to $P_{s,u}(n)$, denoted by $\gamma'_u(n)$, is achieved at A, C, and D for a given $\mathcal{R}_n(u)$ satisfying respectively

1. $b < x_{u,n}$, where $x_{u,n}$ is the smallest i satisfying $G_{s,r_n(i)}(n) > G_{s,u}(n)$.
2. $b \geq x_{u,n}$ and $G_{\mathcal{R}_n(u),u}(n) \leq G_{s,u}(n)$,
3. $b \geq x_{u,n}$ and $G_{\mathcal{R}_n(u),u}(n) > G_{s,u}(n)$.

It can be seen that only in case 3, $\gamma'_u(n)$ is greater than $G_{s,u}(n)P$. Among all $\mathcal{R}_n(u)$'s having the same first relay as $r_n(b)$ and corresponding to case 3, the $\mathcal{R}_n(u)$ with the maximum $G_{\mathcal{R}_n(u),u}(n)$ corresponds to the greatest height of D, which suggests that $\Psi_n(b)$ is the one achieving the maximum $\gamma'_u(n)$ in that set. The corresponding $\gamma'_u(n)$ and $P_{s,u}(n)$ are computed respectively as

$$\gamma'_u(n) = \frac{G_{s,r_n(b)}(n)G_{\Psi_n(b),u}(n)P}{G_{\Psi_n(b),u}(n) + G_{s,r_n(b)}(n) - G_{s,u}(n)} \quad (8.45)$$

and

$$P_{s,u}(n) = \frac{G_{\Psi_n(b),u}(n)}{G_{\Psi_n(b),u}(n) + G_{s,r_n(b)}(n) - G_{s,u}(n)}P. \quad (8.46)$$

Table 8.3 Coordinates of A, B, and C.

	A	B	C
$P_{s,u}(n)$	P	0	P
SNR	$G_{s,r_n(b)}(n)P$	$G_{\mathcal{R}_n(u),u}(n)P$	$G_{s,u}(n)P$

Based on the above analysis, the optimum $\mathcal{R}_n(u)$, $P_{s,u}(n)$, and $P_{r_i,u}(n)$ can be decided. When $G_{\Psi_n(x_{u,n}),u}(n) \leq G_{s,u}(n)$, every possible $\mathcal{R}_n(u)$ corresponds to either case 1 or 2, which means that the $\gamma'_u(n)$ maximized over $\mathcal{R}_n(u)$ and $P_{s,u}(n)$ is equal to $G_{s,u}(n)P$. In this case, the maximum $R_{u,n,1}$ is $R_{u,n,1}(P) = \ln(1 + G_{s,u}(n)P)$, and to achieve it $P_{s,u}(n)$ should be equal to P .

When $G_{\Psi_n(x_{u,n}),u}(n) > G_{s,u}(n)$, every possible $\mathcal{R}_n(u)$ corresponds to one of cases 1, 2, and 3. Obviously, the optimum $\mathcal{R}_n(u)$ should correspond to case 3. To have $\mathcal{R}_n(u)$ corresponding to case 3, b must be between $x_{u,n}$ and $y_{u,n}$, where $y_{u,n}$ is the greatest i satisfying $G_{\Psi_n(i),u}(n) > G_{s,u}(n)$. As pointed out before, $\Psi_n(b)$ is the $\mathcal{R}_n(u)$ achieving the maximum $\gamma'_u(n)$ among all $\mathcal{R}_n(u)$'s having the same first relay as $r_n(b)$. Therefore, the optimum $\mathcal{R}_n(u)$ should be $\Psi_n(z_{u,n})$, where $z_{u,n}$ is the b between $x_{u,n}$ and $y_{u,n}$ and maximizing $\gamma'_u(n)$, namely the maximizer of the right-hand side of (8.17). In this case, the maximum $R_{u,n,1}$ and the optimum RA can be evaluated with the formulas for the third case shown in Section 8.3.2.

