OULU 2018
C 638

ACTA UNIVERSITATIS OULUENSIS

Satya Krishna Joshi

RADIO RESOURCE ALLOCATION TECHNIQUES FOR MISO DOWNLINK CELLULAR NETWORKS

UNIVERSITY OF OULU GRADUATE SCHOOL;
UNIVERSITY OF OULU,
FACULTY OF INFORMATION TECHNOLOGY AND ELECTRICAL ENGINEERING;
CENTRE FOR WIRELESS COMMUNICATIONS

ACTA UNIVERSITATIS OULUENSIS
C Technica 638

SATYA KRISHNA JOSHI

RADIO RESOURCE ALLOCATION TECHNIQUES FOR MISO DOWNLINK CELLULAR NETWORKS

Academic dissertation to be presented with the assent of the Doctoral Training Committee of Technology and Natural Sciences of the University of Oulu for public defence in the OP auditorium (L10), Linnanmaa, on 12 January 2018, at 12 noon

UNIVERSITY OF OULU, OULU 2018

Acta Univ. Oul. C 638, 2018

Supervised by
Professor Matti Latva-aho
Docent Marian Codreanu

Reviewed by
Professor Wei Yu
Professor Carlo Fischione

Opponent
Professor Risto Wichman

ISBN 978-952-62-1742-0 (Paperback)
ISBN 978-952-62-1743-7 (PDF)

ISSN 0355-3213 (Printed)
ISSN 1796-2226 (Online)

Cover Design
Raimo Ahonen

JUVENES PRINT
TAMPERE 2018

Joshi, Satya Krishna, Radio resource allocation techniques for MISO downlink cellular networks.

University of Oulu Graduate School; University of Oulu, Faculty of Information Technology and Electrical Engineering; Centre for Wireless Communications

Acta Univ. Oul. C 638, 2018

University of Oulu, P.O. Box 8000, FI-90014 University of Oulu, Finland

Abstract

This thesis examines radio resource management techniques for multicell multi-input single-output (MISO) downlink networks. Specifically, the thesis focuses on developing linear transmit beamforming techniques by optimizing certain quality-of-service (QoS) features, including, spectral efficiency, fairness, and throughput.

The problem of weighted sum-rate-maximization (WSRMax) has been identified as a central problem to many network optimization methods, and it is known to be NP-hard. An algorithm based on a branch and bound (BB) technique which globally solves the WSRMax problem with an optimality certificate is proposed. Novel bounding techniques via conic optimization are introduced and their efficiency is illustrated by numerical simulations. The proposed BB based algorithm is not limited to WSRMax only; it can be easily extended to maximize any system performance metric that can be expressed as a Lipschitz continuous and increasing function of the signal-to-interference-plus-noise (SINR) ratio.

Beamforming techniques can provide higher spectral efficiency, only when the channel state information (CSI) of users is accurately known. However, in practice the CSI is not perfect. By using an ellipsoidal uncertainty model for CSI errors, both optimal and suboptimal robust beamforming techniques for the worst-case WSRMax problem are proposed. The optimal method is based on a BB technique. The suboptimal algorithm is derived using alternating optimization and sequential convex programming. Through a numerical example it is also shown how the proposed algorithms can be applied to a scenario with statistical channel errors.

Next two decentralized algorithms for multicell MISO networks are proposed. The optimization problems considered are: P1) minimization of the total transmission power subject to minimum SINR constraints of each user, and P2) SINR balancing subject to the total transmit power constraint of the base stations. Problem P1 is of great interest for obtaining a transmission strategy with minimal transmission power that can guarantee QoS for users. In a system where the power constraint is a strict system restriction, problem P2 is useful in providing fairness among the users. Decentralized algorithms for both problems are derived by using a consensus based alternating direction method of multipliers.

Finally, the problem of spectrum sharing between two wireless operators in a dynamic MISO network environment is investigated. The notion of a two-person bargaining problem is used to model the spectrum sharing problem, and it is cast as a stochastic optimization. For this problem, both centralized and distributed dynamic resource allocation algorithms are proposed. The proposed distributed algorithm is more suitable for sharing the spectrum between the operators, as it requires a lower signaling overhead, compared with centralized one. Numerical results show that the proposed distributed algorithm achieves almost the same performance as the centralized one.

Keywords: distributed optimization, dynamic control, global (nonconvex) optimization, radio resource management, robust resource allocation, spectrum sharing

Joshi, Satya Krishna, Radioresurssien hallintatekniikoita laskevan siirtotien moniantennilähetyksiin solukkoverkoissa.
Oulun yliopiston tutkijakoulu; Oulun yliopisto, Tieto- ja sähkötekniikan tiedekunta; Centre for Wireless Communications
Acta Univ. Oul. C 638, 2018
Oulun yliopisto, PL 8000, 90014 Oulun yliopisto

Tiivistelmä

Tässä väitöskirjassa tarkastellaan monisoluisten laskevan siirtotien moniantennilähetystä käyttävien verkkojen radioresurssien hallintatekniikoita. Väitöskirjassa keskitytään erityisesti kehittämään lineaarisia siirron keilanmuodostustekniikoita optimoimalla tiettyjä palvelun laadun ominaisuuksia, kuten spektritehokkuutta, tasapuolisuutta ja välityskykyä.

Painotetun summadatanopeuden maksimoinnin (WSRMax) ongelma on tunnistettu keskeiseksi monissa verkon optimointitavoissa ja sen tiedetään olevan NP-kova. Tässä työssä esitetään yleinen branch and bound (BB) -tekniikkaan perustuva algoritmi, joka ratkaisee WSRMax-ongelman globaalisti ja tuottaa todistuksen ratkaisun optimaalisuudesta. Samalla esitellään uusia conic-optimointia hyödyntäviä suorituskykyrajojen laskentatekniikoita, joiden tehokkuutta havainnollistetaan numeerisilla simuloinneilla. Ehdotettu BB-perusteinen algoritmi ei rajoitu pelkästään WSRMax-ongelmaan, vaan se voidaan helposti laajentaa maksimoimaan mikä tahansa järjestelmän suorituskykyarvo, joka voidaan ilmaista Lipschitz-jatkuvana ja signaali-(häiriö+kohina) -suhteen (SINR) kasvavana funktiona.

Keilanmuodostustekniikat voivat tuottaa suuremman spektritehokkuuden vain, jos käyttäjien kanavien tilatiedot tiedetään tarkasti. Käytännössä kanavan tilatieto ei kuitenkaan ole täydellinen. Tässä väitöskirjassa ehdotetaan WSRMax-ongelman ääritapauksiin sekä optimaalinen että alioptimaalinen keilanmuodostustekniikka soveltaen tilatietovirheisiin ellipsoidista epävarmuusmallia. Optimaalinen tapa perustuu BB-tekniikkaan. Alioptimaalinen algoritmi johdetaan peräkkäistä konveksiohjelmointia käyttäen. Numeerisen esimerkin avulla näytetään, miten ehdotettuja algoritmeja voidaan soveltaa skenaarioon, jossa on tilastollisia kanavavirheitä.

Seuraavaksi ehdotetaan kahta hajautettua algoritmia monisoluisiin moniantennilähetyksellä toimiviin verkkoihin. Tarkastelun kohteena olevat optimointiongelmat ovat: P1) lähetyksen kokonaistehon minimointi käyttäjäkohtaisten minimi-SINR-rajoitteiden mukaan ja P2) SINR:n tasapainottaminen tukiasemien kokonaislähetystehorajoitusten mukaisesti. Ongelma P1 on erittäin kiinnostava, kun pyritään kehittämään mahdollisimman pienen lähetystehon vaativa lähetysstrategia, joka pystyy takaamaan käyttäjien palvelun laadun. Ongelma P2 on hyödyllinen tiukasti tehorajoitetussa järjestelmässä, koska se tarjoaa tasapuolisuutta käyttäjien välillä. Molempien ongelmien hajautetut algoritmit johdetaan konsensusperusteisen vuorottelevan kertoimien suuntaustavan avulla. Lopuksi tarkastellaan kahden langattoman operaattorin välisen spektrinjaon ongelmaa dynaamisessa moniantennilähetystä käyttävässä verkkoympäristössä. Spektrinjakoongelmaa mallinnetaan käyttämällä kahden osapuolen välistä neuvottelua stokastisen optimoinnin näkökulmasta. Tähän ongelmaan ehdotetaan ratkaisuksi sekä keskitettyä että hajautettua resurssien allokoinnin algoritmia. Hajautettu algoritmi sopii paremmin spektrin jakamiseen operaattorien välillä, koska se vaatii vähemmän kontrollisignalointia. Numeeriset tulokset osoittavat, että ehdotetulla hajautetulla algoritmilla saavutetaan lähes sama suorituskyky kuin keskitetyllä algoritmillakin.

Asiasanat: dynaaminen kontrolli, globaali (ei-konveksi) optimointi, hajautettu optimointi, radioresurssien hallinta, spektrin jakaminen, vakaa resurssiallokaatio

To my family

Preface

This thesis is a result of research work that was carried out at the Centre for Wireles Communications (CWC) and the Department of Communications Engineering, University of Oulu, Finland.

Without the continuous suggestions, guidance, and encouragement that I received from many individuals, this thesis would not have been possible. First of all, I would like to thank my supervisor Professor Matti Latva-aho for giving me the opportunity to pursue my PhD studies in the Radio Access Technologies group at CWC. He gave me complete freedom in choosing my research topic, and directed me in the high quality research projects. The continuous encouragement and guidance from him over the years has been invaluable, and I am very grateful to him for that.

I am deeply grateful to my advisor Adjunct Professor Marian Codreanu for his suggestions, in-depth guidance, and all of his advice. Without his comments, advice, and encouragement, I would not have come this far. I also wish to thank Adjunct Professor Premanandana Rajatheva for his invaluable support since my M. Eng. studies. I also wish to thank the reviewer of my thesis, Professor Wei Yu from the University of Toronto, Toronto, Canada, and Professor Carlo Fischione from the KTH Royal Institute of Technology, Stockholm, Sweden, for their constructive comments and feedback. My gratitude also goes to Dr. Ari Pouttu and Dr. Harri Posti, the director of the CWC at the time of my research work, and Professor Markku Juntti, the Dean of University of Oulu Graduate School, for providing such a flexible and encouraging working environment.

The research related to this thesis was conducted on the projects: LOCON, CRUCIAL, 5Gto10G, and CORE++. I would like to thank the managers of these projects Dr. Pekka Pirinen, Dr. Jouko Leinonen, and Tuomo Hänninen. These projects were funded by Finnish Funding Agency for Technology and Innovation, Nokia Networks, Nokia Siemens Networks, Elektrobit, and many other industrial partners. I was also fortunate to receive personal research grants from the Nokia Foundation and Tauno Tönningin Säätiö. I would like to acknowledge them all.

I would like to thank all my past and present colleagues at the CWC, including, Dr. Animesh Yadav, Ayotunde Laiyemo, Dr. Antti Tölli, Dr. Carlos Lima, Dr. Fatih Bayramoglu, Hamidreza Bagheri, Dr. Hirley Alves, Ikram Ashraf, Kalle Lähetkangas, Kien-Giang Nguyen, Dr. Madhusanka Liyanage, Markus Leinonen, Markku Jokinen, Dr. Mehdi Bennis, Mohammed ElBamby, Mohammad Majidzadeh, Dr. Pedro Nardelli, Praneeth Jayasinghe, Prasanth Karunakaran, Dr. Ratheesh Kumar, Tachporn Sanguanpuak, Tamoor Syed, Trung Kien Vu, and many others. They created a supportive and cheerful working environment, which I enjoyed over the years. Special thanks go to Ayswarya Padmanabhan, Bidushi Barua, Dr. Chathuranga Weeraddana, Ganesh Venkatraman, Dr. Keeth Jayasinghe, Inosha Sugathapala, Manosha Kapuruhamy, Pawani Porambage, Sumudu Samarakoon, Dr. Suneth Namal, and Uditha Wijewardhana, whose friendship made the working environment at CWC and my life in Oulu more enjoyable and remarkable. I also like to thank the whole administrative staff from CWC, specifically Anu Niskanen, Elina Komminaho, Hanna Saarela, Jari Sillanpää, Kirsi Ojutkangas, Eija Pajunen, and many others.

I am very grateful to all the members of Nepalese community in Oulu. In particular, I would like to thank all my friends who moved to Oulu from 2010 to 2014, including Saju, Dipti, Bikram, Smita, Bijay, Pragya, Binod, Dinesh, Udgum, and the joyful kids Aarush and Ahana. The wonderful moments that I shared with them throughout these years are unforgettable. Additionally, I would also like to thank friends Bhagya, Buddhika, Dilani, Kanchana, and little friends Aditya, Sahas, and Ashini.

I want to express my deepest gratitude to my mother Keshari Joshi and father Om Krishna Joshi for their love and support provided throughout my life. I would like to thank my brother Sahi for his support. I would also like to express my gratitude to all my closest relatives; thank you all for creating such a mutually caring environment. To my sweet daughter Kesä (Suniti), thank you for bringing such joy into my life, and reminding me of the important things in life. Last, but not least, I would like to thank my wife Sujita for her patience, encouragement, and support.

Satya Krishna Joshi

Abbreviations

Roman-letter notations

$\mathbf{a}_i$	Vertex of a rectangle along the ith standard basis vector used in deriving improved lower bound for the BB algorithm for WSRMax
$a_{nl}(t)$	Amount of data of the lth user of the nth BS allowed to enter the network layer from the transport layer at BS n during time slot t
$\bar{a}_{nl}(t)$	Time average of admitted data rate $a_{nl}(t)$ up to time slot t
b_{ns}	Binary variable that is used to indicate the assignment of the sth subchannel to the nth operator
$\mathbf{b}_n$	Vector of b_{ns} used to indicate the assignment of subchannels to the nth operator, i.e., $\mathbf{b}_n = [b_{n1}, \ldots, b_{nS}]^{\mathrm{T}}$
B	Amount of spectrum band that each operator put in a common spectrum pool in the dynamic RA
$\mathbf{c}_{jl}$	Small scaling fading coefficient between the transmitter of the jth data stream to the receiver of the lth data stream
d_0	Far field reference distance that is used in the channel model
d_l	Information symbol associated with the lth data stream
d_{jl}	Distance from the transmitter of the jth data stream to the receiver of the lth data stream
$\hat{d}$	Coefficient of a monomial function
D_{BS}	Distance between the adjacent BSs
$\hat{\mathbf{e}}_i$	ith standard basis vector
$\mathbf{e}_{jl}$	Channel estimation error between the transmitter node of the jth data stream and the receiver node of lth data stream
$\mathbf{e}_l$	Channel estimation error vector obtained by stacking $\mathbf{e}_{jl}$ for all $j = 1, \ldots, L$, i.e., $\mathbf{e}_l = [\mathbf{e}_{1l}^{\mathrm{T}}, \ldots, \mathbf{e}_{Ll}^{\mathrm{T}}]^{\mathrm{T}}$
$f_0(\cdot)$	Function used to represent the (negative) weighted sum-rate
$\tilde{f}(\cdot)$	Extended-value extension of function $f_0(\cdot)$
$\check{f}(\cdot)$	Objective function of the auxiliary optimization problem used to update SINR and power vectors in the worst-case WSRMax

$f^{\mathrm{WSR-WC}}(\cdot)$	Function used to represent the achieved weighted sum-rate considering the outage in the worst-case WSRMax
$f_n(\cdot)$	Function (extended-value extension) used to represent the total transmit power of the nth BS in the sum-power minimization
$\underline{g}_{ll}$	Minimal channel gain between tran(l) and rec(l), over a channel uncertainty set, along the lth beamforming direction
$\bar{g}_{jl}$	Maximal channel gain between tran(j) and rec(l), over a channel uncertainty set, along the jth beamforming direction
$g(\cdot)$	Function used to represent the ratio between the maximal CSI errors and estimated channel gain.
$g_{nl}(\cdot)$	Function used to represent the utility of the lth user of the nth BS
$\mathbf{h}_{jl}$	Channel vector between the transmitter node of the jth data stream and the receiver node of the lth data stream
$\hat{\mathbf{h}}_{jl}$	Estimated value of the channel vector $\mathbf{h}_{jl}$
$\mathbf{h}_{nl,s}(t)$	Channel vector between the nth BS and its lth user in subchannel s during time slot t
$\mathbf{I}$	Identity matrix
I_{jl}	Out-of-cell interference power from the jth data stream to the receiver of lth data stream
$I_n(\cdot)$	Indicator function used to represent the feasibility of the subproblem associated with the nth BS in SINR balancing
L	Number of data streams (i.e., users) in the system
$\check{L}_n$	Number of users associated with the nth BS
L_k	Lower bound on the optimal value at the kth step of the BB algorithm for WSRMax
$L_\rho(\cdot)$	Function used to represent the augmented Lagrangian
$\tilde{L}(\cdot)$	Lyapunov function
$\mathbf{m}_l$	Transmit beamformer associated with the lth data stream
$\mathbf{m}$	Vector obtained by stacking the transmit beamformer $\mathbf{m}_l$ for all $l = 1, \ldots, L$, i.e., $\mathbf{m} = [\mathbf{m}_1^{\mathrm{T}}, \ldots, \mathbf{m}_L^{\mathrm{T}}]^{\mathrm{T}}$
$\tilde{\mathbf{M}}_l$	Matrix constructed by taking the outer product of beamforming vector $\mathbf{m}_l$, i.e., $\tilde{\mathbf{M}}_l = \mathbf{m}_l\mathbf{m}_l^{\mathrm{H}}$
$\mathbf{M}_n$	Beamformer matrix obtained by concatenating the transmit beamformers associated with the nth BS.

$\mathbf{m}_{nl,s}(t)$	Transmit beamformer associated with the lth user of nth BS in subchannel s during time slot t; the time slot index t is dropped sometimes for simplicity
$\breve{\mathbf{m}}_n$	Vector obtained by stacking the transmit beamformers associated with the nth operator in dynamic RA
$\mathbf{M}_{nl,s}$	Matrix constructed by taking the outer product of the beamforming vector $\mathbf{m}_{nl,s}$, i.e., $\mathbf{M}_{nl,s} = \mathbf{m}_{nl,s}\mathbf{m}_{nl,s}^{\mathrm{H}}$
$\mathbf{M}_{nl}$	Beamforming matrix associated with the lth user of nth BS obtained by concatenating $\mathbf{M}_{nl,s}$ for all $s \in \mathcal{S}$
$\underline{n}$	Opponent of the nth operator, i.e., for $n \in \{1,2\}$, when $n = 1$, $\underline{n} = 2$, and when $n = 2$, $\underline{n} = 1$
n_l	Circular symmetric complex Gaussian noise at the receiver of the lth data stream
$n_{nl,s}(t)$	Circular symmetric complex Gaussian noise at the receiver of lth user of nth BS in subchannel s during time slot t
N	Number of BSs
N_0	Power spectral density of complex Gaussian noise
p_l	Transmit power associated with the lth data stream
$\mathbf{p}$	Transmit power vector obtained by stacking p_l for all $l = 1, \ldots, L$, i.e., $\mathbf{p} = [p_1, \ldots, p_L]^{\mathrm{T}}$
$p_{nl,s}(t)$	Transmit power associated with the lth user of BS n in subchannel s during time slot t
$p_n^{\max}$	Transmit power constraint of the nth BS
$p(\cdot)$	Optimal value function of the auxiliary optimization problem used in SINR balancing
$\tilde{p}(\cdot)$	Function associated with the optimal value function $p(\cdot)$ in SINR balancing
$q_n(t)$	Per-unit price of the spectrum for the opponent of the nth operator during time slot t in the dynamic RA
$q^{\max}$	Per-unit price constraint of the spectrum in the dynamic RA
$\mathbf{Q}_{jl}$	Complex Hermitian positive definite matrix that specifies the size and shape of a complex ellipsoid consisting of the channel estimation errors between tran(j) and rec(l)
$Q_{nl}(t)$	Network layer queue backlog of the lth user of nth BS during time slot t

$\overline{Q}_{nl}$	Time average of queue backlog $Q_{nl}(t)$
R_{BS}	Radius of the transmission region around a BS
R_{int}	Radius of the region around the BS, in which it interferes with the users that belong to other BSs
$r_l(\cdot)$	Function used to represent the transmission rate of the lth data stream
$R_l(\cdot)$	Function used to represent the supported data rate of the lth data stream in the presence of CSI errors in the worst-case WSRMax
$r_{nl}(t)$	Transmission rate of the lth user of the nth BS during time slot t
S	Number of subchannels
$S_n(t)$	Number of subchannels allocated to the nth operator during time slot t
$\mathbf{s}_n$	Antenna signal vector transmitted by the nth BS
T	Number of transmit antennas
$u_{k,nl}$	Lagrange dual variable associated with the (k,nl)th consensus constraint (of the out-of-cell interference value) in sum-power minimization and SINR balancing; variable $u_{k,nl}$ is associated with the kth BS
$\mathbf{u}_n$	Vector obtained by collecting Lagrange dual variables $\{u_{n,bl}\}$ that are associated with the nth BS in sum-power minimization and SINR balancing
$u_{nl,s}$	Variable used to reformulate the auxiliary optimization problem (that update subchannels and beamformers allocation) as a DC program in the dynamic RA
$\mathbf{u}_{nl}$	Vector of $u_{nl,s}$, i.e., $\mathbf{u}_{nl} = [u_{nl,1}, \ldots, u_{nl,S}]^{\mathrm{T}}$
U_k	Upper bound on the optimal value at the kth step of the BB algorithm for WSRMax
$\overline{U}_n$	Time average profit of the nth operator
U_n^0	Utility gain of the nth operator received before sharing its spectrum with other operator
v_n	Lagrange dual variable associated with the nth consensus constraint (of the SINR value) in SINR balancing
$\mathbf{v}_n$	General purpose variable. In sum-power minimization and SINR balancing, this is a scaled version of $\mathbf{u}_n$, i.e., $\mathbf{v}_n = (1/\rho)\mathbf{u}_n$. In

	the worst-case WSRMax, variable $\mathbf{v}_l$ is used for the transmit beamforming direction associated with the lth data stream.
$\mathbf{v}$	Vector obtained by stacking the transmit beamforming direction vector $\mathbf{v}_l$ for all $l = 1, \ldots, L$, i.e., $\mathbf{v} = [\mathbf{v}_l^{\mathrm{T}}, \ldots, \mathbf{v}_L^{\mathrm{T}}]^{\mathrm{T}}$
$\mathbf{V}_l$	Variable used to write the LMI constraint of the auxiliary optimization problem in the worst-case WSRMax
V	Trade-off parameter in a drift-plus penalty expression that is used in the dynamic RA
w_s	Bandwidth of the sth subchannel
$W_n(t)$	Virtual queue associated with the nth operator during time slot t in the dynamic RA
$x_{k,nl}$	Auxiliary variable associated with the kth BS, and it is used to denote the out-of-cell interference power from nth BS to rec(l) in sum-power minimization and SINR balancing
$\mathbf{x}_n$	Vector obtained by collecting variables $\{x_{n,bl}\}$ that are associated with the nth BS in sum-power minimization and SINR balancing
$\tilde{\mathbf{x}}_l$	Subvector of $\mathbf{x}_n$ associated with the lth data stream in sum-power minimization and SINR balancing
x_{ns}	Copy of variable b_{ns}, i.e., $x_{ns} = b_{ns}$ used in the dynamic RA
$x_n^{\mathrm{in}}(t)$	Input to the virtual queue $X_n(t)$ during time slot t in the dynamic RA
$x_n^{\mathrm{out}}(t)$	Output to the virtual queue $X_n(t)$ during time slot t in the dynamic RA
$X_n(t)$	Backlog of virtual queue associated with the nth operator during time slot t in the dynamic RA
y_l	Signal received at the receiver node of the lth data stream
$y_n^{\mathrm{in}}(t)$	Input to the virtual queue $Y_n(t)$ during time slot t in the dynamic RA
$y_n^{\mathrm{out}}(t)$	Output to the virtual queue $Y_n(t)$ during time slot t in the dynamic RA
$Y_n(t)$	Backlog of the virtual queue associated with the nth operator during time slot t in the dynamic RA
z_{nl}^2	Out-of-cell interference power from the nth BS to rec(l) in sum-power minimization and SINR balancing
$\mathbf{z}_n$	Vector of z_{nl}, i.e., $\mathbf{z}_n = [z_{n1}, \ldots, z_{nL}]^{\mathrm{T}}$

$z_{nl,s}$	Variable used to reformulate the auxiliary optimization problem (that updates subchannels and beamformer allocation) as a DC program in the dynamic RA
$\mathbf{z}_{nl}$	Vector of $z_{nl,s}$, i.e., $\mathbf{z}_{nl} = [z_{nl,1}, \ldots, z_{nl,S}]^{\mathrm{T}}$

Mathcal-style notations

$\mathcal{A}$	Subset of the nonnegative orthant $\mathbb{R}_+^L$
$\mathcal{B}_k$	Set of rectangles obtained at the kth step of the BB algorithm for WSRMax
$\mathcal{C}_n$	Set of feasible points of the subproblem associated with the nth BS in SINR balancing
$\check{\mathcal{C}}$	Set of small scale fading coefficients
$\mathcal{CN}(\mathbf{m}, \mathbf{\Sigma})$	Circular symmetric Gaussian random vector with mean $\mathbf{m}$ and covariance matrix $\mathbf{\Sigma}$
$\mathcal{E}_{jl}$	Complex ellipsoid that contains channel estimation errors between tran(j) and rec(l)
$\mathcal{E}_{jl}(\kappa)$	κ-confidence ellipsoid that is used to model the Gaussian distributed CSI errors between tran(j) and rec(l)
$\mathcal{G}$	Set of achievable SINR values in the WSRMax
$\mathcal{I}_{\mathrm{int}}(n)$	Set of data streams that are subject to out-of-cell interference from the nth BS
$\mathcal{L}$	Set of data streams
$\mathcal{L}(n)$	Set of data streams associated with the nth BS
$\mathcal{L}_{\mathrm{int}}$	Set of data streams that are subject to out-of-cell interference
$\mathcal{L}_{\mathrm{int}}(n)$	Set of data streams associated with BS n that are subject to the out-of-cell interference
$\mathcal{M}_n$	Set of feasible points of the subproblem associated with the nth BS in sum-power minimization
$\mathcal{N}$	Set of BSs
$\mathcal{N}_{\mathrm{int}}(l)$	Set of BSs that interfere (out-of-cell interference) rec(l)
$\mathcal{O}_n$	Set of feasible points of the subproblem associated with the nth operator in the dynamic RA
$\mathcal{Q}_{\mathrm{init}}$	Rectangle used to initialize the BB algorithm for WSRMax
$\mathcal{Q}$	Subset of the rectangle $\mathcal{Q}_{\mathrm{init}}$, i.e., $\mathcal{Q} \subseteq \mathcal{Q}_{\mathrm{init}}$

$\mathcal{Q}_k$	Rectangle chosen, for branching, at the kth step of the BB algorithm
$\bar{\mathcal{Q}}^\star$	Rectangle used to obtain the improved lower bound on the optimal value in the BB algorithm for WSRMax
$\mathcal{S}$	Set of subchannels
$\mathcal{S}(n,t)$	Set of subchannels allocated to the nth operator during time slot t
$\mathcal{S}_\kappa$	Optimal solution set obtained by using the κ-confidence ellipsoid in the worst-case WSRMax

Greek-letter notations

α_n	Auxiliary variable associated with the SINR of nth BS data streams in SINR balancing
α	Parameter used to limit the domain of an SINR value in a signomial program for WSRMax
$\hat{\alpha}_l$	Exponent of the lth variable of the monomial function
$\chi_n(\cdot)$	Auxiliary function of subchannels associated with the nth operator in the dynamic RA
$\triangle(\cdot)$	Drift of a Lyapunov function from one slot to the next
$\hat{\chi}_n(\cdot)$	Convex approximation of the auxiliary function $\chi_n(\cdot)$
β_l	Nonnegative weight associated with the lth data stream
ϵ	Accuracy required for the BB algorithm for WSRMax
ϵ_b	Accuracy required for the bisection search method to find an improved lower bound in the BB algorithm for WSRMax
ϵ_γ	Required accuracy of SINR for the signomial program, that updates SINR and power vectors in the worst-case WSRMax
ϵ_g	Accuracy required for the bracketing method in SINR balancing
η	Path loss exponent
γ	Auxiliary variable associated with the SINRs of all data streams in SINR balancing
γ_l	Auxiliary variable associated with the SINR of lth data stream
$\boldsymbol{\gamma}$	Vector of γ_l, i.e., $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_L]^{\mathrm{T}}$
$\gamma_{l,\max}$	Maximum element of rectangle $\mathcal{Q}$ along the lth axis
$\boldsymbol{\gamma}_{\max}$	Vector of $\gamma_{l,\max}$, i.e., $\boldsymbol{\gamma}_{\max} = [\gamma_{1,\max}, \ldots, \gamma_{L,\max}]^{\mathrm{T}}$
$\gamma_{l,\min}$	Minimum element of rectangle $\mathcal{Q}$ along the lth axis

$\boldsymbol{\gamma}_{\min}$	Vector of $\gamma_{l,\min}$, i.e., $\boldsymbol{\gamma}_{\min} = [\gamma_{1,\min}, \ldots, \gamma_{L,\min}]^{\mathrm{T}}$
$\hat{\gamma}_l$	Parameter associated with the lth data stream used to form a GP in the worst-case WSRMax
$\hat{\boldsymbol{\gamma}}$	Vector of $\hat{\gamma}_l$, i.e., $\hat{\boldsymbol{\gamma}} = [\hat{\gamma}_1, \ldots, \hat{\gamma}_L]^{\mathrm{T}}$
$\Gamma_l(\cdot)$	Function used to represent the SINR of the lth data stream
$\tilde{\Gamma}_l(\cdot)$	Function used to represent the lower bound on the SINR of the lth data stream with channel uncertainties
λ_n	Scaled version of variable v_n, i.e., $\lambda_n = (1/\rho)v_n$
λ_{ns}	Lagrange dual variable associated with the (ns)th consensus constraint (of subchannels) of the auxiliary optimization problem in the dynamic RA
μ_{jl}	Variable used to write the LMI constraint of the auxiliary optimization problem in the worst-case WSRMax
$\mu_n(t)$	Auxiliary variable associated with the profit of nth operator during time slot t in the dynamic RA
$\bar{\mu}_n$	Time average of auxiliary variable $\mu_n(t)$
$\phi_{\min}(\cdot)$	Function that returns the optimal value of the (negative) weighted sum-rate over a set $\mathcal{Q}$
$\phi_{\mathrm{lb}}(\cdot)$	Lower bound function of the BB algorithm for WSRMax
$\phi_{\mathrm{ub}}(\cdot)$	Upper bound function of the BB algorithm for WSRMax
$\phi_{\mathrm{lb}}^{\mathrm{Basic}}(\cdot)$	Basic lower bound function of the BB algorithm for WSRMax
$\phi_{\mathrm{ub}}^{\mathrm{Basic}}(\cdot)$	Basic upper bound function of the BB algorithm for WSRMax
$\phi_{\mathrm{lb}}^{\mathrm{Imp}}(\cdot)$	Improved lower bound function of the BB algorithm for WSRMax
$\phi_{nl}(\cdot)$	Function used to reformulate the auxiliary optimization problem (that update subchannels and beamformers allocation) as a DC program in the dynamic RA
$\hat{\phi}_{nl}(\cdot)$	Convex approximation of the auxiliary function $\phi_{nl}(\cdot)$
$\Phi_n(\cdot)$	Auxiliary function used to represent the objective function of the DC program in the dynamic RA
$\psi_{nl}(\cdot)$	Function used to reformulate the auxiliary optimization problem (that update subchannels and beamformers allocation) as a DC program in the dynamic RA
ρ	Parameter used in the augmented Lagrangian function
σ_e^2	Variance of the least-square channel estimator error

σ_l^2	Variance of the white Gaussian noise at the receiver of lth data stream
$\mathbf{\Sigma}$	Covariance matrix of the Gaussian random vector
θ	Subgradient step size used in the dual decomposition method for sum-power minimization
$\theta(\kappa)$	Inverse CDF value of chi-square distribution with $2T$ degree of freedom for a given value of κ.
$\mathbf{\Theta}(t)$	Vector of the actual and virtual queues during time slot t in the dynamic RA
ξ_{jl}	Radius of a ball that contains channel estimation errors between tran(j) and rec(l)
$\zeta_n(\cdot)$	Function associated with the nth operator, introduced in order to relax the binary constraint in the dynamic RA
$\hat{\zeta}_n(\cdot)$	Convex approximation of the auxiliary function $\zeta_n(\cdot)$

Mathematical-operator notations and symbols

cond($\mathcal{Q}$)	Condition number of the rectangle $\mathcal{Q}$
Rank($\mathbf{X}$)	Rank of the matrix $\mathbf{X}$
rec(l)	Receiver node of the lth data stream
size($\mathcal{Q}$)	Maximum half length of the sides of the rectangle $\mathcal{Q}$
Trace($\mathbf{X}$)	Trace of the matrix $\mathbf{X}$
tran(l)	Transmitter node (i.e., BS) of the lth data stream
vec($\mathbf{M}$)	Vector obtained by stacking below each other the columns of the matrix $\mathbf{M}$
vol($\mathcal{Q}$)	Volume of the rectangle $\mathcal{Q}$
$\lvert x\rvert$	Absolute value of the complex number x
$\lvert\mathcal{X}\rvert$	Cardinality of the set $\mathcal{X}$
$\lVert\mathbf{x}\rVert_2$	ℓ_2-norm of the vector $\mathbf{x}$
$\mathbf{X}^{\mathrm{T}}$	Transpose of the matrix $\mathbf{X}$
$\mathbf{X}^{\mathrm{H}}$	Hermitian (complex conjugate) transpose of the matrix $\mathbf{X}$
$\mathbb{C}^n$	Complex n-vectors
$\mathbb{R}$	Real numbers
$\mathbb{R}_+$	Nonnegative real numbers
$\mathbb{R}_+^n$	Nonnegative real n-vectors

$(x)^+$	Denotes the maximum of x and 0, i.e., $\max(x, 0)$
$(\cdot)^\star$	Solution of an optimization problem
$\triangleq$	Defined to be equal to

Acronyms

ADMM	Alternating direction method of multipliers
BB	Branch and bound
BS	Base station
CDF	Cumulative distribution function
CSI	Channel state information
DC	Difference of convex
DDA	Dual decomposition based distributed algorithm
DIA	Distributed interference alignment
GAD	Generalized asynchronous distributed
GP	Geometric program
LB_{Basic}	Basic lower bound
LB_{Imp}	Improved lower bound
LMI	Linear matrix inequality
LMMSE	Linear minimum mean square error
LTE	Long-term evolution
MISO	Multiple-input single-output
MIMO	Multiple-input multiple-output
MSE	Mean square error
NP	Non-deterministic polynomial-time
OFDMA	Orthogonal frequency-division multiple access
QoS	Quality-of-service
RA	Resource allocation
SDP	Semidefinite program
SDR	Semidefinite relaxation
SINR	Signal-to-interference-plus-noise ratio
SISO	Single-input single-output
SNR	Signal-to-noise ratio
SOCP	Second-order cone programming
KKT	Karush-Kuhn-Tucker
UB_{Basic}	Basic upper bound

WiMAX	Worldwide interoperability for microwave access
WLAN	Wireless local area network
WMMSE	Weighted sum mean square error minimization
WSR	Weighted sum-rate
WSRMax	Weighted sum-rate maximization
ZFBF	Zero-forcing beamforming

Contents

1 Introduction

Due to the proliferation of smart wireless devices, like, tablets and smart-phones, and the popularization of many applications that require higher data rates, the demand for wireless services is ever increasing [1, 2]. It is predicted that over the next few years, mobile-connected devices, including machine-to-machine modules [3], will surpass the world population [2]. However, the radio spectrum is a finite natural resource; thus to accommodate the increasing data traffic, wireless cellular networks have been evolving continuously leading to the deployment of various radio access technologies [4]. In spite of these technologies, it has been challenging to handle the ever-growing data traffic with the current wireless networks [5–7].

The successful deployment of future-generation wireless networks will heavily rely on their ability to provide highly efficient and flexible radio resource allocation for the users [8]. Interestingly, the use of multiple antennas at the transmitter and/or receiver and beamforming techniques have been identified as one of the key technology enablers for meeting these challenges [9]. Multiple antennas can significantly improve the capacity of a wireless link, compared to a single-antenna case. Multiple-antenna techniques are considered as an essential feature for many wireless communication standards such as WLAN [10], WiMAX [11], and LTE [12]. However, the use of multiple antennas at the transmitter and/or receiver is fairly well understood only for a point-to-point communication link [9, 13–17]. In a multicell downlink network, coordination between transmitters is required to jointly optimize the transmit beamformers to efficiently utilize radio resources. Thus, developing resource allocation algorithms for multicell multiple-antenna networks is challenging. In fact, in these networks many coordinated beamforming problems of interest, such as the maximization of weighted sum-rate, proportional fairness, and harmonic mean utilities subject to maximum transmit power constraints are known to be NP-hard [18].

Multiple-antenna techniques can provide higher spectral efficiency and enhance the reliability of wireless links, only when the channel state information (CSI) of users is accurately known [9, 13]. However, acquiring CSI is a difficult task in a time-varying environment [19, 20]. Pilot-symbol (training symbols)

assisted channel estimation is one of the popular approaches for estimating CSI [19]. In frequency division duplex systems, the receiver estimates the channel, and it is usually quantized and fedback to the base stations (BS) [21, 22]. In time division duplexing systems the CSI is estimated by the BSs, during uplink transmission, assuming channel reciprocity [23]. Therefore, in practice the CSI is not perfect due to several factors, such as inadequate number of training symbols, quantization errors, outdated CSI feedback, etc. Consequently, the design of resource allocation algorithms that can adapt to time-varying propagation, and provide robustness against CSI errors is challenging and also essential for emerging wireless technologies.

Due to the scarcity of the radio spectrum, along with the use of multiple antennas and beamforming techniques, spectrum sharing between mobile operators is becoming essential [24–26]. In current wireless communications networks, the radio spectrum is divided into disjoint blocks which are assigned (licensed) to different operators on an exclusive basis. The assignment of exclusive spectrum bands to the operators gives each operator the right to control their spectrum bands, and it has well-known advantages including good interference management and guaranteed quality-of-service. However, it often leads to low spectrum utilization [27, 28]. By sharing the spectrum instead of using it individually, wireless operators can improve spectral efficiency, enhance coverage, and also reduce their operating expenditure [26]. Although such spectrum sharing capabilities between operators would yield significant performance gains, detailed studies into it still remains limited.

In this thesis, we study several resource allocation problems addressing the above challenges (e.g., imperfect CSI, distributed implementation, spectrum sharing between operators, etc.) in multi-input single-output (MISO) downlink cellular networks. The work is categorized into four parts, and are briefly described in Section 1.2.

1.1 Literature review

In this section we provide a review of the relevant literature that is associated with the scope of this thesis. In Section 1.1.1, we discuss optimization techniques for the nonconvex weighted sum-rate maximization (WSRMax) problem. Robust resource allocation techniques for imperfect CSI are discussed in Section 1.1.2. A

review of distributed resource allocation methods is presented in Section 1.1.3. Finally, in Section 1.1.4 we discuss the key existing works on spectrum sharing methods between cellular operators.

1.1.1 Resource allocation methods for nonconvex problems

Recognizing or reformulating a problem as a convex optimization problem has great advantages. It is because this sort of problem can be solved very reliably and efficiently, using interior-point methods or other special methods for convex optimization [29]. However, in wireless communications systems many resource allocation problems of great importance are nonconvex. For example, in the interference channel, the general problem of WSRMax, proportional fairness utility maximization, harmonic-rate utility maximization, and min-rate utility maximization are nonconvex, and known to be NP-hard [18, 30].

An exhaustive search method, such as a grid search, is a straightforward approach to find a global optimal solution for a nonconvex problem [31]. However, the exhaustive search method requires exponential complexity, and it becomes computationally intractable with an increase in problem size. A better approach is to apply branch and bound (BB) and polyblock methods [32, 33] to solve nonconvex problems. Both BB and polyblock methods maintain provable upper and lower bounds on the global optimal value for a nonconvex problem, and they terminate when the difference between the upper and lower bounds is smaller than a given threshold. Unfortunately, the BB and polyblock methods can be slow with increase in the problem size (i.e., for large networks) [32, 33]. Therefore, a variety of local optimization techniques, such as alternating optimization [34, 35], sequential convex programming [35], dual relaxation [29, Ch. 5], and difference of convex (DC) programming [36] are frequently used to find a suboptimal solution for nonconvex problems.

Weighted sum-rate maximization

The WSRMax problem is central to many network optimization methods [37]. For example, it appears in network utility maximization [38–41], design of the cross-layer control policies [42, 43], multi-user scheduling [44–46], power and rate control [30, 31, 47–53], etc. In the context of an orthogonal multiple-access

channel [9, 54] the WSRMax problem is convex and it can be solved efficiently. However, with an interference channel, the WSRMax problem is nonconvex [37]. In fact, it is known to be an NP-hard problem [18, 30].

In the case of single-input single-output (SISO) systems, the problem of WSRMax by using global optimization approaches has been addressed in [55–60]. Specifically, in [55] and [56] BB based algorithms are proposed by expressing the WSRMax problem in a form of DC programm [61]. Since DC programming is a central part of these algorithms, they are not adaptable for solving problems whose objective function is not convertible to a DC form (e.g., multicast wireless networks) [59]. A bit loading approach in conjunction with the BB method is used in [57] in the context of digital subscriber lines. At each iteration of the algorithm in [57], the search region is discretized to find the lower and upper bounds on the optimal value of the problem; thus, this algorithm does not have strict control over the accuracy of the solution. In [58, 59], the monotonicity of the users' rate in the signal-to-interference-plus-noise ratio (SINR) values is exploited, and a BB based algorithm is derived. A different approach has been considered in [60], where the WSRMax problem is transformed into a linear fractional program [62], and an outer polyblock approximation method [33, 63] is used to solve it.

In multiple-antenna systems, the decision variables space is large, i.e., the joint optimization of transmit beamforming patterns, transmit powers, and link activation are required. Therefore, designing global optimal methods for the WSRMax problem in multiple-antenna systems is a challenging task [64]. In [65] a global optimization algorithm is proposed by using the outer polyblock approximation method along with the result of [66], which proves that any point in a rate region can be achieved by choosing beamforming vectors that are linear combinations of the zero-forcing and the maximum-ratio transmission beamformers. The algorithm proposed in [65] is for a two-user case and its extension for more than two-users is nontrivial. In [67], a framework to obtain a globally optimal solution is proposed by jointly utilizing the outer polyblock approximation method and rate profile techniques [68–70]. Outer polyblock approximation based algorithms have also been derived in [71–73]. Unfortunately, polyblock based algorithms require a large number of iterations for solving the problem to a high degree of accuracy compared to BB based algorithms [74, 75].