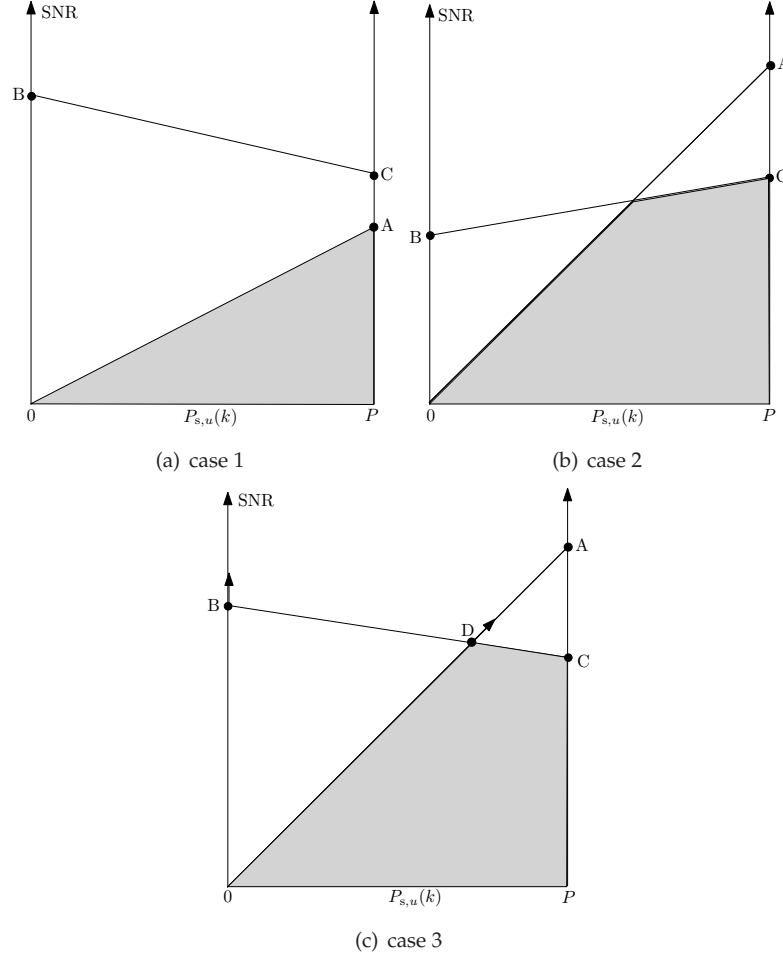


Figure 8.10 Illustrations of $\gamma'_{u,c}(n)$, $\gamma_r(n)$, and $\gamma'_u(n)$ versus $P_{s,u}(n)$. $\gamma'_u(n)$ is highlighted by painting the area it bounds in light grey.

Summary and future work

9

9.1 A summary of the thesis

In Chapters 2 and 3, some fundamental optimization theory has been reviewed. In particular, the focus was put on optimality conditions, duality-gap related concepts and the dual method, which were extensively used in the remaining part of the thesis.

In Chapter 4, the SCA method has been generalized to the SA method for solving a general optimization problem. Then, general methods was proposed to construct a CUBA for a nonconvex function, which is the key to the SCA method. Finally, the EM algorithm and the successive GP approximation algorithm were given as two interesting applications of the SA and CUBA construction methods.

In Chapter 5, the WSR maximized DSM problem has been addressed for a CCI corrupted point-to-point OFDM system under per-transmitter sum power constraints. The SCA-based DSM algorithms, namely the SCALE, CA-DSB and DCA-DSM algorithms, were revisited to show how they can be derived from the SCA and CUBA construction methods. They were also compared with the MIWF and ISB algorithms.

In Chapter 6, the WSMR maximized RA problem has been addressed for the downlink of a multi-cell OFDMA system subject to per BS sum power constraints. A CA-based algorithm, namely Algorithm 5 was proposed to jointly optimize BSs' subcarrier and power allocation alternatively. The effectiveness of the algorithm was illustrated by numerical experiments.