As the WSRMax problem is NP-hard, the convergence speed of global optimization algorithms can be slow for large networks [32, 33, 76]. Hence, these algorithms may not be suitable for practical implementation in large networks. Therefore, fast-converging algorithms, even though suboptimal, may be more desirable in practice.

In the case of SISO systems, the problem of WSRMax by using local optimization methods has been addressed in, e.g., [77–79]. Specifically, in [77] and [78] the WSRMax problem is studied in the high SINR regime. In [78] an approximated WSRMax problem in a high SINR regime is reformulated (recognized) as a geometric program (GP) [80, 81]; hence the WSRMax problem in a high SINR regime can be efficiently solved even with a large number of users. Unfortunately, in medium to low SINR regimes such approximation is inefficient. The WSRMax problem in the medium to low SINR regime is addressed in [79]; there the problem is formulated as a signomial program (SGP) [80, 81], and an iterative algorithm that solves a series of GP is proposed. In the case of SISO orthogonal frequency-division multiple access (OFDMA) networks, suboptimal algorithms for the WSRMax problem have been proposed in [82].

In [83], a single-cell multiple-input and multiple-output (MIMO) system is considered, and an iterative algorithm is proposed by exploiting uplink-downlink SINR duality [84]. In [85], a MIMO-OFDMA system is considered and an iterative algorithm utilizing alternating optimization techniques is proposed. In both works [83] and [85], receivers are equipped with linear minimum mean square error (LMMSE) filters [86], and each step of their algorithms solve an SGP to update the transmit power. In addition, each step of the algorithm in [85] involves solving a second-order-cone program (SOCP) [87]. An iterative SOCP formulation of the WSRMax problem in a multiple-input and single-output (MISO) system has been proposed in [88]. A different approach is considered in [89], where an iterative algorithm is proposed by establishing a connection between the WSRMax and weighted sum mean square error (MSE) minimization problems.

1.1.2 Robust resource allocation methods for imperfect CSI

The uncertainty of the channel at BSs can be modeled either by assuming that channel errors are random variables following a certain statistical distribution [90,

91], or by assuming that CSI errors lie in a bounded uncertainty region (e.g., ellipsoid, polyhedron, etc.) [92–99]. In the case of statistical CSI errors, robust beamformers are obtained by optimizing the average or outage performance of the system, while in the case of the bounded uncertainty model, robust resource allocation strategies are designed for the worst-case scenario. That is, robust beamformers are designed so that the system performance is optimized (and the constraints are satisfied) for all possible CSI errors within the uncertainty region.

A robust optimization method for the WSRMax problem with statistical distributed CSI errors has been investigated in [100]. Specifically, it has considered a MIMO interference broadcast channel, and an iterative algorithm was proposed by exploiting a connection between the WSRMax and weighted sum-MSE minimization problems [89]. The worst-case WSRMax problem with bounded CSI errors has been investigated in [101–103]. Specifically, the work in [101] considered a MIMO interference channel, and it also exploits the connection between the WSRMax and weighted sum-MSE minimization problems in order to arrive at a robust resource allocation algorithm. In [102], a robust algorithm for the worst-case WSRMax problem is obtained by transforming the problem into a minimization of the worst-case MSE problem. In [101] and [102], the bounded region of CSI errors is modeled by using an ellipsoid. To generalize the channel error model, authors in [103] assume that CSI errors lie in an uncertainty set formed by the intersection of ellipsoids. Both works [102] and [103] considered a MISO downlink system.

Various optimization criteria (e.g., transmit power minimization, MSE minimization, SINR balancing, etc.) in the presence of CSI errors have also been studied in the literature. The problem of transmit power minimization subject to minimum SINR constraint of each user has been studied in [96, 98, 104, 105]; instead of SINR constraints as a QoS measure, maximum MSE errors as QoS constraints are considered in [105, 106]. The problem of MSE minimization subject to power constraint has been addressed in [90, 106, 107]. Reference [108] considers an SINR balancing problem, and an SINR maximization problem is considered in [109]. All the above works [90, 96, 98, 100–109] have focused on deriving suboptimal solutions in the presence of CSI errors. An optimal robust resource allocation technique in the presence of bounded CSI errors, by using the branch and cut technique [74], has been investigated in [64].

1.1.3 Distributed resource allocation methods

A study of distributed resource allocation techniques is of great interest for a network that lacks central processing unit (such as ad hoc networks [110–112]), and for a large multicell network in which a central processing is not feasible due to backhaul constraints [113, Sec. 1]. A distributed algorithm finds solution for a problem by solving smaller subproblems either in parallel or sequentially, relying on the local information by each subproblem. In general, it is an iterative procedure, unless a problem is block separable into subproblems [29, 114]. The convexity of a problem is one of the fundamental property that leads the convergence of distributed algorithms toward global optimal solution. For nonconvex problem, such as the WSRMax problem, distributed algorithms usually converge to stationary points, and in general optimality is not guaranteed [113, Sec. 9]. Distributed algorithms that converge fast to optimal (or, near-to-optimal) solutions are of great interest in practice.

Several distributed algorithms for the WSRMax problem have been studied in the literature, e.g., [110, 115–120]. Specifically, in [115] a SISO interference channel is considered, and a distributed algorithm consisting of a two phase iterative procedure has been proposed. In the first phase the algorithm in [115] drops the links that achieve negligible rates, then it operates in the high SINR regime for the remaining links. As in [115], the work in [116] considers the high SINR regime, and the algorithm development is further simplified by assuming that each transmitter uses more antennas than the total number of users in the network. The work in [116] exploits zero-forcing beamforming [121, 122] at each transmitter to derive a distributed algorithm. In [117], a pricing technique is used to arrive at a distributed implementation. Both works [116] and [117] consider a MISO interference channel. For a MISO interference broadcast channel distributed algorithms are provided in [118, 119]. The work in [118] is an extension of the work in [116], while a distributed algorithm in [119] is derived by solving Karush-Kuhn-Tucker (KKT) optimality conditions [29, Sec. 5.5.3] associated with the WSRMax problem. Distributed algorithms for the WSRMax problem in the case of MIMO systems are provided in [110, 120].

Another optimization problem that is of interest from a system-level perspective is a minimization of the total transmission power subject to minimum SINR constraint of each user [83, 123–127]. For this problem, a distributed

algorithm using the primal decomposition method [114] is proposed in [128], and a distributed algorithm using the dual decomposition method [114] is proposed in [129]. In the primal and dual decomposition methods the original problem is decomposed into multiple subproblems, and each one of the subproblems is solved (by subsystem) separately [114]. Then the subsystems coordinate to solve a master problem in order to achieve consensus among the subsystems. In both works [128] and [129], the master problem is solved by using the subgradient method [130], and hence these algorithms are sensitive to the choice of a subgradient step length. Uplink-downlink duality is exploited in [131] to arrive at a distributed algorithm, which is suitable for time-division duplex systems. The algorithm proposed in [131] is a multicell generalization of that proposed in [132] for a single-cell case. A game theoretic approach is considered in [133]. By considering imperfect CSI at BSs, a distributed algorithm using the alternating direction method of multipliers (ADMM) is proposed in [98].

In a system where a power constraint is a strict system restriction, an SINR balancing problem can provide fairness among the users [134–137]. For this problem, in [138] a distributed algorithm is proposed by using the bisection search method in conjunction with uplink-downlink SINR duality. The algorithm in [138] is a hierarchical iterative algorithm which consists of inner and outer iterations, where a bisection search is carried out in the outer iteration and uplink-downlink SINR duality is used in the inner iteration.

1.1.4 Spectrum sharing between multiple operators

Operators can share their spectrum band with each other in two basic ways [139]: orthogonal sharing and non-orthogonal sharing. In orthogonal sharing, operators are allowed to operate in each other's spectrum bands; but at any time instance one spectrum band can be used only by one operator. In contrast, in non-orthogonal sharing, multiple operators are allowed to transmit on the same spectrum band at the same time and location. Here, the operators are required to coordinate their operation and choose transmission strategies to mitigate inter-operator interference [140]. The performance of these two spectrum sharing models depends on the users' locations from BS [141].

Several orthogonal and non-orthogonal inter-operator spectrum sharing algorithms have been proposed in literature, e.g. [142–147]. Specifically in [142,

143] a time division multiple access technique is considered, and the operators are allowed to lease their unused time slots to each other. The principle of last resort sharing is adopted in [142, 143], i.e., an operator hires time slots only if its private portion is not sufficient to satisfy the QoS of its users. Moreover, the work in [143] considers sharing of infrastructure (BSs) between the operators. The use of a common spectrum pool [148, 149] to share the spectrum between operators is investigated in [144, 145, 147]. Two operators are considered in [144, 145] and distributed algorithms using non-cooperative game theory [150] are proposed. In general, the non-cooperative game theoretic approach leads to a stable operating point, and it is known as Nash equilibrium. However, in the context of spectrum sharing the Nash equilibrium point is often seen as an inefficient operating point, because the operators' performance can be further improved over it [146, 151, 152]. In [147] the co-primary shared access model [25] is adopted, and several heuristic centralized and distributed algorithms are proposed. All works in [142–147] consider single antennal transmitters and receivers.

The problem of spectrum sharing between operators in MISO wireless systems has been considered in [153–156]. Specifically, in [153] a distributed algorithm is proposed using cooperative game theory [150]. In [154] the problem is studied by using both cooperative and non-cooperative (competitive) game theoretic approach, and a significant gain by operators cooperation has been demonstrated. For cooperative spectrum sharing, various transmit beamforming techniques to manage the inter-operator interference have been proposed in [155, 156].

1.2 Aims and the outline of the thesis

The aim of this thesis is to develop optimization techniques for managing the radio resources in MISO downlink networks. In particular, we focus on developing linear transmit beamforming techniques by optimizing certain QoS features, including, spectral efficiency, fairness, and throughput. In the following we briefly outline the problems that we study and the main contributions of this thesis, which are presented in four different chapters.

Chapter 2 considers the problem of WSRMax for multicell MISO downlink networks. The problem of WSRMax has been identified as a central problem to many network optimization methods, and it is known to be NP-hard. We propose a globally optimal solution method based on the BB technique for the

NP-hard WSRMax problem. Specifically, the proposed algorithm computes a sequence of asymptotically tight upper and lower bounds, and it terminates when a difference between them falls below a pre-specified tolerance. Novel bounding techniques via conic optimization are introduced, and their efficiency is demonstrated by numerical simulations. The proposed BB based algorithm is not limited to the WSRMax problem only; it can be easily extended to maximize any system performance metric that can be expressed as a Lipschitz continuous and increasing function of an SINR ratio. Numerically, we also use the proposed algorithm to evaluate the performance loss of several suboptimal algorithms. The results are presented in [157, 158].

Chapter 3 investigates robust resource allocation methods for multicell MISO downlink networks, in the cases where there is uncertainty in the users' CSI at the BSs. Assuming a bounded ellipsoidal model for CSI errors, we propose both optimal and suboptimal algorithms for the worst-case WSRMax problem. The optimal algorithm is derived by adopting the BB algorithm proposed in Chapter 2. The main difficulty in adopting the algorithm in Chapter 2 is to define the bounding functions that are used in the BB algorithm. We provide an efficient method based on a semidefinite relaxation technique (SDR) to compute the bounding functions. As the convergence speed of the BB algorithm can be slow for large networks, we also provide a fast but possibly suboptimal algorithm using the alternating optimization technique and sequential convex programming. Numerical results are provided to show the performance of both proposed optimal and suboptimal algorithms. Through a numerical example we have also shown how our design methodology can be applied to a scenario with statistical channel errors. The results are presented in [159–161].

Chapter 4 investigates distributed resource allocation methods for multicell MISO downlink networks. The optimization problems considered are: P1) minimization of the total transmission power subject to minimum SINR constraints of each user, and P2) SINR balancing subject to the total transmit power constraint of BSs. Decentralized algorithms for both problems are derived by using consensus based ADMM. For problem P1, the proposed distributed algorithm converges to an optimal centralized solution. A heuristic method is provided, for problem P1, to find an ADMM penalty parameter that leads to faster convergence of the algorithm. Problem P2 is not amendable to a convex formulation, and in this case ADMM need not converge to an optimal point.

Numerical results are provided to demonstrate the performance of the proposed distributed algorithms in comparison to the optimal centralized solutions. The results are presented in [162–164].

Chapter 5 considers the problem of spectrum sharing between two operators in a dynamic network. We allow both operators to share (a fraction of) their licensed spectrum band with each other by forming a common spectrum band. The objective is to maximize the gain in profits for both operators by sharing their licensed spectrum bands rather than using them exclusively, while considering fairness between the operators. The notion of a two-person bargaining problem is used, and the spectrum sharing problem is cast as a stochastic optimization problem. To solve this problem, we propose both centralized and distributed dynamic control algorithms. Numerically, we show that the proposed distributed algorithm achieves almost the same performance as the centralized one. The results are documented in [165, 166].

Chapter 6 concludes the thesis and discusses possible future work directions.

1.3 The author's contribution to the publications

The thesis is based on four journal papers [157, 159, 162, 165], and five related conference papers [158, 160, 163, 164, 166]. The author of this thesis had the main responsibility for carrying out the analysis, writing the MATLAB simulation codes, generating numerical results, and writing papers [157–160, 162–166]. This was except for conference paper [158], where the second and third co-authors took the responsibility of manuscript preparation. In all other papers, including [158], the role of other authors was to provide comments, criticism, and support during the process.

In addition to papers [157–160, 162–166], the author also published a conference paper [167] which is not included in this thesis. The author has also contributed to the journal paper [168] and conference papers [161, 169, 170].

2 Global optimization method for nonconvex problem in MISO cellular networks

The main contribution of this chapter is to propose a globally optimal solution method for the WSRMax problem in multicell MISO downlink networks. The problem is known to be NP-hard [30]. We propose a method, based on the branch and bound technique [76], which solves globally the nonconvex WSRMax problem within a pre-defined accuracy ϵ. Specifically, the proposed algorithm computes a sequence of asymptotically tight upper and lower bounds for the maximum weighted sum-rate, and it terminates when the difference between the upper and lower bound is smaller than ϵ. Thus, our solution is certified to be at most ϵ-away from the global optimal value.

Our proposed branch and bound method is somewhat similar to the one used in [59] in the context of SISO systems. Unlike in [59], the WSRMax problem considered in this chapter (in MISO systems), is a higher dimensional problem, i.e., it requires finding the optimal transmit beamformers and power allocation. Reformulating this higher dimensional problem appropriately to allow applying a BB method in a reduced dimensional space, and finding efficient bounding techniques which ensures the convergence of BB algorithm are challenging tasks. The key contribution of this chapter is the reformulation of the original problem into a smaller dimensional space (i.e., the SINR domain) by noticing that transmit powers and precoders can be easily computed by using standard convex optimization techniques [132, Sec. IV-B], if one knows the SINR point. Specifically, instead of implementing a BB algorithm over a higher dimensional space (i.e., in the power and precoder domain), we search over a SINR domain to find the optimal SINR point, which maximizes the weighted sum-rate. Then standard convex optimization techniques are applied to recover the optimal powers and precoders. Furthermore, we have introduced an improved bounding technique which increases significantly the convergence speed of the BB algorithm.

The proposed method can be used to provide performance benchmarks for many existing network design problems which relys on solving the WSRMax problem. Moreover, the proposed framework is not restricted to the WSRMax

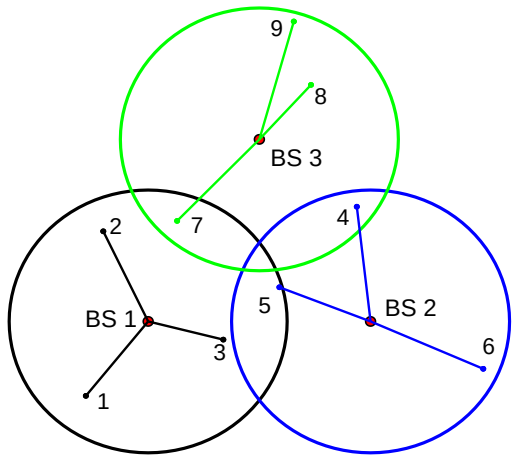


Fig. 2.1. Multicell network, $N = 3$**,** $L = 9$**,** $\mathcal{N} = \{1, 2, 3\}$**,** $\mathcal{L} = \{1, \ldots, 9\}$**,** $\mathcal{L}(1) = \{1, 2, 3\}$**,** $\mathcal{L}(2) = \{4, 5, 6\}$**, and** $\mathcal{L}(3) = \{7, 8, 9\}$**.**

problem; it can be used to maximize *any* system performance that is Lipschitz continuous and increasing function of SINR values.

2.1 System model and problem formulation

A multicell MISO downlink system, with N BSs each equipped with T transmit antennas, is considered. The set of all BSs is denoted by $\mathcal{N}$ and we label them with the integer values $n = 1, \ldots, N$. The transmission region of each BS is modeled as a disc with radius R_{BS} centered at the location of the BS[1]. A single data stream is transmitted for each user. We denote the set of all data streams in the system by $\mathcal{L}$, and we label them with the integer values $l = 1, \ldots, L$. The transmitter node (i.e., the BS) of lth data stream is denoted by $\mathrm{tran}(l)$ and the receiver node of lth data stream is denoted by $\mathrm{rec}(l)$. We have $\mathcal{L} = \cup_{n \in \mathcal{N}} \mathcal{L}(n)$, where $\mathcal{L}(n)$ denotes the set of data streams transmitted by BS n (see Fig. 2.1).

[1]For simplicity, the transmission region of each BS is modeled as a disc. This assumption is not restrictive for the work carried out in this thesis.

The antenna signal vector transmitted by nth BS is given by

$$\mathbf{s}_n = \sum_{l \in \mathcal{L}(n)} d_l \mathbf{m}_l, \tag{2.1}$$

where $d_l \in \mathbb{C}$ represents the information symbol and $\mathbf{m}_l \in \mathbb{C}^T$ denotes the transmit beamformer associated with lth data stream. We assume that the data streams for different users are independent, i.e., $\mathrm{E}\{d_l d_j^*\} = 0$ for all $l, j \in \mathcal{L}$, where $l \neq j$; and we also assume that d_l is normalized to $\mathrm{E}|d_l|^2 = 1$.

The signal received at rec(l) can be expressed as

$$y_l = d_l \mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l + \sum_{j \in \mathcal{L}, j \neq l} d_j \mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j + n_l, \tag{2.2}$$

where $\mathbf{h}_{jl}^{\mathrm{H}} \in \mathbb{C}^{1 \times T}$ is the channel vector between tran(j) and rec(l), and n_l is circular symmetric complex Gaussian noise with variance σ_l^2. The received SINR of lth data stream is given by

$$\Gamma_l(\mathbf{m}) = \frac{|\mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l|^2}{\sigma_l^2 + \sum_{j \in \mathcal{L}, j \neq l} |\mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j|^2}, \tag{2.3}$$

where we use the notation $\mathbf{m}$ to denote a vector obtained by stacking $\mathbf{m}_l$ for all $l \in \mathcal{L}$ in top of each other, i.e., $\mathbf{m} = [\mathbf{m}_1^{\mathrm{T}}, \ldots, \mathbf{m}_L^{\mathrm{T}}]^{\mathrm{T}}$.

Let β_l be an arbitrary nonnegative weight associated with data stream l, $l \in \mathcal{L}$. We consider the case where all receivers are using *single-user detection* (i.e., a receiver decodes its intended signal by treating all other interfering signals as noise). Assuming that the power allocation is subject to a maximum power constraint $\sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}$ for each BS $n \in \mathcal{N}$, the problem of WSRMax can be expressed as

$$\begin{aligned} \text{maximize} \quad & \sum_{l \in \mathcal{L}} \beta_l \log_2 \left(1 + \frac{|\mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l|^2}{\sigma_l^2 + \sum_{j \in \mathcal{L}, j \neq l} |\mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j|^2} \right) \\ \text{subject to} \quad & \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \end{aligned} \tag{2.4}$$

where the optimization variable is $\{\mathbf{m}_l\}_{\in \mathcal{L}}$.

2.2 Solution via branch and bound method

In this section we start by equivalently reformulating problem (2.4) as a minimization of a nonconvex function over an L-dimensional rectangle. Then we apply the BB method [76] to minimize the nonconvex function over the L-dimensional rectangle.

2.2.1 Equivalent reformulation

By introducing a new variable γ_l, for all $l \in \mathcal{L}$, and changing the sign of the objective function of problem (2.4), it can be expressed equivalently as

$$\begin{array}{ll} \text{minimize} & \sum_{l\in\mathcal{L}} -\beta_l \log_2(1+\gamma_l) \\ \text{subject to} & \gamma_l \leq \dfrac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum_{\ j\in\mathcal{L}, j\neq l\ } |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2}, \quad l \in \mathcal{L} \\ & \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \end{array} \tag{2.5}$$

with variables $\{\gamma_l\}_{l\in\mathcal{L}}$ and $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$. Note that the equivalence between problems (2.4) and (2.5) follows from the monotonically increasing property of the $\log_2(\cdot)$ function, which ensures that the first set of inequality constraints of problem (2.5) are tight (i.e., they hold with equality at the optimal solution).

Now let us rewrite problem (2.5) in a compact form as

$$\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{\gamma}) \\ \text{subject to} & \boldsymbol{\gamma} \in \mathcal{G}, \end{array} \tag{2.6}$$

with variable $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_L]^{\mathrm{T}}$, where the objective function $f_0(\boldsymbol{\gamma})$ is

$$f_0(\boldsymbol{\gamma}) = \sum_{l\in\mathcal{L}} -\beta_l \log_2(1+\gamma_l) \tag{2.7}$$

and the feasible set (or the achievable SINR values) $\mathcal{G}$ for variable $\boldsymbol{\gamma}$ is

$$\mathcal{G} = \left\{ \boldsymbol{\gamma} \;\middle|\; \begin{array}{ll} \gamma_l \leq \dfrac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum_{\ j\in\mathcal{L}, j\neq l\ } |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2}, & l \in \mathcal{L} \\ \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, & n \in \mathcal{N} \end{array} \right\}. \tag{2.8}$$

Note that the function $f_0(\boldsymbol{\gamma})$ is nonpositive and zero is the maximum value over $\boldsymbol{\gamma} \in \mathcal{G}$. Thus, we define a new function $\tilde{f} : \mathbb{R}_+^L \to \mathbb{R}$ as

$$\tilde{f}(\boldsymbol{\gamma}) = \begin{cases} f_0(\boldsymbol{\gamma}) & \text{if } \boldsymbol{\gamma} \in \mathcal{G} \\ 0 & \text{otherwise}, \end{cases} \tag{2.9}$$

Hence, for any $\mathcal{A} \subseteq \mathbb{R}_+^L$ such that $\mathcal{G} \subseteq \mathcal{A}$, we have

$$\inf_{\boldsymbol{\gamma}\in\mathcal{A}} \tilde{f}(\boldsymbol{\gamma}) = \inf_{\boldsymbol{\gamma}\in\mathcal{G}} f_0(\boldsymbol{\gamma}) = p^\star\ , \tag{2.10}$$

where p^* is the optimal value of problem (2.5). It is worth noting that the function $\tilde{f}$ is nonconvex over convex set $\mathcal{A}$ and f_0 is a global lower bound on $\tilde{f}$, i.e., $f_0(\boldsymbol{\gamma}) \leq \tilde{f}(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \mathcal{A}$.

We now show that problem (2.5) can be equivalently expressed as a minimization of the nonconvex function $\tilde{f}(\boldsymbol{\gamma})$ over an L-dimensional rectangle. To do this, let us define an L-dimensional rectangle $\mathcal{Q}_{\text{init}}$ as

$$\mathcal{Q}_{\text{init}} = \left\{ \boldsymbol{\gamma} \,\middle|\, 0 \leq \gamma_l \leq \frac{\|\mathbf{h}_{ll}\|_2^2}{\sigma_l^2} p_{\text{tran}(l)}^{\max}, l \in \mathcal{L} \right\}. \tag{2.11}$$

It is easy to check that $\mathcal{G} \subseteq \mathcal{Q}_{\text{init}}$ [2] . Therefore, from (2.10), it follows that $\inf_{\boldsymbol{\gamma} \in \mathcal{Q}_{\text{init}}} \tilde{f}(\boldsymbol{\gamma}) = p^\star$. Thus, we have reformulated problem (2.5) equivalently as a minimization of the nonconvex function $\tilde{f}(\boldsymbol{\gamma})$ over the rectangle $\mathcal{Q}_{\text{init}}$.

2.2.2 Branch and bound algorithm

In this section we apply BB method [76] to minimize the nonconvex function $\tilde{f}(\boldsymbol{\gamma})$ over the L-dimensional rectangle $\mathcal{Q}_{\text{init}}$. We first review briefly the BB method. Then we summarize a BB algorithm to minimize $\tilde{f}(\boldsymbol{\gamma})$ over the rectangle $\mathcal{Q}_{\text{init}}$.

For any L-dimension rectangle $\mathcal{Q} = \{\boldsymbol{\gamma} | \gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max}, l \in \mathcal{L}\}$ such that $\mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}$, let us define a function $\phi_{\min}(\mathcal{Q})$ as

$$\phi_{\min}(\mathcal{Q}) = \inf_{\boldsymbol{\gamma} \in \mathcal{Q}} \tilde{f}(\boldsymbol{\gamma}) \ . \tag{2.12}$$

By using (2.10) and (2.12), it can be easily verified that

$$\phi_{\min}(\mathcal{Q}_{\text{init}}) = \inf_{\boldsymbol{\gamma} \in \mathcal{Q}_{\text{init}}} \tilde{f}(\boldsymbol{\gamma}) = p^\star. \tag{2.13}$$

The key idea of the BB algorithm is to generate a sequence of asymptotically tight upper and lower bounds for $\phi_{\min}(\mathcal{Q}_{\text{init}})$. At each iteration k, the lower bound L_k and the upper bound U_k are updated by partitioning $\mathcal{Q}_{\text{init}}$ into smaller rectangles. To ensure convergence, the bounds should become tight as the number of rectangles in the partition of $\mathcal{Q}_{\text{init}}$ grows. To do this, the BB uses two functions $\phi_{\text{ub}}(\mathcal{Q})$ and $\phi_{\text{lb}}(\mathcal{Q})$, defined for any rectangle $\mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}$ such that the following conditions are satisfied [76, 171]:

C1: The functions $\phi_{\text{lb}}(\mathcal{Q})$ and $\phi_{\text{ub}}(\mathcal{Q})$ compute a lower bound and an upper bound, respectively on $\phi_{\min}(\mathcal{Q})$, i.e.,

$$\phi_{\text{lb}}(\mathcal{Q}) \leq \phi_{\min}(\mathcal{Q}) \leq \phi_{\text{ub}}(\mathcal{Q}) \ .$$

[2] It follows from Cauchy-Schwartz inequality (i.e., $|\mathbf{h}^{\text{H}}{}_{ll}\mathbf{m}_l|^2 \leq \|\mathbf{h}_{ll}\|_2^2 p_{\text{tran}(l)}^{\max}$ for all $\|\mathbf{m}_l\|_2^2 \leq p_{\text{tran}(l)}^{\max}$) after neglecting the interference terms in the denominator of SINR constraints in (2.8).

C2: As the maximum half length of the sides of $\mathcal{Q}$ (i.e., $\text{size}(\mathcal{Q}) = \frac{1}{2}\max_{l\in\mathcal{L}}\{\gamma_{l,\max} - \gamma_{l,\min}\}$) goes to zero, the difference between the upper and lower bounds converges to zero, i.e.,

$$\begin{aligned} &\forall \epsilon > 0 \ \exists \delta > 0 \ \text{ s.t. } \ \forall \mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}, \\ &\text{size}(\mathcal{Q}) \leq \delta \Rightarrow \phi_{\text{ub}}(\mathcal{Q}) - \phi_{\text{lb}}(\mathcal{Q}) \leq \epsilon \ . \end{aligned} \tag{2.14}$$

Finding accurate and easy to compute upper and lower bound functions $\phi_{\text{ub}}(\mathcal{Q})$ and $\phi_{\text{lb}}(\mathcal{Q})$ is one of the most difficult part in deriving a BB algorithm. For clarity, we first summarize the generic BB algorithm and the bounding functions are defined in Section 2.3.

The BB algorithm starts by computing $U_1 = \phi_{\text{ub}}(\mathcal{Q}_{\text{init}})$ and $L_1 = \phi_{\text{lb}}(\mathcal{Q}_{\text{init}})$, which are upper and lower bounds on $p^\star$, respectively. Let ϵ be the pre-defined tolerance. Then if $U_1 - L_1 \leq \epsilon$, the algorithm terminates with a certificate that the upper bound U_1 is at most ϵ-away from the optimal value $p^\star$. Otherwise, the initial rectangle $\mathcal{Q}_{\text{init}}$ is partitioned into smaller rectangles. At the kth partitioning step, $\mathcal{Q}_{\text{init}}$ is split into k rectangles such that $\mathcal{Q}_{\text{init}} = \cup_{i=1}^{k}\mathcal{Q}_i$. The upper and lower bounds on $p^\star$ at kth partitioning step are updated as

$$U_k = \min_{i=1,\ldots,k} \phi_{\text{ub}}(\mathcal{Q}_i), \qquad L_k = \min_{i=1,\ldots,k} \phi_{\text{lb}}(\mathcal{Q}_i). \tag{2.15}$$

The BB algorithm terminates if the difference between U_k and L_k is smaller than ϵ. Otherwise, further partitioning of $\mathcal{Q}_{\text{init}}$ is required, until the difference between U_k and L_k is smaller than ϵ. The basic BB algorithm to minimize $\tilde{f}$ over $\mathcal{Q}_{\text{init}}$ can be summarized as follows:

Algorithm 2.1. *Branch and bound algorithm*

1. Initialization: given tolerance ϵ>0. Set k=1, $\mathcal{B}_1$=$\{\mathcal{Q}_{\text{init}}\}$, U_1=$\phi_{\text{ub}}(\mathcal{Q}_{\text{init}})$, and L_1=$\phi_{\text{lb}}(\mathcal{Q}_{\text{init}})$.
2. Stopping criterion: if $U_k - L_k > \epsilon$ go to Step 3, otherwise STOP.
3. Branching:
   a) pick $\mathcal{Q} \in \mathcal{B}_k$ for which $\phi_{\text{lb}}(\mathcal{Q}) = L_k$ and set $\mathcal{Q}_k = \mathcal{Q}$.
   b) split $\mathcal{Q}_k$ along one of its longest edge into $\mathcal{Q}_I$ and $\mathcal{Q}_{II}$.
   c) Let $\mathcal{B}_{k+1} = (\mathcal{B}_k \backslash \{\mathcal{Q}_k\}) \cup \{\mathcal{Q}_I, \mathcal{Q}_{II}\}$.

4. Bounding:
   a) set $U_{k+1} = \min_{\mathcal{Q}\in\mathcal{B}_{k+1}}\{\phi_{\rm ub}(\mathcal{Q})\}$.
   b) set $L_{k+1} = \min_{\mathcal{Q}\in\mathcal{B}_{k+1}}\{\phi_{\rm lb}(\mathcal{Q})\}$.
5. Set $k = k + 1$ and go to step 2.

The first step initializes the algorithm, and computes the upper and lower bounds on $p^\star$ over the initial rectangle $\mathcal{Q}_{\rm init}$. The second step checks the stopping criterion, and the algorithm terminates if the difference between the upper and lower bounds is smaller than ϵ. Otherwise, the algorithm repeats step 2 to 5, until $U_k - L_k \leq \epsilon$. Step 3 is a branching step; here, a rectangle is further partitioned into smaller rectangles. Note that in Algorithm 2.1, we have used the notation $\mathcal{B}_k$ to denote a set of k smaller rectangles that are obtained by partitioning $\mathcal{Q}_{\rm init}$. At step 3, we pick the rectangle $\mathcal{Q}$ with the smallest lower bound from the set $\mathcal{B}_k$, and split it into two smaller rectangles along its longest edge. Splitting of the rectangle along its longest edge ensures the convergence of the algorithm [76]. Step 4 updates the best upper and the best lower bounds according to (2.15).

Note that in step 3 of Algorithm 2.1, from the set $\mathcal{B}_k$ any rectangle for which the lower bound $\phi_{\rm lb}(\mathcal{Q})$ is larger than U_k is never selected for further splitting. This is because in such a rectangle all points are worse than the current best upper bound U_k on the optimal value $p^\star$. Thus, from the set $\mathcal{B}_k$ any rectangle that satisfies $\phi_{\rm lb}(\mathcal{Q}) > U_k$ can be eliminated (i.e., *pruned*), without affecting the algorithm. Even though pruning does not affect the algorithm, it can be useful in reducing the storage requirements of Algorithm 2.1.

2.2.3 Convergence of the branch and bound algorithm

In this section we show that the branch and bound algorithm (i.e., Algorithm 2.1) converges within a finite number of iterations. The convergence of Algorithm 2.1 is established by the following theorem:

Theorem 2.1. *If for any $\mathcal{Q} \subseteq \mathcal{Q}_{\rm init}$ with $\mathcal{Q} = \{\boldsymbol{\gamma}\,|\gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max},\ l \in \mathcal{L}\}$, the functions $\phi_{\rm ub}(\mathcal{Q})$ and $\phi_{\rm lb}(\mathcal{Q})$ satisfy the conditions C1 and C2, then Algorithm 2.1 converges in a finite number of iterations to a value arbitrarily close to $p^\star$, i.e., $\forall\epsilon > 0,\ \exists K > 0$ s.t $U_K - p^\star \leq \epsilon$.*

Proof. The proof is similar to the one provided in [76, 171], and it is provided here for the sake of completeness. First note that there are k rectangles in the set $\mathcal{B}_k$. Let $\text{vol}(\mathcal{Q}_{\text{init}})$ denote the volume of rectangle $\mathcal{Q}_{\text{init}}$. Thus, we have

$$\min_{\mathcal{Q}\in\mathcal{B}_k} \text{vol}(\mathcal{Q}) \leq \frac{\text{vol}(\mathcal{Q}_{\text{init}})}{k} \ . \tag{2.16}$$

Therefore, as k increases, at least one rectangle in the partition becomes small. Then it is required to show that the smaller $\text{vol}(\mathcal{Q})$ the smaller $\text{size}(\mathcal{Q})$. To do this, we first define the condition number of rectangle

$$\mathcal{Q} = \{\boldsymbol{\gamma} \,|\gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max},\ l \in \mathcal{L}\}$$

as

$$\text{cond}(\mathcal{Q}) = \frac{\max_l(\gamma_{l,\max} - \gamma_{l,\min})}{\min_l(\gamma_{l,\max} - \gamma_{l,\min})} \ . \tag{2.17}$$

Note that the branching rule we use (see Algorithm 2.1 , step 3), always ensures that for any k and any rectangle $\mathcal{Q} \in \mathcal{B}_k$ [76, Lem. 1]

$$\text{cond}(\mathcal{Q}) \leq \max\{\text{cond}(\mathcal{Q}_{\text{init}}), 2\} \ . \tag{2.18}$$

Moreover, we have,

$$\begin{aligned}
\text{vol}(\mathcal{Q}) &= \textstyle\prod_{l=1}^{L}(\gamma_{l,\max} - \gamma_{l,\min}) && (2.19)\\
&\geq \max_l(\gamma_{l,\max} - \gamma_{l,\min})\Big(\min_l(\gamma_{l,\max} - \gamma_{l,\min})\Big)^{L-1} && (2.20)\\
&= \frac{(2\ \text{size}(\mathcal{Q}))^L}{(\text{cond}(\mathcal{Q}))^{L-1}} && (2.21)\\
&\geq \left(\frac{2\ \text{size}(\mathcal{Q})}{\text{cond}(\mathcal{Q})}\right)^L \ , && (2.22)
\end{aligned}$$

where the last inequality follows by noting that $\text{cond}(\mathcal{Q}) \geq 1$. Thus, from (2.22) we have

$$\text{size}(\mathcal{Q}) \leq \frac{1}{2}\text{cond}(\mathcal{Q})\text{vol}(\mathcal{Q})^{1/L} \ . \tag{2.23}$$

By using (2.16), (2.18), and (2.23) we get

$$\min_{\mathcal{Q}\in\mathcal{B}_k} \text{size}(\mathcal{Q}) \leq \frac{1}{2}\max\{\text{cond}(\mathcal{Q}_{\text{init}}), 2\}\frac{\text{vol}(\mathcal{Q}_{\text{init}})}{k} \ . \tag{2.24}$$

We are now ready to show that there exists a positive integer K such that for any $\epsilon > 0$, $U_K - p^\star \leq \epsilon$. To see this, we select K as the maximum number of

iterations so that

$$\frac{1}{2}\max\{\text{cond}(\mathcal{Q}_{\text{init}}), 2\}\frac{\text{vol}(\mathcal{Q}_{\text{init}})}{K} \leq \delta \ . \tag{2.25}$$

Thus from (2.24), for some $\tilde{\mathcal{Q}} \in \mathcal{B}_K$, $\text{size}(\tilde{\mathcal{Q}}) \leq \delta$ and from C2 (see (2.14)), we have $\phi_{\text{ub}}(\tilde{\mathcal{Q}}) - \phi_{\text{lb}}(\tilde{\mathcal{Q}}) \leq \epsilon$. However, note that $U_K \leq \phi_{\text{ub}}(\tilde{\mathcal{Q}})$ (since $U_K = \min_{\mathcal{Q}\in\mathcal{B}_K}\{\phi_{\text{ub}}(\mathcal{Q})\}$) and $p^\star \geq \phi_{\text{lb}}(\tilde{\mathcal{Q}})$. Thus, $U_K - p^\star \leq \epsilon$ and the result follows. □

2.3 Upper and lower bound functions

In this section we derive the bounding functions $\phi_{\text{ub}}(\mathcal{Q})$ and $\phi_{\text{lb}}(\mathcal{Q})$ for Algorithm 2.1 by exploiting the monotonic nonincreasing property of f_0. First, basic bounding functions are established, and then a method to improve the basic lower bounding function is proposed.

2.3.1 Basic upper and lower bounds

The basic bounding functions are built upon the general expression of the basic lower and upper bound functions used in [59], which can be formally expressed as

$$\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q}) = \begin{cases} f_0(\boldsymbol{\gamma}_{\max}) & \boldsymbol{\gamma}_{\min} \in \mathcal{G} \\ 0 & \text{otherwise} \ , \end{cases} \tag{2.26}$$

and

$$\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q}) = \tilde{f}(\boldsymbol{\gamma}_{\min}) = \begin{cases} f_0(\boldsymbol{\gamma}_{\min}) & \boldsymbol{\gamma}_{\min} \in \mathcal{G} \\ 0 & \text{otherwise} \ , \end{cases} \tag{2.27}$$

where $\mathcal{Q} = \{\boldsymbol{\gamma} | \gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max}, l \in \mathcal{L}\}$ such that $\mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}$, $\boldsymbol{\gamma}_{\max} = \left[\gamma_{1,\max}, \ldots, \gamma_{L,\max}\right]^T$, $\boldsymbol{\gamma}_{\min} = \left[\gamma_{1,\min}, \ldots, \gamma_{L,\min}\right]^T$, and $\mathcal{G}$ is defined in (2.8). These general expressions hold true for the case of MISO system as well. However, checking the condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$, which is central to computing $\phi_{\text{lb}}^{\text{Basic}}$ and $\phi_{\text{ub}}^{\text{Basic}}$, is much more difficult in the case of multiple transmit antennas. In the sequel, we first show that the functions $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ and $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ satisfy conditions C1 and C2; which are essential for the convergence of Algorithm 2.1. Then we provide a computationally efficient method to check the condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$.

Lemma 2.1. *The functions $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ and $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ satisfy the conditions C1 and C2.*

Proof. The proof is similar to the one provided in [59], and it is provided here for the sake of completeness. First, we prove that the functions $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ and $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ satisfy the condition C1; then we prove that the functions $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ and $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ satisfy the condition C2.