In Chapter 7, the sum rate maximized RA problem has been addressed for a point-to-point OFDM system assisted by multiple DF relays subject to per

Table 9.1 A summary of algorithms studied in this thesis

	based on	converg.	complexity
EM algorithm	SA	yes	problem specific
Successive GP approx.	SCA	yes	problem specific
SCALE algorithm	SCA	unproved	low
CA-DSB algorithm	SCA	unproved	low
DCA-DSM algorithm	SCA	yes	low
Algorithm 5	CA	yes	high
Algorithm 7	dual	yes	high
Algorithm 10	CA	yes	high
the two-step algorithm	dual	yes	high

device sum power constraints of the source and relays. A duality-based algorithm, namely Algorithm 7, and a CA-based algorithm, namely Algorithm 10, were proposed. The effectiveness of these algorithms was illustrated by numerical experiments.

In Chapter 8, the WSR maximized RA problem has been addressed for an OFDMA system assisted by multiple DF relays subject to a system sum power constraints. A two-step algorithm was proposed to find the globally optimum RA. The effectiveness of the proposed algorithm was illustrated by numerical experiments.

A summary of the algorithms studied in this thesis is given in Table 9.1. Note that an algorithm is marked as being of high complexity when its average execution time is greater than 5 seconds.

9.2 Perspectives for future work

In Chapter 5, interesting future work can be carried out to design SCA-based DSM algorithms using the general SCA and CUBA construction methods.

In Chapter 6, the power allocation is optimized by using lower bounds built in the same way as that for the SCALE algorithm. It is interesting to investigate if using lower bounds built in other ways, e.g., as those for the CA-DSB or DCA-DSM algorithms, can lead to better performance.

In Chapters 7-8, a carrier in the first time slot is always used together with the same carrier in the second slot, when either the direct or relay-aided transmission mode is used. In fact, carrier pairing optimization, i.e., finding the best carrier in the second slot used along with every carrier in the first slot, can be further incorporated to improve performance. In addition, source may be allowed to participate beamforming in the second slot for the relay-aided mode. In future, it is interesting to design improved RA algorithms considering carrier pairing optimization and source's participation in beamforming. Moreover, it deserves much attention to develop RA algorithms under per-device sum power constraints for the system studied in Chapter 8.

As shown in Table 9.1, the algorithms developed in Chapters 6-8 have high complexity. Moreover, all the algorithms in Chapters 5-8 require perfect CSI. In future, it is interesting to develop low-complexity algorithms using imperfect CSI.

Bibliography

- [AADM07] A. Abrardo, A. Alessio, P. Detti, and M. Moretti, "Centralized radio resource allocation for OFDMA cellular systems," *IEEE International Conference on Communications*, pp. 5738–5743, Jun. 2007.
- [AW70] M. Avriel and A. C. Williams, "Complementary geometric programming," *SIAM J. Appl. Math.*, vol. 19, pp. 125–141, 1970.
- [Ber03] D. P. Bertsekas, *Nonlinear programming, 2nd edition*. Athena Scientific, 2003.
- [BKVH05] S. P. Boyd, S. J. Kim, L. Vandenberghe, and A. Hassibi, "A tutorial on geometric programming," *Optimization and Engineering*, July 2005.
- [Boy04] S. Boyd, *EE392o course notes*. Stanford university, available at: <http://www.stanford.edu/class/ee392o>, 2004.
- [BV04] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.
- [CG79] T. Cover and A. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inf. Theory*, vol. 25, no. 5, pp. 572–584, Sep. 1979.
- [CLW09] Y. Cui, V. K. N. Lau, and R. Wang, "Distributive subband allocation, power and rate control for relay-assisted OFDMA cellular system with imperfect system state knowledge," *IEEE Transactions on Wireless Communications*, vol. 8, no. 10, pp. 5096–5102, Oct. 2009.

- [CM05] R. Cendrillon and M. Moonen, "Iterative spectrum balancing for digital subscriber lines," in *IEEE Int. Conf. Commun.*, vol. 3, May 2005, pp. 1937–1941.
- [CTP⁺07] M. Chiang, C. W. Tan, D. Palomar, D. O'Neill, and D. Julian, "Power control by geometric programming," *IEEE Transactions on Wireless Communications*, vol. 6, no. 7, pp. 2640–2651, July 2007.
- [CYM⁺06] R. Cendrillon, W. Yu, M. Moonen, J. Verlinden, and T. Bostoen, "Optimal multiuser spectrum balancing for digital subscriber lines," *IEEE Trans. Commun.*, vol. 54, no. 5, pp. 922–933, May 2006.
- [DHH⁺10] O. Duval, Z. Hasan, E. Hossain *et al.*, "Subcarrier selection and power allocation for amplify-and-forward relaying over OFDM links," *IEEE Transactions on Wireless Communications*, vol. 9, no. 4, pp. 1293–1297, Apr. 2010.
- [DP72] R. J. Duffin and E. L. Peterson, "Reversed geometric programs treated by harmonic means," *Indiana Univ. Math. J.*, vol. 22, pp. 531–550, 1972.
- [DTM⁺10] W. Dang, M. Tao, H. Mu *et al.*, "Subcarrier-pair based resource allocation for cooperative multi-relay OFDM systems," *IEEE Transactions on Wireless Communications*, vol. 9, no. 5, pp. 1640–1649, May 2010.
- [FK88] K. Fazel and S. Kaiser, *Multi-Carrier and Spread Spectrum Systems: From OFDM and MC-CDMA to LTE and WiMAX*. John Wiley & Sons, 2nd edition, 2008.
- [GCJ08] B. Gui and L. J. Cimini-Jr, "Bit loading algorithms for cooperative OFDM systems," *Eurasip Journal on Wireless Communications and Networking*, 2008.
- [GGbK08] A. Gjendemsjφ, D. Gesbert, G. Øien, and S. Kiani, "Binary power control for sum rate maximization over multiple interfering links," *IEEE Transactions on Wireless Communications*, vol. 7, no. 8, pp. 3164–3173, August 2008.

- [GKGb07] D. Gesbert, S. Kiani, A. Gjendemsj , and G.  ien, "Adaptation, coordination, and distributed resource allocation in interference-limited wireless networks," *Proceedings of the IEEE*, vol. 95, no. 12, pp. 2393–2409, Dec. 2007.
- [GKH⁺07] D. Gesbert, M. Kountouris, R. W. Heath, C. B. Chae, and T. Salzer, "Shifting the MIMO paradigm," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 36–46, Sept. 2007.
- [Her06] C. Herzet, *Code-Aided Synchronization for Digital Burst Communications*. Doctoral thesis in Universit  catholique de Louvain, Jul. 2006.
- [HJ85] R. Horn and C. R. Johnson, *Matrix analysis*. Cambridge University Press, 1985.
- [HLLN] S. Huberman, C. Leung, and T. Le-Ngoc, "Dynamic spectrum management (DSM) algorithms for multi-user xDSL," to appear in *IEEE Commun. Surveys & Tutor* (please find it among early access papers of this journal).
- [JL10] Y. Jang, E. Jeong, and Y. H. Lee, "A two-step approach to power allocation for OFDM signals over two-way amplify-and-forward relay," *IEEE Transactions on Signal Processing*, vol. 58, no. 4, pp. 2426–2430, Apr. 2010.
- [JL03] J. Jang and K. B. Lee, "Transmit power adaptation for multiuser OFDM systems," *IEEE J. Selected Areas in Communications*, vol. 21, no. 2, pp. 171–178, Feb 2003.
- [KC07] R. Kwak and J. Cioffi, "Resource-allocation for OFDMA multi-hop relaying downlink systems," in *IEEE Global Telecommunications Conference*, 2007, pp. 3225–3229.
- [KG08] S. Kiani and D. Gesbert, "Optimal and distributed scheduling for multicell capacity maximization," *IEEE Transactions on Wireless Communications*, vol. 7, no. 1, pp. 288–297, Jan. 2008.
- [KKMB06] K. Koh, S. J. Kim, A. Mutapcic, and S. P. Boyd, "gpcvx: a matlab solver for geometric programs in convex form," 2006, www.stanford.edu/~boyd/ggplab/gpcvx.pdf.