1. *The functions* $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ *and* $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ *satisfy the condition* C1

   In the case of $\boldsymbol{\gamma}_{\min} \notin \mathcal{G}$ we can easily see that $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q}) = \phi_{\min}(\mathcal{Q}) = \phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q}) = 0$, and therefore the inequalities in C1 hold with the equalities. In the case of $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$ we notice that

   $$\phi_{\min}(\mathcal{Q}) = \inf_{\boldsymbol{\gamma} \in \mathcal{Q}} \tilde{f}(\boldsymbol{\gamma}) \leq \tilde{f}(\boldsymbol{\gamma}_{\min}) = f_0(\boldsymbol{\gamma}_{\min}) = \phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q}) \ . \tag{2.28}$$

   The first equality follows from (2.12), the inequality follows since $\boldsymbol{\gamma}_{\min} \in \mathcal{Q}$, and the second equality follows from (2.9). Moreover, we have

   $$\phi_{\min}(\mathcal{Q}) = \inf_{\boldsymbol{\gamma} \in \mathcal{Q}} \tilde{f}(\boldsymbol{\gamma}) \geq \inf_{\boldsymbol{\gamma} \in \mathcal{Q}} f_0(\boldsymbol{\gamma}) = f_0(\boldsymbol{\gamma}_{\max}) = \phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q}) \ , \tag{2.29}$$

   where the inequality follows from the fact that $\tilde{f}(\boldsymbol{\gamma}) \geq f_0(\boldsymbol{\gamma})$ and the second equality is from the fact that $\mathcal{Q}$ is a rectangle and $f_0(\boldsymbol{\gamma})$ is monotonically decreasing in each variable γ_l, $l \in \mathcal{L}$. From (2.28) and (2.29) we conclude that $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q}) \leq \phi_{\min}(\mathcal{Q}) \leq \phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$. This completes the proof of the first part of Lemma 2.1

2. *The functions* $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$ *and* $\phi_{\text{ub}}^{\text{Basic}}(\mathcal{Q})$ *satisfy the condition* C2

   We first show that the function $f_0(\boldsymbol{\gamma}) = \sum_{l \in \mathcal{L}} -\beta_l \log_2(1 + \gamma_l)$ is Lipschitz continuous on $\mathbb{R}_+^L$ with the constant $H = \sqrt{\sum_{l \in \mathcal{L}} \beta_l^2} / \log(2)$, i.e.,

   $$|f_0(\boldsymbol{\mu}) - f_0(\boldsymbol{\nu})| \leq H \, ||\boldsymbol{\mu} - \boldsymbol{\nu}||_2 \tag{2.30}$$

   for all $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathbb{R}_+^L$. We start by noting that $f_0(\boldsymbol{\gamma})$ is convex. Therefore, for all $\boldsymbol{\mu}, \boldsymbol{\nu} \in \mathbb{R}_+^L$ we have [29, Sec. 3.1.3]

   $$f_0(\boldsymbol{\mu}) - f_0(\boldsymbol{\nu}) \leq \nabla f_0(\boldsymbol{\mu})^{\mathrm{T}} (\boldsymbol{\mu} - \boldsymbol{\nu}) \ . \tag{2.31}$$

   Without loss of generality, we can assume that $f_0(\boldsymbol{\mu}) - f_0(\boldsymbol{\nu}) \geq 0$. Otherwise, we can obtain exactly the same results by interchanging $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ in (2.31),

i.e., $f_0(\boldsymbol{\nu}) - f_0(\boldsymbol{\mu}) \leq \nabla f_0(\boldsymbol{\nu})^{\mathrm{T}}(\boldsymbol{\nu} - \boldsymbol{\mu})$. Thus, we see that

$$
\begin{aligned}
|f_0(\boldsymbol{\mu}) - f_0(\boldsymbol{\nu})| &\leq \left|\nabla f_0(\boldsymbol{\mu})^{\mathrm{T}}(\boldsymbol{\mu} - \boldsymbol{\nu})\right| && (2.32)\\
&\leq ||\nabla f_0(\boldsymbol{\mu})||_2 \, ||(\boldsymbol{\mu} - \boldsymbol{\nu})||_2 && (2.33)\\
&\leq \max\nolimits_{\boldsymbol{\gamma} \in \mathbb{R}_+^L} ||\nabla f_0(\boldsymbol{\gamma})||_2 \, ||(\boldsymbol{\mu} - \boldsymbol{\nu})||_2 && (2.34)\\
&= \max_{\boldsymbol{\gamma} \in \mathbb{R}_+^L} \frac{1}{\log(2)} \sqrt{\sum_{l \in \mathcal{L}} \frac{\beta_l^2}{(1+\gamma_l)^2}} \; ||(\boldsymbol{\mu} - \boldsymbol{\nu})||_2 && (2.35)\\
&= H \, ||(\boldsymbol{\mu} - \boldsymbol{\nu})||_2 \;, && (2.36)
\end{aligned}
$$

where (2.32) follows from (2.31), (2.33) follows from the Cauchy-Schwarz inequality, (2.34) follows from the maximization operation, (2.35) follows by noting that $[\nabla f_0(\gamma)]_l = \frac{\beta_l}{(1+\gamma_l)\log(2)}$, $l \in \mathcal{L}$, and (2.36) follows by setting $\gamma_l = 0$ for all $l \in \mathcal{L}$.

Now we can write the following relations:

$$
\begin{aligned}
\phi_{\mathrm{ub}}^{\mathrm{Basic}}(\mathcal{Q}) - \phi_{\mathrm{lb}}^{\mathrm{Basic}}(\mathcal{Q}) &\leq f_0(\boldsymbol{\gamma}_{\min}) - f_0(\boldsymbol{\gamma}_{\max}) && (2.37)\\
&\leq H \, ||\boldsymbol{\gamma}_{\min} - \boldsymbol{\gamma}_{\max}||_2 && (2.38)\\
&= H \Big|\Big| \sum_{l \in \mathcal{L}} (\gamma_{l,\max} - \gamma_{l,\min}) \hat{\mathbf{e}}_l \Big|\Big|_2 && (2.39)\\
&\leq H \textstyle\sum_{l \in \mathcal{L}} (\gamma_{l,\max} - \gamma_{l,\min}) && (2.40)\\
&\leq 2HL \; \mathrm{size}(\mathcal{Q}) \; . && (2.41)
\end{aligned}
$$

The first inequality (2.37) follows from (2.26) and (2.27) by noting that f_0 is nonincreasing, (2.38) follows from (2.30), (2.39) follows clearly by noting that $\hat{\mathbf{e}}_l$ is lth standard unit vector, (2.40) follows from the triangle inequality, and (2.41) follows from the definition of size($\mathcal{Q}$) (see C2). Thus, for any given $\epsilon > 0$, we can select δ such that $\delta \leq \epsilon/2HL$, which in turn implies that condition C2 is satisfied. This completes the proof of the second part of Lemma 2.1

□

We now provide a computationally efficient method based on SOCP to check the condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$. Let $\boldsymbol{\gamma} = \left[\gamma_1, \ldots, \gamma_L\right]^{\mathrm{T}}$ be a specified set of SINR values. To test if these values are achievable (i.e., to test if $\boldsymbol{\gamma} \in \mathcal{G}$) is equivalent to

solving the following feasibility problem [29, Sec. 4.1.1]:

$$\begin{array}{ll} \text{find} & \mathbf{m}_1, \ldots, \mathbf{m}_L \\ \text{subject to} & \dfrac{|\mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l|^2}{\sigma_l^2 + \sum_{\ j \in \mathcal{L}, j \neq l\ } |\mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j|^2} \geq \gamma_l, \qquad l \in \mathcal{L} \\ & \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \qquad n \in \mathcal{N}, \end{array} \tag{2.42}$$

with variable $\{\mathbf{m}_l\}_{l \in \mathcal{L}}$. The feasibility problem (2.42) determines whether the SINR constraints are achievable, and if so, returns a set of feasible transmit beamformers $\{\mathbf{m}_l^\star\}_{l \in \mathcal{L}}$ that satisfies them.

Problem (2.42) is not convex as such, but following the approach of [132, Sec. IV-B], it can be reformulated as a standard SOCP and solved efficiently via interior points methods [29, 172]. To do this, let us define the matrix $\mathbf{M}_n = [\mathbf{m}_l]_{l \in \mathcal{L}(n)}$ obtained by concatenating the column vectors $\mathbf{m}_l$. Then problem (2.42) can be cast as the following SOCP feasibility problem:

$$\begin{array}{ll} \text{find} & \mathbf{m}_1, \ldots, \mathbf{m}_L \\ \text{subject to} & \begin{bmatrix} \sqrt{\left(1 + \frac{1}{\gamma_l}\right)} \mathbf{m}_l^{\mathrm{H}} \mathbf{h}_{ll} \\ \left[\mathbf{m}_1^{\mathrm{H}} \mathbf{h}_{1l}, \ldots, \mathbf{m}_L^{\mathrm{H}} \mathbf{h}_{Ll}\right]^T \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{L} \\ & \begin{bmatrix} \sqrt{p_n^{\max}} \\ \mathrm{vec}(\mathbf{M}_n) \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \qquad n \in \mathcal{N}, \end{array} \tag{2.43}$$

where the optimization variable is $\{\mathbf{m}_l\}_{l \in \mathcal{L}}$; and we use notation $\succeq_{\mathrm{SOC}}$ to denote the generalized inequalities with respect to the second order cone [132], [29, Sec. 2.2.3], i.e., for any $x \in \mathbb{R}$ and $\mathbf{y} \in \mathbb{C}^T$, $[x, \mathbf{y}^T]^T \succeq_{\mathrm{SOC}} 0$ is equivalent to $x \geq \|\mathbf{y}\|_2$.

2.3.2 Improved lower bound

Tighter bounds[3] are very important because they can substantially increase the convergence speed of the BB algorithm. By exploiting the monotonically nonincreasing property of f_0, an improved lower bound is proposed in this subsection.

[3]We say a bound is tighter in the following sense: $\phi_{\mathrm{lb}}(\mathcal{Q})$ is a tighter lower bound if for any $\mathcal{Q} \subseteq \mathcal{Q}_{\mathrm{init}}$, we have $\phi_{\min}(\mathcal{Q}) \geq \phi_{\mathrm{lb}}(\mathcal{Q}) \geq \phi_{\mathrm{lb}}^{\mathrm{Basic}}(\mathcal{Q})$.

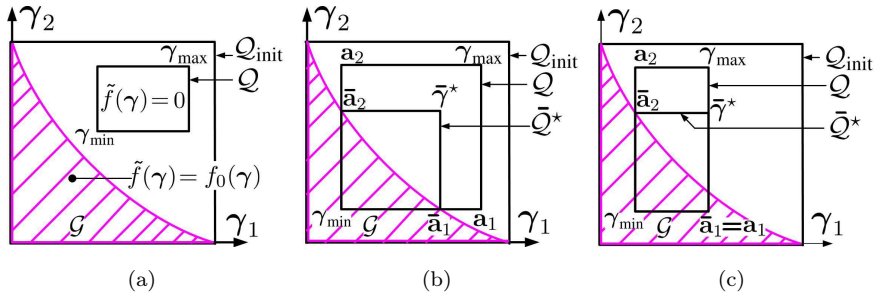


Fig. 2.2. Illustration of $\mathcal{G}$, $\mathcal{Q}_{\text{init}}$, $\mathcal{Q}$, and $\bar{\mathcal{Q}}^\star$ in a $2-$dimensional space, [157] ©2012, IEEE.

Note that in the case of $\boldsymbol{\gamma}_{\min} \notin \mathcal{G}$ (i.e., $\mathcal{Q} \cap \mathcal{G} = \emptyset$, see Fig. 2.2(a), $\tilde{f}(\boldsymbol{\gamma}) = 0$ for any $\boldsymbol{\gamma} \in \mathcal{Q}$. Thus, both the basic lower bound (2.26) and the basic upper bound (2.27) are trivially zero and no further improvement is possible since they are tight. Consequently, tighter bounds can be found only in the case $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$ (i.e., $\mathcal{Q} \cap \mathcal{G} \neq \emptyset$, see Fig. 2.2(b)). Thus, we consider only this case in the sequel.

Roughly speaking, a tighter lower bound can be obtained as follows. We first construct the smallest rectangle $\bar{\mathcal{Q}}^\star \subseteq \mathcal{Q}$ which encloses the intersection $\mathcal{Q} \cap \mathcal{G}$ (see Fig. 2.2(b)). Let us denote this rectangle as

$$\bar{\mathcal{Q}}^\star = \{\boldsymbol{\gamma} \,|\gamma_{l,\min} \leq \gamma_l \leq \bar{\gamma}_l^\star,\ l \in \mathcal{L}\}\,. \tag{2.44}$$

Then the improved lower bound is given by $f_0(\bar{\gamma}_1^\star, \ldots, \bar{\gamma}_L^\star)$.

Recall that $\mathcal{Q} = \{\boldsymbol{\gamma} \,|\gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max},\ l \in \mathcal{L}\}$. For any $\mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}$, the improved lower bound can be formally expressed as

$$\phi_{\text{lb}}^{\text{Imp}}(\mathcal{Q}) = \begin{cases} f_0(\bar{\boldsymbol{\gamma}}^\star) & \text{if } \boldsymbol{\gamma}_{\min} \in \mathcal{G} \\ 0 & \text{otherwise}\,, \end{cases} \tag{2.45}$$

where $\bar{\boldsymbol{\gamma}}^\star = [\bar{\gamma}_1^\star, \ldots, \bar{\gamma}_L^\star]^T$ is the maximum corner of the rectangle $\bar{\mathcal{Q}}^\star$, and $\bar{\gamma}_i^\star$ can be found by using bisection search on each edge of the rectangle $\mathcal{Q}$ as discussed below.

Let us define a corner point along the $\hat{\mathbf{e}}_i$ edge of the rectangle $\mathcal{Q}$ as $\mathbf{a}_i = \boldsymbol{\gamma}_{\min} + (\gamma_{i,\max} - \gamma_{i,\min})\hat{\mathbf{e}}_i$. If a corner point $\mathbf{a}_i$ lies inside $\mathcal{G}$, i.e., $\mathbf{a}_i \in \mathcal{G}$ (see $\mathbf{a}_1$ in Fig. 2.2(c)), then $\bar{\gamma}_i^\star = \gamma_{i,\max}$. Otherwise (i.e., $\mathbf{a}_i \notin \mathcal{G}$), a bisection search over

the line segment between the points $\boldsymbol{\gamma}_{\min}$ and $\mathbf{a}_i$ can be used to find $\bar{\gamma}_i^\star$. The bisection search used to find $\bar{\gamma}_i^\star$ (when $\mathbf{a}_i \notin \mathcal{G}$) is summarized in Algorithm 2.2 [4].

Algorithm 2.2. *Bisection search for finding* $\bar{\gamma}_i^\star$

1. Initialization: $\mathbf{l} = \boldsymbol{\gamma}_{\min}$ and $\mathbf{u} = \mathbf{a}_i$, and tolerance $\epsilon_b > 0$.
2. If $\|\mathbf{u} - \mathbf{l}\|_2 < \epsilon_b$ return $\bar{\gamma}_i^\star = [\mathbf{u}]_i$ and STOP.
3. Set $\mathbf{t} = (\mathbf{l} + \mathbf{u})/2$.
4. If $\mathbf{t} \in \mathcal{G}$ set $\mathbf{l} = \mathbf{t}$. Otherwise, set $\mathbf{u} = \mathbf{t}$. Go to step 2.

Note that the SOCP feasibility problem formulation (2.43) is used for checking if $\mathbf{t} \in \mathcal{G}$ at the step 4 of Algorithm 2.2. In Algorithm 2.2, the interval $[\mathbf{l}, \mathbf{u}]$ is divided into two at each iteration. Hence, for a tolerance $\epsilon_b > 0$ exactly $\lceil \log_2(\|\mathbf{l} - \mathbf{u}\|_2/\epsilon_b) \rceil$ iterations are required for the algorithm to terminate [29, Ch. 4.2.5].

2.4 Numerical examples

In this section we first evaluate the impact of the proposed bounds (Section 2.3) on the convergence of Algorithm 2.1. Then, we use the proposed Algorithm 2.1 to evaluate the performance loss of several suboptimal algorithms.

We consider a multicell wireless downlink system as shown in Fig. 2.3. There are $N = 2$ BSs each with $T = 2$ transmit antennas. The distance between the BSs is denoted by D_{BS}. We assume circular cells, where the radius of each one is denoted by R_{BS}. For simplicity, we assume 2 users per cell. The locations of the users associated with the BSs are arbitrarily chosen as shown in Fig. 2.3.

We assume an exponential path loss model, where the channel vector between the BSs and users is modeled as

$$\mathbf{h}_{jl} = \left(\frac{d_{jl}}{d_0}\right)^{-\eta/2} \mathbf{c}_{jl}, \tag{2.46}$$

where d_{jl} is the distance from tran(j) to rec(l), d_0 is the far field reference distance [173], η is the path loss exponent, and $\mathbf{c}_{jl} \in \mathbb{C}^T$ is arbitrarily chosen from

[4] For any $0 < \epsilon_b < \|\mathbf{l} - \mathbf{u}\|_2$, bound $\phi_{\text{lb}}^{\text{Imp}}(\mathcal{Q})$ is less than $\phi_{\text{lb}}^{\text{Basic}}(\mathcal{Q})$. Hence, instead of solving Algorithm 2.2 accurately, a value of ϵ_b can be selected in order to find improved lower bound, and reduce the execution time of BB Algorithm 2.1 .

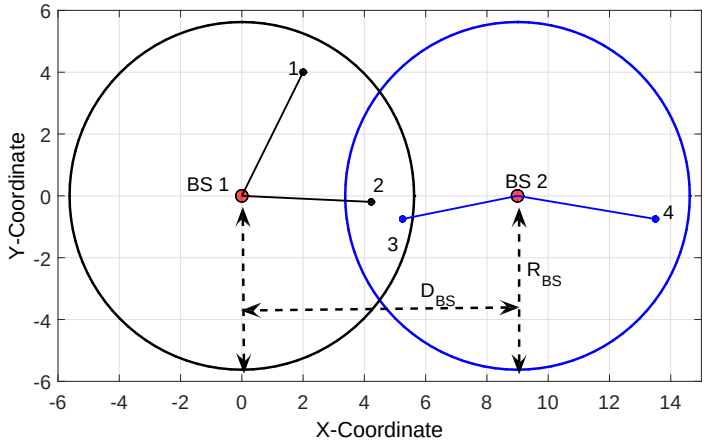


Fig. 2.3. MISO downlink wireless network with $N = 2$ BSs and $L = 4$ users. $\mathcal{N} = \{1, 2\}$, $\mathcal{L}(1) = \{1, 2\}$, and $\mathcal{L}(2) = \{3, 4\}$, [157] ©2012, IEEE.

the distribution $\mathcal{CN}(0, \mathbf{I})$ (i.e., frequency-flat fading channel with uncorrelated antennas). Here, we refer an arbitrarily generated set of fading coefficients $\check{\mathcal{C}} = \{\mathbf{c}_{jl} | j, l \in \mathcal{L}\}$ as a single fading realization. Note that in expression (2.46) the term $(d_{jl}/d_0)^{-\eta/2}$ denotes large scale fading, and the term $\mathbf{c}_{jl}$ denotes small scale fading.

We set $p_n^{\max} = p_0^{\max}$ for all $n \in \mathcal{N}$, and $\sigma_l = \sigma$ for all $l \in \mathcal{L}$. We define the signal-to-noise ratio (SNR) operating point at a distance r as

$$\mathrm{SNR}(r) = \left(\frac{r}{d_0}\right)^{-\eta} \frac{p_0^{\max}}{\sigma^2}. \tag{2.47}$$

In the following simulations, we set $d_0 = 1$, $\eta = 4$, and the cell radius R_{BS} is fixed throughout the simulations such that $\mathrm{SNR}(R_{\mathrm{BS}}) = 10$ dB for $p_0^{\max}/\sigma^2 = 40$ dB. Furthermore, we let $D_{\mathrm{BS}}/R_{\mathrm{BS}} = 1.6$.

Fig. 2.4 shows the evolution of upper and lower bounds for the optimal value of problem (2.5) for a single fading realization, and $\beta_l = 0.25$ for all $l \in \mathcal{L}$. Specifically, in Fig. 2.4 we used the basic upper bound ($\mathrm{UB}_{\mathrm{Basic}}$) in conjunction with both the basic lower bound ($\mathrm{LB}_{\mathrm{Basic}}$) and the improved lower bound ($\mathrm{LB}_{\mathrm{Imp}}$). Results show that both lower/upper bound pairs ($\mathrm{LB}_{\mathrm{Imp}}, \mathrm{UB}_{\mathrm{Basic}}$) and ($\mathrm{LB}_{\mathrm{Basic}}, \mathrm{UB}_{\mathrm{Basic}}$) become tighter as the number of iterations grows. However, the convergence speed of Algorithm 2.1 is substantially increased by the improved lower bound as compared to the basic one. For example, when $\epsilon = 0.2$, the basic

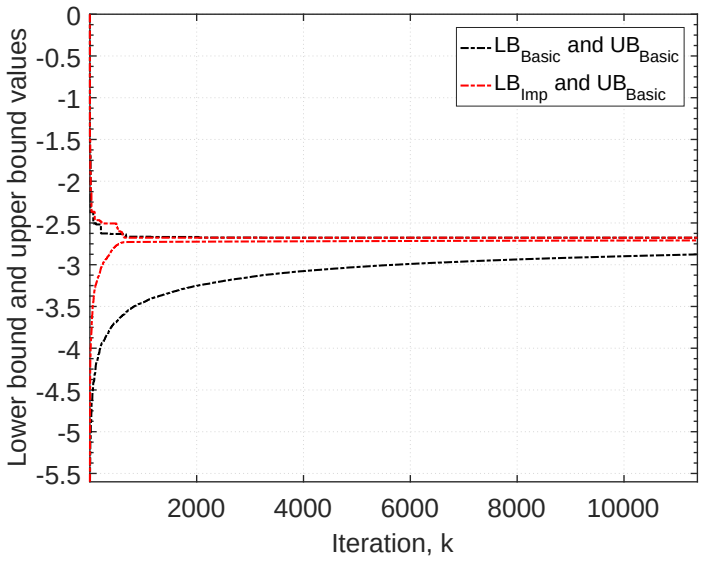


Fig. 2.4. Upper and lower bound evolution, [157] ©2012, IEEE.

bound requires more than 10^4 iterations to converge, where as the improved lower bound, with the bisection search tolerance $\epsilon_b = 0.1$, achieves the same level of accuracy in just 525 iterations.

In order to provide a statistical description of the speed of convergence, we run Algorithm 2.1 for 100 fading realizations. For each one we store the number of iterations required to find the optimal value of problem (2.5) within an accuracy of $\epsilon = 0.1$ with both lower/upper bound pairs $(\mathrm{LB}_{\mathrm{Imp}}, \mathrm{UB}_{\mathrm{Basic}})$ and $(\mathrm{LB}_{\mathrm{Basic}}, \mathrm{UB}_{\mathrm{Basic}})$, respectively.

Fig. 2.5 shows the empirical cumulative distribution function (CDF) plots of the total number of iterations required to terminate Algorithm 2.1. Results show that the improved lower bound significantly increases (by about 100 times) the convergence speed of Algorithm 2.1. For example, when the improved lower bound is used the algorithm finishes in less than 1500 iterations for more than 90% of the simulated cases, but with the basic lower bound the algorithm needs about 1.5×10^5 iterations to find the optimal solution with the same probability.

In the sequel, we use the proposed Algorithm 2.1 to numerically evaluate the performance loss of the following suboptimal algorithms: 1) generalized asynchronous distributed (GAD) algorithm [174, Sec. 2], 2) distributed interfer-

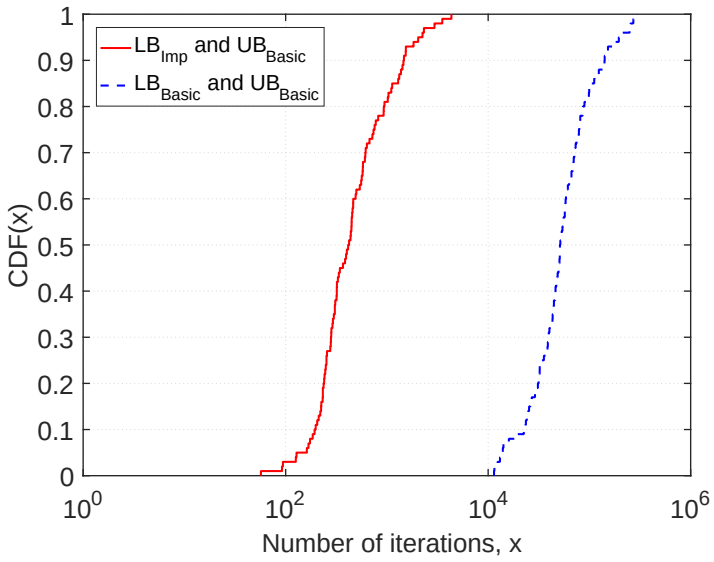


Fig. 2.5. Empirical CDF plot of total number of iterations, [157] ©2012, IEEE.

ence alignment (DIA) algorithm [175], and 3) weighted sum-MSE minimization (WMMSE) algorithm [120]. The GAD and DIA algorithms can handle only interference channels, and therefore we limit our simulations to 2-user MISO interference channels. Specifically, in the following simulation we consider one user per BS in Fig. 2.3, i.e., only user 2 of BS 1 and user 3 of BS 2 are considered.

Fig. 2.6 shows the weighted sum-rate of the considered algorithms for different SNR values[5]. Each curve is averaged over 500 fading realizations. Transmit beamforming vectors with full transmit power are used for initializing the beamformers of suboptimal algorithms. For Algorithm 2.1 the accuracy ϵ is set to 0.01. Results show that the performance of the GAD algorithm is very close to the optimal value irrespective of SNR values for the considered system setup. The DIA algorithm has a noticeable performance loss at low SNR values, however, it approaches to the optimal value at high SNR values. In contrast, the performance of WMMSE algorithm is close to the optimal value at low SNR values, and exhibits a noticeable performance loss at high SNR values.

[5]For fixed radius R_{BS} in Fig. 2.3, different SNRs (i.e., different SNR(R_{BS})) are obtained by changing $p_0^{\max}/\sigma^2$ in (2.47).

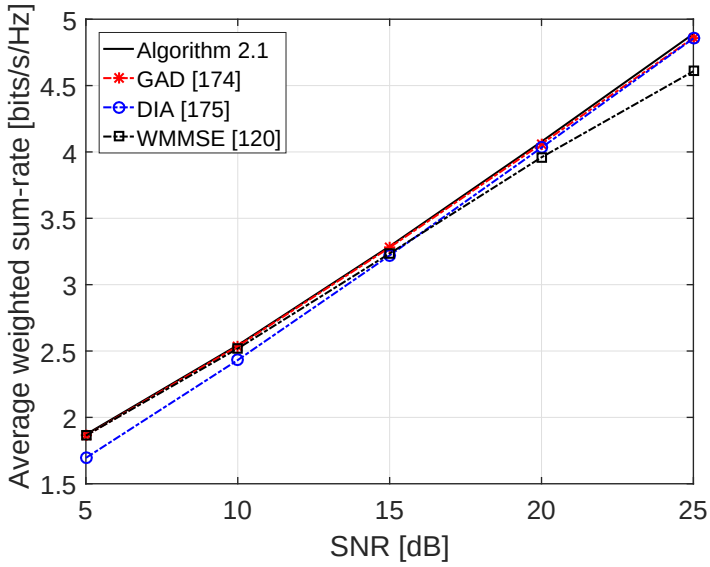
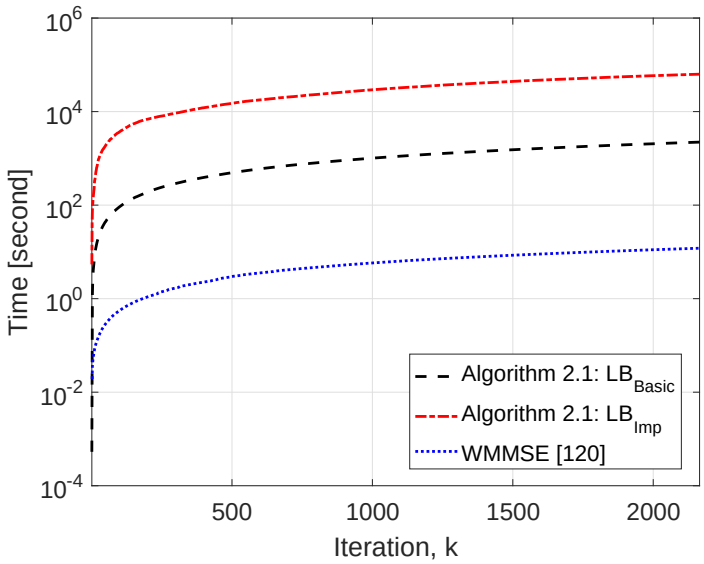


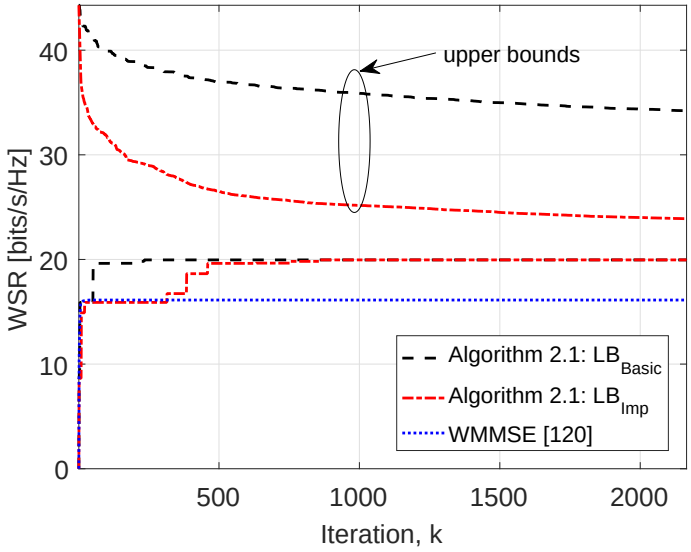
Fig. 2.6. Average weighted sum-rate for SNR, [157] ©2012, IEEE.

Next, we plot the empirical complexity of BB Algorithm 2.1 as compared to the WMMSE [120] algorithm, in order to illustrate the complexity-performance tradeoff. We consider $N = 3$ BSs, each with 2 users. We set $\beta_l = 1$ for all $l \in \mathcal{L}$, and run Algorithm 2.1 for the both basic lower bound (LB_{Basic}) and improved lower bound (LB_{Imp}). Fig. 2.7(a) shows the execution time[6] [seconds] versus the number of iterations for SNR = 15 dB, and the evolution of the corresponding WSR value is shown in Fig. 2.7(b). Results show that the WMMSE [120] algorithm can converge fast (e.g., for the simulated case, it converges in about 10 iterations, taking around 7.8×10^{-2} seconds). However, the BB Algorithm 2.1 can be slow as shown in Fig 2.7(a). But, the BB algorithm is a useful tool for evaluating the performance loss of the suboptimal algorithm; see Fig. 2.7(b).

[6]The simulation was run in MATLAB 2015b using CVX package[172], on Linux system with an Intel Xeon CPU E5-2640 v4 @ 2.40GHz.



(a)



(b)

Fig. 2.7. Complexity-performance tradeoff: (a) Execution time versus iteration; (b) Evolution of WSR versus iteration.

2.5 Summary and discussion

We have considered the problem of WSRMax in multicell downlink MISO systems. The problem is known to be NP-hard. We have proposed a method, based on a BB technique, which globally solves the nonconvex WSRMax problem with an optimality certificate. Bounding methods based on conic optimization were proposed. The bounding functions at each step of the algorithm can be efficiently solved by using interior-point method [29, 172]. The convergence speed of the proposed algorithm can be substantially increased by improving the lower bound. It is worth to note that the BB technique, basically, implements some sort of exhaustive search in a systematic manner. Hence, Algorithm 2.1 can be (and often are) slow. In the worst case its complexity can grow exponentially with the problem size.

The proposed method can be used to provide performance benchmarks by back-substituting it into many existing network design problems which rely on the WSRMax problem. Moreover, the method proposed here is not restricted to the WSRMax problem. It can also handle *any* system performance metric that can be expressed as a Lipschitz continuous and increasing function of SINR values.

3 Robust resource allocation with imperfect CSI for MISO cellular networks

In this chapter the problem of WSRMax in the presence of imperfect CSI at BSs for multicell MISO downlink networks is considered. Specifically, assuming bounded ellipsoidal model for the CSI errors, the problem of worst-case WSRMax for multicell MISO downlink networks is considered. The problem is known to be NP-hard even in the case of perfect CSI.

We propose both global and local solution methods for the worst-case WSRMax problem with CSI errors. The optimal global solution method is built upon the work in Chapter 2 which address the WSRMax problem in the case of perfect CSI via the BB method [76]. The extension of the BB algorithm of Chapter 2 to the case of *imperfect* CSI is not straightforward. This is because the upper and lower bound functions, which are the most challenging components of the BB algorithm, must be redesigned from the scratch to take into account the CSI errors.

As the problem is NP-hard, the convergence speed of the BB method can be slow for large networks [76]. Hence, even though the BB method provides an optimal solution, the algorithm may not be suitable for practical implementation in large networks. Nonetheless, the optimal BB based algorithm is a very useful tool for providing performance benchmarks for any suboptimal algorithm.

Next, we propose a possibly suboptimal but fast-converging algorithm for the worst-case WSRMax problem. The proposed suboptimal algorithm is based on the alternating optimization technique [34, 35] and sequential convex programming [35]. The main idea is to solve the worst-case WSRMax problem with respect to different subsets of variables by considering the others to be fixed. Specifically, the proposed algorithm decomposes the original worst-case WSRMax problem into two smaller subproblems and solves them sequentially. Unfortunately, the subproblems are still not convex. Hence, to derive a fast suboptimal algorithm we approximate these subproblems with convex problems, such that the efficient interior-point methods [29] can be used to solve them. It is worth noting that somewhat similar techniques were used in [83, 176] in the context of multi user MIMO downlink systems with perfect CSI at BSs (and

generalized in [177]). However, the WSRMax problem addressed in this chapter is substantially more challenging due to CSI uncertainty.

In practice, the channel estimation errors may have a statistical distribution [90, 178]. Thus, numerically we also show how we can apply our proposed algorithms for a scenario with such statistical channel errors.

3.1 System model and problem formulation

3.1.1 Network model

A multicell MISO downlink system, with N BSs each equipped with T transmit antennas is considered. The set of all BSs is denoted by $\mathcal{N}$, and we label them with the integer values $n = 1, \ldots, N$. The transmission region of each BS is modeled as a disc with radius R_{BS} centered at the location of the BS. A single data stream is transmitted for each user. We denote the set of all data streams in the system by $\mathcal{L}$, and we label them with the integer values $l = 1, \ldots, L$. The transmitter node (i.e., the BS) of lth data stream is denoted by $\mathrm{tran}(l)$ and the receiver node of lth data stream is denoted by $\mathrm{rec}(l)$. We have $\mathcal{L} = \cup_{n \in \mathcal{N}} \mathcal{L}(n)$, where $\mathcal{L}(n)$ denotes the set of data streams transmitted by nth BS.

The antenna signal vector transmitted by nth BS is given by

$$\mathbf{s}_n = \sum_{l \in \mathcal{L}(n)} d_l \mathbf{m}_l, \tag{3.1}$$

where $d_l \in \mathbb{C}$ represents the information symbol and $\mathbf{m}_l \in \mathbb{C}^T$ denotes the transmit beamformer associated with lth data stream. We assume that d_l is normalized such that $\mathrm{E}|d_l|^2 = 1$. Moreover, we assume that the data streams are independent, i.e., $\mathrm{E}\{d_l d_j^*\} = 0$ for all $l, j \in \mathcal{L}$, where $l \neq j$.

The signal received at $\mathrm{rec}(l)$ can be expressed as

$$\begin{aligned} y_l \;=\; & d_l \mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l + \underbrace{\sum_{j \,\in\, \mathcal{L}(\mathrm{tran}(l)),\, j \neq l} d_j \mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_j}_{\text{(intra-cell interference)}} \\ & + \underbrace{\sum_{n \in \mathcal{N} \setminus \{\mathrm{tran}(l)\}} \; \sum_{j \,\in\, \mathcal{L}(n)} d_j \mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j}_{\text{(out-of-cell interference)}} + n_l, \end{aligned} \tag{3.2}$$

where $\mathbf{h}_{jl}^{\mathrm{H}} \in \mathbb{C}^{1 \times T}$ is the channel vector between $\mathrm{tran}(j)$ and $\mathrm{rec}(l)$, and n_l is circular symmetric complex Gaussian noise with variance σ_l^2. Note that the

second right hand term in (3.2) represents the intra-cell interference[7] and the third right hand term represents the out-of-cell interference. The received SINR of lth data stream is given by

$$\Gamma_l(\mathbf{m}) = \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\in\mathcal{L}(\mathrm{tran}(l)),j\neq l} |\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum\limits_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2}, \tag{3.3}$$

where we use the notation $\mathbf{m}$ to denote a vector obtained by stacking $\mathbf{m}_l$ for all $l \in \mathcal{L}$ on top of each other, i.e., $\mathbf{m} = [\mathbf{m}_1^{\mathrm{T}}, \ldots, \mathbf{m}_L^{\mathrm{T}}]^{\mathrm{T}}$.

3.1.2 Channel uncertainty model

We assume that all the channels are imperfectly known at the network controller, but they belong to a known compact set of possible values. We model the channel vector $\mathbf{h}_{jl}$ as the sum of two components, i.e.,

$$\mathbf{h}_{jl} = \hat{\mathbf{h}}_{jl} + \mathbf{e}_{jl}, \quad j, l \in \mathcal{L}, \tag{3.4}$$

where $\hat{\mathbf{h}}_{jl} \in \mathbb{C}^T$ denotes the estimated value of the channel and $\mathbf{e}_{jl} \in \mathbb{C}^T$ represents the corresponding channel estimation error. It is assumed that $\mathbf{e}_{jl}$ can take any value inside a T-dimensional complex ellipsoid, which is defined as

$$\mathcal{E}_{jl} = \left\{\mathbf{e}_{jl} | \mathbf{e}_{jl}^{\mathrm{H}}\mathbf{Q}_{jl}\mathbf{e}_{jl} \leq 1\right\}, \tag{3.5}$$

where $\mathbf{Q}_{jl}$ is a complex Hermitian positive definite matrix, assumed to be known, which specifies the size and shape of the ellipsoid. For example, when $\mathbf{Q}_{jl} = (1/\xi_{jl}^2)\mathbf{I}$, the ellipsoidal channel error model (3.5) reduces to a ball uncertainty region with uncertainty radius ξ_{jl} [29]. Note that the ellipsoidal model can serve as a conservative approximation of the stochastic model, where we can take the ellipsoids to be the confidence ellipsoids for some high confidence, e.g., 95% [96, 179]. We refer the interested reader to [180] for the relationship between the stochastic and the ellipsoidal CSI error models. Now, with the channel uncertainty model (3.4), the received SINR (3.3) of lth data steam can

[7]Note that in the expression of intra-cell interference we have used the notation $\mathbf{h}_{ll}$ instead of $\mathbf{h}_{jl}$, since $\mathrm{tran}(j) = \mathrm{tran}(l)$ for all $j \in \mathcal{L}(\mathrm{tran}(l))$.

be expressed as

$$\Gamma_l(\mathbf{m},\mathbf{e}_l) = \frac{|(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2+\sum\limits_{j\in\mathcal{L}(\mathrm{tran}(l)),j\neq l}|(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{m}_j|^2+\sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}}\sum\limits_{j\in\mathcal{L}(n)}|(\hat{\mathbf{h}}_{jl}+\mathbf{e}_{jl})^{\mathrm{H}}\mathbf{m}_j|^2}, \quad (3.6)$$

where we use the notation $\mathbf{e}_l$ to denote a vector obtained by stacking $\mathbf{e}_{jl}$ for all $j \in \mathcal{L}$ on top of each other, i.e., $\mathbf{e}_l = [\mathbf{e}_{1l}^{\mathrm{T}},\ldots,\mathbf{e}_{Ll}^{\mathrm{T}}]^{\mathrm{T}}$.

3.1.3 Problem formulation

Let β_l be an arbitrary nonnegative weight associated with data stream l, $l \in \mathcal{L}$. We consider the case where all receivers are using *single-user detection* (i.e., a receiver decodes its intended signal by treating all other interfering signals as noise). Assuming that the power allocation is subject to a maximum power constraint $\sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l\|_2^2 \leq p_n^{\max}$ for each BS $n \in \mathcal{N}$, the problem of worst-case WSRMax can be expressed as[8]

$$\begin{array}{ll}\text{maximize} & \inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl},j,l\in\mathcal{L}}\left(\sum_{l\in\mathcal{L}}\beta_l\log_2\left(1+\Gamma_l(\mathbf{m},\mathbf{e}_l)\right)\right)\\ \text{subject to} & \sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n\in\mathcal{N},\end{array} \quad (3.7)$$

with variable $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$, where function $\Gamma_l(\mathbf{m},\mathbf{e}_l)$ is defined in (3.6). Note that if there are no CSI errors (i.e., $\mathbf{e}_{jl} = 0$ for all $j,l \in \mathcal{L}$), then problem (3.7) is reduced to problem (2.4) of Chapter 2, i.e., the WSRMax problem with no CSI error.

3.2 Optimal solution via branch and bound method

We extend the BB Algorithm 2.1 presented in Section 2.2.2 to solve the worst-case WSRMax problem (3.7). The main difficulty in using Algorithm 2.1 in the case of imperfect CSI is to define the bounding functions that are used in the BB algorithm. In this section we start by equivalently reformulating problem (3.7) as a minimization of a nonconvex function over an L-dimensional rectangle so

[8] In this Chapter, by the worst-case optimization criteria we provide a framework to handle CSI uncertainty in a robust manner. That is a certain level of performance is ensured for the CSI error model (3.4).

that BB Algorithm 2.1 can be used. Then we provide an efficient method based on a semidefinite program (SDP) to compute the bounding functions that are used in the BB algorithm.

3.2.1 Equivalent reformulation

Sets of error vectors associated with data streams are disjoint, i.e., set $\{\mathbf{e}_{1l}, \ldots, \mathbf{e}_{Ll}\}$ and set $\{\mathbf{e}_{1i}, \ldots, \mathbf{e}_{Li}\}$ for all $l, i \in \mathcal{L}$, where $l \neq i$, are disjoint. Hence, in the objective function of problem (3.7), we can move $\inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}}$ for all $l \in \mathcal{L}$ inside the sum function. Then express the objective function of problem (3.7) as[9] $\sum_{l \in \mathcal{L}} \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \left(\beta_l \log_2 \left(1 + \Gamma_l(\mathbf{m}, \mathbf{e}_l)\right)\right)$. Furthermore, by noting that $\log_2(\cdot)$ is a monotonically increasing function, we can move the infimum over $\mathbf{e}_{jl} \in \mathcal{E}_{jl}$ for all $j \in \mathcal{L}$ inside the $\log_2(\cdot)$ function. Then by introducing a new variable $\gamma_l = \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l)$, for all $l \in \mathcal{L}$, and changing the sign of the objective function of problem (3.7), it can be equivalently reformulated as

$$\begin{array}{ll} \text{minimize} & \sum_{l \in \mathcal{L}} -\beta_l \log_2(1 + \gamma_l) \\ \text{subject to} & \gamma_l \leq \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l), \quad l \in \mathcal{L} \\ & \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \end{array} \tag{3.8}$$

with variables $\{\gamma_l\}_{l \in \mathcal{L}}$ and $\{\mathbf{m}_l\}_{l \in \mathcal{L}}$. Note that the first inequality constraints of problem (3.8) hold with equality at the optimal solution due to a monotonic decreasing property of the objective function.

Now let us rewrite problem (3.8) in a compact form as

$$\begin{array}{ll} \text{minimize} & f_0(\boldsymbol{\gamma}) \\ \text{subject to} & \boldsymbol{\gamma} \in \mathcal{G}, \end{array} \tag{3.9}$$

with variable $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_L]^{\mathrm{T}}$, where the objective function $f_0(\boldsymbol{\gamma})$ is

$$f_0(\boldsymbol{\gamma}) = \sum_{l \in \mathcal{L}} -\beta_l \log_2(1 + \gamma_l) \tag{3.10}$$

and the feasible set (or the achievable SINR values) $\mathcal{G}$ for variable $\boldsymbol{\gamma}$ is

$$\mathcal{G} = \left\{ \boldsymbol{\gamma} \;\middle|\; \begin{array}{ll} \gamma_l \leq \inf\limits_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l), & l \in \mathcal{L} \\ \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, & n \in \mathcal{N} \end{array} \right\}. \tag{3.11}$$

[9] Using the fact that the infimum of the sum of separable functions is equal to the sum of the infimum of the functions, i.e., $\inf_{x,y} \left(f(x) + f(y)\right) = \inf_x f(x) + \inf_y f(y)$.

Note that the function $f_0(\boldsymbol{\gamma})$ is nonpositive and zero is the maximum value over $\boldsymbol{\gamma} \in \mathcal{G}$. Thus, we define a new function $\tilde{f} : \mathbb{R}_+^L \to \mathbb{R}$ as

$$\tilde{f}(\boldsymbol{\gamma}) = \begin{cases} f_0(\boldsymbol{\gamma}) & \text{if } \boldsymbol{\gamma} \in \mathcal{G} \\ 0 & \text{otherwise} , \end{cases} \tag{3.12}$$

Hence, for any $\mathcal{A} \subseteq \mathbb{R}_+^L$ such that $\mathcal{G} \subseteq \mathcal{A}$, we have

$$\inf_{\boldsymbol{\gamma} \in \mathcal{A}} \tilde{f}(\boldsymbol{\gamma}) = \inf_{\boldsymbol{\gamma} \in \mathcal{G}} f_0(\boldsymbol{\gamma}) = p^\star , \tag{3.13}$$

where p^* is the optimal value of problem (3.8). It is worth noting that the function $\tilde{f}$ is nonconvex over convex set $\mathcal{A}$ and f_0 is a global lower bound on $\tilde{f}$, i.e., $f_0(\boldsymbol{\gamma}) \leq \tilde{f}(\boldsymbol{\gamma})$ for all $\boldsymbol{\gamma} \in \mathcal{A}$.

We now show that problem (3.8) can be equivalently expressed as a minimization of the nonconvex function $\tilde{f}$ over an L-dimensional rectangle. To do this, let us define the following proposition:

Proposition 3.1. *Define the L-dimensional rectangle $\mathcal{Q}_{\text{init}}$ as*

$$\mathcal{Q}_{\text{init}} = \left\{ \boldsymbol{\gamma} \middle| \; 0 \leq \gamma_l \leq \frac{\|\hat{\mathbf{h}}_{ll}\|_2^2}{\sigma_l^2} p_{\text{tran}(l)}^{\max}, \quad l \in \mathcal{L} \; \right\}. \tag{3.14}$$

Then the feasible set of problem (3.9) is enclosed in $\mathcal{Q}_{\text{init}}$, i.e., $\mathcal{G} \subseteq \mathcal{Q}_{\text{init}}$.

Proof. By neglecting the interference terms in the denominator of SINR expression (3.6), let us define L-dimensional rectangle $\mathcal{Q}_0$ as

$$\mathcal{Q}_0 = \left\{ \boldsymbol{\gamma} \middle| \, 0 \leq \gamma_l \leq \sup_{\|\mathbf{m}_l\|_2^2 \leq p_{\text{tran}(l)}^{\max}} \inf_{\mathbf{e}_{ll} \in \mathcal{E}_{ll}} \frac{|(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{m}_l|^2}{\sigma_l^2}, l \in \mathcal{L} \right\}. \tag{3.15}$$

Clearly $\mathcal{G} \subseteq \mathcal{Q}_0$. Now we note that

$$\begin{aligned} &\sup_{\|\mathbf{m}_l\|_2^2 \leq p_{\text{tran}(l)}^{\max}} \inf_{\mathbf{e}_{ll} \in \mathcal{E}_{ll}} |(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{m}_l|^2 \\ &\qquad \leq \inf_{\mathbf{e}_{ll} \in \mathcal{E}_{ll}} \sup_{\|\mathbf{m}_l\|_2^2 \leq p_{\text{tran}(l)}^{\max}} |(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{m}_l|^2 && (3.16) \\ &\qquad = \inf_{\mathbf{e}_{ll} \in \mathcal{E}_{ll}} \|\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll}\|_2^2 p_{\text{tran}(l)}^{\max} && (3.17) \\ &\qquad \leq \|\hat{\mathbf{h}}_{ll}\|_2^2 p_{\text{tran}(l)}^{\max}, && (3.18) \end{aligned}$$

where the first inequality (3.16) follows by using max-min inequality [29], the second equality (3.17) follows by maximizing $|(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{m}_l|^2$ subject to the

constraint $\|\mathbf{m}_l\|_2^2 \leq p_{\text{tran}(l)}^{\max}$ (which can be obtained by using Cauchy-Schwartz inequality), and the third inequality (3.18) follows by choosing $\mathbf{e}_{ll} = 0$ [10]. Using inequality (3.18) in expression (3.15), it follows that $\mathcal{Q}_0 \subseteq \mathcal{Q}_{\text{init}}$, and hence, $\mathcal{G} \subseteq \mathcal{Q}_{\text{init}}$. □

Therefore, from Proposition 3.1 and expression (3.13), it follows that

$$\inf_{\boldsymbol{\gamma} \in \mathcal{Q}_{\text{init}}} \tilde{f}(\boldsymbol{\gamma}) = p^\star. \tag{3.19}$$

Thus, we have reformulated problem (3.8) equivalently as a minimization of the nonconvex function $\tilde{f}$ over the L-dimensional rectangle $\mathcal{Q}_{\text{init}}$. Hence, the BB Algorithm 2.1, presented in Section 2.2.2, can be used to solve problem (3.8). For clarity, we present the generic BB Algorithm 2.1 below.

Algorithm 3.1. *(Algorithm 2.1) Branch and bound algorithm*

1. Initialization: given tolerance $\epsilon > 0$. Set $k = 1$, $\mathcal{B}_1 = \{\mathcal{Q}_{\text{init}}\}$, $U_1 = \phi_{\text{ub}}(\mathcal{Q}_{\text{init}})$, and $L_1 = \phi_{\text{lb}}(\mathcal{Q}_{\text{init}})$.
2. Stopping criterion: if $U_k - L_k > \epsilon$ go to Step 3, otherwise STOP.
3. Branching:
   a) pick $\mathcal{Q} \in \mathcal{B}_k$ for which $\phi_{\text{lb}}(\mathcal{Q}) = L_k$ and set $\mathcal{Q}_k = \mathcal{Q}$.
   b) split $\mathcal{Q}_k$ along one of its longest edges into $\mathcal{Q}_I$ and $\mathcal{Q}_{II}$.
   c) Let $\mathcal{B}_{k+1} = (\mathcal{B}_k \backslash \{\mathcal{Q}_k\}) \cup \{\mathcal{Q}_I, \mathcal{Q}_{II}\}$.
4. Bounding:
   a) set $U_{k+1} = \min_{\mathcal{Q} \in \mathcal{B}_{k+1}} \{\phi_{\text{ub}}(\mathcal{Q})\}$.
   b) set $L_{k+1} = \min_{\mathcal{Q} \in \mathcal{B}_{k+1}} \{\phi_{\text{lb}}(\mathcal{Q})\}$.
5. Set $k = k + 1$ and go to step 2.

Let $\mathcal{Q} = \{\boldsymbol{\gamma} | \gamma_{l,\min} \leq \gamma_l \leq \gamma_{l,\max}, l \in \mathcal{L}\}$ denote an L-dimension rectangle, such that $\mathcal{Q} \subseteq \mathcal{Q}_{\text{init}}$. Then the lower bound function $\phi_{\text{lb}}(\mathcal{Q})$ and the upper bound function $\phi_{\text{ub}}(\mathcal{Q})$ for Algorithm 3.1 can be expressed as (see Section 2.3

[10] Note that we can choose any $\mathbf{e}_{ll}$ inside the uncertainty region $\mathcal{E}_{ll}$ to obtain an upper bound for (3.17).

for detailed explanation)

$$\phi_{\mathrm{lb}}(\mathcal{Q}) = \begin{cases} f_0(\boldsymbol{\gamma}_{\max}) & \boldsymbol{\gamma}_{\min} \in \mathcal{G} \\ 0 & \text{otherwise} , \end{cases} \tag{3.20}$$

and

$$\phi_{\mathrm{ub}}(\mathcal{Q}) = \tilde{f}(\boldsymbol{\gamma}_{\min}) = \begin{cases} f_0(\boldsymbol{\gamma}_{\min}) & \boldsymbol{\gamma}_{\min} \in \mathcal{G} \\ 0 & \text{otherwise} , \end{cases} \tag{3.21}$$

where $\boldsymbol{\gamma}_{\max} = \left[\gamma_{1,\max}, \ldots, \gamma_{L,\max}\right]^{\mathrm{T}}$, $\boldsymbol{\gamma}_{\min} = \left[\gamma_{1,\min}, \ldots, \gamma_{L,\min}\right]^{\mathrm{T}}$, and $\mathcal{G}$ is defined in (3.11). Note that the general expressions (3.20) and (3.21) satisfy the required conditions C1 and C2 that are defined in Section 2.2.2 (see Lemma 2.1), and they ensure the convergence of the BB algorithm (see Theorem 2.1) in the case of imperfect CSI. However, checking the condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$, which is central to calculating ϕ_{lb} and ϕ_{ub}, is much more difficult in the case of imperfect CSI. A method to check condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$ is derived in the next section.

3.2.2 Computation of upper and lower bounds

In this section we provide an efficient method based on SDR to check the condition $\boldsymbol{\gamma}_{\min} \in \mathcal{G}$, in order to compute the bounding functions (3.20) and (3.21).

Let $\boldsymbol{\gamma} = \left[\gamma_1, \ldots, \gamma_L\right]^{\mathrm{T}}$ be a specified set of SINR values. Checking the condition that these values are achievable (i.e., testing if $\boldsymbol{\gamma} \in \mathcal{G}$) is equivalent to solving the following feasibility problem [29, Sec. 4.1.1]:

$$\begin{array}{ll} \text{find} & \mathbf{m}_1, \ldots, \mathbf{m}_L \\ \text{subject to} & \gamma_l \leq \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l), \quad l \in \mathcal{L} \\ & \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \end{array} \tag{3.22}$$

with variable $\{\mathbf{m}_l\}_{l \in \mathcal{L}}$. Note that the SINR expression $\Gamma_l(\mathbf{m}, \mathbf{e}_l)$ has common variable $\mathbf{e}_{ll}$ in the numerator and denominator $\big($see (3.6)$\big)$. Hence, it is hard to solve $\inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l)$, and express it in a tractable representation.

By using the definition of set $\mathcal{E}_{jl}$ $\big($see (3.5)$\big)$ let us first equivalently rewrite problem (3.22) as

$$\begin{array}{ll} \text{find} & \mathbf{m}_1, \ldots, \mathbf{m}_L \\ \text{subject to} & \gamma_l \leq \Gamma_l(\mathbf{m}, \mathbf{e}_l), \quad l \in \mathcal{L} \\ & \mathbf{e}_{jl}^{\mathrm{H}} \mathbf{Q}_{jl} \mathbf{e}_{jl} \leq 1, \quad j, l \in \mathcal{L} \\ & \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \end{array} \tag{3.23}$$

with variables $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ and $\{\mathbf{e}_{jl}\}_{j,l\in\mathcal{L}}$. Note that in problem (3.23) we have used a standard trick to express the first inequality constraints of problem (3.22) as a set of separate inequalities [29, Sec. 4.3.1].

Next, we use $\mathcal{S}$-lemma [29, 181] to replace the first and second inequality constraints of problem (3.23) with linear matrix inequalities (LMIs), and then apply a semidefinite relaxation (SDR) technique [182] to determine the feasibility of problem (3.23). Furthermore, we replace the dummy objective function of problem (3.23) with a minimization of the total sum-power of the network. Note that the new objective function will ensure that the SDR is tight, as will be clear shortly.

Let us first introduce a new variable I_{jl} to denote the power of the out-of-cell interference from jth data stream to rec(l) as

$$I_{jl} = |(\hat{\mathbf{h}}_{jl} + \mathbf{e}_{jl})^{\mathrm{H}}\mathbf{m}_j|^2,$$

for all $l \in \mathcal{L}, n \in \mathcal{N}\backslash\{\text{tran}(l)\}, j \in \mathcal{L}(n)$. Then by using the expression of $\Gamma_l(\mathbf{m}, \mathbf{e}_l)$ (see (3.6)) and replacing the objective function of problem (3.23) with the minimization of the total sum-power of the network, we modify problem (3.23) as

$$\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{l\in\mathcal{L}} \|\mathbf{m}_l\|_2^2 \\
\text{subject to} \quad & (\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \left[\frac{\mathbf{m}_l\mathbf{m}_l^{\mathrm{H}}}{\gamma_l} - \sum_{\substack{j\in\mathcal{L}(\text{tran}(l)) \\ j\neq l}} \mathbf{m}_j\mathbf{m}_j^{\mathrm{H}} \right] (\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll}) \\
& \geq \sum_{n\in\mathcal{N}\backslash\{\text{tran}(l)\}} \textstyle\sum_{j\in\mathcal{L}(n)} I_{jl} + \sigma_l^2, \quad l \in \mathcal{L} && \text{(3.24a)} \\
& \mathbf{e}_{ll}^{\mathrm{H}}\mathbf{Q}_{ll}\mathbf{e}_{ll} \leq 1, \quad l \in \mathcal{L} && \text{(3.24b)} \\
& (\hat{\mathbf{h}}_{jl} + \mathbf{e}_{jl})^{\mathrm{H}}\mathbf{m}_j\mathbf{m}_j^{\mathrm{H}}(\hat{\mathbf{h}}_{jl} + \mathbf{e}_{jl}) \leq I_{jl}, \quad l \in \mathcal{L}, \\
& \qquad\qquad n \in \mathcal{N}\backslash\{\text{tran}(l)\}, j \in \mathcal{L}(n) && \text{(3.24c)} \\
& \mathbf{e}_{jl}^{\mathrm{H}}\mathbf{Q}_{jl}\mathbf{e}_{jl} \leq 1, \quad l \in \mathcal{L}, n \in \mathcal{N}\backslash\{\text{tran}(l)\}, j \in \mathcal{L}(n) && \text{(3.24d)} \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \mathbf{m}_l^{\mathrm{H}}\mathbf{m}_l \leq p_n^{\max}, \quad n \in \mathcal{N}, && \text{(3.24e)}
\end{aligned}$$

where the optimization variables are $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$, $\{I_{jl}\}_{j,l\in\mathcal{L}}$, and $\{\mathbf{e}_{jl}\}_{j,l\in\mathcal{L}}$. Note that in problem (3.24) we have written the second inequality constraint of problem (3.23) as two separate inequalities (3.24b) and (3.24d). Since the objective function of problem (3.24) is decreasing in variable $\mathbf{m}_l$, it can be easily

shown (e.g., by contradiction) that constraints (3.24c) hold with equality at the optimal point. Hence, the feasibility of problem (3.22) can be determined by solving problem (3.24).

Let us now introduce a new variable $\tilde{\mathbf{M}}_l = \mathbf{m}_l \mathbf{m}_l^{\mathrm{H}}$ such that $\mathrm{Rank}(\tilde{\mathbf{M}}_l) = 1$, for all $l \in \mathcal{L}$. Then by using $\mathcal{S}$-lemma [29, 181] and then applying SDR technique [182], we determine a solution to problem (3.24) by solving the following SDP (the detail derivation is provided in Appendix 1).

$$
\begin{aligned}
\text{minimize} \quad & \textstyle\sum_{l\in\mathcal{L}} \mathrm{Trace}(\tilde{\mathbf{M}}_l) \\
\text{subject to} \quad & \begin{bmatrix} \mathbf{V}_l & \mathbf{V}_l\hat{\mathbf{h}}_{ll} \\ \hat{\mathbf{h}}_{ll}^{\mathrm{H}}\mathbf{V}_l & \hat{\mathbf{h}}_{ll}^{\mathrm{H}}\mathbf{V}_l\hat{\mathbf{h}}_{ll} - \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum\limits_{j\in\mathcal{L}(n)} I_{jl} - \sigma_n^2 \end{bmatrix} \\
& + \mu_{ll} \begin{bmatrix} \mathbf{Q}_{ll} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \quad l \in \mathcal{L} \\
& \begin{bmatrix} -\tilde{\mathbf{M}}_j & -\tilde{\mathbf{M}}_j\hat{\mathbf{h}}_{jl} \\ -\hat{\mathbf{h}}_{jl}^{\mathrm{H}}\tilde{\mathbf{M}}_j & I_{jl} - \hat{\mathbf{h}}_{jl}^{\mathrm{H}}\tilde{\mathbf{M}}_j\hat{\mathbf{h}}_{jl} \end{bmatrix} + \mu_{jl} \begin{bmatrix} \mathbf{Q}_{jl} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \\
& \qquad\qquad l \in \mathcal{L}, n \in \mathcal{N}\setminus\{\mathrm{tran}(l)\}, j \in \mathcal{L}(n) \\
& \mu_{jl} \geq 0, \quad j,l \in \mathcal{L} \\
& \tilde{\mathbf{M}}_l \succeq 0, \quad l \in \mathcal{L} \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \mathrm{Trace}(\tilde{\mathbf{M}}_l) \leq p_n^{\max}, \quad n \in \mathcal{N},
\end{aligned}
\tag{3.25}
$$

with variables $\{\mu_{jl}\}_{j,l\in\mathcal{L}}$, $\{I_{jl}\}_{j,l\in\mathcal{L}}$, and $\{\tilde{\mathbf{M}}_l\}_{l\in\mathcal{L}}$, where $\mathbf{V}_l$ is

$$
\mathbf{V}_l = \frac{\tilde{\mathbf{M}}_l}{\gamma_l} - \sum_{j\in\mathcal{L}(\mathrm{tran}(l)), j\neq l} \tilde{\mathbf{M}}_j. \tag{3.26}
$$

Note that in problem (3.25) constraint $\mathrm{Rank}(\tilde{\mathbf{M}}_l) = 1$ for all $l \in \mathcal{L}$ is dropped.

Let us denote $\tilde{\mathbf{M}}_l^\star$, for all $l \in \mathcal{L}$, the optimal solution of problem (3.25). From $\tilde{\mathbf{M}}_l^\star$ we can find $\mathbf{m}_l^\star$ that is feasible for the original problem (3.22), if the rank of $\tilde{\mathbf{M}}_l^\star$ is one for all $l \in \mathcal{L}$. Note that due to the minimization of the total sum-power as the objective function, problem (3.25) yields a low rank solution $\tilde{\mathbf{M}}_l^\star$ [11]. In fact, the existence of a rank one solution (for the sum-power minimization problem) has been recently proven in [98, 183] under some technical assumptions in the system model. The technical assumptions that guarantee the

[11] $\mathrm{Trace}(\tilde{\mathbf{M}}_l)$ is the sum of the eigenvalues of $\tilde{\mathbf{M}}_l$. Thus, the objective of problem (3.25) is equivalent to the minimization of the sum of eigenvalues of the positive semidefinite matrix $\tilde{\mathbf{M}}_l$. Hence, problem (3.25) is similar to ℓ_1-norm minimization problem [29, Sec. 6.2]. Therefore, problem (3.25) yields a low rank solution $\tilde{\mathbf{M}}_l^\star$.

rank one solution are: a) only a single user in each cell, b) error in inter-cell CSI, but perfect intra-cell CSI is available, and c) small errors with the bound, provided in [183], in the length of the principal semi-axis of the ellipsoids. Thus, if any one of these conditions are satisfied, problem (3.25) yields a rank one solution. We refer the interested reader to [98, Proposition 1] and [183] for detailed explanations.

3.3 Suboptimal fast-converging algorithm

In this section we derive a fast but possibly suboptimal algorithm for the worst-case WSRMax problem (3.7). The proposed algorithm is based on the alternating optimization technique, and is derived in conjunction with sequential convex programming [35].

We start by expressing the transmit beamformer $\mathbf{m}_l$ associated with lth data stream as

$$\mathbf{m}_l = \sqrt{p}_l \mathbf{v}_l, \tag{3.27}$$

where $p_l \in \mathbb{R}_+$ and $\mathbf{v}_l \in \mathbb{C}^T$ denote the power and transmit direction associated with lth data stream, i.e., $p_l = \|\mathbf{m}_l\|_2^2$ and $\mathbf{v}_l = \mathbf{m}_l/\|\mathbf{m}_l\|_2$. Then the SINR expression (3.6) can be expressed as

$$\Gamma_l(\mathbf{p}, \mathbf{v}, \mathbf{e}_l) = \frac{p_l |(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{v}_l|^2}{\sigma_l^2 + \sum\limits_{j \in \mathcal{L}(\mathrm{tran}(l)), j \neq l} p_j |(\hat{\mathbf{h}}_{ll} + \mathbf{e}_{ll})^{\mathrm{H}} \mathbf{v}_j|^2 + \sum\limits_{n \in \mathcal{N} \setminus \{\mathrm{tran}(l)\}} \sum_{j \in \mathcal{L}(n)} p_j |(\hat{\mathbf{h}}_{jl} + \mathbf{e}_{jl})^{\mathrm{H}} \mathbf{v}_j|^2}, \tag{3.28}$$

where $\mathbf{p} = [p_1, \ldots, p_L]^{\mathrm{T}}$ and the notation $\mathbf{v}$ denotes a vector obtained by stacking $\mathbf{v}_l$ for all $l \in \mathcal{L}$ on top of each other, i.e., $\mathbf{v} = [\mathbf{v}_1^{\mathrm{T}}, \ldots, \mathbf{v}_L^{\mathrm{T}}]^{\mathrm{T}}$.

Thus, problem (3.8) can be equivalently expressed as

$$\begin{array}{ll} \text{minimize} & \sum_{l \in \mathcal{L}} -\beta_l \log_2(1 + \gamma_l) \\ \text{subject to} & \gamma_l \leq \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{p}, \mathbf{v}, \mathbf{e}_l), \quad l \in \mathcal{L} \\ & \sum_{l \in \mathcal{L}(n)} p_l \|\mathbf{v}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N} \\ & \|\mathbf{v}_l\|_2^2 = 1, p_l \geq 0, \quad l \in \mathcal{L}, \end{array} \tag{3.29}$$

with variables $\{\gamma_l\}_{l \in \mathcal{L}}$, $\{p_l\}_{l \in \mathcal{L}}$, and $\{\mathbf{v}_l\}_{l \in \mathcal{L}}$. Note that problem (3.29) is NP-hard [30]. Hence, to find a fast-converging algorithm we have to rely on local optimization methods [184].

The basic idea of the proposed algorithm is to solve problem (3.29) with respect to different subsets of variables by considering the others to be fixed. Here, we first fix variable $\{\mathbf{v}_l\}_{l\in\mathcal{L}}$ and solve problem (3.29) for variable $\{\gamma_l, p_l\}_{l\in\mathcal{L}}$; we refer to this problem as *Subproblem 1*. Then by keeping $\{\gamma_l\}_{l\in\mathcal{L}}$ fixed we update variable $\{p_l, \mathbf{v}_l\}_{l\in\mathcal{L}}$, so that the objective of problem (3.29) can be further decreased; we refer to this problem as *Subproblem 2*. Unfortunately, both subproblems are not convex problems. To derive a fast-converging algorithm we approximate Subproblem 1 and Subproblem 2 by convex problems. The proposed suboptimal algorithm alternatively iterates between the approximated Subproblem 1 and Subproblem 2. Note that a somewhat similar technique has been used in [83], [176, Sec. 4.3] in the context of multi user MIMO downlink systems with perfect CSI at BSs. However, the WSRMax problem addressed in this section is substantially more challenging due to CSI uncertainty.

3.3.1 Subproblem 1: update SINR and power

For fixed beamformers $\mathbf{v}_l$ for all $l \in \mathcal{L}$, problem (3.29) can be expressed as

$$\begin{array}{ll} \text{minimize} & \prod_{l\in\mathcal{L}}(1+\gamma_l)^{-\beta_l} \\ \text{subject to} & \gamma_l \leq \inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l), \quad l\in\mathcal{L} \\ & \sum_{l\in\mathcal{L}(n)} p_l \leq p_n^{\max}, \quad n\in\mathcal{N} \\ & p_l \geq 0, \quad l\in\mathcal{L}. \end{array} \tag{3.30}$$

with variables $\{\gamma_l\}_{l\in\mathcal{L}}$ and $\{p_l\}_{l\in\mathcal{L}}$. Recall that the SINR expression $\Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$ consists of common variable $\mathbf{e}_{ll}$ in the numerator and denominator (see (3.28)). Hence, it is hard to solve $\inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$, and express the first inequality constraints of problem (3.30) in a tractable representation. Thus, we replace $\inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$ with its lower bound.

To find a lower bound for $\inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$, we ignore the dependence of numerator and denominator terms of $\Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$ due to the common variable $\mathbf{e}_{ll}$. Then by independently finding the worst-case numerator and the worst-case denominator terms of $\Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$, the lower bound for $\inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$ can be expressed (a similar approach is also used in [102]) as

$$\tilde{\Gamma}_l(\mathbf{p},\mathbf{v}) = \frac{\inf\limits_{\mathbf{e}_{ll}\in\mathcal{E}_{ll}} \left(p_l|(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{v}_l|^2\right)}{\sigma_l^2 + \sup\limits_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \left(\sum\limits_{\substack{j\in\mathcal{L}(\mathrm{tran}(l)) \\ j\neq l}} p_j|(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{v}_j|^2 + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum\limits_{j\in\mathcal{L}(n)} p_j|(\hat{\mathbf{h}}_{jl}+\mathbf{e}_{jl})^{\mathrm{H}}\mathbf{v}_j|^2 \right)}. \tag{3.31}$$

To simplify (3.31), let us introduce new variables $\underline{g}_{ll}$ and $\bar{g}_{jl}$ for all $j \in \mathcal{L}$ defined as

$$\underline{g}_{ll} \triangleq \inf_{\mathbf{e}_{ll}\in\mathcal{E}_{ll}} |(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{v}_l|^2, \tag{3.32}$$

$$\bar{g}_{jl} \triangleq \sup_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}} |(\hat{\mathbf{h}}_{jl}+\mathbf{e}_{jl})^{\mathrm{H}}\mathbf{v}_j|^2, \quad j\in\mathcal{L}. \tag{3.33}$$

Now we note that

$$\begin{aligned} \inf_{\mathbf{e}_{ll}\in\mathcal{E}_{ll}} |(\hat{\mathbf{h}}_{ll}+\mathbf{e}_{ll})^{\mathrm{H}}\mathbf{v}_l| &\geq \inf_{\mathbf{e}_{ll}\in\mathcal{E}_{ll}} \left(|\hat{\mathbf{h}}_{ll}^{\mathrm{H}}\mathbf{v}_l| - |\mathbf{e}_{ll}^{\mathrm{H}}\mathbf{v}_l|\right) \\ &= |\hat{\mathbf{h}}_{ll}^{\mathrm{H}}\mathbf{v}_l| - \sqrt{\mathbf{v}_l^{\mathrm{H}}\mathbf{Q}_{ll}^{-1}\mathbf{v}_l}, \end{aligned} \tag{3.34}$$

and

$$\begin{aligned} \sup_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}} |(\hat{\mathbf{h}}_{jl}+\mathbf{e}_{jl})^{\mathrm{H}}\mathbf{v}_j| &\leq \sup_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}} \left(|\hat{\mathbf{h}}_{jl}^{\mathrm{H}}\mathbf{v}_j| + |\mathbf{e}_{jl}^{\mathrm{H}}\mathbf{v}_j|\right) \\ &= |\hat{\mathbf{h}}_{jl}^{\mathrm{H}}\mathbf{v}_j| + \sqrt{\mathbf{v}_j^{\mathrm{H}}\mathbf{Q}_{jl}^{-1}\mathbf{v}_j}, \quad j\in\mathcal{L}, \end{aligned} \tag{3.35}$$

where the first inequality of (3.34) and (3.35) follows by using triangle inequality, and the second inequality of (3.34) and (3.35) follows by using Cauchy-Schwarz inequality along with $\mathbf{e}_{ll}^{\mathrm{H}}\mathbf{Q}_{ll}\mathbf{e}_{ll} \leq 1$ and $\mathbf{e}_{jl}^{\mathrm{H}}\mathbf{Q}_{jl}\mathbf{e}_{jl} \leq 1$, respectively (see expression (3.5) for the definition of $\mathcal{E}_{jl}$). Then by applying the results of (3.34) and (3.35) in expressions (3.32) and (3.33), respectively, we get

$$\underline{g}_{ll} = \left|\left(|\hat{\mathbf{h}}_{ll}^{\mathrm{H}}\mathbf{v}_l| - \sqrt{\mathbf{v}_l^{\mathrm{H}}\mathbf{Q}_{ll}^{-1}\mathbf{v}_l}\right)^+\right|^2, \tag{3.36}$$

$$\bar{g}_{jl} = \left||\hat{\mathbf{h}}_{jl}^{\mathrm{H}}\mathbf{v}_j| + \sqrt{\mathbf{v}_j^{\mathrm{H}}\mathbf{Q}_{jl}^{-1}\mathbf{v}_j}\right|^2, \quad j\in\mathcal{L}, \tag{3.37}$$

where we have used $(x)^+$ for $\max(x,0)$. Therefore, by using a lower bound function $\tilde{\Gamma}_l(\mathbf{p},\mathbf{v})$ for $\inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p},\mathbf{v},\mathbf{e}_l)$, the solution to problem (3.30) can

be approximated by solving the following optimization problem:

$$\begin{array}{ll} \text{minimize} & \prod_{l\in\mathcal{L}}(1+\gamma_l)^{-\beta_l} \\ \text{subject to} & \gamma_l \le \dfrac{p_l \underline{g}_{ll}}{\sigma_l^2 + \sum\limits_{\substack{j\,\in\,\mathcal{L}(\mathrm{tran}(l)),\\ j\neq l}} p_j \bar{g}_{jl} + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum_{j\in\mathcal{L}(n)} p_j \bar{g}_{jl}}, \quad l\in\mathcal{L} \\ & \sum_{l\in\mathcal{L}(n)} p_l \le p_n^{\max}, \quad n\in\mathcal{N} \\ & p_l \ge 0, \quad l\in\mathcal{L}, \end{array} \tag{3.38}$$

with variables $\{\gamma_l\}_{l\in\mathcal{L}}$ and $\{p_l\}_{l\in\mathcal{L}}$.

Note that problem (3.38) is still a nonconvex problem. However, it can be easily expressed as a signomial optimization problem [79, 80]. Hence, a close local solution for problem (3.38) can be efficiently obtained by solving a sequence of GP which locally approximates the original problem.

To do this, we approximate the objective function of problem (3.38) near an arbitrary positive point $\hat{\boldsymbol{\gamma}} = [\hat{\gamma}_1, \ldots, \hat{\gamma}_L]^{\mathrm{T}}$ by a monomial function. Let $\check{f}(\boldsymbol{\gamma})$ denote the objective function of problem (3.38), i.e, $\check{f}(\boldsymbol{\gamma}) = \prod_{l\in\mathcal{L}}(1+\gamma_l)^{-\beta_l}$. Then a monomial function $m(\boldsymbol{\gamma}) = \hat{d}\prod_{l\in\mathcal{L}}\gamma_l^{\hat{\alpha}_l}$ is the best local approximation of $\check{f}(\boldsymbol{\gamma})$ near $\hat{\boldsymbol{\gamma}}$ if [80, Sec. 8], [83, Lem. 3]

$$m(\hat{\boldsymbol{\gamma}}) = \check{f}(\hat{\boldsymbol{\gamma}}), \text{ and } \nabla m(\hat{\boldsymbol{\gamma}}) = \nabla \check{f}(\hat{\boldsymbol{\gamma}}). \tag{3.39}$$

By solving expressions in (3.39), we can obtain parameters $\hat{d}$ and $\hat{\alpha}_l$, and they are given by

$$\hat{d} = \prod_{l\in\mathcal{L}} \big(\hat{\gamma}_l^{-\hat{\gamma}_l/(1+\hat{\gamma}_l)}(1+\hat{\gamma}_l)\big)^{-\beta_l}, \quad \hat{\alpha}_l = -\beta_l\hat{\gamma}_l/(1+\hat{\gamma}_l). \tag{3.40}$$

Therefore, the solution of problem (3.38), by using monomial function $m(\boldsymbol{\gamma})$, near an arbitrary positive point $\hat{\boldsymbol{\gamma}}$ can be approximated by solving following GP:

$$\begin{array}{ll} \text{minimize} & \prod\limits_{l\in\mathcal{L}} \big(\hat{\gamma}_l^{-\hat{\gamma}_l/(1+\hat{\gamma}_l)}(1+\hat{\gamma}_l)\big)^{-\beta_l} \prod\limits_{l\in\mathcal{L}} \gamma_l^{-\beta_l\hat{\gamma}_l/(1+\hat{\gamma}_l)} \\ \text{subject to} & \underline{g}_{ll}^{-1} p_l^{-1}\gamma_l \Big(\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(\mathrm{tran}(l)),\, j\neq l} p_j\bar{g}_{jl} \\ & \quad + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum_{j\in\mathcal{L}(n)} p_j\bar{g}_{jl}\Big) \le 1, \quad l\in\mathcal{L} \\ & \sum_{l\in\mathcal{L}(n)} p_l \le p_n^{\max}, \quad n\in\mathcal{N} \\ & p_l \ge 0, \quad l\in\mathcal{L} \\ & \alpha^{-1}\hat{\gamma}_l \le \gamma_l \le \alpha\hat{\gamma}_l, \quad l\in\mathcal{L}, \end{array} \tag{3.41}$$

with variables $\{\gamma_l\}_{l\in\mathcal{L}}$ and $\{p_l\}_{l\in\mathcal{L}}$, where $\alpha > 0$. In problem (3.41), the fourth set of constraints is called the trust region [80], and it limits the domain of γ_l such that the monomial approximation of the objective function near $\hat{\boldsymbol{\gamma}}$ is accurate enough.

Note that problem (3.41) approximates the solution for problem (3.38) near an arbitrary point $\hat{\boldsymbol{\gamma}}$. Hence, to obtain the best local solution to problem (3.38), we need to solve problem (3.41) repeatedly for different values of $\hat{\boldsymbol{\gamma}}$. Thus, we take the solution $\gamma_l^\star$ of problem (3.41) as the next iterate $\hat{\gamma}_l$ (i.e., we set $\hat{\gamma}_l = \gamma_l^\star$ for all $l \in \mathcal{L}$), then solve problem (3.41). This step is repeated until convergence.

3.3.2 Subproblem 2: update beamformers

Here, we fix variable $\{\gamma_l\}_{l\in\mathcal{L}}$ and update $\{p_l, \mathbf{v}_l\}_{l\in\mathcal{L}}$. To do this, we formulate an optimization problem that finds a power margin such that the SINR value γ_l, for all $l \in \mathcal{L}$, are preserved. This can be cast as the following optimization problem:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \gamma_l \leq \inf_{\mathbf{e}_{jl}\in\mathcal{E}_{jl}, j\in\mathcal{L}} \Gamma_l(\mathbf{p}, \mathbf{v}, \mathbf{e}_l), \quad l \in \mathcal{L} \\ & \sum_{l\in\mathcal{L}(n)} p_l \|\mathbf{v}_l\|_2^2 \leq t p_n^{\max}, \quad n \in \mathcal{N} \\ & \|\mathbf{v}_l\|_2^2 \ = 1, p_l \geq 0 \quad l \in \mathcal{L}. \end{array} \tag{3.42}$$

where the optimization variables are t, $\{\mathbf{v}_l\}_{l\in\mathcal{L}}$, and $\{p_l\}_{l\in\mathcal{L}}$. Note that the solutions $t^\star$, $\{\mathbf{v}_l^\star\}_{l\in\mathcal{L}}$, and $\{p_l^\star\}_{l\in\mathcal{L}}$ of problem (3.42) do not directly decrease the objective of the original problem (3.29). However, they provide the power margin, i.e., $p_n^{\max} - \sum_{l\in\mathcal{L}(n)} p_l^\star \geq 0$ for all $n \in \mathcal{N}$. This positive power margin implies that the objective of problem (3.29) can be decreased, further, by solving Subproblem 1 for $\mathbf{v}_l$ fixed to $\mathbf{v}_l^\star$ for all $l \in \mathcal{L}$. It is worth noting that $t^\star \leq 1$ for any $\{\gamma_l\}_{l\in\mathcal{L}}$ that are feasible for problem (3.29).

Note that by changing variables p_l and $\mathbf{v}_l$ back to $\mathbf{m}_l$ $\big($see (3.27)$\big)$, we can express the constraint set of problem (3.42) similar to that of problem (3.22). Then the method proposed in Section 3.2.2 to solve problem (3.22) can be applied to solve problem (3.42). Hence, by using expression (3.27), let us rewrite problem (3.42) equivalently as

$$
\begin{aligned}
&\text{minimize} \quad t \\
&\text{subject to} \quad \gamma_l \leq \inf_{\mathbf{e}_{jl} \in \mathcal{E}_{jl}, j \in \mathcal{L}} \Gamma_l(\mathbf{m}, \mathbf{e}_l), \quad l \in \mathcal{L} \\
&\qquad\qquad \textstyle\sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq t p_n^{\max}, \quad n \in \mathcal{N},
\end{aligned} \tag{3.43}
$$

with variables t and $\{\mathbf{m}_l\}_{l \in \mathcal{L}}$, where $\Gamma_l(\mathbf{m}, \mathbf{e}_l)$ is defined in (3.6). Then by following the approach of steps (3.23)-(3.25) to handle the constraints of problem (3.43), the solution for problem (3.43) can be determined by solving the following SDP:

$$
\begin{aligned}
&\text{minimize} \quad t \\
&\text{subject to} \quad \begin{bmatrix} \mathbf{V}_l & \mathbf{V}_l \hat{\mathbf{h}}_{ll} \\ \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l & \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l \hat{\mathbf{h}}_{ll} - \sum\limits_{n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}} \sum\limits_{j \in \mathcal{L}(n)} I_{jl} - \sigma_n^2 \end{bmatrix} \\
&\qquad + \mu_{ll} \begin{bmatrix} \mathbf{Q}_{ll} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \quad l \in \mathcal{L} \\
&\qquad \begin{bmatrix} -\tilde{\mathbf{M}}_j & -\tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \\ -\hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j & I_{jl} - \hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \end{bmatrix} + \mu_{jl} \begin{bmatrix} \mathbf{Q}_{jl} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \\
&\qquad\qquad\qquad\qquad l \in \mathcal{L}, n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}, j \in \mathcal{L}(n) \\
&\qquad \mu_{jl} \geq 0, \quad j, l \in \mathcal{L} \\
&\qquad \tilde{\mathbf{M}}_l \succeq 0, \quad l \in \mathcal{L} \\
&\qquad \textstyle\sum_{l \in \mathcal{L}(n)} \mathrm{Trace}(\tilde{\mathbf{M}}_l) \leq t p_n^{\max}, \quad n \in \mathcal{N},
\end{aligned} \tag{3.44}
$$

with variables t, $\{\tilde{\mathbf{M}}_l\}_{l \in \mathcal{L}}$, $\{\mu_{jl}\}_{j,l \in \mathcal{L}}$, and $\{I_{jl}\}_{j,l \in \mathcal{L}}$, where $\tilde{\mathbf{M}}_l = \mathbf{m}_l \mathbf{m}_l^{\mathrm{H}}$ and $\mathbf{V}_l = \tilde{\mathbf{M}}_l / \gamma_l - \sum_{j \in \mathcal{L}(\mathrm{tran}(l)), j \neq l} \tilde{\mathbf{M}}_j$. Note that we have used SDR technique to arrive at problem (3.44), hence $\mathrm{Rank}(\tilde{\mathbf{M}}_l) = 1$ constraint for all $l \in \mathcal{L}$ has been dropped in problem (3.44).

If the solution of problem (3.44) is rank one (i.e, $\mathrm{Rank}(\tilde{\mathbf{M}}_l^\star) = 1$ for all $l \in \mathcal{L}$), we can recover optimal $\mathbf{m}_l^\star$ from $\tilde{\mathbf{M}}_l^\star$ for problem (3.43) for all $l \in \mathcal{L}$. However, if $\mathrm{Rank}(\tilde{\mathbf{M}}_l^\star) \geq 1$, we approximate the solution $\mathbf{m}_l^\star$ by using the dominant eigenvector and eigenvalue of $\tilde{\mathbf{M}}_l^\star$ (interestingly, in all our numerical simulations we have noted that the solution is always rank one). Then by using relation (3.27) we recover $p_l^\star$ and $\mathbf{v}_l^\star$ from $\mathbf{m}_l^\star$ for problem (3.42).

Finally, we summarize the proposed suboptimal fast-converging algorithm for the worst-case WSRMax problem (3.29) in Algorithm 3.2.

Algorithm 3.2. *Suboptimal algorithm for the worst-case WSRMax problem (3.29)*

1. Initialization: given positive definite matrix $\{\mathbf{Q}_{jl}\}_{j,l\in\mathcal{L}}$, feasible initial beamformer $\{\mathbf{v}_l^{(0)}\}_{l\in\mathcal{L}}$, power allocation $\{p_l^{(0)}\}_{l\in\mathcal{L}}$ for problem (3.29), and tolerance $\epsilon_\gamma > 0$. Set iteration index $k = 0$.
2. Solve Subproblem 1: By fixing $\mathbf{v}_l = \mathbf{v}_l^{(k)}$ for all $l \in \mathcal{L}$,
   a) compute $\hat{\gamma}_l$ for all $l \in \mathcal{L}$ as follows

$$\hat{\gamma}_l = \frac{p_l^{(k)} \underline{g}_{ll}}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(\mathrm{tran}(l)),\, j\neq l} p_j^{(k)} \bar{g}_{jl} + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} \sum_{j\in\mathcal{L}(n)} p_j^{(k)} \bar{g}_{jl}},$$

   where $\underline{g}_{ll} = \left| \left(|\hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{v}_l| - \sqrt{\mathbf{v}_l^{\mathrm{H}} \mathbf{Q}_{ll}^{-1} \mathbf{v}_l} \right)^+ \right|^2$ and $\bar{g}_{jl} = \left| |\hat{\mathbf{h}}_{jl}^{\mathrm{H}} \mathbf{v}_j| + \sqrt{\mathbf{v}_j^{\mathrm{H}} \mathbf{Q}_{jl}^{-1} \mathbf{v}_j} \right|^2$.
   b) solve problem (3.41) for variables $\{p_l, \gamma_l\}_{l\in\mathcal{L}}$. Denote the solution by $\{p_l^\star, \gamma_l^\star\}_{l\in\mathcal{L}}$.
   c) if $\max_{l\in\mathcal{L}} |\hat{\gamma}_l - \gamma_l^\star| > \epsilon_\gamma$ set $\hat{\gamma}_l = \gamma_l^\star$ for all $l \in \mathcal{L}$ and repeat step 2(b). Otherwise, set $p_l^{(k)} = p_l^\star$ and $\gamma_l^{(k)} = \gamma_l^\star$ for $l \in \mathcal{L}$, and go to step 3.
3. Stopping criterion: if the stopping criterion is satisfied STOP by returning the suboptimal solution $\{p_l^{(k)}, \gamma_l^{(k)}, \mathbf{v}_l^{(k)}\}_{l\in\mathcal{L}}$. Otherwise, go to step 4.
4. Solve Subproblem 2: By fixing $\gamma_l = \gamma_l^{(k)}$ for all $l \in \mathcal{L}$, solve problem (3.44). Denote the solution by $t^\star$ and $\{\tilde{\mathbf{M}}_l^\star\}_{l\in\mathcal{L}}$. Set $\mathbf{m}_l^\star = \sqrt{\lambda_l}\mathbf{u}_l$, where λ_l is the dominant eigenvalue of $\tilde{\mathbf{M}}_l^\star$ and $\mathbf{u}_l$ is eigenvector that corresponds to λ_l, for all $l \in \mathcal{L}$.
5. Update $p_l^{(k+1)} = \|\mathbf{m}_l^\star\|_2^2 / t^\star$ and $\mathbf{v}_l^{(k+1)} = \mathbf{m}_l^\star / \|\mathbf{m}_l^\star\|_2$ for all $l \in \mathcal{L}$. Set $k = k + 1$ and go to step 2.