- [KOG07] S. Kiani, G. Oien, and D. Gesbert, "Maximizing multicell capacity using distributed power allocation and scheduling," in *Wireless Communications and Networking Conference*, March 2007, pp. 1690–1694.
- [KP07] M. Kaneko and P. Popovski, "Radio resource allocation algorithm for relay-aided cellular OFDMA system," in *IEEE International Conference on Communications*, 2007, pp. 4831–4836.
- [KPL06] I. Kim, I.-S. Park, and Y. Lee, "Use of linear programming for dynamic subcarrier and bit allocation in multiuser OFDM," *IEEE Trans. Vehicular Technology*, vol. 55, no. 4, pp. 1195–1207, July 2006.
- [LP12] Z.-Q. Luo and J.-S. Pang, "Analysis of iterative waterfilling algorithm for multiuser power control in digital subscriber lines," *EURASIP J. Appl. Sig. Proc.*, 2006, Article ID 24012.
- [LTW04] J. N. Laneman, D. N. C. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: efficient protocols and outage behavior," *IEEE Transaction on Information Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.
- [LW03] J. N. Laneman and G. W. Wornell, "Distributed space-time-coded protocols for exploiting cooperative diversity in wireless networks," *IEEE Transaction on Information Theory*, vol. 49, no. 10, pp. 2415–2425, Oct. 2003.
- [LWK⁺08] Y. Li, W. Wang, J. Kong *et al.*, "Power allocation and subcarrier pairing in OFDM-based relaying networks," in *IEEE International Conference on Communications*, 2008, pp. 2602–2606.
- [LY05] R. Lui and W. Yu, "Low-complexity near-optimal spectrum balancing for digital subscriber lines," in *IEEE Int. Conf. Commun.*, vol. 3, May 2005, pp. 1947–1951.
- [LZ08] Z.-Q. Luo and S. Zhang, "Dynamic spectrum management: Complexity and duality," *IEEE J. Select. Topics Sig. Proc.*, vol. 2, no. 1, pp. 57–73, Feb. 2008.

- [MKP07] M. Morelli, C. Kuo, and M. Pun, "Synchronization techniques for orthogonal frequency division multiple access (ofdma): a tutorial review," *Proceeding of IEEE*, vol. 95, no. 7, pp. 1394–1427, July 2007.
- [MS00] T. K. Moon and W. C. Stirling, *Mathematical methods and algorithms for signal processing*. Prentice Hall, 2000.
- [MW78] B. Marks and G. P. Wright, "A general inner approximation algorithm for nonconvex mathematical programs," *Operations Research*, vol. 26, no. 4, pp. 681–683, 1978.
- [NCC⁺07] W. Nam, W. Chang, S.-Y. Chung *et al.*, "Transmit optimization for relay-based cellular OFDMA systems," in *IEEE International Conference on Communications*, 2007, pp. 5714–5719.
- [NY07] T. C.-Y. Ng and W. Yu, "Joint optimization of relay strategies and resource allocations in cooperative cellular networks," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 2, pp. 328–339, Feb. 2007.
- [PE06] J. Papandriopoulos and J. Evans, "Low-complexity distributed algorithms for spectrum balancing in multi-user DSL networks," in *IEEE Int. Conf. Commun.*, vol. 7, June 2006, pp. 3270–3275.
- [PE09] —, "SCALE: A low-complexity distributed protocol for spectrum balancing in multiuser DSL networks," *IEEE Trans. Inf. Theory*, vol. 55, no. 8, pp. 3711–3724, Aug. 2009.
- [PH09] S. Peters and R. Heath, "The future of wimax: Multihop relaying with ieee 802.16j," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 104–111, Jan. 2009.
- [Pha88] D. T. Pham, "Duality in d.c. (difference of convex functions) optimization. subgradient methods," *Trends in math. optim., Intern. Series Numer. Math.*, vol. 84, pp. 277–293, 1988.
- [PJ04] S. Pietrzyk and G. J. M. Janssen, "Radio resource allocation for cellular networks based on OFDMA with QoS guarantees," *IEEE Global Telecommunications Conference*, pp. 2694–2699, Nov. 2004.