The first step initializes the algorithm. Step 2 solves the approximated Subproblem 1, i.e., problem (3.41). In step 2, problem (3.41) is solved repeatedly such that the solution $\{\gamma_l^\star\}_{l\in\mathcal{L}}$ is within a desired tolerance[12]. Step 3 checks the stopping criteria; here, the algorithm is stopped when the difference between the achieved WSR values between the successive iterations is less than a given

[12]In order to reduce the computational cost of Subproblem 1, in step 2, problem (3.41) can be solved for a fixed number of iteration.

threshold. Step 4 solves the approximated Subproblem 2, i.e., problem (3.44). Note that problem (3.41) is a GP and problem (3.44) is a SDP, hence both problems can be solved efficiently by using interior point methods [29]. Finally, step 5 updates the transmit power and beamforming direction, then repeats the iteration from step 2.

3.3.3 Convergence of suboptimal algorithm

Algorithm 3.2 monotonically converges, if the solution of Subproblem 2 (i.e., variable $\tilde{\mathbf{M}}_l^\star$ for all $l \in \mathcal{L}$ in step 4 of the algorithm) is rank one. This is because with a rank one solution of Subproblem 2, there is no loss in the beamforming gain by the rank one approximation, and the saved power by solving Subproblem 2 can be used, further, to decrease the objective of Subproblem 1 [13]. This can be realized by setting $\mathbf{v}_l = \mathbf{m}_l^\star / \|\mathbf{m}_l^\star\|_2$, $p_l = \|\mathbf{m}_l^\star\|_2^2 / t^\star$, and increasing γ_l such that the SINR constraints of problem (3.38) are tight for all $l \in \mathcal{L}$. Unfortunately, if Subproblem 2 does not have a rank one solution, the monotonic convergence of Algorithm 3.2 cannot be guaranteed. In such situations, we can run the algorithm for a finite number of iterations, and take the best solution.

3.4 Numerical examples

We illustrate the performance of the proposed BB Algorithm 3.1 and Algorithm 3.2 with the setup $N = 3$ BSs and $T = 2$ transmit antennas at each one (see Fig. 3.1). The BSs are placed in such a way that they form an equilateral triangle. The distance between the BSs is denoted by D_{BS}. We assume circular cells, where the radius of each one is denoted by R_{BS}. For simplicity, we assume that the network is transmitting $L = 6$ data streams, 2 streams per each BS. The locations of the users associated with each data stream are arbitrarily chosen as shown in Fig. 3.1.

We assume an exponential path loss model, where the channel vector from the transmitter of data stream j (i.e., BS tran(j)) to the receiver of data stream l (i.e., user rec(l)) is modeled as

$$\hat{\mathbf{h}}_{jl} = \left(\frac{d_{jl}}{d_0}\right)^{-\eta/2} \mathbf{c}_{jl}, \tag{3.45}$$

[13] Note that we solve problem (3.38) for solving Subproblem 1.

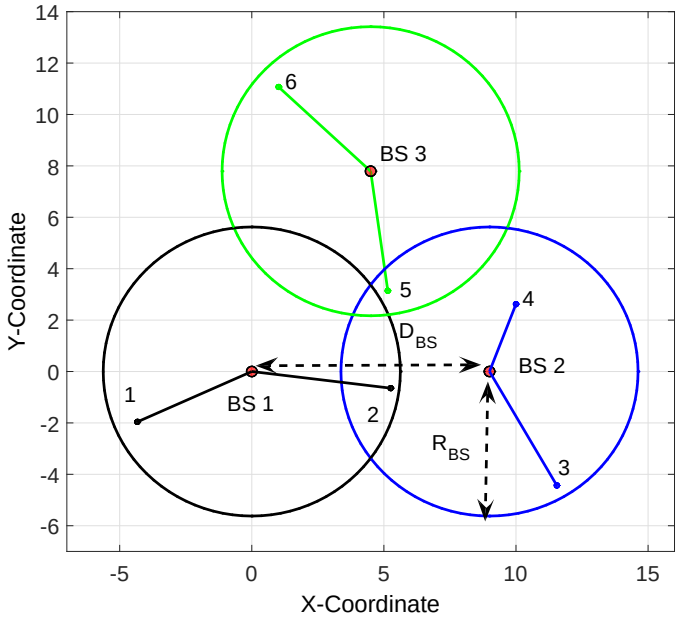


Fig. 3.1. MISO downlink wireless network with $N = 3$ BSs and $L = 6$ users. $\mathcal{N} = \{1, 2, 3\}$, $\mathcal{L}(1) = \{1, 2\}$, $\mathcal{L}(2) = \{3, 4\}$, and $\mathcal{L}(3) = \{5, 6\}$, [159] ©2015, IEEE.

where d_{jl} is the distance from tran(j) to rec(l), d_0 is the far field reference distance [173], η is the path loss exponent, and $\mathbf{c}_{jl} \in \mathbb{C}^T$ is arbitrarily chosen from the distribution $\mathcal{CN}(0, \mathbf{I})$ (i.e., frequency-flat fading channel with uncorrelated antennas). Here, we refer an arbitrarily generated set of fading coefficients $\check{\mathcal{C}} = \{\mathbf{c}_{jl} | j, l \in \mathcal{L}\}$ as a single fading realization. Note that in expression (3.45) the term $(d_{jl}/d_0)^{-\eta/2}$ denotes large scale fading, and the term $\mathbf{c}_{jl}$ denotes small scale fading.

We set $p_n^{\max} = p_0^{\max}$ for all $n \in \mathcal{N}$, and $\sigma_l = \sigma$ for all $l \in \mathcal{L}$. We define the SNR operating point at a distance r as

$$\text{SNR}(r) = \left(\frac{r}{d_0}\right)^{-\eta} \frac{p_0^{\max}}{\sigma^2}. \tag{3.46}$$

In the following simulations, we set $d_0 = 1$, $\eta = 4$, and the cell radius R_{BS} is fixed throughout the simulations such that $\text{SNR}(R_{\text{BS}}) = 10$ dB for $p_0^{\max}/\sigma^2 = 40$ dB. We let $D_{\text{BS}}/R_{\text{BS}} = 1.6$. The weight β_l associated with lth stream for all $l \in \mathcal{L}$

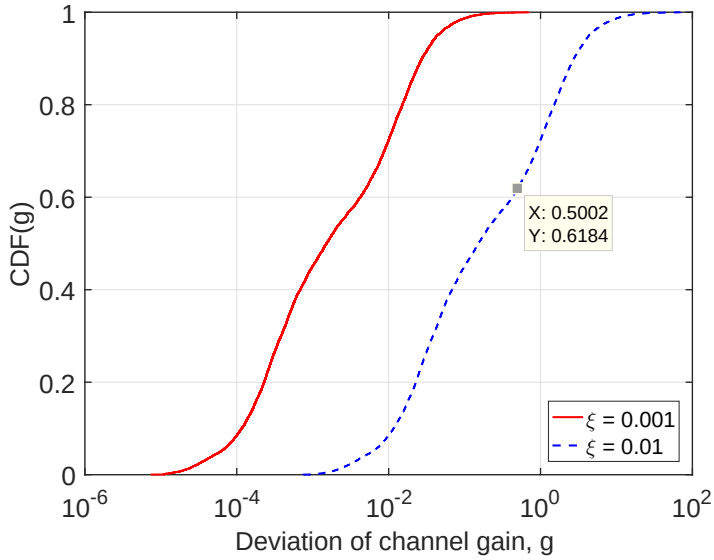


Fig. 3.2. CDF plot of channel gain deviation, [159] ©2015, IEEE.

are randomly generated. Furthermore, we assume that $\mathbf{Q}_{jl} = (1/\xi^2)\mathbf{I}$ such that the channel estimation error can take any value inside a ball with a radius ξ.

Note that because of the large scale fading coefficient in channel model (3.45), even a small value of ξ imposes considerable uncertainty on the estimated value of channel $\hat{\mathbf{h}}_{jl}$. To illustrate the effect of a value of ξ, let us define a metric $g(\hat{\mathbf{h}}_{jl}, \xi) = \xi^2/\|\hat{\mathbf{h}}_{jl}\|_2^2$, which is a ratio between the maximal CSI error and estimated channel gain.

Fig. 3.2 shows the CDF of the metric $g(\hat{\mathbf{h}}_{jl}, \xi)$, for all $j, l \in \mathcal{L}$, computed over 500 fading realizations for $\xi = 0.001$ and 0.01. Results show that when a value of $\xi = 0.001$, for almost all the channels the value of metric $g(\hat{\mathbf{h}}_{jl}, \xi)$ is less than 0.5. But when a value of $\xi = 0.01$, only for about 61% of the channels the value of metric g (i.e., a ratio between the maximal CSI error and estimated channel gain) is less than 0.5. Hence, the channel estimation error is significantly high when the channel uncertainty radius is set to $\xi = 0.01$.

To illustrate the convergence behavior of BB Algorithm 3.1, we consider a single fading realization. We set the algorithm tolerance $\epsilon = 0.2$ and the channel uncertainty radius $\xi = 0.01$. Fig. 3.3 shows the evolution of the upper and lower

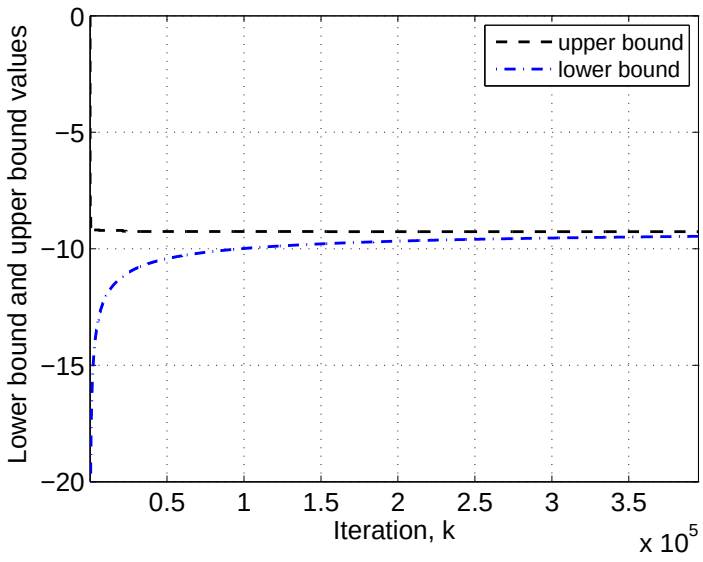


Fig. 3.3. Upper and lower bound evolution, [159] ©2015, IEEE.

bound for the optimal value of problem (3.8) for SNR = 5 dB [14]. Results show that the upper and lower bound becomes tight with an increase in the number of iterations. Furthermore, we can see that the proposed algorithm can find the best upper bound value very fast. However, the algorithm takes more iterations to certify that the achieved upper bound is close to the optimal value. For example, to certify that the achieved upper bound is $\epsilon = 0.2$ away from the optimal value, the algorithm requires more than 3.5×10^5 iterations.

We next evaluate the performance of Algorithm 3.2. Here, we first consider a single fading realization. As benchmarks, we the consider proposed optimal BB Algorithm 3.1 and a suboptimal algorithm proposed in recent paper [103]. To make a fair comparison with [103] on a required computational complexity per iteration for convergence, we solve problem (3.41) only once at step 2 of Algorithm 3.2 (with this setting Algorithm 3.2 solves one SDP and one GP at each iteration, and algorithm [22] solves one SDP at each iteration). Furthermore, we remove the trust region of problem (3.41). Algorithm 3.2 is initialized with

[14] For fixed radius R_{BS}, different SNR (i.e., different SNR(R_{BS})) are obtained by changing $p_0^{\max}/\sigma^2$.

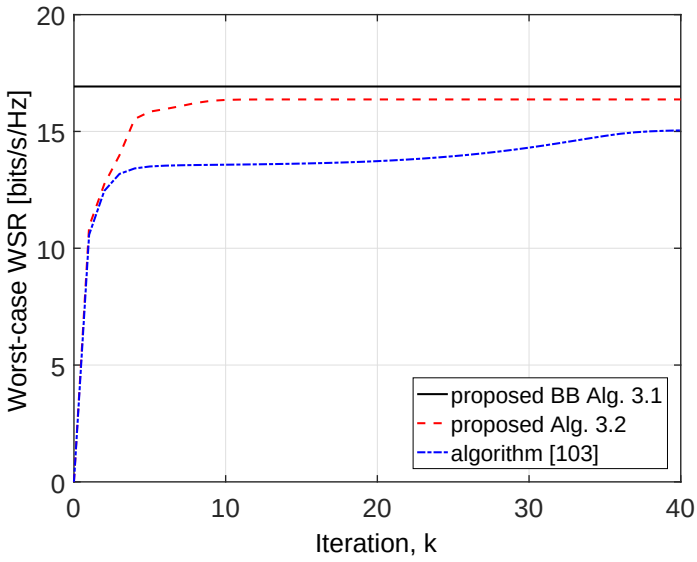
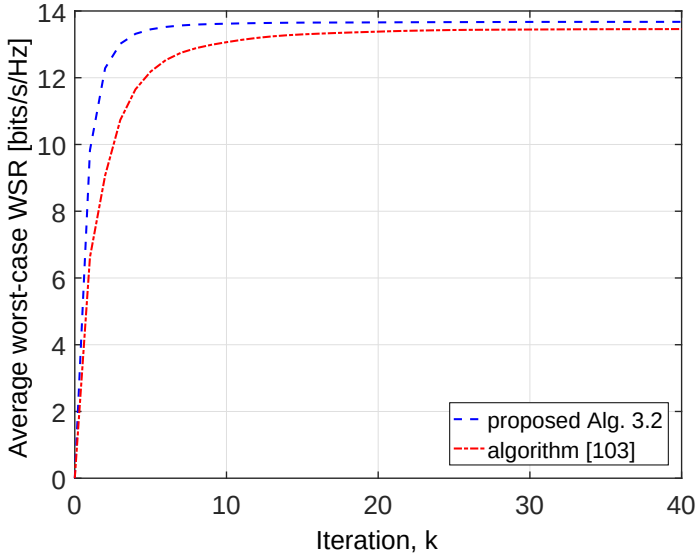


Fig. 3.4. Worst-case WSR (bits/s/Hz) versus iteration for $\mathrm{SNR} = 5$ dB **and error radius** $\xi = 0.01$**, [159] ©2015, IEEE.**

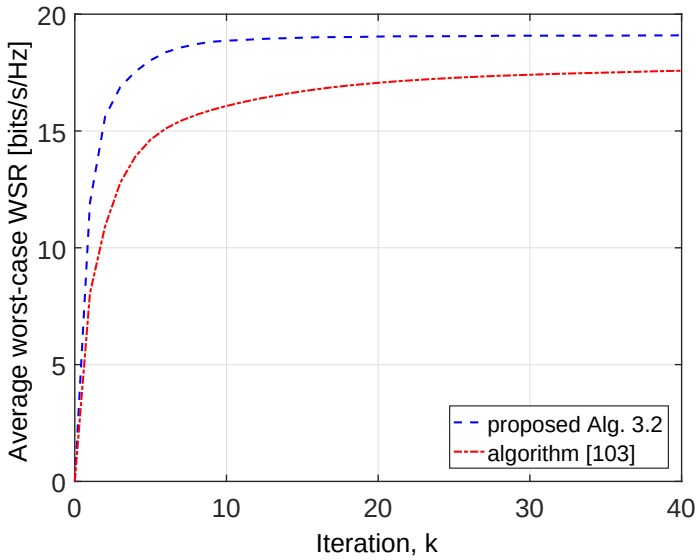
the transmit beamforming vectors with equal transmit power for all data streams. For the BB Algorithm 3.1, we set the algorithm tolerance $\epsilon = 0.4$.

Fig. 3.4 shows the convergence behavior of Algorithm 3.2 and algorithm [103], along with the upper bound on the optimal value given by the proposed BB Algorithm 3.1. We set SNR = 5 dB and the channel uncertainty radius $\xi = 0.01$. For Algorithm 3.2, the achieved objective value is computed after step 2 of the algorithm. Results show that both suboptimal algorithms (i.e, proposed Algorithm 3.2 and algorithm [103]) can achieve the worst-case WSR close to the optimal value. Moreover, results show that the convergence speed of proposed Algorithm 3.2 is faster compared to that of algorithm [103].

In order to see the average behavior of proposed Algorithm 3.2, next, we run Algorithm 3.2 and algorithm [103] for 300 fading realizations with the channel uncertainty radius $\xi = 0.01$. Fig. 3.5 shows the average worst-case WSR versus iteration for SNR = 5 dB and SNR = 20 dB. Plots are drawn for the first 40 iterations. Results show that for a low SNR value (i.e., SNR = 5 dB), both Algorithm 3.2 and algorithm [103] nearly converge to the same objective value. However, for high SNR value (i.e., SNR = 20 dB), the proposed algorithm



(a)



(b)

Fig. 3.5. Convergence of the average worst-case WSR (bits/s/Hz): (a) $\mathrm{SNR} = 5$ dB**; (b)** $\mathrm{SNR} = 20$ dB**, [159] ©2015, IEEE.**

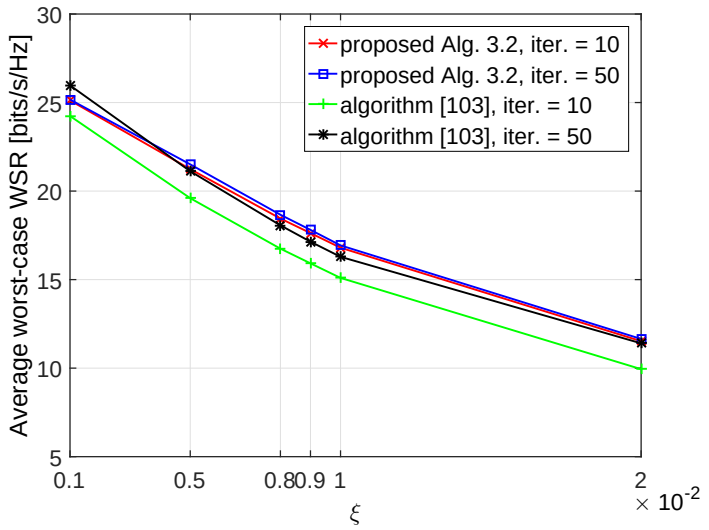


Fig. 3.6. Average worst-case WSR (bits/s/Hz) versus channel uncertainty radius ξ for $\mathrm{SNR} = 10$ dB**, [159] ©2015, IEEE.**

achieves higher WSR compared to that of algorithm [103]. Moreover, results show that the convergence speed of the proposed algorithm is faster compared to that of algorithm [103]. Note that this gain comes at an additional cost of solving one more GP per iteration compared to algorithm [103].

Fig. 3.6 shows the average worst-case WSR versus channel uncertainty radius. Plots are drawn for the objective values obtained at iteration number 10 and 50 for a wide range of ξ values. Results show that the average objective values achieved by the proposed algorithm at iteration number 10 and 50 is very close for the wide range of ξ values. However, for algorithm [103] there is a noticeable difference between the objective values achieved at iteration number 10 and 50. Hence, results show that the proposed Algorithm 3.2 provides an efficient solution very fast compared to algorithm [103] for a wide range of the channel uncertainty radius ξ values.

Fig. 3.7 shows the average worst-case WSR for different SNR values. To solve Subproblem 1, in step 2 of Algorithm 3.2, we set $\epsilon_\gamma = 0.01$. Plots are drawn for $\alpha = 1.1, 3$, and without the trust region (which we denote by $\alpha = \infty$) in problem (3.41). For a comparison, we use algorithm [103], and algorithm [176, Sec. 4.3] that assumes perfect CSI. For a fair comparison, all algorithms are

stopped when the difference between the achieved objective values between successive iterations is less than 10^{-3}.

Results show that for a small channel uncertainty region (i.e., Fig. 3.7(a)) algorithm [103] performs better than the proposed algorithm at low SNR values, and both algorithms have the same performance at high SNR values. On the other hand, in the case of a larger channel uncertainty region (i.e., Fig. 3.7(b)), the performance of both algorithms is close at low SNR values. However, at high SNR values, the proposed algorithm achieves better objectives values than algorithm [103]. Note that this gain comes at the cost of solving one additional signomial program (i.e., solving step 2 of Algorithm 3.2 several times to achieve given ϵ_γ) per iteration compared to algorithm [103]. But this additional computation complexity may be worthwhile as the proposed algorithm can provide better objective value. Furthermore, results show that the performance of the proposed algorithm can be improved by using a small value of α in problem (3.41).

In practice, the channel estimation errors may have a statistical distribution [90, 96, 178]. Next, we show how we can apply our design methodology (Algorithm 3.2)[15] to a scenario with statistical channel errors. We consider the least-squares training-based channel estimator [19]. Then the CSI errors are zero-mean circular symmetric complex Gaussian random variables with covariance $\mathbf{\Sigma}$, i.e., $\mathbf{e}_{jl} \sim \mathcal{CN}(0, \mathbf{\Sigma})$ for all $j, l \in \mathcal{L}$.

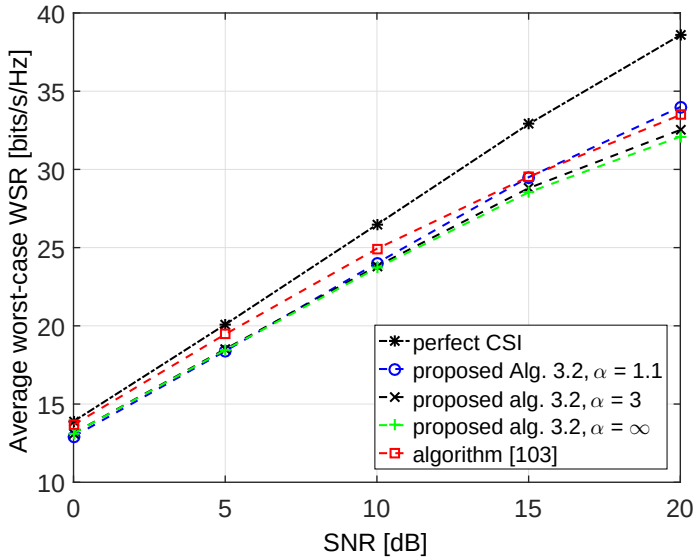
The key idea behind applying Algorithm 3.2 in a scenario with Gaussian distributed CSI errors is to design the ellipsoidal uncertainty region so that it includes a certain percentage of the errors [179, 180]. We design the ellipsoidal uncertainty region by utilizing the confidence ellipsoids [185, Sec. 3], [186]. The κ-confidence ellipsoids $\mathcal{E}_{jl}(\kappa)$ for CSI errors $\mathbf{e}_{jl} \sim \mathcal{CN}(0, \mathbf{\Sigma})$ can be written as

$$\mathcal{E}_{jl}(\kappa) = \left\{\mathbf{e}_{jl} | \mathbf{e}_{jl}^{\mathrm{H}} \mathbf{\Sigma}^{-1} \mathbf{e}_{jl} \leq \theta(\kappa)\right\}, \quad j, l \in \mathcal{L}, \tag{3.47}$$

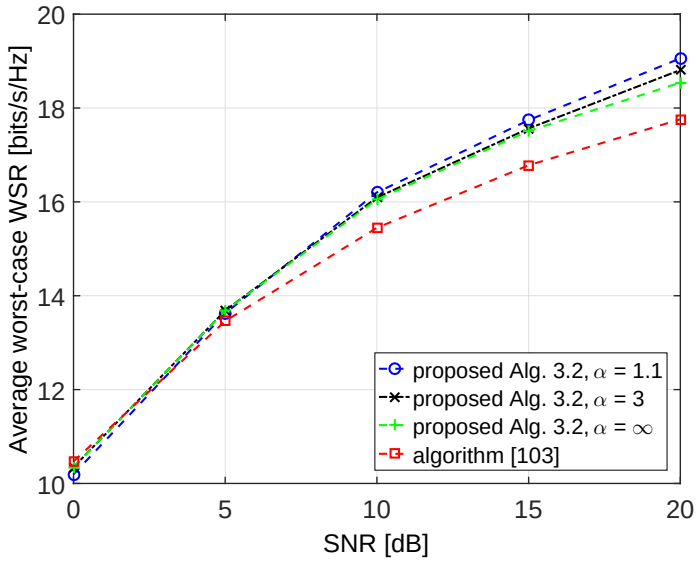
such that $\mathbf{Prob}\big(\mathbf{e}_{jl} \in \mathcal{E}_{jl}(\kappa)\big) \geq \kappa$ [16]. Note that $\mathcal{E}_{jl}(\kappa)$ is the smallest volume ellipsoid that contains $\kappa\%$ samples of the CSI errors $\mathbf{e}_{jl}$ generated from the distribution $\mathcal{CN}(0, \mathbf{\Sigma})$ [185, Sec. 3], [186]. Then by using κ-confidence ellipsoids

[15] Here, we evaluate the performance of Algorithm 3.2. The performance of BB Algorithm 3.1 can be evaluated in a similar way.

[16] The nonnegative random variable $\mathbf{e}_{jl}^{\mathrm{H}} \mathbf{\Sigma}^{-1} \mathbf{e}_{jl}$ has a chi-square distribution with $2T$ degrees of freedom. Thus, the value of $\theta(\kappa)$ for a given κ can be calculated by using the inverse CDF of chi-square distribution.



(a)



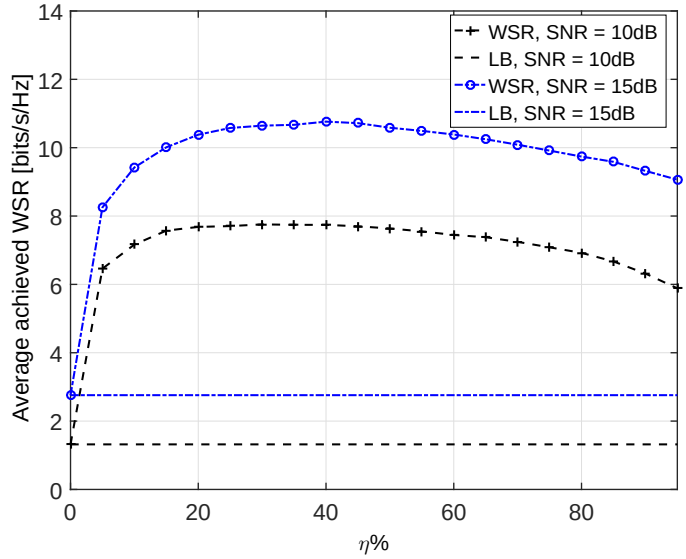
(b)

Fig. 3.7. Average worst case WSR (bits/s/Hz) versus SNR: (a) CSI uncertainty radius $\xi = 0.001$; (b) CSI uncertainty radius $\xi = 0.01$, [159] ©2015, IEEE.

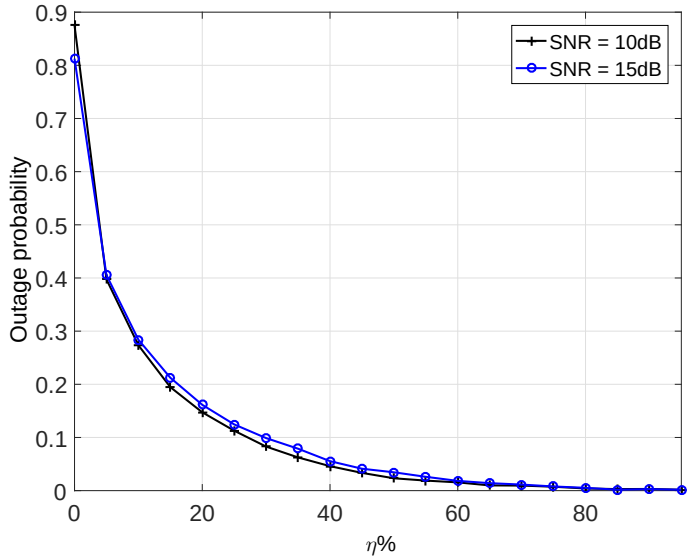
in Algorithm 3.2, a transmission rate and a beamformer associated with each data stream is obtained. If the CSI errors lie inside the considered uncertainty region, the SINRs associated with the obtained beamformers guarantee that the allocated rates of all data streams are achieved. However, if the CSI errors lie outside the considered κ-confidence ellipsoids, the allocated rates of all data streams may not be achieved by using the beamformers, and hence there can be an outage. Therefore, for a small confidence ellipsoid there may be a high outage rate, since a larger subset of channel errors lie outside the considered confidence ellipsoid. On the other hand, for a larger confidence ellipsoid the outage rate is very small, but the transmission rates would be small due to the conservative design of the considered κ-confidence ellipsoid. Thus, there is a trade-off between the transmission rates and the outage, when the κ-confidence ellipsoid is used in Algorithm 3.2 in a scenario with the statistical channel errors.

In order to elaborate the above mentioned trade off when utilizing κ-confidence ellipsoids for Gaussian distributed CSI errors, we run Algorithm 3.2 for different sizes of confidence ellipsoids $\big($i.e., different values of κ in (3.47)$\big)$ and evaluate the performance of Algorithm 3.2. For a given κ and a fading realization, let us denote the optimal solution set $\mathcal{S}_\kappa = \{p_l^\star, \gamma_l^\star, \mathbf{v}_l^\star\}_{l\in\mathcal{L}}$ obtained from step 3 of Algorithm 3.2. Then the transmission rate and the associated beamformer of lth data stream is set to $r_l(\mathcal{S}_\kappa) = \log_2(1+\gamma_l^\star)$ and $\mathbf{m}_l = \sqrt{p_l^\star}\mathbf{v}_l^\star$, respectively, for all $l \in \mathcal{L}$. However, the actual supported rates of data streams using the beamformers $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ depends on the actual SINR values (3.6) $\big($i.e., by using SINR (3.6) the supported rate of lth data stream is $R_l(\mathcal{S}_\kappa) = \log_2(1+\Gamma_l(\mathbf{m}, \mathbf{e}_l))\big)$; and these rates are unknown to the BSs since the CSI errors are unknown. Hence, if the actual supported rate $R_l(\mathcal{S}_\kappa)$ is smaller than the allocated transmission rate $r_l(\mathcal{S}_\kappa)$, there will be an outage. Thus, the (actual) achieved WSR considering the outage is $f^{\mathrm{WSR-WO}}(\mathcal{S}_\kappa) = \sum_{l\in\mathcal{L}} \beta_l r_l(\mathcal{S}_\kappa) I_l$, where, $I_l = 1$ if $R_l(\mathcal{S}_\kappa) \geq r_l(\mathcal{S}_\kappa)$ and $I_l = 0$ otherwise.

Fig. 3.8(a) shows the average achieved WSR $f^{\mathrm{WSR-WO}}(\mathcal{S}_\kappa)$ versus κ for SNR = 10 dB and 15 dB. Each curve is averaged over 1000 fading realizations (estimated CSI values), and for each realization the channel errors $\mathbf{e}_{jl}$ are arbitrarily chosen from the distribution $\mathcal{CN}(0, \mathbf{\Sigma})$. The horizontal lines in Fig. 3.8(a) extend the average WSR values obtained by using $\kappa = 0$ (i.e., assuming the estimated CSI values as the actual channels); and those serve as lower bounds (LBs). Fig. 3.8(b) shows the outage probability versus κ. In the



(a)



(b)

Fig. 3.8. (a) Average achieved WSR $f^{\mathrm{WSR-WO}}(\mathcal{S}_\kappa)$ versus size of confidence ellipsoids η; (b) outage probability versus size of confidence ellipsoids κ, [159] ©2015, IEEE.

simulations, we use a least-squares channel estimator with a minimum length optimal training sequence [19]. Hence, the channel error covariance matrix $\mathbf{\Sigma} = (\sigma_e^2/T)\mathbf{I}$, where $\sigma_e^2 = \sigma^2 T^2 / p_0^{\max}$ is the variance of the channel estimation error [19, eq. (12)], [178]. Furthermore, we use equal weights for all data streams, i.e., $\beta_l = 1$ for all $l \in \mathcal{L}$.

Fig. 3.8(a) shows that the confidence ellipsoids in between 20% to 50% achieve the maximal WSR, and there is performance loss when the confidence ellipsoid is either too small or large. In the case of smaller values of κ, the performance loss is due to large outage (see Fig. 3.8(b)) caused by the optimistic design (because it assumes that the CSI errors are small and the actual channels are very close to the estimated CSI values). In the case of larger values of κ the outage is small, but still there is performance loss (see Fig. 3.8(a)) due to the overly conservative design (because the ellipsoidal uncertainty region is designed considering that the CSI error could be very high). Results suggest that for any value of κ within the interval of 20 to 50 percent, Algorithm 3.2 performs quite well and achieves the trade-off between the transmission rates and the outage.

3.5 Summary and discussion

We have considered a robust WSRMax problem in multicell downlink MISO systems. The CSI of all relevant users is assumed to be imperfectly known at the base stations. The problem is known to be NP-hard even in the case of perfect CSI. We have considered a bounded ellipsoidal model for the CSI errors. By using the notion of worst-case optimization, we have proposed both optimal and suboptimal algorithms for the WSRMax problem with CSI errors. The proposed optimal Algorithm 3.1 is based on a BB method, and it globally solves the worst-case WSRMax problem with an optimality certificate. Note that the BB method basically implements some sort of exhaustive search in a systematic manner. Hence, Algorithm 3.1 can be (and often is) slow. In the worst case, the complexity of Algorithm 3.1 grows exponentially with the problem size. Finding efficient (improved) bounding techniques could be one way to reduce computational complexity of BB based Algorithm 3.1. As the convergence speed of the BB method can be slow for large networks, we have also provided a fast but possibly suboptimal algorithm based on the alternating optimization technique in conjunction with sequential convex programming. The optimal

BB based algorithm can be used to provide performance benchmarks for any suboptimal algorithm. Furthermore, through a numerical example we have shown how our design methodology can be applied to a scenario with statistical channel errors.

4 Distributed resource allocation for MISO cellular networks

In this chapter we propose decentralized resource allocation methods for multicell MISO downlink networks. Specifically, we consider the following optimization problems: P1 - minimization of the total transmission power subject to minimum SINR constraints of each user, and P2 - SINR balancing subject to total transmit power constraint of each BS.

The main contribution of this chapter is to propose consensus-based distributed algorithms for problems P1 and P2, and a fast solution method via ADMM [113]. The ADMM turns the original problem into a series of iterative steps, namely, *local variable* update, *global variable* update, and *dual variable* update [113]. The local and dual variable updates are carried out independently in parallel by all BSs, while the global variable update is carried out by BSs coordination. We first derive the distributed algorithm for problem P1. Then we extend the formulation of problem P1 to derive the distributed algorithm for problem P2. In particular, we recast the problem into a more tractable form and combine a bracketing method (a golden ratio search) [187, 188] with ADMM to derive the distributed algorithm for problem P2.

By considering the uncertainty in the channel measurements, for problem P1, an algorithm based on ADMM has been proposed in [98]. In this chapter, we consider perfect CSI, and use the consensus technique to solve the problem. Then we apply ADMM to derive the distributed algorithm. The consensus technique can be easily decomposed into a set of subproblems suitable for distributed implementation [113, 189]. Hence, the algorithm formulation in this chapter is more intuitive than that provided in [98]. For problem P1, we show that the proposed distributed algorithm converges to the optimal centralized solution. Moreover, for problem P1, we also provide a method to find ADMM penalty parameter [113] which leads faster convergence of the algorithm.

Problem P2 is a quasiconvex problem. To the best of our knowledge there is no convergence theory to the ADMM method for a quasiconvex problem. However, if each step of the ADMM iteration is tractable, the ADMM algorithm can still be used to derive (possibly suboptimal) distributed methods for problem

P2 [113, Sec. 9]. Numerical results show that the proposed distributed algorithm for problem P2 can find a close-to-optimal centralized solution.

4.1 System model and problem formulation

A multicell MISO downlink system, with N BSs each equipped with T transmit antennas is considered. The set of all BSs is denoted by $\mathcal{N}$, and we label them with the integer values $n = 1, \ldots, N$. The transmission region of each BS is modeled as a disc with radius R_{BS} centered at the location of the BS. A single data stream is transmitted for each user. We denote the set of all data streams in the system by $\mathcal{L}$, and we label them with the integer values $l = 1, \ldots, L$. The transmitter node (i.e., the BS) of lth data stream is denoted by $\mathrm{tran}(l)$ and the receiver node of lth data stream is denoted by $\mathrm{rec}(l)$. We have $\mathcal{L} = \cup_{n \in \mathcal{N}} \mathcal{L}(n)$, where $\mathcal{L}(n)$ denotes the set of data streams transmitted by nth BS. Note that the intended users of the data streams transmitted by each BS are necessarily located inside the transmission region of the BS (see Fig. 4.1).

The antenna signal vector transmitted by nth BS is given by

$$\mathbf{s}_n = \sum_{l \in \mathcal{L}(n)} d_l \mathbf{m}_l, \tag{4.1}$$

where $d_l \in \mathbb{C}$ represents the information symbol and $\mathbf{m}_l \in \mathbb{C}^T$ denotes the transmit beamformer associated with lth data stream. We assume that d_l is normalized such that $\mathrm{E}|d_l|^2 = 1$. Moreover, we assume that the data streams are independent, i.e., $\mathrm{E}\{d_l d_j^*\} = 0$ for all $l \neq j$, where $l, j \in \mathcal{L}$.

The signal received at $\mathrm{rec}(l)$ can be expressed as

$$\begin{aligned} y_l \;=\; & d_l \mathbf{h}_{ll}^{\mathrm{H}} \mathbf{m}_l + \underbrace{\sum_{j \,\in\, \mathcal{L}(\mathrm{tran}(l)),\, j \neq l} d_j \mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j}_{\text{(intra-cell interference)}} \\ & + \underbrace{\sum_{n \in \mathcal{N} \setminus \{\mathrm{tran}(l)\}} \; \sum_{j \,\in\, \mathcal{L}(n)} d_j \mathbf{h}_{jl}^{\mathrm{H}} \mathbf{m}_j}_{\text{(out-of-cell interference)}} + n_l, \end{aligned} \tag{4.2}$$

where $\mathbf{h}_{jl}^{\mathrm{H}} \in \mathbb{C}^{1 \times T}$ is the channel vector between $\mathrm{tran}(j)$ and $\mathrm{rec}(l)$, and n_l is circular symmetric complex Gaussian noise with variance σ_l^2. Note that the second right hand term in (4.2) represents the intra-cell interference and the third right hand term represents the out-of-cell interference. The received SINR

of lth data stream is given by

$$\Gamma_l(\mathbf{m}) = \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(\mathrm{tran}(l)),\,j\,\neq\,l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} z_{nl}^2}, \tag{4.3}$$

where we use the notation $\mathbf{m}$ to denote a vector obtained by stacking $\mathbf{m}_l$ for all $l \in \mathcal{L}$ on top of each other, i.e., $\mathbf{m} = [\mathbf{m}_1^{\mathrm{T}}, \ldots, \mathbf{m}_L^{\mathrm{T}}]^{\mathrm{T}}$, and variable $z_{nl}^2 = \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2$ represents the power of the out-of-cell interference from nth BS to rec(l).

The out-of-cell interference term in (4.3) (i.e., $\sum_{n\in\mathcal{N}\setminus\{\mathrm{tran}(l)\}} z_{nl}^2$) prevents resource allocation on an intra-cell basis and demands BSs coordination. To avoid unnecessary coordination between distantly located BSs, we make the following assumption: transmission from nth BS interferes with the lth data stream (transmitted by BS $b \neq n$) only if the distance between nth BS and rec(l) is smaller than a threshold R_{int}[17]. A disc with radius R_{int} centered at

[17]Similar assumptions are made, e.g., in[190] in the context of arbitrary wireless networks.

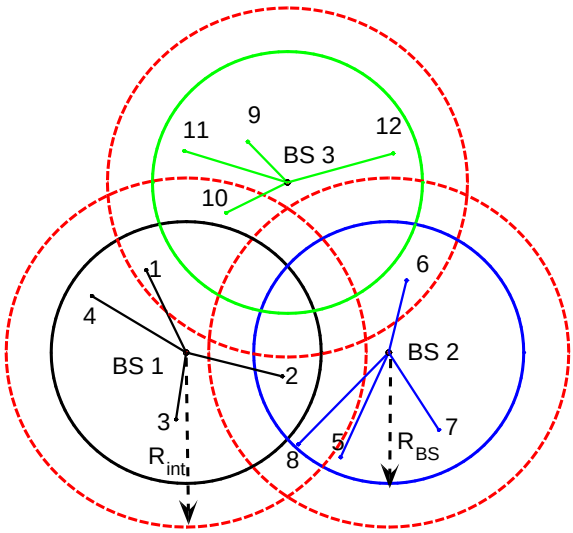


Fig. 4.1. Multicell network, $N = 3$**,** $L = 12$**,** $\mathcal{N} = \{1, 2, 3\}$**,** $\mathcal{L} = \{1, \ldots, 12\}$**,** $\mathcal{L}(1) = \{1, \ldots, 4\}$**,** $\mathcal{L}(2) = \{5, \ldots, 8\}$**,** $\mathcal{L}(3) = \{9, \ldots, 12\}$**, and** $\mathcal{L}_{\mathrm{int}} = \{1, 2, 6, 8, 10\}$**.**

the location of any BS is referred to as the interference region of the BS (see Fig. 4.1). Thus, if nth BS located at a distance larger than R_{int} to rec(l), the associated z_{nl} components are set to zero[18]. Based on the assumption above, we can express $\Gamma_l(\mathbf{m})$ as

$$\Gamma_l(\mathbf{m}) = \frac{|\mathbf{h}_{ll}^{\text{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j \in \mathcal{L}(\text{tran}(l)),\, j \neq l} |\mathbf{h}_{jl}^{\text{H}}\mathbf{m}_j|^2 + \sum\limits_{n \in \mathcal{N}_{\text{int}}(l)} z_{nl}^2},$$

where $\mathcal{N}_{\text{int}}(l) \subseteq \mathcal{N} \backslash \{\text{tran}(l)\}$ is the set of out-of-cell interfering BSs that are located at a distance less than R_{int} to rec(l). For example, in Fig. 4.1, we have $\mathcal{N}_{\text{int}}(1) = \{3\}$, $\mathcal{N}_{\text{int}}(2) = \{2\}$, $\mathcal{N}_{\text{int}}(6) = \{3\}$, $\mathcal{N}_{\text{int}}(8) = \{1\}$, $\mathcal{N}_{\text{int}}(10) = \{1\}$ and $\mathcal{N}_{\text{int}}(l) = \emptyset$ for all $l \in \{3, 4, 5, 7, 9, 11, 12\}$. Moreover, it is useful to define the set $\mathcal{L}_{\text{int}}$ of all data streams that are subject to the out-of-cell interference, i.e., $\mathcal{L}_{\text{int}} = \{l | l \in \mathcal{L}, \mathcal{N}_{\text{int}}(l) \neq \emptyset\}$. For example, in Fig. 4.1, we have $\mathcal{L}_{\text{int}} = \{1, 2, 6, 8, 10\}$.

The total transmitted power of the multicell downlink system can be expressed as

$$P = \sum_{n \in \mathcal{N}} \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2.$$

Assuming that the SINR $\Gamma_l(\mathbf{m})$ is subject to the constraint $\Gamma_l(\mathbf{m}) \geq \gamma_l$ for each user $l \in \mathcal{L}$, the problem of minimizing the total transmitted power (i.e., P1) can be expressed as

$$\begin{aligned} \text{P1}: \quad \text{minimize} \quad & \sum_{n \in \mathcal{N}} \sum_{l \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \\ \text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\text{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j \in \mathcal{L}(\text{tran}(l)),\, j \neq l} |\mathbf{h}_{jl}^{\text{H}}\mathbf{m}_j|^2 + \sum\limits_{n \in \mathcal{N}_{\text{int}}(l)} z_{nl}^2} \geq \gamma_l, \quad l \in \mathcal{L} \\ & z_{nl}^2 = \sum_{j \in \mathcal{L}(n)} |\mathbf{h}_{jl}^{\text{H}}\mathbf{m}_j|^2, \quad l \in \mathcal{L}_{\text{int}}, n \in \mathcal{N}_{\text{int}}(l), \end{aligned} \tag{4.4}$$

[18]The value of R_{int} is chosen so that the power of the interference term is below the noise level and this commonly used approximation is made to avoid unnecessary coordination between distant BSs. The appropriate value of R_{int} can be chosen to trade-off between the required backhaul signaling and the optimality of the solution. The effect of nonzero z_{nl} terms can be modeled by changing the statistical characteristics of noise n_l at rec(l). However, those issues are extraneous to the main focus of this chapter.

with variables $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ and $\{z_{nl}\}_{l\in\mathcal{L}_{\rm int},n\in\mathcal{N}_{\rm int}(l)}$ [19].

Providing fairness among the users with per BS power constraint is another important resource allocation problem. One way[20] of providing fairness among the users is by maximizing the minimum SINR (i.e., P2) [132], which can be formulated as

$$\begin{aligned} \text{P2}: \quad & \text{maximize} \quad \min_{l\in\mathcal{L}} \left(\frac{|\mathbf{h}_{ll}^{\rm H}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(\text{tran}(l)),\, j\neq l} |\mathbf{h}_{jl}^{\rm H}\mathbf{m}_j|^2 + \sum\limits_{n\in\mathcal{N}_{\rm int}(l)} z_{nl}^2} \right) \\ & \text{subject to} \quad z_{nl}^2 = \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\rm H}\mathbf{m}_j|^2, \quad l\in\mathcal{L}_{\rm int}, n\in\mathcal{N}_{\rm int}(l) \\ & \qquad\qquad \sum_{j\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n\in\mathcal{N}, \end{aligned} \tag{4.5}$$

where the variables are $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ and $\{z_{nl}\}_{l\in\mathcal{L}_{\rm int},n\in\mathcal{N}_{\rm int}(l)}$. In problem (4.5), the second inequality constraints denote the per BS power constraints.

4.2 Sum-power minimization

In this section we derive a distributed algorithm for problem (4.4), i.e., P1. First, we equivalently reformulate the original problem (4.4) in the form of a global consensus problem. Then we derive the proposed distributed algorithm based on ADMM [113].

[19] In problem (4.4) and (4.5), the set $\{z_{nl}\}_{l\in\mathcal{L}_{\rm int},n\in\mathcal{N}_{\rm int}(l)}$ is a collection of z_{nl} for which the lth user is inside the interference region of BS n. Thus, the constrained for unconsidered out-of-cell interference term (i.e., $z_{nl}^2 = 0$) for lth user that is outside the interference region of BS n is dropped in problem (4.4) and (4.5).

[20] A more general SINR balancing problem which can set the priority of users (keeping the SINR values of the data stream to fixed ratios) [83, Sec. IV-C] can be formulated. To simplify the presentation, we consider the maximization of the minimum SINR. Note that the proposed decentralized method can be easily generalized to the more general problem considered in [83, Sec. IV-C].

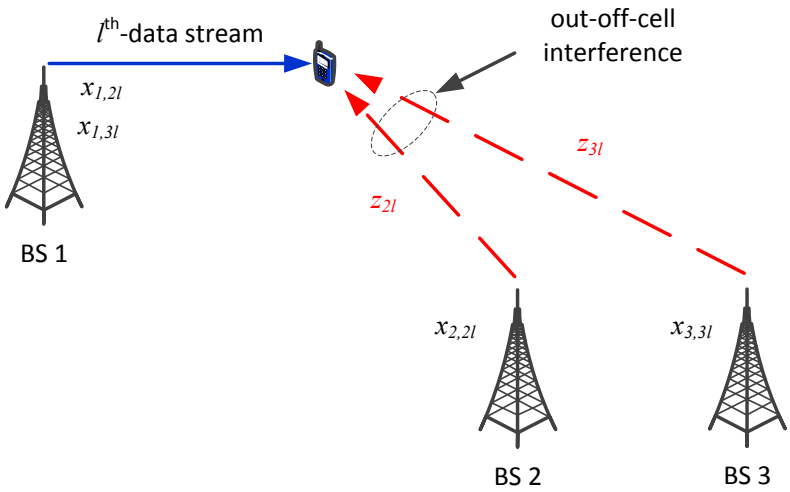


Fig. 4.2. BS 2 and BS 3 are coupled with BS 1 due to coupling variables z_{2l} and z_{3l}, respectively. To distribute the problem, local copy $x_{1,2l}$ of z_{2l} at BS 1 and local copy $x_{2,2l}$ of z_{2l} at BS 2 are introduced. Similarly, local copy $x_{1,3l}$ of z_{3l} at BS 1 and local copy $x_{3,3l}$ of z_{3l} at BS 3 are introduced.

4.2.1 Equivalent reformulation

We start by reformulating sum-power minimization problem (4.4) as

$$\begin{aligned} \text{minimize} \quad & \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \\ \text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(\text{tran}(l)),\, j\neq l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{n\in\mathcal{N}_{\text{int}}(l)} z_{nl}^2} \geq \gamma_l, \quad l\in\mathcal{L} \\ & z_{nl}^2 \geq \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, \quad l\in\mathcal{L}_{\text{int}}, n\in\mathcal{N}_{\text{int}}(l), \end{aligned} \tag{4.6}$$

where the variables are $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ and $\{z_{nl}\}_{l\in\mathcal{L}_{\text{int}},n\in\mathcal{N}_{\text{int}}(l)}$. Problem (4.4) and (4.6) are equivalent as it can be easily shown (e.g., by contradiction) that the second inequality holds with equality at the optimal point.

Recall that z_{nl}^2 in problem (4.6) represents the power of the out-of-cell interference caused by nth BS at rec(l), and hence, variable z_{nl} couples exactly two BSs (i.e., BS n and BS tran(l)). We use a consensus technique to distribute problem (4.6) over the BSs. The method consists of introducing local copies of the coupling variables z_{nl} for all $l\in\mathcal{L}_{\text{int}}$, $n\in\mathcal{N}_{\text{int}}(l)$ at each BS (see Fig. 4.2).

Let us define $x_{k,nl}$ as the local copy of z_{nl} at BS k. Thus for each z_{nl}, we make two local copies, i.e., $x_{n,nl}$ at BS n and $x_{\text{tran}(l),nl}$ at BS tran(l). Then

problem (4.6) can be written equivalently in a global consensus form as

$$\begin{aligned}
\text{minimize} \quad & \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \\
\text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(t_l),\, j\,\neq\, l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{b\in\mathcal{N}_{\mathrm{int}}(l)} x_{n,bl}^2} \geq \gamma_l, \quad n\in\mathcal{N}, l\in\mathcal{L}(n) \\
& x_{n,nl}^2 \geq \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, \quad l\in\mathcal{L}_{\mathrm{int}}, n\in\mathcal{N}_{\mathrm{int}}(l) \\
& x_{k,nl} = z_{nl}, \quad k\in\{n,\mathrm{tran}(l)\}, l\in\mathcal{L}_{\mathrm{int}}, n\in\mathcal{N}_{\mathrm{int}}(l),
\end{aligned} \tag{4.7}$$

with variables $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$, $\{x_{k,nl}\}_{k\in\{n,\mathrm{tran}(l)\},l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$, and $\{z_{nl}\}_{l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$. Note that in the first set of constraints of problem (4.7) we have replaced z_{bl} with the local copy $x_{n,bl}$, and used the fact that $\mathcal{L} = \cup_{n\in\mathcal{N}}\mathcal{L}(n)$. In the second inequality constraints of (4.7), we have replaced z_{nl} by the local copy $x_{n,nl}$. The last set of equality constraints of (4.7) are called consistency constraints, and they ensure that the local copies $x_{n,nl}$ and $x_{\mathrm{tran}(l),nl}$ are equal to the corresponding global variable z_{nl}.

Problem (4.7) is not a convex problem. However, it can be equivalent expressed as a SOCP [132, Sec. IV-B]. To do this, let us define the matrix $\mathbf{M}_n = [\mathbf{m}_l]_{l\in\mathcal{L}(n)}$ obtained by concatenating the column vectors $\mathbf{m}_l$. Then by following the approach of [132, Sec. IV-B], problem (4.7) can be equivalently reformulated as

$$\begin{aligned}
\text{minimize} \quad & \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \\
\text{subject to} \quad & \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad n\in\mathcal{N}, l\in\mathcal{L}(n) \\
& \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l\in\mathcal{L}_{\mathrm{int}}, n\in\mathcal{N}_{\mathrm{int}}(l) \\
& x_{k,nl} = z_{nl}, \quad k\in\{n,\mathrm{tran}(l)\}, l\in\mathcal{L}_{\mathrm{int}}, n\in\mathcal{N}_{\mathrm{int}}(l),
\end{aligned} \tag{4.8}$$

with the optimization variables $\{\mathbf{m}_l\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{x_{k,nl}\}_{k\in\{n,\mathrm{tran}(l)\},l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$, and $\{z_{nl}\}_{l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$, where vector $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b\in\mathcal{N}_{\mathrm{int}}(l)}$, the vector $\mathbf{h}_{jl}$ in the second set of constraints denotes the channel from BS n to link l (i.e., the index j in the second set of constraints denotes an arbitrary link in $\mathcal{L}(n)$),

and the notation $\succeq_{\text{SOC}}$ denotes the generalized inequalities with respect to the second-order cone [132], [29, Sec. 2.2.3].

In problem (4.8), the objective function and the first set of inequality constraints are separable in $n \in \mathcal{N}$ (one for each BS). Also, it can be easily shown that the second set of inequality constraints of (4.8) are separable in $n \in \mathcal{N}$. To do this, let us denote $\mathcal{I}_{\text{int}}(n)$ the set of links for which BS n acts as an out-of-cell interferer, i.e., $\mathcal{I}_{\text{int}}(n) = \{l | l \in \mathcal{L}_{\text{int}}, n \in \mathcal{N}_{\text{int}}(l)\}$. Then by noting that the sets $\{(n,l) | l \in \mathcal{L}_{\text{int}}, n \in \mathcal{N}_{\text{int}}(l)\}$ and $\{(n,l) | n \in \mathcal{N}, l \in \mathcal{I}_{\text{int}}(n)\}$ are identical, the second set of inequality constraints of (4.8) can be written as

$$\begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\text{H}} \mathbf{h}_{jl} \end{bmatrix} \succeq_{\text{SOC}} 0, \quad n \in \mathcal{N}, l \in \mathcal{I}_{\text{int}}(n), \tag{4.9}$$

which is separable in $n \in \mathcal{N}$. Observe that without the consistency constraints, problem (4.8) can now be easily decoupled into N subproblems, one for each BS.

We next express problem (4.8) more compactly. To do this, we first express the consistency constraints of problem (4.8) more compactly by using vector notations, which denote a collection of the local and global variables associated with BS n. By using the equivalence between the sets $\{(n,l) | l \in \mathcal{L}_{\text{int}}, n \in \mathcal{N}_{\text{int}}(l)\}$ and $\{(n,l) | n \in \mathcal{N}, l \in \mathcal{I}_{\text{int}}(n)\}$, let us express the consistency constraints of problem (4.8) as

$$\begin{array}{rcll} x_{n,nl} & = & z_{nl}, & n \in \mathcal{N}, l \in \mathcal{I}_{\text{int}}(n) \\ x_{\text{tran}(l),nl} & = & z_{nl}, & l \in \mathcal{L}_{\text{int}}, n \in \mathcal{N}_{\text{int}}(l). \end{array} \tag{4.10}$$

In the first set of equalities of (4.10), $\{x_{n,nl}\}_{l \in \mathcal{I}_{\text{int}}(n)}$ is a set of local variables that are associated with BS n. Similarly, to find a set of local variables that are associated with BS n in the second set of equalities of (4.10), let us define $\mathcal{L}_{\text{int}}(n)$ as the set of links in BS n that are affected by the out-of-cell interference, i.e., $\mathcal{L}_{\text{int}}(n) = \{l | l \in \mathcal{L}_{\text{int}} \cap \mathcal{L}(n)\}$. Then by noting that the set $\mathcal{L}_{\text{int}} = \cup_{n \in \mathcal{N}} \mathcal{L}_{\text{int}}(n)$, we can rewrite expression (4.10) as

$$\begin{array}{rcll} x_{n,nl} & = & z_{nl}, & n \in \mathcal{N}, l \in \mathcal{I}_{\text{int}}(n) \\ x_{\text{tran}(l),bl} & = & z_{bl}, & n \in \mathcal{N}, l \in \mathcal{L}_{\text{int}}(n), b \in \mathcal{N}_{\text{int}}(l). \end{array} \tag{4.11}$$

Clearly, in the second set of equalities of (4.11) [21], $\{x_{\text{tran}(l),bl}\}_{l \in \mathcal{L}_{\text{int}}(n), b \in \mathcal{N}_{\text{int}}(l)}$ is a set of local variables that are associated with BS n.

[21] Note that $\mathcal{L}_{\text{int}}(n) \subseteq \mathcal{L}(n)$. Hence, $\text{tran}(l) = n$ for all $l \in \mathcal{L}_{\text{int}}(n)$.

We now denote (4.11) compactly using vector notation. Let us define vectors $\mathbf{x}_n$ and $\mathbf{z}_n$ as [22]

$$\begin{array}{rcl} \mathbf{x}_n & = & \big((x_{n,nl})_{l\in\mathcal{I}_{\rm int}(n)}, (x_{{\rm tran}(l),bl})_{l\in\mathcal{L}_{\rm int}(n),b\in\mathcal{N}_{\rm int}(l)}\big) \\ \mathbf{z}_n & = & \big((z_{nl})_{l\in\mathcal{I}_{\rm int}(n)}, (z_{bl})_{l\in\mathcal{L}_{\rm int}(n),b\in\mathcal{N}_{\rm int}(l)}\big). \end{array} \tag{4.12}$$

Then (4.11) can be compactly written as

$$\mathbf{x}_n = \mathbf{z}_n, \quad n \in \mathcal{N}. \tag{4.13}$$

Note that $\mathbf{x}_n$ is a collection of the local variables that are associated with BS n, and $\mathbf{z}_n$ is a collection of the global variables that are associated with the components of variable $\mathbf{x}_n$.

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

$$\mathcal{M}_n = \left\{ \mathbf{M}_n, \mathbf{x}_n \;\middle|\; \begin{array}{l} \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\rm H}\mathbf{m}_l \\ \mathbf{M}_n^{\rm H}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\rm SOC} 0, \quad l \in \mathcal{L}(n) \\ \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\rm H}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\rm SOC} 0, \quad l \in \mathcal{I}_{\rm int}(n), \end{array} \right\}, \tag{4.14}$$

where $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b\in\mathcal{N}_{\rm int}(l)}$ is a subvector of $\mathbf{x}_n$, and the following function

$$f_n(\mathbf{M}_n, \mathbf{x}_n) = \begin{cases} \sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 & (\mathbf{M}_n, \mathbf{x}_n) \in \mathcal{M}_n \\ \infty & \text{otherwise} \end{cases}. \tag{4.15}$$

Then by using notations (4.13), (4.14), and (4.15), consensus problem (4.8) can be written compactly as

$$\begin{array}{ll} \text{minimize} & \sum\limits_{n\in\mathcal{N}} f_n(\mathbf{M}_n, \mathbf{x}_n) \\ \text{subject to} & \mathbf{x}_n = \mathbf{z}_n, \quad n \in \mathcal{N}, \end{array} \tag{4.16}$$

where the variables are $\{\mathbf{M}_n\}_{n\in\mathcal{N}}$, $\{\mathbf{x}_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{z}_n\}_{n\in\mathcal{N}}$.

4.2.2 Distributed algorithm via ADMM

In this section we derive distributed algorithm for problem (4.16). The proposed algorithm is based on ADMM [113]. We start by writing the augmented

[22] To simplify the presentation, here we have slightly abused the notation, i.e., we have considered that the sets in (4.12) are ordered.

Lagrangian [191] for problem (4.16) as

$$L_\rho(\{\mathbf{M}_n, \mathbf{x}_n, \mathbf{z}_n, \mathbf{u}_n\}_{n\in\mathcal{N}}) = \sum_{n\in\mathcal{N}} f_n(\mathbf{M}_n, \mathbf{x}_n) + \sum_{n\in\mathcal{N}} \left(\mathbf{u}_n^{\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n\|_2^2\right), \quad (4.17)$$

where $\mathbf{u}_n$ for all $n \in \mathcal{N}$ are the dual variables[23] associated with the equality constraints of (4.16), and $\rho > 0$ is a penalty parameter that adds the quadratic penalty to the standard Lagrangian L_0 for the violation of the equality constraints of problem (4.16).

Each iteration of the ADMM algorithm consists of the following three steps [113]:

$$\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1} = \underset{\mathbf{M}_n, \mathbf{x}_n}{\operatorname{argmin}} \; L_\rho\left(\mathbf{M}_n, \mathbf{x}_n, \mathbf{z}_n^i, \mathbf{u}_n^i\right), \quad n \in \mathcal{N} \quad (4.18)$$

$$\{\mathbf{z}_n^{i+1}\}_{n\in\mathcal{N}} = \underset{\{\mathbf{z}_n\}_{n\in\mathcal{N}}}{\operatorname{argmin}} \; L_\rho\left(\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}, \mathbf{z}_n, \mathbf{u}_n^i\}_{n\in\mathcal{N}}\right) \quad (4.19)$$

$$\mathbf{u}_n^{i+1} = \mathbf{u}_n^i + \rho\left(\mathbf{x}_n^{i+1} - \mathbf{z}_n^{i+1}\right), \quad n \in \mathcal{N}, \quad (4.20)$$

where superscript i is the iteration counter. Steps (4.18) and (4.20) are completely decentralized, and hence, can be carried out independently in parallel in each BS. Note that each component of $\mathbf{z}_n$ couples two local variables that are associated with the adjacent BSs (see consistency constraint of problem (4.8)) [24]. Thus, step (4.19) requires to gather the updated local variables $\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}\}$ and the dual variables $\mathbf{u}_n^i$ from all N BSs. In the sequel, we first explain in detail to solve the ADMM steps in (4.18) and (4.19). Then we summarize the proposed ADMM based distributed algorithm.

The local variable update $\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}\}$ in (4.18) is a solution of the following optimization problem:

$$\text{minimize} \quad f_n(\mathbf{M}_n, \mathbf{x}_n) + \mathbf{u}_n^{i\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n^i) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i\|_2^2 \quad (4.21)$$

with variables $\mathbf{M}_n$ and $\mathbf{x}_n$. Here, we write $\mathbf{u}_n^{i\mathrm{T}}$ instead of $(\mathbf{u}_n^i)^{\mathrm{T}}$ to lighten the notation.

[23] Let $u_{k,nl}$ for all $k \in \{n, \mathrm{tran}(l)\}, l \in \mathcal{L}_{\mathrm{int}}$, and $n \in \mathcal{N}_{\mathrm{int}}(l)$ be the dual variables associated with the equality constraints of problem (4.8), then by following steps (4.10) to (4.12), we can easily express $\mathbf{u}_n = \left((u_{n,nl})_{l\in\mathcal{I}_{\mathrm{int}}(n)}, (u_{n,bl})_{l\in\mathcal{L}_{\mathrm{int}}(n), b\in\mathcal{N}_{\mathrm{int}}(l)}\right)$ for all $n \in \mathcal{N}$.

[24] Variable z_{nl} (component of $\mathbf{z}_n$) couples two local variables $x_{n,nl}$ (component of $\mathbf{x}_n$) and $x_{\mathrm{tran}(l),nl}$ (component of $\mathbf{x}_{\mathrm{tran}(l)}$). Hence, in step (4.19) to update z_{nl}, coordination between BS n and BS $\mathrm{tran}(l)$ is required.

Let $\mathbf{v}_n = (1/\rho)\mathbf{u}_n$ be a scaled dual variable. For convenience we can combine the second and third terms of the objective function of problem (4.21) as

$$\mathbf{u}_n^{i\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n^i) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i\|_2^2 = \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 - \frac{\rho}{2}\|\mathbf{v}_n^i\|_2^2. \quad (4.22)$$

Then by using notations (4.14), (4.15), and (4.22), problem (4.21) can be equivalently expressed as

$$\begin{array}{ll} \text{minimize} & \displaystyle\sum_{l\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 \\ \text{subject to} & \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{L}(n) \\ & \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{I}_{\mathrm{int}}(n) \end{array} \quad (4.23)$$

with variables $\mathbf{M}_n = [\mathbf{m}_l]_{l\in\mathcal{L}(n)}$ and $\mathbf{x}_n$, where $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b\in\mathcal{N}_{\mathrm{int}}(l)}$ is a subvector of $\mathbf{x}_n$ (see (4.12)), the vector $\mathbf{h}_{jl}$ in the second set of constraints denotes the channel from BS n to link l (i.e., index j in the third set of constraints denotes an arbitrary link in $\mathcal{L}(n)$), and the notation $\succeq_{\mathrm{SOC}}$ denotes the generalized inequalities with respect to the second-order cone [132], [29, Sec. 2.2.3]. Note that in the objective function of (4.23) we have dropped a constant term $\frac{\rho}{2}\|\mathbf{v}_n^i\|_2^2$, since it does not affect the solution of the problem.

Moreover, by writing problem (4.23) in the epigraph form, and then following the approach of [132, Sec. IV-B], problem (4.23) can be equivalently reformulated in a form of SOCP as

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} t \\ \mathrm{vec}(\mathbf{M}_n) \\ \sqrt{(\rho/2)}(\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i) \end{bmatrix} \succeq_{\mathrm{SOC}} 0 \\ & \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{L}(n) \\ & \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{I}_{\mathrm{int}}(n), \end{array} \quad (4.24)$$

with variables t, $\mathbf{M}_n$, and $\mathbf{x}_n$. Let us denote $t^\star$, $\mathbf{M}_n^\star$, and $\mathbf{x}_n^\star$ the solutions of problem (4.24), then the update $\mathbf{M}_n^{i+1} = \mathbf{M}_n^\star$ and $\mathbf{x}_n^{i+1} = \mathbf{x}_n^\star$.

Now we turn to the second step of the ADMM algorithm and provide a solution for the global variable update (4.19). The update $\{\mathbf{z}_n^{i+1}\}_{n\in\mathcal{N}}$ is a solution of the following optimization problem:

$$\text{minimize} \quad \sum_{n\in\mathcal{N}} \Big(\mathbf{u}_n^{i\mathrm{T}}(\mathbf{x}_n^{i+1} - \mathbf{z}_n) + \frac{\rho}{2}\|\mathbf{x}_n^{i+1} - \mathbf{z}_n\|_2^2 \Big), \tag{4.25}$$

with variable $\{\mathbf{z}_n\}_{n\in\mathcal{N}}$. By using the notations in (4.12), and further noting that equalities (4.13) and the equality constraints of problem (4.8) are equivalent, problem (4.25) in the components of $\mathbf{x}_n$, $\mathbf{z}_n$, and $\mathbf{u}_n$ can be expressed as

$$\text{minimize} \quad \sum_{l\in\mathcal{L}_{\text{int}}} \sum_{n\in\mathcal{N}_{\text{int}}(l)} \sum_{k\in\{n,\text{tran}(l)\}} \Big(u_{k,nl}^i (x_{k,nl}^{i+1} - z_{nl}) + \frac{\rho}{2}(x_{k,nl}^{i+1} - z_{nl})^2 \Big), \tag{4.26}$$

with variable $\{z_{nl}\}_{l\in\mathcal{L}_{\text{int}}, n\in\mathcal{N}_{\text{int}}(l)}$, where $\{u_{k,nl}\}_{k\in\{n,\text{tran}(l)\}, l\in\mathcal{L}_{\text{int}}, n\in\mathcal{N}_{\text{int}}(l)}$ are the dual variables associated with the equality constraints of problem (4.7) [25].

Problem (4.26) decouples across z_{nl}, for each $l \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$. Moreover, the objective function of problem (4.26) is quadratic in z_{nl}. Hence, by setting the gradient of (4.26) with respect to z_{nl} equal to zero, we can obtain the solution $z_{nl}^\star$ which can be expressed as

$$z_{nl}^\star \quad = \quad \frac{1}{2}\Big(x_{n,nl}^{i+1} + x_{\text{tran}(l),nl}^{i+1} + \frac{1}{\rho}(u_{n,nl}^i + u_{\text{tran}(l),nl}^i) \Big), \tag{4.27}$$

for all $l \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$. Therefore, the update $z_{nl}^{i+1} = z_{nl}^\star$ for all $l \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$. Moreover, by substituting z_{nl}^{i+1} in (4.20) [26], we can show that the sum of the dual variables $u_{n,nl}^i + u_{\text{tran}(l),nl}^i$ is equal to zero. Thus, the update z_{nl}^{i+1} further simplifies to

$$z_{nl}^{i+1} \quad = \quad \frac{1}{2}\Big(x_{n,nl}^{i+1} + x_{\text{tran}(l),nl}^{i+1} \Big), \tag{4.28}$$

for all $l \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$. Hence, the global variable update z_{nl}^{i+1} is simply the average of its local copies $x_{n,nl}^{i+1}$ and $x_{\text{tran}(l),nl}^{i+1}$.

[25] Note that variables $\mathbf{u}_n$ for all $n \in \mathcal{N}$ are the dual variables associate with the consistency constraints of problem (4.16). By following steps (4.10) to (4.12), we can easily show that $\mathbf{u}_n = \big((u_{n,nl})_{l\in\mathcal{I}_{\text{int}}(n)}, (u_{n,bl})_{l\in\mathcal{L}_{\text{int}}(n), b\in\mathcal{N}_{\text{int}}(l)}\big)$ for all $n \in \mathcal{N}$.

[26] Note that (4.20) in the components of $\mathbf{u}_n$, $\mathbf{x}_n$, and $\mathbf{z}_n$ can be expressed as $u_{k,nl}^{i+1} = u_{k,nl}^i + \rho(x_{k,nl}^{i+1} - z_{nl}^{i+1})$ for all $k \in \{n, \text{tran}(l)\}, l \in \mathcal{L}_{\text{int}}$, and $n \in \mathcal{N}_{\text{int}}(l)$.

Finally, we summarize the proposed ADMM based distributed algorithm for sum-power minimization problem (4.8) as follows [27]

Algorithm 4.1. *Proposed ADMM based distributed algorithm for sum-power minimization*

1. Initialization: given SINR target $\{\gamma_l\}_{l\in\mathcal{L}}$ and penalty parameter $\rho > 0$. Set $\mathbf{u}_n^0 = 0$, and $\mathbf{z}_n^0 = 0$ for all $n \in \mathcal{N}$, and iteration index $i = 0$.
2. BS $n = 1 \dots N$ update local variables $\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}\}$ by solving problem (4.24).
3. BS n and BS tran(l) exchange their local copies $x_{n,nl}^{i+1}$ and $x_{\text{tran}(l),nl}^{i+1}$ for all $l \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$.
4. BS $n = 1 \dots N$ update global variable $\mathbf{z}_n^{i+1}$ by using expression (4.28).
5. BS $n = 1 \dots N$ update dual variable $\mathbf{u}_n^{i+1}$ by using expression (4.20).
6. If stopping criteria is satisfied, STOP. Otherwise set $i = i + 1$, and go to step 2.

The first step initializes the algorithm. Step 2 updates the local variables of each BS by solving problem (4.24). Step 2 is completely decentralized. In step 3, the neighboring BSs that are coupled by variable z_{nl}, i.e., BS n and BS tran(l), exchange their local copies $x_{n,nl}^{i+1}$ and $x_{\text{tran}(l),nl}^{i+1}$. In step 4, each BS updates the global variable $\mathbf{z}_n^{i+1}$. Note that the global variable update $\mathbf{z}_n^{i+1}$ in its component is simply the average of its local copies, see (4.28). In step 5, the dual variables are updated by each BS, by solving (4.20). Step 6 checks the stopping criteria[28], and the algorithm stops if the stopping criteria is satisfied. Otherwise, the algorithm continues in an iterative manner. Note that in deriving Algorithm 4.1 we have considered perfect CSI in all relevant channels between BSs and receivers. The impact of imperfect CSI in the derivation of the algorithm can be found in [98].

[27] Algorithm 4.1 progresses when all BSs solve their respective subproblems. Thus, in practice, a BS with the slowest processing speed (or communication delay) can affect the algorithm convergence time. This delay in the progress of the algorithm can be minimized by adopting the concept of an asynchronous ADMM [192].

[28] In the ADMM algorithm, standard stopping criteria is to check primal and dual residuals [113]. However, it is often the case that the ADMM can produce acceptable results of practical use within a few tens of iterations [113]. As, a finite number of iterations is more favorable for practical implementation, we adopt a fixed number of iterations to stop the algorithm.

4.2.3 Finding feasible solution at each iteration

In many practical applications we have to stop the distributed algorithm after a finite number of iterations before converging the algorithm. On the other hand, the intermediate solutions provided by Algorithm 4.1 do not necessarily result feasible solution. In particular, the SINR constraints of problem (4.4) may not hold, since the local copies $x_{n,nl}$ and $x_{\mathrm{tran}(l),nl}$ that correspond to the global variable z_{nl}, for all $l \in \mathcal{L}_{\mathrm{int}}$ and $n \in \mathcal{N}_{\mathrm{int}}(l)$, may not be equal. Thus, we can get SINR $\Gamma_l(\mathbf{m}) \leq \gamma_l$ as a solution of step 2 of Algorithm 4.1 for some $l \in \mathcal{L}$.

At the cost of solving one additional subproblem by each BS in each iteration, we can find a set of feasible beamformers $\mathbf{M}_n$ for all $n \in \mathcal{N}$. For this, in order to make the local copies $x_{n,nl}$ and $x_{\mathrm{tran}(l),nl}$ equal, we fix them to the consensus value z_{nl}^i (i.e., $x_{n,nl} = z_{nl}^i$ and $x_{\mathrm{tran}(l),nl} = z_{nl}^i$) for all $l \in \mathcal{L}_{\mathrm{int}}$ and $n \in \mathcal{N}_{\mathrm{int}}(l)$. Then solve problem (4.24) in variables t and $\mathbf{M}_n$ by each BS, which can be expressed as

$$\begin{aligned}
&\text{minimize} \quad t \\
&\text{subject to} \quad \begin{bmatrix} t \\ \mathrm{vec}(\mathbf{M}_n) \end{bmatrix} \succeq_{\mathrm{SOC}} 0 \\
&\qquad \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{L}(n) \\
&\qquad \begin{bmatrix} x_{n,nl}^i \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{I}_{\mathrm{int}}(n),
\end{aligned} \tag{4.29}$$

where $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b \in \mathcal{N}_{\mathrm{int}}(l)}$. Note that at iteration i the set of beamformer $\mathbf{M}_n$ for all $n \in \mathcal{N}$ is feasible for the original problem (4.4), if problem (4.29) is feasible for all BSs.

4.2.4 Convergence

The convergence of Algorithm 4.1 to the global optimal solution of problem P1 (i.e., problem (4.4)) can be established by using proposition 4.2 in [193].

First, by applying proposition 4.2 in [193] to problem (4.16), we can show that the ADMM Algorithm 4.1 converges to the global optimal solution of

problem (4.16) (note that problem (4.16) is a compact representation of problem (4.8)). Next, we note that the phase of the optimization variable $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$ in problems (4.8) and (4.4) does not change the objective and the constraints of both problems. Thus, the optimal solution obtained by Algorithm 4.1 for problem (4.8) is also optimal for problem (4.4) (i.e., problem P1).

4.3 SINR balancing

In this section we derive a distributed algorithm for problem (4.5), i.e., P2. As before in the sum-power minimization problem, we begin by reformulating problem (4.5) in the global consensus form. Then we apply the ADMM [113] to derive the distributed algorithm.

4.3.1 Equivalent reformulation

We start by equivalently reformulating SINR balancing problem (4.5) in the epigraph form [29] as

$$\begin{aligned} \text{minimize} \quad & -\gamma \\ \text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(t_l),\,j\,\neq\,l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{n\in\mathcal{N}_{\mathrm{int}}(l)} z_{nl}^2} \geq \gamma, \quad l\in\mathcal{L} \\ & z_{nl}^2 \geq \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, \quad l\in\mathcal{L}_{\mathrm{int}}, n\in\mathcal{N}_{\mathrm{int}}(l) \\ & \sum_{j\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n\in\mathcal{N}, \end{aligned} \tag{4.30}$$

with variables γ, $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$, and $\{z_{nl}\}_{l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$.

We now follow a similar approach as in Section 4.2.1 to express problem (4.30) in a global consensus form (i.e., we introduce the local copies of the coupling variables γ and z_{nl} for each BS). Since the SINR variable γ couples all BSs via SINR constraints, we introduce local copies α_n for each BS such that $\alpha_n = \gamma$ for all $n\in\mathcal{N}$. For the out-of-cell interference variable z_{nl}, we introduce local copies $x_{k,nl}$ and $x_{\mathrm{tran}(l),nl}$, respectively, for BS n and BS $\mathrm{tran}(l)$ as in problem (4.7). Then problem (4.30) in the global consensus form can be expressed equivalently as

$$
\begin{aligned}
\text{minimize} \quad & -\gamma \\
\text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(n),\,j\,\neq\,l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{b\in\mathcal{N}_{\mathrm{int}}(l)} x_{n,bl}^2} \geq \alpha_n, \quad n \in \mathcal{N}, l \in \mathcal{L}(n) \\
& x_{n,nl}^2 \geq \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, \quad n \in \mathcal{N}, l \in \mathcal{I}_{\mathrm{int}}(n) \\
& \sum_{j\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N} \\
& x_{k,nl} = z_{nl}, \quad k \in \{n, \mathrm{tran}(l)\}, l \in \mathcal{L}_{\mathrm{int}}, n \in \mathcal{N}_{\mathrm{int}}(l) \\
& \alpha_n = \gamma, \quad n \in \mathcal{N},
\end{aligned}
\tag{4.31}
$$

where the variables are γ, $\{\mathbf{m}_l\}_{l\in\mathcal{L}}$, $\{\alpha_n\}_{n\in\mathcal{N}}$, $\{x_{k,nl}\}_{k\in\{n,\mathrm{tran}(l)\},n\in\mathcal{N},l\in\mathcal{I}_{\mathrm{int}}(n)}$, and $\{z_{nl}\}_{l\in\mathcal{L}_{\mathrm{int}},n\in\mathcal{N}_{\mathrm{int}}(l)}$. Note that in the second set of inequality constraints we use the equivalence between the sets $\{(n,l)|l \in \mathcal{L}_{\mathrm{int}}, n \in \mathcal{N}_{\mathrm{int}}(l)\}$ and $\{(n,l)|n \in \mathcal{N}, l \in \mathcal{I}_{\mathrm{int}}(n)\}$ (see (4.10)).

Now we express problem (4.31) more compactly. Note that except the third and fifth constraints of problem (4.31), the constraints set of problem (4.31) is identical to that of problem (4.7). Hence, we can use variables $\mathbf{M}_n$, $\mathbf{x}_n$, and $\mathbf{z}_n$ to define a set

$$
\mathcal{C}_n = \left\{ \mathbf{M}_n, \mathbf{x}_n, \alpha_n \;\middle|\; \begin{array}{l} \dfrac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j\,\in\,\mathcal{L}(n),\,j\,\neq\,l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{b\in\mathcal{N}_{\mathrm{int}}(l)} x_{n,bl}^2} \geq \alpha_n, l \in \mathcal{L}(n) \\ x_{n,nl}^2 \geq \sum\limits_{j\in\mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, l \in \mathcal{I}_{\mathrm{int}}(n) \\ \sum\limits_{j\in\mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max} \end{array} \right\},
\tag{4.32}
$$

and the following indicator function:

$$
I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) = \begin{cases} 0 & (\mathbf{M}_n, \mathbf{x}_n, \alpha_n) \in \mathcal{C}_n \\ \infty & \text{otherwise.} \end{cases}
\tag{4.33}
$$

Then by using notations (4.32) and (4.33), consensus problem (4.31) can be rewritten compactly as

$$
\begin{aligned}
\text{minimize} \quad & -\gamma + \sum_{n\in\mathcal{N}} I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) \\
\text{subject to} \quad & \mathbf{x}_n = \mathbf{z}_n, \quad n \in \mathcal{N} \\
& \alpha_n = \gamma, \quad n \in \mathcal{N},
\end{aligned}
\tag{4.34}
$$

where the variables are γ, $\{\mathbf{M}_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{x}_n, \mathbf{z}_n, \alpha_n\}_{n\in\mathcal{N}}$. Furthermore, from the second equality constraints of problem (4.34), we can write $\sum_{n\in\mathcal{N}} \alpha_n = N\gamma$. Thus, problem (4.34) can be expressed equivalently as

$$\begin{aligned} \text{minimize} \quad & \sum_{n\in\mathcal{N}} \Big(-\frac{\alpha_n}{N} + I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) \Big) \\ \text{subject to} \quad & \mathbf{x}_n = \mathbf{z}_n, \quad n \in \mathcal{N} \\ & \alpha_n = \gamma, \quad n \in \mathcal{N}, \end{aligned} \tag{4.35}$$

with variables γ, $\{\mathbf{M}_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{x}_n, \mathbf{z}_n, \alpha_n\}_{n\in\mathcal{N}}$.

4.3.2 Distributed algorithm via ADMM

To derive the ADMM algorithm we first form the augmented Lagrangian [191] of problem (4.35). Let $\mathbf{u}_n$ and v_n be the dual variables associated with the first and second consensus constraints of problem (4.35), respectively. Then the augmented Lagrangian can be written as

$$\begin{aligned} L_\rho\big(\{\mathbf{M}_n, \mathbf{x}_n, \alpha_n, \mathbf{u}_n, v_n, \mathbf{z}_n\}_{n\in\mathcal{N}}, \gamma\big) &= \sum_{n\in\mathcal{N}} \big(-\frac{\alpha_n}{N} + I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n)\big) \\ &+ \sum_{n\in\mathcal{N}} \big(\mathbf{u}_n^{\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n) + v_n(\alpha_n - \gamma) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n\|_2^2 + \frac{\rho}{2}(\alpha_n - \gamma)^2\big), \end{aligned} \tag{4.36}$$

where $\rho > 0$ is the penalty parameter. Each iteration of ADMM consists of the following steps [113]:

$$\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}, \alpha_n^{i+1} = \underset{\mathbf{M}_n, \mathbf{x}_n, \alpha_n}{\operatorname{argmin}} \; L_\rho\big(\mathbf{M}_n, \mathbf{x}_n, \alpha_n, \mathbf{u}_n^i, v_n^i, \mathbf{z}_n^i, \gamma^i\big), \quad n \in \mathcal{N} \tag{4.37}$$

$$\{\mathbf{z}_n^{i+1}\}_{n\in\mathcal{N}}, \gamma^{i+1} = \underset{\{\mathbf{z}_n\}_{n\in\mathcal{N}}, \gamma}{\operatorname{argmin}} \; L_\rho\big(\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}, \alpha_n^{i+1}, \mathbf{u}_n^i, v_n^i, \mathbf{z}_n\}_{n\in\mathcal{N}}, \gamma\big) \tag{4.38}$$

$$\mathbf{u}_n^{i+1} = \mathbf{u}_n^i + \rho\big(\mathbf{x}_n^{i+1} - \mathbf{z}_n^{i+1}\big), \quad n \in \mathcal{N} \tag{4.39}$$

$$v_n^{i+1} = v_n^i + \rho\big(\alpha_n^{i+1} - \gamma^{i+1}\big), \quad n \in \mathcal{N}. \tag{4.40}$$

Note that the first step is completely decentralized. Each BS $n \in \mathcal{N}$ updates the local variables $\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}, \alpha_n^{i+1}\}$ by solving the following optimization problem:

$$\begin{aligned} \text{minimize} \quad & -\frac{\alpha_n}{N} + I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) + \mathbf{u}_n^{i\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n^i) + v_n^i(\alpha_n - \gamma^i) \\ & + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i\|_2^2 + \frac{\rho}{2}(\alpha_n - \gamma^i)^2, \end{aligned} \tag{4.41}$$

with variables α_n, $\mathbf{M}_n$, and $\mathbf{x}_n$. Let $\mathbf{v}_n = (1/\rho)\mathbf{u}_n$ and $\lambda_n = (1/\rho)v_n$, then by simplifying the terms of the objective function of problem (4.41) [29], it can be written as

$$\text{minimize} \quad -\frac{\alpha_n}{N} + I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 + \frac{\rho}{2}(\alpha_n - \gamma^i + \lambda_n^i)^2, \tag{4.42}$$

with variables α_n, $\mathbf{M}_n$, and $\mathbf{x}_n$. Note that in the objective function of problem (4.42) the constant terms $(\rho/2)\|\mathbf{v}_n^i\|_2^2$ and $(\rho/2)(\lambda_n^i)^2$ are dropped, since they do not affect the solution of the optimization problem.