- [PNT10] T. T. Pham, H. H. Nguyen, and H. D. Tuan, "Power allocation in orthogonal wireless relay networks with partial channel state information," *IEEE Transactions on Signal Processing*, vol. 58, no. 2, pp. 869–878, Feb. 2010.
- [PT97] D. T. Pham and H. A. L. Thi, "Convex analysis approach to d.c. programming: theory, algorithms and applications," *Acta Math. Vietnamica*, vol. 22, pp. 289–355, 1997.
- [PWS⁺04] R. Pabst, B. Walke, D. Schultz *et al.*, "Relay-based deployment concepts for wireless and mobile broadband radio," *IEEE Communications Magazine*, vol. 42, no. 9, pp. 80–89, sept. 2004.
- [RC00] W. Rhee and J. M. Cioffi, "Increase in capacity of multiuser OFDM system using dynamic subchannel allocation," in *Proc. Vehicular Technology Conference Proceedings*, vol. 2, 2000, pp. 1085–1089.
- [RW74] G. V. Reklaitis and D. J. Wilde, "Geometric programming via a primal auxiliary problem," *AIIE Trans.*, vol. 6, pp. 308–317, 1974.
- [SAE05] Z. Shen, J. Andrews, and B. Evans, "Adaptive resource allocation in multiuser OFDM systems with proportional rate constraints," *IEEE Trans. Wireless Communications*, vol. 4, no. 6, pp. 2726–2737, Nov. 2005.
- [SAR⁺10] M. Salem, A. Adinoyi, M. Rahman *et al.*, "Fairness-aware radio resource management in downlink OFDMA cellular relay networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 5, pp. 1628–1639, May 2010.
- [SCS99] T. Starr, J. Cioffi, and P. Silverman, *Understanding Digital Subscriber Line Technology*. Prentice Hall, 1999.
- [SEA03] A. Sendonaris, E. Erkip, and B. Aazhang, "User cooperation diversity. part i. system description," *IEEE Trans. Commun.*, vol. 51, no. 11, pp. 1927–1938, Nov. 2003.
- [SLV08] T. Sartenauer, J. Louveaux, and L. Vandendorpe, "Balanced capacity of wireline multiple access channels with individual power constraints," *IEEE Trans. on Communications*, vol. 56, no. 6, pp. 925–936, June 2008.

- [SPB08] G. Scutari, D. Palomar, and S. Barbarossa, "Optimal linear precoding strategies for wideband noncooperative systems based on game theory. I. nash equilibria," *IEEE Trans. Sig. Proc.*, vol. 56, no. 3, pp. 1230–1249, Mar. 2008.
- [Ste98] G. Stewart, "Matrix algorithms: Basic decompositions," *SIAM*, p. 55, 1998.
- [SVL05] T. Sartenauer, L. Vandendorpe, and J. Louveaux, "Balanced capacity of wireline multiuser channels," *IEEE Trans. on Communications*, vol. 53, no. 12, pp. 2029–2042, Dec. 2005.
- [TDM08] P. Tsiaflakis, M. Diehl, and M. Moonen, "Distributed spectrum management algorithms for multiuser DSL networks," *IEEE Trans. Sig. Proc.*, vol. 56, no. 10, pp. 4825–4843, Oct. 2008.
- [TV05] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge University Press, 2005.
- [VDL⁺08] L. Vandendorpe, R. Duran, J. Louveaux *et al.*, "Power allocation for OFDM transmission with DF relaying," in *IEEE International Conference on Communications*, 2008, pp. 3795–3800.
- [vdM71] E. C. van der Meulen, "Three-terminal communication channels," *Adv. Appl. Prob.*, vol. 3, pp. 120–154, 1971.
- [VLO⁺] L. Vandendorpe, J. Louveaux, O. Oguz *et al.*, "Rate-optimized power allocation for DF-relayed OFDM transmission under sum and individual power constraints," *Eurasip Journal on Wireless Communications and Networking*, vol. 2009.
- [VLO⁺08] —, "Improved OFDM transmission with DF relaying and power allocation for a sum power constraint," in *ISWPC*, 2008, pp. 665–669.
- [VLO⁺09a] —, "Power allocation for improved DF relayed OFDM transmission: the individual power constraint case," in *IEEE International Conference on Communications*, 2009, pp. 1–6.