Problem (4.42) is not a convex problem, because the indicator function $I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n)$ is a function of nonconvex set $\mathcal{C}_n$ (see expression (4.32)). However, for fixed α_n set $\mathcal{C}_n$ is a convex set, and hence problem (4.42) can be solved easily. Therefore, to solve problem (4.42) we first find the optimal $\alpha_n^\star$, and determine the corresponding $\mathbf{M}_n^\star$ and $\mathbf{x}_n^\star$.

For fixed α_n, let us denote the optimal value function of problem (4.42) as

$$\begin{aligned} p(\alpha_n) = &\inf_{\mathbf{M}_n, \mathbf{x}_n} \Big(-\frac{\alpha_n}{N} + I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 \\ &\qquad + \frac{\rho}{2}(\alpha_n - \gamma^i + \lambda_n^i)^2 \Big) \qquad (4.43) \\ = &\inf_{\mathbf{M}_n, \mathbf{x}_n} \Big(I_n(\mathbf{M}_n, \mathbf{x}_n, \alpha_n) + \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 \Big) - \frac{\alpha_n}{N} \\ &\qquad + \frac{\rho}{2}(\alpha_n - \gamma^i + \lambda_n^i)^2, \qquad (4.44) \end{aligned}$$

where (4.44) follows by noting that α_n/N and $(\rho/2)(\alpha_n - \gamma^i + \lambda_n^i)^2$ are independent of the optimization variables $\mathbf{M}_n$ and $\mathbf{x}_n$. Then the optimal value of problem (4.42) is given by

$$p^\star = \inf_{\alpha_n} p(\alpha_n). \tag{4.45}$$

For ease of presentation, let us express the optimal value function $p(\alpha_n)$ in (4.44) as

$$p(\alpha_n) = \tilde{p}(\alpha_n) - \frac{\alpha_n}{N} + \frac{\rho}{2}(\alpha_n - \gamma^i + \lambda_n^i)^2, \tag{4.46}$$

[29] For convenience we can combine the terms in problem (4.41) as a) $\mathbf{u}_n^{i\mathrm{T}}(\mathbf{x}_n - \mathbf{z}_n^i) + (\rho/2)\|\mathbf{x}_n - \mathbf{z}_n^i\|_2^2 = (\rho/2)\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 - (\rho/2)\|\mathbf{v}_n^i\|_2^2$ and b) $v_n^i(\alpha_n - \gamma^i) + (\rho/2)(\alpha_n - \gamma^i)^2 = (\rho/2)(\alpha_n - \gamma^i + \lambda_n^i)^2 - (\rho/2)(\lambda_n^i)^2$.

where $\tilde{p}(\alpha_n)$ is the optimal value of the following optimization problem:

$$\begin{aligned}
\text{minimize} \quad & \frac{\rho}{2}\|\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i\|_2^2 \\
\text{subject to} \quad & \frac{|\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l|^2}{\sigma_l^2 + \sum\limits_{j \,\in\, \mathcal{L}(n),\, j \neq l} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2 + \sum\limits_{b \in \mathcal{N}_{\mathrm{int}}(l)} x_{n,bl}^2} \geq \alpha_n, \quad l \in \mathcal{L}(n) \\
& x_{n,nl}^2 \geq \sum_{j \in \mathcal{L}(n)} |\mathbf{h}_{jl}^{\mathrm{H}}\mathbf{m}_j|^2, \quad l \in \mathcal{I}_{\mathrm{int}}(n) \\
& \sum_{j \in \mathcal{L}(n)} \|\mathbf{m}_l\|_2^2 \leq p_n^{\max},
\end{aligned} \tag{4.47}$$

with optimization variables $\mathbf{x}_n$ and $\{\mathbf{m}_l\}_{l \in \mathcal{L}(n)}$. Note that to write problem (4.47) we have used the notations defined in (4.32) and (4.33). Furthermore, by writing problem (4.47) in the epigraph form, and then following the approach of [132, Sec. IV-B], it can be formulated equivalently in an SOCP form as

$$\begin{aligned}
\text{minimize} \quad & t \\
\text{subject to} \quad & \begin{bmatrix} t \\ \sqrt{\rho/2}(\mathbf{x}_n - \mathbf{z}_n^i + \mathbf{v}_n^i) \end{bmatrix} \succeq_{\mathrm{SOC}} 0 \\
& \begin{bmatrix} \sqrt{1 + \frac{1}{\alpha_n}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_n \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{L}(n) \\
& \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l \in \mathcal{I}_{\mathrm{int}}(n) \\
& \begin{bmatrix} \sqrt{p_n^{\max}} \\ \mathrm{vec}(\mathbf{M}_n) \end{bmatrix} \succeq_{\mathrm{SOC}} 0,
\end{aligned} \tag{4.48}$$

with variables t, $\mathbf{x}_n$, and $\mathbf{M}_n$, where $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b \in \mathcal{N}_{\mathrm{int}}(l)}$ is a subvector of $\mathbf{x}_n$ (see expression (4.12)), the vector $\mathbf{h}_{jl}$ in the third set of constraints denotes the channel from BS n to link l (i.e., the index j in the third set of constraints denotes an arbitrary link in $\mathcal{L}(n)$). Note that to write problem (4.47) in the SOCP form (4.48), we first took the square root of the objective function of (4.47). Hence, the optimal value of problem (4.47) is given by $t^{\star 2}$ (i.e., $\tilde{p}(\alpha_n) = t^{\star 2}$), where $t^\star$ is the solution of problem (4.48).

We now propose a method to solve problem (4.45). Let an interval $[0, \alpha_n^{\max}]$ denotes a range of feasible α_n for problem (4.45). Note that the optimal value $\tilde{p}(\alpha_n)$ is a nondecreasing function of $\alpha_n \in [0, \alpha_n^{\max}]$. Based on this observation,

in Appendix 2 we have provided the condition for which $p(\alpha_n)$ is a unimodal function, and we use the bracketing method [187, 188] to solve problem (4.45). In Algorithm 4.2, we summarize the bracketing method (golden ratio search) [187, Sec. 8.1] to find the optimal $\alpha_n^\star$ for problem (4.45).

Algorithm 4.2. *Bracketing method to find optimal $\alpha_n^\star$ for problem (4.45)*

1. Initialization: given SINR interval $[0, \alpha_n^{\max}]$, $k = (\sqrt{5}-1)/2$, and $\epsilon_g > 0$. Set $a = 0$, $b = \alpha_n^{\max}$, $c = ka + (1-k)b$, and $d = (1-r)a + kb$.
2. Compute $p(c)$ and $p(d)$ using (4.46).
3. Squeeze the search SINR range: if $p(c) \leq p(d)$, set $b = d$, else set $a = c$.
4. Compute $c = ka + (1-k)b$ and $d = (1-k)a + kb$.
5. Stopping criterion: if $b - a < \epsilon_g$, STOP, and set $\alpha_n^\star = c$. Otherwise, go to step 2.

At step 2 of Algorithm 4.2, to compute expression (4.46), we need to solve problem (4.48). Thus, the optimal $\mathbf{x}_n^\star$ and $\mathbf{M}_n^\star = [\mathbf{m}_l^\star]_{l \in \mathcal{L}(n)}$ that are associated with $\alpha_n^\star$ are obtained as a by-product of Algorithm 4.2.

We now turn to the second step of ADMM in (4.38), where the global variables $\{\mathbf{z}_n^{i+1}\}_{n\in\mathcal{N}}$ and γ^{i+1} are updated. By dropping the constant terms which do not affect the solution, problem (4.38) can be written as

$$\begin{aligned} \text{minimize} \quad & \sum_{n\in\mathcal{N}} \Big(\mathbf{u}_n^{iT}(\mathbf{x}_n^{i+1} - \mathbf{z}_n) + v_n^i(\alpha_n^{i+1} - \gamma) \\ & \qquad + \frac{\rho}{2}\|\mathbf{x}_n^{i+1} - \mathbf{z}_n\|_2^2 + \frac{\rho}{2}(\alpha_n^{i+1} - \gamma)^2 \Big), \end{aligned} \tag{4.49}$$

with variables $\{\mathbf{z}_n\}_{n\in\mathcal{N}}$ and γ.

Problem (4.49) is separable in variables $\{\mathbf{z}_n\}_{n\in\mathcal{N}}$ and γ. Note that the minimization of problem (4.49) with respect to $\{\mathbf{z}_n\}_{n\in\mathcal{N}}$ yields problem (4.25), and hence, the solution $\{\mathbf{z}_n^\star\}_{n\in\mathcal{N}}$ is given by (4.28). Here, we provide the solution for γ. The minimization of problem (4.49) with respect to γ yields the following optimization problem:

$$\text{minimize} \quad \sum_{n\in\mathcal{N}} \Big(v_n^i(\alpha_n^{i+1} - \gamma) + \frac{\rho}{2}(\alpha_n^{i+1} - \gamma)^2 \Big). \tag{4.50}$$

Problem (4.50) is an unconstrained quadratic optimization problem in γ. Therefore, by setting the gradient of problem (4.50) with respect to γ equal to zero,

we can get

$$\gamma^{\star} = \frac{\sum_{n\in\mathcal{N}} v_n^i + \rho\alpha_n^{i+1}}{\rho N}. \tag{4.51}$$

Hence, the update $\gamma^{i+1} = \gamma^{\star}$. Moreover, by substituting γ^{i+1} in (4.40), we can show that the sum of the dual variables $\sum_{n\in\mathcal{N}} v_n^i$ is equal to zero. Thus, the update γ^{i+1} (i.e., expression (4.51)) further simplifies to

$$\gamma^{i+1} = \frac{\sum_{n\in\mathcal{N}} \alpha_n^{i+1}}{N}. \tag{4.52}$$

We now summarize the proposed ADMM based distributed algorithm for SINR balancing problem P2 in Algorithm 4.3.

Algorithm 4.3. *Proposed ADMM based distributed algorithm for SINR balancing problem P2*

1. Initialization: given maximum transmit power $\{p_n^{\max}\}_{n\in\mathcal{N}}$ and penalty parameter $\rho > 0$. Set $\mathbf{u}_n^0 = 0$, $v_n^0 = 0$ for all $n \in \mathcal{N}$, and iteration index $i = 0$.
2. BS $n = 1 \dots N$ update local variables $\{\mathbf{M}_n^{i+1}, \mathbf{x}_n^{i+1}, \alpha_n^{i+1}\}$ by using Algorithm 4.2.
3. Exchange local updates:
   a) BS n and BS tran(l) exchange their local copies $x_{n,nl}^{i+1}$ and $x_{\text{tran}(l),nl}^{i+1}$ for all $n \in \mathcal{L}_{\text{int}}$ and $n \in \mathcal{N}_{\text{int}}(l)$.
   b) BS n transmits local copy α_n^{i+1} to all other BSs, for all $n \in \mathcal{N}$.
4. BS $n = 1 \dots N$ update global variables $\mathbf{z}_n^{i+1}$ and γ by using expressions (4.28) and (4.52), respectively.
5. BS $n = 1 \dots N$ update dual variables $\mathbf{u}_n^{i+1}$ and v_n^{i+1} by using expressions (4.39) and (4.40).
6. If stopping criteria is satisfied, STOP. Otherwise set $i = i + 1$, and go to step 2.

The computational steps of Algorithm 4.3 are similar to those of Algorithm 4.1. As in Algorithm 4.1, step 1 initializes the algorithm. Step 2 updates the local variables. In step 3, BSs exchange their local copies to update global variables. Local copies $x_{n,nl}^{i+1}$ and $x_{\text{tran}(l),nl}^{i+1}$ are exchanged between the adjacent BS n and

BS tran(l), while local copy α_n is broadcasted to all other BSs. Steps 4 and 5 update the global and dual variables, respectively. Note that steps 2, 4 and 5 are completely decentralized. Step 6 checks the stopping criteria[30].

4.3.3 Finding feasible solution at each iteration

Note that at each step of Algorithm 4.3, the locally obtained SINR α_n for all $n \in \mathcal{N}$ are not necessarily balanced (i.e., α_n for all $n \in \mathcal{N}$ are not necessarily equal) before converging the algorithm. So, we can take the global variable γ^i, which is the average of α_n for all $n \in \mathcal{N}$, as the intermediate solution of Algorithm 4.3. However, due to the difference in the local copies $x_{n,nl}^{i+1}$ at BS n and $x_{\text{tran}(l),nl}^{i+1}$ at BS tran(l), and the maximum transmit power constraint of the BSs, the intermediate solution γ^i may not be feasible for all BSs.

Therefore, it is necessary to check the feasibility of γ^i to use it as an intermediate solution at each step of Algorithm 4.3. The SINR γ^i is feasible for BS n, if their exist a feasible solution of problem (4.48) for $\alpha_n = \gamma^i$ and given out-of-cell interference value $\mathbf{x}_n$. Thus, we set $\alpha_n = \gamma^i$ and $\mathbf{x}_n = \mathbf{z}_n^i$ for all $n \in \mathcal{N}$ (i.e., α_n and $\mathbf{x}_n$ are set equal to the consensus value). Then the feasibility of problem (4.48) is checked by each BS in between steps 4 and 5 of Algorithm 4.3, which is equivalent to the following SOCP feasibility problem:

$$
\begin{aligned}
\text{find} \quad & \{\mathbf{m}_l\}_{l \in \mathcal{L}(n)} \\
\text{subject to} \quad & \begin{bmatrix} \sqrt{1+\frac{1}{\alpha_n}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\text{SOC}} 0, \quad l \in \mathcal{L}(n) \\
& \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\text{SOC}} 0, \quad l \in \mathcal{I}_{\text{int}}(n) \\
& \begin{bmatrix} \sqrt{p_n^{\max}} \\ \text{vec}(\mathbf{M}_n) \end{bmatrix} \succeq_{\text{SOC}} 0, \quad l \in \mathcal{I}_{\text{int}}(n)
\end{aligned}
\tag{4.53}
$$

[30] In the ADMM algorithm, standard stopping criteria is to check primal and dual residuals [113]. However, it is often the case that the ADMM can produce acceptable results of practical use within a few tens of iterations [113]. As, a finite number of iterations is favorable for practical implementation, we adopt a fixed number of iterations to stop the algorithm.

with variable $\mathbf{M}_n = [\mathbf{m}_l]_{l\in\mathcal{L}(n)}$, where $\tilde{\mathbf{x}}_l = (x_{n,bl})_{b\in\mathcal{N}_{\text{int}}(l)}$ is a subvector of $\mathbf{x}_n$ (see (4.12)), the vector $\mathbf{h}_{jl}$ in the third set of constraints denotes the channel from BS n to link l (i.e., the index j in the third set of constraints denotes an arbitrary link in $\mathcal{L}(n)$). Note that γ^i is feasible for the original problem (4.5) only if problem (4.53) is feasible for all BSs. Thus, in Algorithm 4.3 we can update the feasible SINR γ^i_{feas} as (assume $\gamma^0_{\text{feas}} = 0$)

$$\gamma^i_{\text{feas}} = \begin{cases} \gamma^i & \text{if problem (4.53) is feasible for all } n \in \mathcal{N} \\ \gamma^{i-1}_{\text{feas}} & \text{otherwise,} \end{cases} \tag{4.54}$$

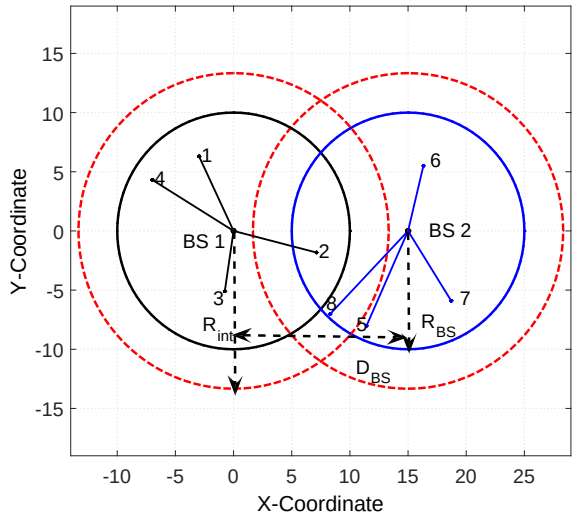
4.4 Numerical examples

In this section we numerically evaluate the performance of proposed Algorithm 4.1 and Algorithm 4.3. In our simulations two multicell wireless networks as shown in Fig. 4.3 are considered. In the case of the first network [i.e., Fig. 4.3(a)], there are $N = 2$ BSs with $T = 4$ antennas at each one. The distance between BSs is denoted by D_{BS}. In the case of second network [i.e., Fig. 4.3(b)], there are $N = 7$ BSs with $T = 6$ antennas at each one. The BSs are located such that they form a hexagon, and the distance between the adjacent BSs is denoted by D_{BS}. We assume that the BSs have circular transmission and interference regions, where the radius of the transmission region of each BS is denoted by R_{BS}, and the radius of the interference region of each BS is denoted by R_{int}. For simplicity, we assume 4 users per cell in the first network, and 3 users per cell in the second network. The location of users associated with the BSs are arbitrarily chosen as shown in Fig. 4.3.

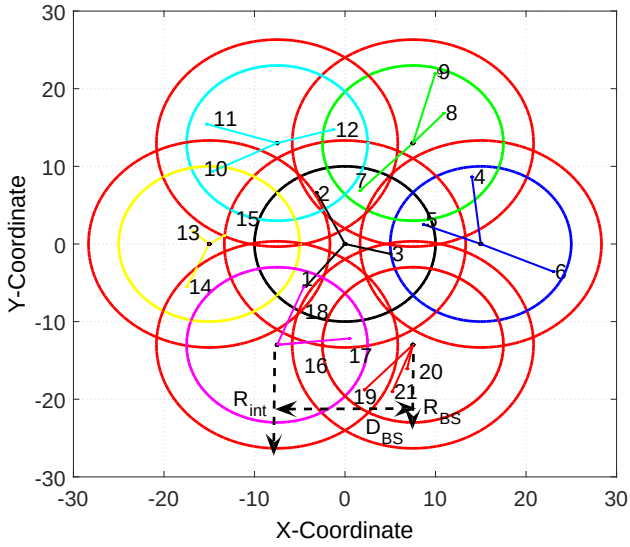
We assume an exponential path loss model, where the channel vector between BSs and users is modeled as

$$\mathbf{h}_{jl} = \left(\frac{d_{jl}}{d_0}\right)^{-\eta/2} \mathbf{c}_{jl},$$

where d_{jl} is the distance from the transmitter of data stream j (i.e., BS $\text{tran}(j)$) to the receiver of data stream l (i.e., user $\text{rec}(l)$), d_0 is the far field reference distance [173], η is the path loss exponent, and $\mathbf{c}_{jl} \in \mathbb{C}^T$ is arbitrarily chosen from the distribution $\mathcal{CN}(0, \mathbf{I})$ (i.e., frequency-flat fading channel with uncorrelated antennas). Here, we refer to an arbitrarily generated set of fading coefficients $\check{\mathcal{C}} = \{\mathbf{c}_{jl} | j, l \in \mathcal{L}\}$ as a single fading realization.



(a)



(b)

Fig. 4.3. (a) Multicell network 1, $\mathcal{N} = \{1, 2\}$**,** $\mathcal{L}(1) = \{1, 2, 3, 4\}$**,** $\mathcal{L}(2) = \{5, 6, 7, 8\}$**,** $\mathcal{L}_{\text{int}} = \{2, 8\}$**; (b) Multicell network 2,** $\mathcal{N} = \{1, 2, 3, 4, 5, 6, 7\}$**,** $\mathcal{L}(1) = \{1, 2, 3\}$**,** $\mathcal{L}(2) = \{4, 5, 6\}$**,** $\mathcal{L}(3) = \{7, 8, 9\}$**,** $\mathcal{L}(4) = \{10, 11, 12\}$**,** $\mathcal{L}(5) = \{13, 14, 15\}$**,** $\mathcal{L}(6) = \{16, 17, 18\}$**,** $\mathcal{L}(7) = \{19, 20, 21\}$**,** $\mathcal{L}_{\text{int}} = \{1, 2, 3, 4, 5, 7, 10, 12, 14, 15, 16, 17, 18, 19\}$**.**

We assume that the maximum power constraint is same for each BS, i.e., $p_n^{\max} = p_0^{\max}$ for all $n \in \mathcal{N}$, and $\sigma_l = \sigma$ for all $l \in \mathcal{L}$. We define the SNR operating point at a distance r as

$$\mathrm{SNR}(r) = \left(\frac{r}{d_0}\right)^{-\eta} \frac{p_0^{\max}}{\sigma^2}. \tag{4.55}$$

In the following simulations, we set $d_0 = 1$ and $\eta = 4$; the cell radius R_{BS} and the radius of the interference region R_{int} are fixed throughout the simulations such that $\mathrm{SNR}(R_{\mathrm{BS}}) = 5$ dB and $\mathrm{SNR}(R_{\mathrm{int}}) = 0$ dB, respectively, for $p_0^{\max}/\sigma^2 = 45$ dB. We let $D_{\mathrm{BS}} = 1.5 \times R_{\mathrm{BS}}$.

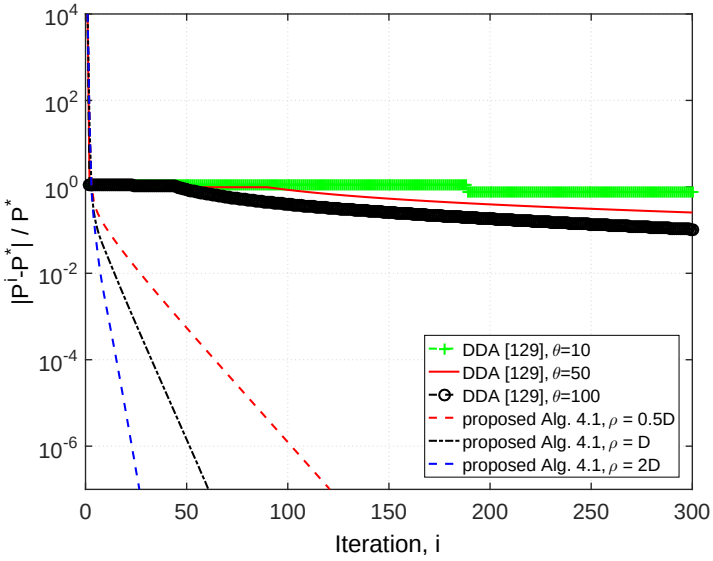
To illustrate the convergence behavior of Algorithm 4.1, we consider a single fading realization and run the algorithm for both networks shown in Fig. 4.3. For a comparison, we consider a dual decomposition based distributed algorithm (DDA) proposed in [129]. For DDA [129] we consider a fixed step size θ to solve a master problem (see in [129]), which is based on the subgradient method [130].

Fig. 4.4 shows the normalized power accuracy $|P^i - P^\star|/P^\star$, where P^i is the objective value at ith iteration, and $P^\star$ is the optimal objective value obtained by using the centralized algorithm [132, Sec. IV]. The SINR target is set to $\gamma_l = 5$ dB for all $l \in \mathcal{L}$. DDA [129] plots are drawn for the subgradient step size $\theta = 10, 50, 100$. For Algorithm 4.1, the penalty parameter is set to $\rho = 0.5D, D, 2D$, where D is defined as (detailed in Appendix 3)

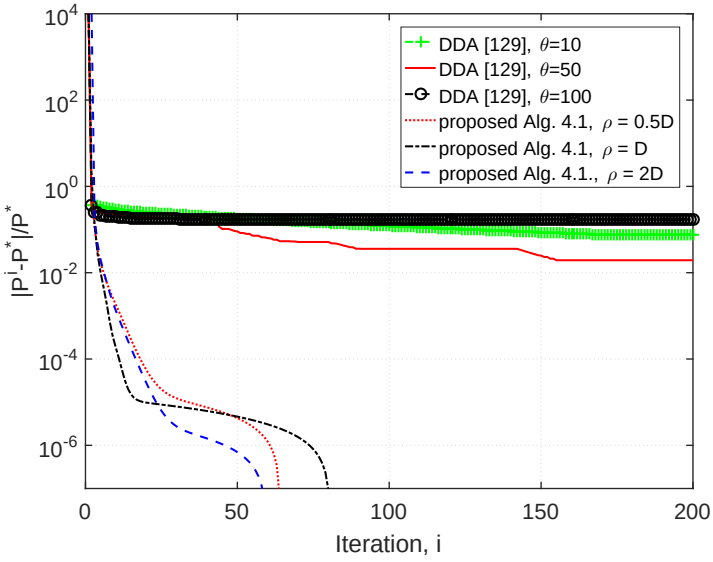
$$D = \max_{n \in \mathcal{N}} \left\{ \textstyle\sum_{l \in \mathcal{L}(n)} (10^{0.1 \times \gamma_l})/\|\mathbf{h}_{ll}\|_2^2 \right\}. \tag{4.56}$$

Results show that the proposed Algorithm 4.1 converges much faster than DDA [129]. For example, in both multicell networks Algorithm 4.1 can achieve normalized power accuracy 10^{-2} in less than 10 iterations. However, in order to achieve the same accuracy (i.e., normalized power accuracy of 10^{-2}), the DDA [129] requires more than 200 iterations for all simulated cases in both networks. Results also show that Algorithm 4.1 performs very well for a wide range of values of ρ. Hence Algorithm 4.1 is less sensitive to the variation of values of ρ, while the results show that the convergence speed of DDA [129] is quite sensitive to the variation of the subgradient step size θ.

In order to see the average behavior of the proposed Algorithm 4.1, we next run Algorithm 4.1 for 500 fading realizations with the algorithm parameter $\rho = 2D$ for both networks shown in Fig. 4.3. We first present the feasibility rate



(a)



(b)

Fig. 4.4. Normalized power accuracy versus iteration for SINR $\gamma_l = 5$ dB for all $l \in \mathcal{L}$: (a) Multicell network 1; (b) Multicell network 2.

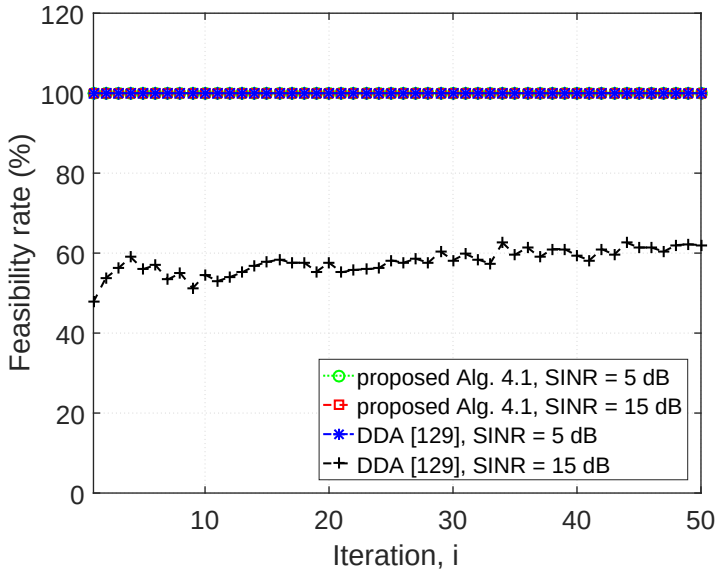
of the proposed algorithm, and then provide the average performance of the algorithm.

Fig. 4.5 shows the feasibility rate of Algorithm 4.1 versus iteration for SINR target $\gamma_l = 5$ dB and 15 dB for all $l \in \mathcal{L}$. For a comparison, we consider DDA [129] with a subgradient step size $\theta = 50$. Plots are drawn for the first 50 iterations. Results show that in the case of multicell network 4.3(a), the proposed algorithm achieves the feasible solution for all the simulated cases (see Fig. 4.5(a)); and for multicell network 4.3(b) the feasibility rate improves with iteration (see Fig. 4.5(b)). That is the proposed algorithm can achieves the feasible solution for all channel realizations. However, for DDA [129] feasibility rate depends on the network size and the SINR target. For example, in the case of small network and low SINR target (i.e., multicell network 4.3(a) and SINR target $\gamma_l = 5$ dB), DDA [129] can achieves the feasible solution for all simulated cases. But, with increase in the SINR target and the network size, the feasibility rate of DDA [129] drops significantly. For example, in multicell network 4.3(b) for SINR target $\gamma_l = 15$ dB, DDA [129] is not able to find a feasible solution for any of the fading realization (see Fig. 4.5(b)).

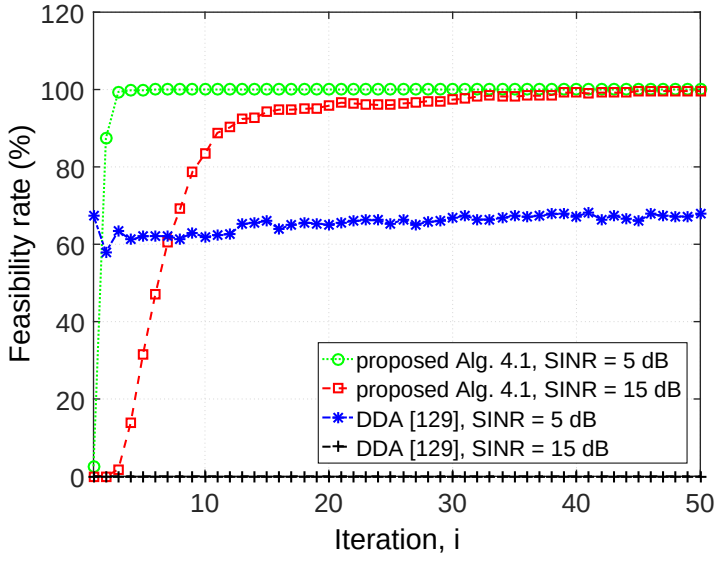
Fig. 4.6 shows the average sum-power versus iteration for multicell network 4.3(a). The SINR target γ_l is set to 15 dB for all $l \in \mathcal{L}$. For a comparison, we consider centralized algorithm [132, Sec. IV] and DDA [129]. DDA [129] plots are drawn for the subgradient step size $\theta = 10, 50$. For a fair comparison of Algorithm 4.1, DDA [129], and the centralized algorithm [132, Sec. IV], the plots are drawn for the fading realizations that are feasible for all considered algorithms. Results show that the convergence speed of proposed Algorithm 4.1 compared with DDA [129] is fast, and can achieves the centralized solution in less than 10 iterations.

Fig. 4.7 shows the average sum-power versus SINR target for multicell network 4.3(b). For a comparison, we consider centralized algorithm [132, Sec. IV]. To note a fair progress of the proposed algorithm for a wide SINR target values, each curve is averaged for the fading realizations that are feasible for all the SINR values. Plots are drawn for the average sum-power at iteration number 20 and 50. Results show that the proposed Algorithm 4.1 can achieve the centralized solution over a wide range of SINR target values.

We next evaluate the performance of Algorithm 4.3 for the SINR balancing problem (P2). We first consider a single fading realization and run the algorithm



(a)



(b)

Fig. 4.5. Feasibility rate versus iteration for SINR target $\gamma_l = 5$ dB and 15 dB for all $l \in \mathcal{L}$: (a) Multicell network 1; (b) Multicell network 2.

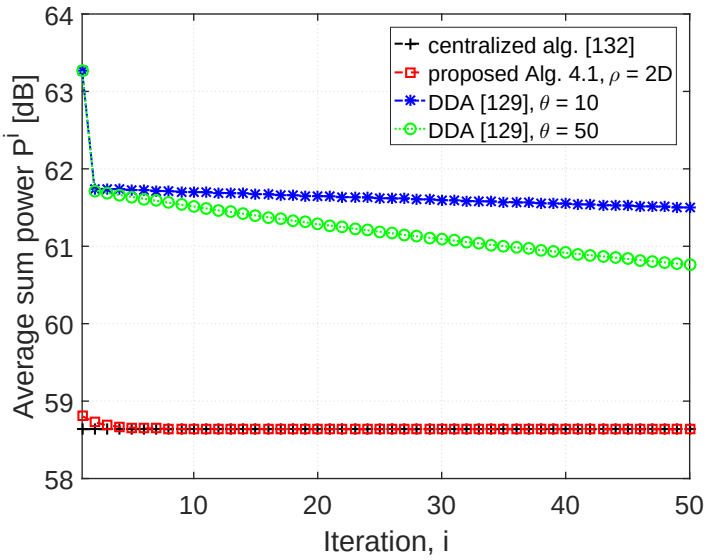


Fig. 4.6. Multicell network 1: Average sum-power versus iteration for SINR target $\gamma_l = 15$ dB for all $l \in \mathcal{L}$.

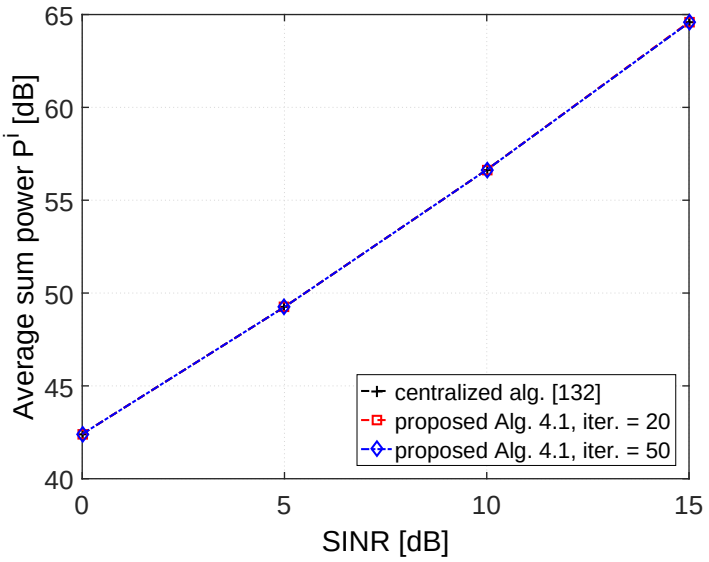


Fig. 4.7. Multicell network 2: Average sum-power versus SINR for $\rho = 2D$.

for both networks shown in Fig. 4.3. As a benchmark, we consider centralized optimal algorithm proposed in [132, Sec. V]. In the simulation, we set SNR = 5 dB, and for Algorithm 4.2, we set $\epsilon_g = 0.1$ and $\alpha_n^{\max} = 2 \times 10^{0.1 \times \text{SNR}}$ for all $n \in \mathcal{N}$. Plots are drawn for $\rho = 0.5, 1, 2$.

Fig. 4.8 shows the progress of the global variable γ by iteration. Note that the global variable γ is the average of SINR values $\alpha_1, \ldots, \alpha_N$ that are obtained independently by all N BSs (see expression (4.52)). Results show that for all the considered values of ρ, Algorithm 4.3 can obtain an SINR value γ that converges to the optimal centralized solution. Since γ is the average of the SINR values obtained independently in all N BSs, the intermediate values of γ may not be feasible for all BSs before the algorithm converges. For example, the value of γ for $\rho = 0.5$ is clearly infeasible at iteration step $i = 4, 5, 6, 7, 8$ in Fig. 4.8(a). Therefore, to illustrate the convergence of a feasible value of γ, we define the following metric:

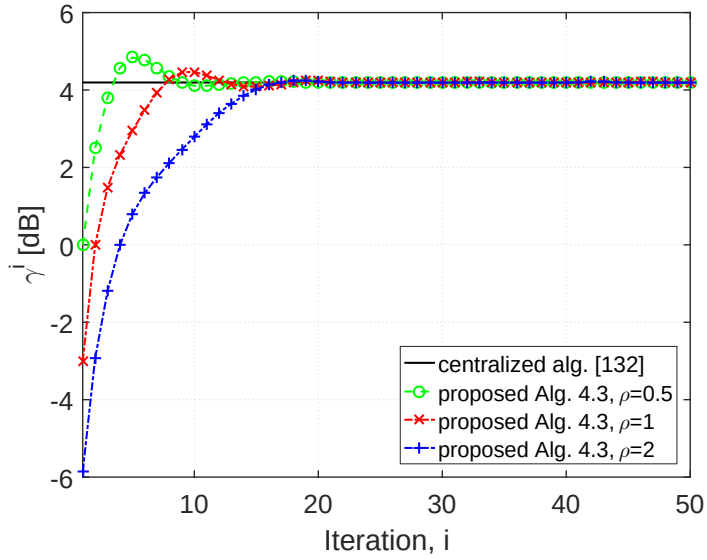
$$\gamma_{\text{best}}^i = \max_{t=1,\ldots,i} \{\gamma_{\text{feas}}^t\}, \tag{4.57}$$

where γ_{best}^i is the best feasible SINR value at ith iteration and γ_{feas}^t is the feasible SINR at tth iteration (see expression (4.54)).

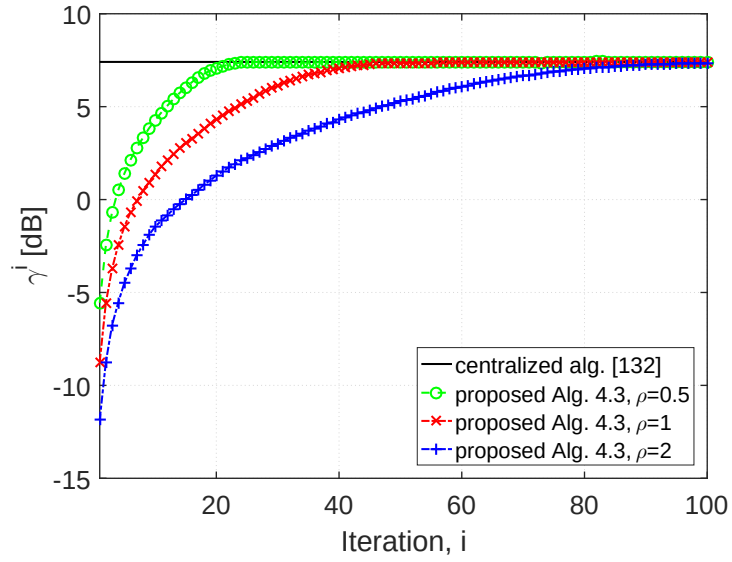
Fig. 4.9 shows the behavior of γ_{best}^i by iteration. Results show that Algorithm 4.3 can achieve the feasible values of γ that converge to the centralized solution. For example, with $\rho = 0.5$ the algorithm converges to the centralized solution in just 10th iterations (see Fig. 4.9(a)).

Fig. 4.10 shows the SINR γ_{best}^i for different SNR values[31]. Each curve is averaged over 300 fading realizations. In the simulation, a penalty parameter ρ is set to 0.5. Plots are drawn for the SINR values obtained at iterations 20, 30, and 50 of Algorithm 4.3. Results show that the proposed Algorithm 4.3 can achieve close to the centralized solution over the wide range of SNR values without any tuning of ρ.

[31]For fixed radius R_{BS} in Fig. 4.3, different SNRs (i.e., different SNR(R_{BS})) are obtained by changing $\frac{p_0^{\max}}{\sigma^2}$ in (4.55).

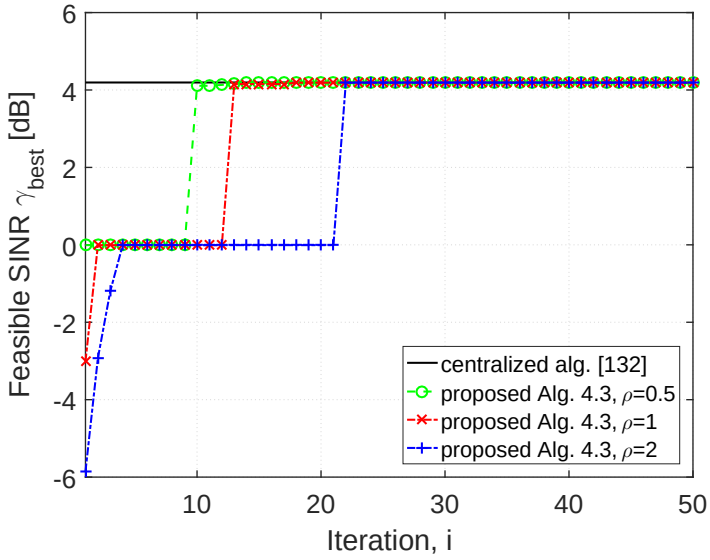


(a)

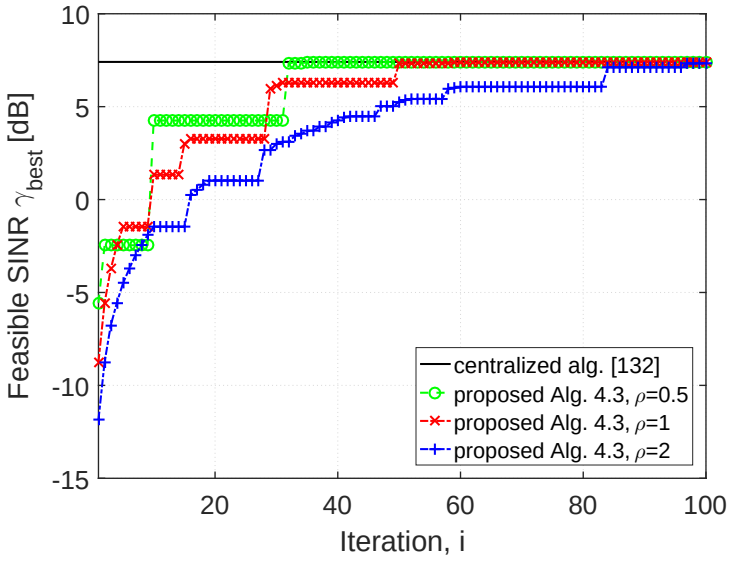


(b)

Fig. 4.8. Progress of global variable γ for SNR $= 5$ dB: (a) Multicell network 1; (b) Multicell network 2.