- [VLO⁺09b] —, “Rate-optimized power allocation for OFDM transmission with multiple DF/regenerative relays and an improved protocol,” in *IEEE Wireless Communications and Networking Conference*, 2009, pp. 1–6.
- [VPW07] L. Venturino, N. Prasad, and X. Wang, “An improved iterative water-filling algorithm for multi-cell interference mitigation in downlink OFDMA networks,” in *Asilomar Conference on Signals, Systems and Computers*, Nov. 2007, pp. 1718–1722.
- [VPW08] —, “A successive convex approximation algorithm for weighted sum-rate maximization in downlink OFDMA networks,” in *Conference on Information Sciences and Systems*, Mar. 2008, pp. 379–384.
- [Wan08] T. Wang, *Phase estimation and ambiguity resolution for coded communication systems*. Master thesis in Universitat Politècnica de Catalunya, Sep. 2008.
- [WCLM99] C. Y. Wong, R. Cheng, K. Lataief, and R. Murch, “Multiuser OFDM with adaptive subcarrier, bit, and power allocation,” *IEEE J. Selected Areas in Communications*, vol. 17, no. 10, pp. 1747–1758, Oct 1999.
- [WV10a] T. Wang and L. Vandendorpe, “Coordinate ascent based resource allocation algorithm for maximizing wsmr in downlink cellular ofdma systems,” in *Proc. PIMRC*, Sep. 2010, pp. 380–385.
- [WV10b] —, “Iterative power and subcarrier allocation for maximizing WSMR in cellular OFDMA systems,” in *Proc. RAWNET*, Jun. 2010, pp. 324–329.
- [WV10c] —, “Resource allocation for maximizing weighted sum min-rate in downlink cellular OFDMA systems,” in *Proc. ICC*, May 2010, pp. 1–6.
- [WV11a] —, “Iterative resource allocation for maximizing weighted sum min-rate in downlink cellular OFDMA systems,” *IEEE Trans. Sig. Proc.*, vol. 59, no. 1, pp. 223–234, Jan. 2011.

- [WV11b] —, “On the SCALE algorithm for multiuser multicarrier power spectrum management,” *submitted to IEEE Trans. Sig. Proc.*, Jun. 2011.
- [WV11c] —, “Successive convex approximation based methods for dynamic spectrum management,” *submitted to ICC 2012*, Sep. 2011.
- [WV11d] —, “Sum rate maximized resource allocation in multiple DF relays aided OFDM transmission,” *IEEE J. Sel. Areas on Commun.*, vol. 29, no. 8, pp. 1559–1571, Sep. 2011.
- [WV11e] —, “WSR maximized resource allocation in multiple DF relays aided OFDMA downlink transmission,” *IEEE Trans. Sig. Proc.*, vol. 59, no. 8, pp. 3964–3976, Aug. 2011.
- [WV11f] —, “A two-step RA algorithm for multiple DF relays aided OFDMA downlink transmission,” in *Proc. Joint NEW-COM++/COST 2100 Workshop*, Mar. 2011.
- [YGC02] W. Yu, G. Ginis, and J. Cioffi, “Distributed multiuser power control for digital subscriber lines,” *IEEE J. Selec. Areas Commun.*, vol. 20, no. 5, pp. 1105–1115, Jun. 2002.
- [YL00] H. Yin and H. Liu, “An efficient multiuser loading algorithm for OFDM-based broadband wireless systems,” in *IEEE Global Telecommunications Conference*, vol. 1, 2000, pp. 103–107.
- [YL06] W. Yu and R. Lui, “Dual methods for nonconvex spectrum optimization of multicarrier systems,” *IEEE Trans. Commun.*, vol. 54, no. 7, pp. 1310–1322, July 2006.
- [Yu07] W. Yu, “Multiuser water-filling in the presence of crosstalk,” in *Proc. Information Theory and Applications Workshop*, Feb. 2007, pp. 414–420.
- [ZL04] Y. J. Zhang and K. Letaief, “Multiuser adaptive subcarrier-and-bit allocation with adaptive cell selection for OFDM systems,” *IEEE Trans. Wireless Communications*, vol. 3, no. 5, pp. 1566–1575, Sept. 2004.