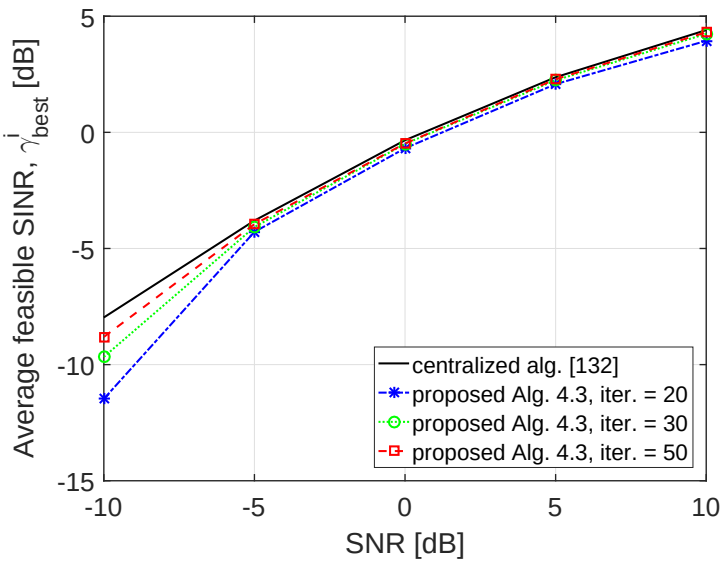


(a)

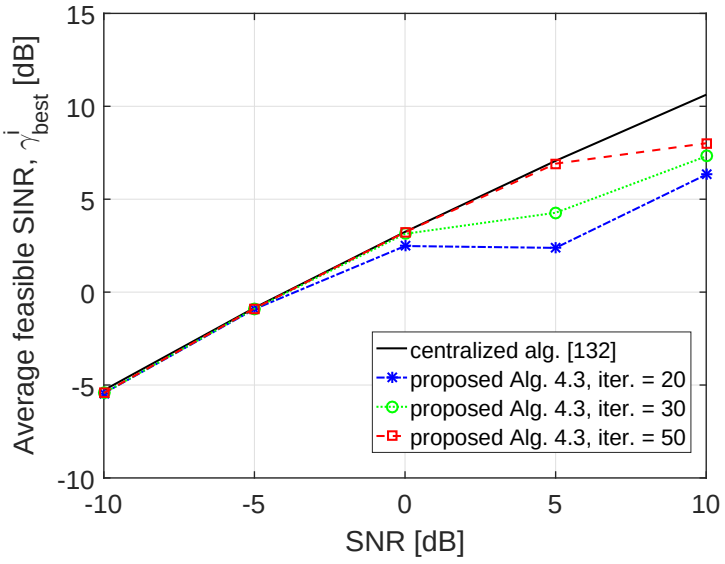


(b)

Fig. 4.9. Feasible SINR γ^i_{best} versus iteration for SNR $= 5$ dB: (a) Multicell network 1; (b) Multicell network 2.



(a)



(b)

Fig. 4.10. Feasible SINR γ^i_{best} versus iteration for SNR $= 5$ dB: (a) Multicell network 1; (b) Multicell network 2.

4.5 Summary and discussion

We have provided distributed algorithms for the radio resource allocation problems in multicell downlink MISO systems. Specifically, we have considered two optimization problems: P1 - minimization of the total transmission power subject to SINR constraints of each user, and P2 - SINR balancing subject to total transmit power constraint of BSs. We have proposed consensus-based distributed algorithms, and the fast solution method via ADMM. First, we have derived a distributed algorithm for problem P1. Then in conjunction with the bracketing method, the algorithm is extended for problem P2. The proposed distributed algorithm for problem P1 converges to an optimal centralized solution, and this has been also demonstrated by numerical example. However, problem P2 is not amendable to a convex formulation, and in this case the ADMM does not need to converge to an optimal point. Numerical results show that the proposed distributed algorithm for problem P2 finds a close-to-optimal solution. The consensus based ADMM technique proposed in this chapter can also be extended to derive a distributed algorithm for the WSRMax problem in multicell MISO downlink networks.

5 Dynamic resource allocation for cellular network operators

In this chapter the problem of dynamic spectrum sharing in a MISO downlink cellular network is considered. We assume a co-primary setup [25, 194], where operators share a fraction of their licensed spectrum by forming a common spectrum pool, which is dynamically allocated to the operators according to their channel qualities, traffic demands, past activity, etc.

Clearly, by sharing the spectrum instead of using it individually the operators can increase their profits, but this gives rise to a *new problem*: how to distribute the surplus (i.e., the increase in the profit) *fairly* between the operators, so that they have incentive to share their licensed spectrum with each other. For example, the total (or sum) profit can be maximized by allocating each portion of the shared spectrum to the operator that can make the most profit out of it. However, such a simplistic strategy does not make much sense since it may lead to a highly unfair outcome, where some operators may even suffer from reduced profit instead of increasing it.

This problem of fair surplus management has been intensively studied, especially in economics, and a solution has been proposed by Nash in 50's [195]. The bargaining problem proposed in [195] has shown that under a reasonable set of axioms, the fair operating point can be obtained by maximizing the product of the incremental profit gains, with respect to the case when the agents do not cooperate. We adopt this definition of fairness to share the licensed spectrum band between two cellular network operators in this chapter.

Our goal is to propose a dynamic network control algorithm which decides at each time slot: 1) the portion of spectrum band for each operator from the common spectrum pool (i.e., divide the common spectrum pool orthogonally between the operators), 2) beamformers, powers, transmission rates, admitted data, etc., for users, and 3) possible inter-operator payments, such that the surplus obtained from cooperation is fairly shared between the operators.

To achieve fairness in the surplus sharing between operators, we formulate (or cast) the spectrum sharing problem as a Nash bargaining game [151, 195–198], and then we use the Lyapunov drift plus penalty framework [42, 43] to derive a

dynamic control algorithm which approximates its solution. The solution (control algorithm) consists of solving the following (decoupled) tasks at each time slot: 1) determine the spectrum price for the operators, 2) make flow control decisions for the users' data, and 3) jointly allocate the common spectrum pool orthogonally to the operators and design the transmit beamformers and powers, which is known as resource allocation (RA). The RA problem is the most challenging one (combinatorial, NP-hard) and for this we propose both centralized and distributed algorithms, and hence we have both centralized and distributed versions of the dynamic control algorithm. The centralized algorithm is derived by solving the RA problem via sequential convex programming [35], and the distributed algorithm is derived by solving the RA problem via the ADMM [113] in conjunction with the sequential convex programming. Numerically, we show that the proposed distributed algorithm achieves almost the same performance as the centralized one. Furthermore, the results show that there is a trade-off between the achieved profits of the operators and network congestion.

5.1 System model and problem formulation

We consider a downlink wireless network consisting of a cell with two coexisting BSs, belonging to two different operators. The set of BSs is denoted by $\mathcal{N}$, and we label them with the integer values $n = 1, 2$. The transmission region of the BSs is modeled as a disc with a radius R_{BS} centered at the location of the BS. Each BS is equipped with T transmit antennas, and each user is equipped with single receive antenna. We denote the set of all users in nth BS by $\mathcal{L}(n)$, and we label them with the integer values $l = 1, \ldots, \check{L}_n$. Let each operator share an equal[32] amount of spectrum band B Hz with the other operator. Hence, a total spectrum of bandwidth $2B$ Hz is available for both operators. Furthermore, we assume that the total spectrum band $2B$ Hz is split into S subchannels. The set of subchannels is denoted by $\mathcal{S}$, and we label them with the integer values $s = 1, \ldots, S$. Let the bandwidth of sth subchannel be w_s Hz, and we assume that it is smaller than a coherence bandwidth. See Fig. 5.1 for the functional architecture of the considered co-primary spectrum sharing setup.

[32]The work can be easily generalized to the case where operators share different portions of the spectrum bands with each other.

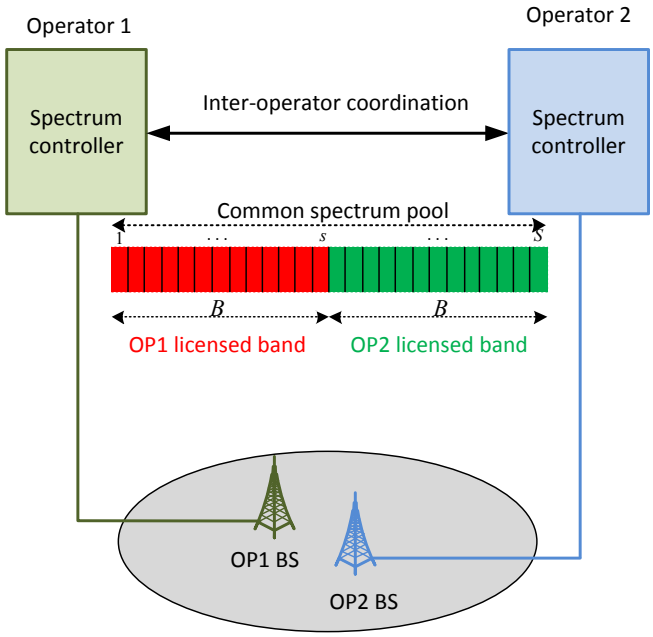


Fig. 5.1. High level functional architecture for co-primary spectrum sharing [25, 194, 199]: Two operators (OP) share a common spectrum pool in the same geographical area. The usage conditions of the common spectrum pool are defined with mutual agreement between the operators. The spectrum controller is responsible for inter-operator spectrum coordination. We refer the interested reader to [194] for more detail on co-primary sharing architecture for a denser network, [165] ©2017, IEEE.

The network is assumed to be operating in slotted time with slots normalized to integer values $t \in \{0, 1, 2, \ldots\}$. At each time slot, a network controller partitions the S subchannels orthogonally between the operators (i.e., between the two BSs)[33]. Let the set of subchannels allocated to nth BS during time slot t be $\mathcal{S}(n,t)$, and we label them with the integer values $s = 1, \ldots, S_n(t)$.

Let $p_{nl,s}(t)$ and $\mathbf{v}_{nl,s}(t)$ denote the power and direction of the transmit beamformer associated with lth user of BS n in subchannel s during time slot t. We assume $\mathbf{v}_{nl,s}(t)$ is normalized such that $\|\mathbf{v}_{nl,s}(t)\|_2 = 1$ for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$. Furthermore, let $\mathbf{m}_{nl,s}(t) = \sqrt{p_{nl,s}(t)}\mathbf{v}_{nl,s}(t)$, then the SINR of lth user of BS n in subchannel s during time slot t is given by

$$\Gamma_{nl,s}(\breve{\mathbf{m}}_n(t)) = \frac{|\mathbf{h}^{\mathrm{H}}_{nl,s}(t)\mathbf{m}_{nl,s}(t)|^2}{N_0 w_s + \sum_{j\in\mathcal{L}(n), j\neq l} |\mathbf{h}^{\mathrm{H}}_{nl,s}(t)\mathbf{m}_{nj,s}(t)|^2}, \tag{5.1}$$

[33] We use the terminologies BS and operator interchangeably.

where $\mathbf{h}_{nl,s}^{\mathrm{H}}(t) \in \mathbb{C}^{1\times T}$ is the channel vector from nth BS to its lth user in subchannel s, N_0 is a noise power spectral density, and the notation $\breve{\mathbf{m}}_n(t)$ denotes a vector obtained by stacking $\mathbf{m}_{nl,s}(t)$ for all $l \in \mathcal{L}(n)$ and $s \in \mathcal{S}(n,t)$ on top of each other, i.e.,

$$\breve{\mathbf{m}}_n(t) = [\mathbf{m}_{n1,1}(t)^{\mathrm{T}}, \ldots, \mathbf{m}_{n\breve{L}_n,S_n(t)}(t)^{\mathrm{T}}]^{\mathrm{T}}.$$

We consider the case where all receivers are using single-user detection (i.e., a receiver decodes its intended signal by treating all other interfering signals as noise), and assume that the achievable rate of lth user of nth BS during time slot t is given by [9, Ch. 5]

$$\begin{aligned} r_{nl}(t) &\triangleq r_{nl}\big(\mathcal{S}(n,t), \breve{\mathbf{m}}_n(t)\big) \\ &= \sum_{s\in\mathcal{S}(n,t)} w_s \log_2\left(1 + \frac{|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nl,s}(t)|^2}{N_0 w_s + \sum_{j\in\mathcal{L}(n),j\neq l} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}(t)|^2}\right), \end{aligned} \quad (5.2)$$

Furthermore, we assume that the power allocation is subject to a maximum power constraint $\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}(t)\|_2^2 \leq p_n^{\max}$ for each BS $n \in \mathcal{N}$.

5.1.1 Spectrum pricing

At each time slot, the common spectrum pool $2B$ Hz (i.e., the set of S subchannels) is partitioned between the operators. The total spectrum band allocated to operator $n \in \mathcal{N}$ during time slot t is $\sum_{s\in\mathcal{S}(n,t)} w_s$. We assume that both operators can use up to the amount of spectrum that they put into the common spectrum pool without any payment. But, the operator pays for an extra band of spectrum, if it uses more spectrum than it has put into the common spectrum pool, to the other operator. Specifically, if a spectrum band used by nth operator $\sum_{s\in\mathcal{S}(n,t)} w_s$ is more than B Hz, operator n pays the other operator (i.e., the opponent of nth operator) for the extra band of spectrum $(\sum_{s\in\mathcal{S}(n,t)} w_s - B)$ Hz. The amount to be paid is determined by the pricing rule established by a control algorithm.

Let $q_n(t)$ be the per-unit price of spectrum during time slot t to charge the nth operator's opponent for using the extra spectrum band. To simplify the notation, let us use $\underline{n}$ to denote the opponent of nth operator[34]. Then the

[34]For operator $n = 1$, its opponent is $\underline{n} = 2$. Similarly, for operator $n = 2$, its opponent is $\underline{n} = 1$.

payment from operator $n \in \mathcal{N}$, for using the extra band of spectrum, to its opponent is $q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+$.

With this spectrum pricing rule, operators with both low and high spectrum demands can be benefitted. For example, an operator with high spectrum demand can have access to more spectrum than it owns; while an operator with low spectrum demand (or, no spectrum demand at all) gets paid for leasing its spectrum. Even in the case when both operators use the same amount of spectrum that they have put into the common spectrum pool (i.e., in the case of equal spectrum demand), they can still gain as the operators have an opportunity to access subchannels with better channel quality from the common spectrum pool, free of charge.

5.1.2 Network queuing and time average profit

We consider a network utility maximization framework similar to the one considered in [42, Sec. 5.1], [43, Ch. 5]. Specifically, exogenously arriving data is not immediately admitted to the network layer of BSs. Instead, the exogenous data is first placed in transport layer storage reservoirs. Let $D_{nl}(t)$ represent the transport layer storage backlog of lth user of BS n during time slot t, and $\lambda_{nl}(t)$ represents the amount of data that exogenously arrives to it. Then, at each time slot a flow control decision is made, and the amount of each user data to be admitted to the network layer from reservoir $D_{nl}(t)$ is decided. Let $a_{nl}(t)$ denote the amount of data of lth user of nth BS that is admitted to the network layer from the reservoir $D_{nl}(t)$ during time slot t.

We assume that only the data currently available in $D_{nl}(t)$ at the beginning of slot t can be admitted to the network layer during that slot. The transport layer storage reservoir may not always have data to be admitted to the network layer, and hence the flow rate $a_{nl}(t)$ is subject to a constraint $a_{nl}(t) \leq D_{nl}(t)$ for each user. Then the dynamics of the transport layer storage reservoir $D_{nl}(t)$ from one time slot to the next can be expressed as[35]

$$D_{nl}(t+1) = \max[D_{nl}(t) - a_{nl}(t), 0] + \lambda_{nl}(t), \quad n \in \mathcal{N}, l \in \mathcal{L}(n). \tag{5.3}$$

[35] In the case that the transport layer storage reservoir has a finite size $D_{nl}^{\max} \geq 0$, expression (5.3) can be expressed as $D_{nl}(t+1) = \min\big[\max[D_{nl}(t) - a_{nl}(t), 0] + \lambda_{nl}(t), D_{nl}^{\max}\big]$ for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$.

Here, we assume that the exogenously arriving data $\{\lambda_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$ can have arbitrary input rates (i.e., input rates can be inside or outside of the network capacity region).

Furthermore, at the network layer each BS maintains a set of internal queues for storing the current backlog (or unfinished work) of its users. Let $Q_{nl}(t)$ represent the current backlog of lth user of nth BS. The evolution of the size of $Q_{nl}(t)$ can be expressed as [42]

$$Q_{nl}(t+1) = \max[Q_{nl}(t) - r_{nl}(t), 0] + a_{nl}(t), \quad n \in \mathcal{N}, l \in \mathcal{L}(n), \tag{5.4}$$

where $r_{nl}(t)$ is the transmission rate (defined in (5.2)) offered to lth user of nth BS during time slot t. Here, we adopt the notion of *strong stability*[36], and we say that the network is strongly stable if[37] [42, 43]

$$\overline{Q}_{nl} \triangleq \limsup_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathrm{E}\{Q_{nl}(\tau)\} < \infty, \quad n \in \mathcal{N}, l \in \mathcal{L}(n), \tag{5.5}$$

where the expectation depends on the control policy, and is with respect to the random channel states and the control actions made in reaction to these channel states[38]. Intuitively, expression (5.5) means that a queue is strongly stable if its time average backlog is finite; and a network is strongly stable if all individual queues in the network are strongly stable.

At each time slot, for lth user of BS n the network controller admits $a_{nl}(t)$ data into the internal queue for transmission. Note that under network stability, the admitted data $a_{nl}(t)$ for all t in the internal queue is transmitted to the corresponding user over a finite period of time [43]. Thus, we define the utility of the user in terms of the admitted data rate $a_{nl}(t)$, instead of the transmission rate $r_{nl}(t)$. To define the utility of lth user of BS n, let $\overline{a}_{nl}(t)$ denote the admitted time average rate up to time slot t, i.e., $\overline{a}_{nl}(t) \triangleq \frac{1}{t}\sum_{\tau=0}^{t-1} \mathrm{E}\{a_{nl}(\tau)\}$. Then, associated with each user, we define a non-decreasing concave utility function $g_{nl}(\overline{a}_{nl}(t))$. The utility function $g_{nl}(\overline{a}_{nl}(t))$ represents a monetary measure of the satisfaction that nth operator receives by sending data to its lth user based

[36] A definition of strong stability is general, and it also implies other forms of stability [43, Theorem 2.8].

[37] Note that we use a commonly used procedure, and express the long term average as the time average of expectation, which leads to a tractable algorithm [46, 200–205].

[38] Throughout this chapter, all expectations are taken with respect to the random channel states and the control actions made in reaction to these channel states, unless stated otherwise.

on its current data rate $\overline{a}_{nl}(t)$. Finally, we define the *time average expected* profit of nth operator as

$$\overline{U}_n \triangleq \liminf_{t\to\infty} \Big(\sum_{l\in\mathcal{L}(n)} g_{nl}(\overline{a}_{nl}(t)) + \tfrac{1}{t} \sum_{\tau=0}^{t-1} \mathrm{E}\big\{ q_n(\tau) \big(\sum_{s\in\mathcal{S}(\underline{n},\tau)} w_s - B \big)^+ \big\} - \tfrac{1}{t} \sum_{\tau=0}^{t-1} \mathrm{E}\big\{ q_{\underline{n}}(\tau) \big(\sum_{s\in\mathcal{S}(n,\tau)} w_s - B \big)^+ \big\} \Big), \quad (5.6)$$

where the second right hand term in (5.6) represents an amount that nth operator receives by leasing its spectrum band to its opponent; and the third right hand term represents an amount that nth operator pays to its opponent for renting the extra spectrum band. Note that during any given time slot, only one operator can use more than B Hz from the spectrum pool $2B$ Hz. Hence, during any given time slot, operators either lease or rent a portion of the common spectrum band. Specifically, during time slot τ, either term $q_n(\tau)\big(\sum_{s\in\mathcal{S}(\underline{n},\tau)} w_s - B\big)^+$ or term $q_{\underline{n}}(\tau)\big(\sum_{s\in\mathcal{S}(n,\tau)} w_s - B\big)^+$ is nonzero.

5.1.3 Problem formulation

Our objective is to maximize the gain in profits of both operators by sharing their licensed spectrum bands with each other, rather than using them exclusively. That is, we want to distribute the surplus (i.e., the increase in the profit) fairly between the operators, so that they have an incentive to share their licensed spectrum with each other. To do this, we model the spectrum sharing between two operators as a two-person bargaining problem [195–198] and cast it as a stochastic optimization problem.

Let U_n^0 denote the utility gain of nth operator that it receives before sharing its spectrum band with the other operator. In the context of bargaining problem, the utility U_n^0 is commonly known as a *disagreement point*, and it is assumed to be known. We assume that each operator knows a value of U_n^0 from their past experience. Thus the benefits of the operators obtained by sharing their licensed spectrum bands with each other is $\overline{U}_n - U_n^0$ for all $n \in \mathcal{N}$. For tractability, we assume that a per-unit price of the spectrum band set by each operator is bounded, i.e., $0 \le q_n(t) \le q^{\max}$ for all $n \in \mathcal{N}$. Then the optimization problem to maximize the operators' profit (i.e., $\overline{U}_n - U_n^0$ for all $n \in \mathcal{N}$) fairly, subject to network stability and the maximum power constraint for each BS can be

expressed as[39] [40]

$$
\begin{aligned}
\text{maximize} \quad & \textstyle\sum_{n\in\mathcal{N}} \log(\overline{U}_n - U_n^0) \\
\text{subject to} \quad & \overline{U}_n \geq U_n^0, \quad n \in \mathcal{N} && \text{(5.7a)} \\
& \overline{Q}_{nl} < \infty, \quad n \in \mathcal{N}, l \in \mathcal{L}(n) && \text{(5.7b)} \\
& 0 \leq q_n(t) \leq q^{\max}, \quad n \in \mathcal{N}, \forall t && \text{(5.7c)} \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}(t)\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \forall t && \text{(5.7d)} \\
& \mathcal{S}(1,t) \cap \mathcal{S}(2,t) = \emptyset, \quad \mathcal{S}(1,t), \mathcal{S}(2,t) \subseteq \mathcal{S}, \forall t, && \text{(5.7e)}
\end{aligned}
$$

with variables $\{q_n(t)\}_{n\in\mathcal{N}}$, $\{a_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{\mathbf{m}_{nl,s}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$, $\mathcal{S}(1,t)$, and $\mathcal{S}(2,t)$ for all $t \in \{0,1,2,\ldots\}$; where $\overline{U}_n$ and $\overline{Q}_{nl}$ are defined in (5.6) and (5.5), respectively. The constraint (5.7a) ensures that the profits of the operators obtained by sharing their spectrum bands are greater than without sharing their spectrum bands with each other. The constraint (5.7b) ensures that the network is stable. The constraint (5.7d) limits the total transmit power of each BS, and constraint (5.7e) ensures that a subchannel is allocated only to a single operator.

5.2 Dynamic algorithm via Lyapunov optimization

In this section we use the cross-layer stochastic optimization framework of [42, 43] to solve problem (5.7). We start by transforming problem (5.7) so that it conforms to the structure required for the drift-plus-penalty method of [42, 43]. Then, we apply the drift-plus-penalty minimization method to the transformed problem to derive the steps of the proposed dynamic control algorithms.

[39] Efficient utilization of the common spectrum pool can be obtained by maximizing the social welfare objective $\sum_{n\in\mathcal{N}} \overline{U}_n$, without regards to the spectrum prices $\{q_n\}_{n\in\mathcal{N}}$ because the payment will be canceled out. However, in sharing the spectrum between operators, such a strategy does not make much sense since it may lead to a highly unfair outcome, where some operators may even suffer from reduced profit instead of increasing it.

[40] In the formulation (5.7), we have used the notion of a bargaining problem [195–198], where the fair operating point is obtained by maximizing the product of the incremental profit gains, with respect to the disagreement point. This is easy to see, as we can equivalently express the objective function of problem (5.7) as $\log \prod_{n\in\mathcal{N}} (\overline{U}_n - U_n^0)$; and omit the $\log(\cdot)$ function.

5.2.1 Transformed problem via auxiliary variables

We start by equivalently reformulating problem (5.7) by introducing an auxiliary variable $\overline{\mu}_n$, for all $n \in \mathcal{N}$, as

$$\begin{aligned}
\text{maximize} \quad & \textstyle\sum_{n\in\mathcal{N}} \log(\overline{\mu}_n) \\
\text{subject to} \quad & \overline{\mu}_n \leq \overline{U}_n - U_n^0, \quad n \in \mathcal{N} \\
& \overline{U}_n \geq U_n^0, \quad n \in \mathcal{N} \\
& \overline{Q}_{nl} < \infty, \quad n \in \mathcal{N}, l \in \mathcal{L}(n) \\
& 0 \leq q_n(t) \leq q^{\max}, \quad n \in \mathcal{N}, \forall t \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}(t)\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \forall t \\
& \mathcal{S}(1,t) \cap \mathcal{S}(2,t) = \emptyset, \quad \mathcal{S}(1,t), \mathcal{S}(2,t) \subseteq \mathcal{S}, \forall t,
\end{aligned} \tag{5.8}$$

with variables $\{\overline{\mu}_n\}_{n\in\mathcal{N}}$, $\{q_n(t)\}_{n\in\mathcal{N}}$, $\{a_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{\mathbf{m}_{nl,s}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$, $\mathcal{S}(1,t)$, and $\mathcal{S}(2,t)$ for all $t \in \{0,1,2,\ldots\}$. Note that the first inequality constraints of problem (5.8) hold with equality at the optimal solution due to a monotonic increasing property of the objective function.

Note that problem (5.8) involves the statistics of random channel states, and that we do not know. Our objective is to provide a dynamic control algorithm that makes a decision in each time slot and solves problem (5.8). We achieve this by using the drift-plus-penalty minimization method [42, 43], which converts a long-term objective of problem (5.8) into a series of myopic optimizations. To do this, we now assume that the auxiliary variable $\overline{\mu}_n$ is a time average of auxilary variables $\mu_n(t)$ for all $t = \{0,1,2,\ldots\}$, i.e., $\overline{\mu}_n \triangleq \lim_{t\to\infty} \frac{1}{t} \sum_{\tau=0}^{t-1} \mathrm{E}\{\mu_n(\tau)\}$. Then by following the approach of [43, Ch. 5.0.5], we modify problem (5.8) as the following optimization problem:

$$\begin{aligned}
\text{maximize} \quad & \textstyle\sum_{n\in\mathcal{N}} \overline{\log(\mu_n)} \\
\text{subject to} \quad & \overline{\mu}_n \leq \overline{U}_n - U_n^0, \quad n \in \mathcal{N} && \text{(5.9a)} \\
& \overline{U}_n \geq U_n^0, \quad n \in \mathcal{N} && \text{(5.9b)} \\
& \overline{Q}_{nl} < \infty, \quad n \in \mathcal{N}, l \in \mathcal{L}(n) && \text{(5.9c)} \\
& 0 \leq q_n(t) \leq q^{\max}, \quad n \in \mathcal{N}, \forall t && \text{(5.9d)} \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}(t)\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N}, \forall t && \text{(5.9e)} \\
& \mathcal{S}(1,t) \cap \mathcal{S}(2,t) = \emptyset, \quad \mathcal{S}(1,t), \mathcal{S}(2,t) \subseteq \mathcal{S}, \forall t, && \text{(5.9f)}
\end{aligned}$$

with variables $\{\mu_n(t)\}_{n\in\mathcal{N}}$, $\{q_n(t)\}_{n\in\mathcal{N}}$ $\{a_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\mathcal{S}(1,t)$, $\mathcal{S}(2,t)$, and $\{\mathbf{m}_{nl,s}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ for all $t\in\{0,1,2,\ldots\}$; where $\overline{\log(\mu_n)}$ is defined as

$$\overline{\log(\mu_n)} \triangleq \lim_{t\to\infty} \tfrac{1}{t}\sum_{\tau=0}^{t-1} \mathrm{E}\big\{\log(\mu_n(\tau))\big\}. \tag{5.10}$$

Note that by using Jensen's inequality we can easily verify that $\overline{\log(\mu_n)}$ is lower bound on $\log(\overline{\mu}_n)$. Thus, the solution of problem (5.9) is also feasible for the original problem (5.7), and hence problem (5.9) provides a reasonable lower bound for the original problem (5.7).

5.2.2 Solving the transformed problem

In this section we use the drift-plus-penalty minimization method introduced in [42, 43] to solve problem (5.9). In the drift-plus-penalty minimization method, the inequality constraints (5.9a) and (5.9b) are enforced by transforming them into a queue stability problem. In other words, for each inequality constraint, in (5.9a) and (5.9b), a virtual queue is introduced in such a way that the stability of these virtual queues implies the feasibility of constraints (5.9a) and (5.9b).

Let $\{X_n(t)\}_{n\in\mathcal{N}}$ be virtual queues associated with constraint (5.9a). We update the virtual queue $X_n(t)$ for all $n\in\mathcal{N}$ at each time slot as

$$X_n(t+1) = \max[X_n(t) - x_n^{\mathrm{out}}(t), 0] + x_n^{\mathrm{in}}(t), \tag{5.11}$$

where,

$$x_n^{\mathrm{out}}(t) = \textstyle\sum_{l\in\mathcal{L}(n)} g_{nl}(a_{nl}(t)) + q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n},t)} w_s - B\big)^+, \tag{5.12}$$

$$x_n^{\mathrm{in}}(t) = \mu_n(t) + U_n^0 + q_{\underline{n}}(t)\big(\textstyle\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+. \tag{5.13}$$

Note that $X_n(t)$ can be viewed as a backlog in a virtual queue with an input rate $x_n^{\mathrm{in}}(t)$ and a service rate $x_n^{\mathrm{out}}(t)$. If virtual queues $\{X_n(t)\}_{n\in\mathcal{N}}$ are strongly stable (see expression (5.5) for the definition of strong stability), then constraint (5.9a) is satisfied, see Appendix 4.

Likewise, to ensure inequality constraint (5.9b), we define virtual queues $\{Y_n(t)\}_{n\in\mathcal{N}}$; and update $Y_n(t)$ for all $n\in\mathcal{N}$ according to the following dynamics:

$$Y_n(t+1) = \max[Y_n(t) - y_n^{\mathrm{out}}(t), 0] + y_n^{\mathrm{in}}(t), \tag{5.14}$$

where,

$$y_n^{\text{out}}(t) = \sum_{l\in\mathcal{L}(n)} g_{nl}(a_{nl}(t)) + q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n},t)} w_s - B\big)^+, \tag{5.15}$$

$$y_n^{\text{in}}(t) = U_n^0 + q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+. \tag{5.16}$$

The stability of virtual queues $\{Y_n(t)\}_{n\in\mathcal{N}}$ ensures constraint (5.9b), and this can be shown by following the approach in Appendix 4.

We now define Lyapunov function and its drift, which will be used to define a queue stability problem for the actual queues $\{Q_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$ and the virtual queues $\{X_n(t), Y_n(t)\}_{n\in\mathcal{N}}$. For a compact representation, let us use $\mathbf{\Theta}(t)$ to denote a vector of the actual and the virtual queues, i.e., $\mathbf{\Theta}(t) = [Q_{11}(t), \ldots, Q_{1L_1}(t), Q_{21}(t), \ldots, Q_{2L_2}(t), X_1(t), X_2(t), Y_1(t), Y_2(t)]^{\mathrm{T}}$. Then we define a quadratic Lyapunov function $\tilde{L}(\mathbf{\Theta}(t))$ as [42, 43]

$$\tilde{L}\mathbf{\Theta}(t)) = \frac{1}{2}\Big[\sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)^2 + \sum_{n\in\mathcal{N}} X_n(t)^2 + \sum_{n\in\mathcal{N}} Y_n(t)^2\Big]. \tag{5.17}$$

The Lyapunov function $\tilde{L}(\mathbf{\Theta}(t))$ is a scalar measure of network congestion. Intuitively, if $\tilde{L}(\mathbf{\Theta}(t))$ is small then all the queues are small; and if $\tilde{L}(\mathbf{\Theta}(t))$ is large then at least one queue is large. Thus, by minimizing a drift in the Lyapunov function (i.e., by minimizing a difference in the Lyapunov function from one slot to the next) queues $\{Q_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$ and $\{X_n(t), Y_n(t)\}_{n\in\mathcal{N}}$ can be stabilized [43]. By using expression (5.17), the drift in the Lyapunov function (i.e., the expected change in the Lyapunov function from one slot to the next) can be written as

$$\Delta(\mathbf{\Theta}(t)) \triangleq \mathrm{E}\big\{\tilde{L}(\mathbf{\Theta}(t+1)) - \tilde{L}(\mathbf{\Theta}(t))|\mathbf{\Theta}(t)\big\}. \tag{5.18}$$

We now use the drift-plus-penalty minimization method introduced in [42, 43] to solve problem (5.9). In this method, a control policy that solves problem (5.9) is obtained by minimizing an upper bound on the following drift-plus-penalty expression [42, 43]:

$$\Delta(\mathbf{\Theta}(t)) - V\sum_{n\in\mathcal{N}} \mathrm{E}\{\log(\mu_n(t))|\mathbf{\Theta}(t)\}, \tag{5.19}$$

where $V \geq 0$, subject to the constraints (5.7c)-(5.7e) in each time slot, i.e.,

$$0 \leq q_n(t) \leq q^{\max}, \quad n \in \mathcal{N} \tag{5.20}$$

$$\sum_{l\in\mathcal{L}(n)}\sum_{s\in\mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}(t)\|_2^2 \leq p_n^{\max}, \quad n \in \mathcal{N} \tag{5.21}$$

$$\mathcal{S}(1,t) \cap \mathcal{S}(2,t) = \emptyset, \quad \mathcal{S}(1,t), \mathcal{S}(2,t) \subseteq \mathcal{S}. \tag{5.22}$$

Note that the expression (5.19) has two terms. The first term is the drift $\Delta(\mathbf{\Theta}(t))$; and in [42, 43] it is shown that by minimizing the drift $\Delta(\mathbf{\Theta}(t))$, at each time slot, we can satisfy inequality constrains (5.9a)-(5.9c). The second term is $-\sum_{n\in\mathcal{N}} \mathrm{E}\{\log(\mu_n(t))|\mathbf{\Theta}(t)\}$, and by minimizing it at each time slot, the objective of problem (5.9) is maximized. Thus, by varying parameter V we can obtain a desired trade-off between the size of the queue backlogs and the profits of the operators.

In the rest of this section, to simplify algorithm development, we first find an upper bound of the expression (5.19). Then we present a dynamic control algorithm to solve problem (5.9) that, at each time slot, minimizes the upper bound of expression (5.19) subject to the constraints (5.20)-(5.22).

By using expressions (5.4), (5.11), and (5.14), we note that [41]

$$Q_{nl}(t+1)^2 \leq Q_{nl}(t)^2 + a_{nl}(t)^2 + r_{nl}(t)^2 + 2Q_{nl}(t)[a_{nl}(t) - r_{nl}(t)], \quad (5.23)$$

$$X_n(t+1)^2 \leq X_n(t)^2 + x_n^{\mathrm{in}}(t)^2 + x_n^{\mathrm{out}}(t)^2 + 2X_n(t)[x_n^{\mathrm{in}}(t) - x_n^{\mathrm{out}}(t)], \quad (5.24)$$

$$Y_n(t+1)^2 \leq Y_n(t)^2 + y_n^{\mathrm{in}}(t)^2 + y_n^{\mathrm{out}}(t)^2 + 2Y_n(t)[y_n^{\mathrm{in}}(t) - y_n^{\mathrm{out}}(t)]. \quad (5.25)$$

Then, by using expressions (5.18) and inequalities (5.23)-(5.25), an upper bound of expression (5.19) can be expressed as

$$\begin{aligned}\Delta(\mathbf{\Theta}(t)) - V \sum_{n\in\mathcal{N}} \mathrm{E}\{\log(\mu_n(t))|\mathbf{\Theta}(t)\} \;\leq\; & C - V \sum_{n\in\mathcal{N}} \mathrm{E}\{\log(\mu_n(t))|\mathbf{\Theta}(t)\} \\ & + \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\mathrm{E}\{a_{nl}(t) - r_{nl}(t)|\mathbf{\Theta}(t)\} \\ & + \sum_{n\in\mathcal{N}} X_n(t)\mathrm{E}\{x_n^{\mathrm{in}}(t) - x_n^{\mathrm{out}}(t)|\mathbf{\Theta}(t)\} \\ & + \sum_{n\in\mathcal{N}} Y_n(t)\mathrm{E}\{y_n^{\mathrm{in}}(t) - y_n^{\mathrm{out}}(t)|\mathbf{\Theta}(t)\}, \quad (5.26)\end{aligned}$$

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

$$\begin{aligned}C \geq \frac{1}{2}\Big[& \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} \mathrm{E}\{a_{nl}(t)^2 + r_{nl}(t)^2|\mathbf{\Theta}(t)\} + \\ & + \sum_{n\in\mathcal{N}} \mathrm{E}\{x_n^{\mathrm{in}}(t)^2 + x_n^{\mathrm{out}}(t)^2 + y_n^{\mathrm{in}}(t)^2 + y_n^{\mathrm{out}}(t)^2|\mathbf{\Theta}(t)\}\Big]. \quad (5.27)\end{aligned}$$

[41] To write inequalities (5.23)-(5.25), we have used the fact that $(\max[Q-b,0]+A)^2 \leq Q^2 + A^2 + b^2 + 2Q(A-b)$ for any $Q \geq 0$, $b \geq 0$, and $A \geq 0$.

Furthermore, by substituting expressions (5.12), (5.13), (5.15), and (5.16) in (5.26), we get

$$\begin{aligned}\Delta(\boldsymbol{\Theta}(t)) - V \sum_{n\in\mathcal{N}} \mathrm{E}\big\{ \log(\mu_n(t))|\boldsymbol{\Theta}(t)\big\} &\leq C - V \sum_{n\in\mathcal{N}} \mathrm{E}\{\log(\mu_n(t))|\boldsymbol{\Theta}(t)\big\} \\ &+ \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\mathrm{E}\big\{a_{nl}(t) - r_{nl}(t)|\boldsymbol{\Theta}(t)\big\} + \\ &\sum_{n\in\mathcal{N}} X_n(t)\mathrm{E}\big\{\mu_n(t)\big|\boldsymbol{\Theta}(t)\big\} \\ &+ \sum_{n\in\mathcal{N}} W_n(t)\mathrm{E}\big\{U_n^0 + q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+\big|\boldsymbol{\Theta}(t)\big\} \\ &- \sum_{n\in\mathcal{N}} W_n(t)\mathrm{E}\big\{\sum_{l\in\mathcal{L}(n)} g_{nl}(a_{nl}(t)) + q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n},t)} w_s - B\big)^+\big|\boldsymbol{\Theta}(t)\big\}, \quad (5.28)\end{aligned}$$

where $W_n(t) = X_n(t) + Y_n(t)$ for all $n \in \mathcal{N}$.

Finally, we summarize the steps of the proposed dynamic control algorithms based on the drift-plus-penalty minimization method [42, 43] to solve problem (5.9) in Algorithm 5.1. The proposed algorithms observe queue backlogs $\boldsymbol{\Theta}(t)$ and the channel states $\{\mathbf{h}_{nl,s}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$, and makes a control action to minimize the righthand side of expression (5.28) subject to constraints (5.20)-(5.22). The minimization of the righthand side of expression (5.28) can be decoupled across variables $\{q_n(t)\}_{n\in\mathcal{N}}$, $\{a_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{\mu_n(t)\}_{n\in\mathcal{N}}$, and $\{\{\mathbf{m}_{nl,s}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}(n,t)}, \{\mathcal{S}(n,t)\}_{n\in\mathcal{N}}\}$, resulting in subproblems as shown in Algorithm 5.1. Note that the drift-plus-penalty minimization method [42, 43] uses the concept of *opportunistically minimizing an expectation* [43, Ch. 1.8] to solve each subproblems.

Algorithm 5.1. *Algorithm for the spectrum sharing problem (5.9)*

1. Pricing: for each $n \in \mathcal{N}$, per-unit price $q_n(t)$ is chosen as

$$q_n(t) = \begin{cases} q^{\max} & \text{if } W_n(t) > W_{\underline{n}}(t) \\ 0 & \text{otherwise} \end{cases} \quad (5.29)$$

2. Flow control: for each $n \in \mathcal{N}$, flow rate $a_{nl}(t) = a_{nl}$ for all $l \in \mathcal{L}(n)$, where $\{a_{nl}\}_{l\in\mathcal{L}(n)}$ solves the following optimization problem:

$$\begin{array}{ll} \text{maximize} & W_n(t)\sum_{l\in\mathcal{L}(n)} g_{nl}(a_{nl}) - \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)a_{nl} \\ \text{subject to} & 0 \leq a_{nl} \leq \min[D_{nl}(t), A^{\max}], \quad l \in \mathcal{L}(n), \end{array} \quad (5.30)$$

with variables $\{a_{nl}\}_{l\in\mathcal{L}(n)}$, where $A^{\max} > 0$ is the algorithm parameter as described in [42, Sec. 4.2.1]. In the inequality constraint of problem (5.30), the term $D_{nl}(t)$ denotes the available data in the transport layer reservoir (see Section 5.1.2).

3. Auxiliary variable: for each $n \in \mathcal{N}$, auxiliary variable $\mu_n(t) = \mu_n$, where μ_n solves the following optimization problem:

$$\begin{array}{ll} \text{maximize} & V\log(\mu_n) - X_n(t)\mu_n \\ \text{subject to} & 0 \le \mu_n \le \mu^{\max}, \end{array} \tag{5.31}$$

with variables μ_n, where $\mu^{\max} > 0$ is the algorithm parameter as described in [43, Ch. 5] .

4. Resource allocation: solve the following optimization problem:

$$\begin{array}{ll} \text{maximize} & \sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{L}(n)} Q_{nl}(t) r_{nl}\big(\mathcal{S}_n, \mathbf{m}_n\big) \\ & + \sum_{n\in\mathcal{N}} W_n(t) q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n})} w_s - B\big)^+ \\ & - \sum_{n\in\mathcal{N}} W_n(t) q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n)} w_s - B\big)^+ \\ \text{subject to} & \sum_{l\in\mathcal{L}(n)}\sum_{s\in\mathcal{S}(n)} \|\mathbf{m}_{nl,s}\|_2^2 \le p_n^{\max}, \quad n\in\mathcal{N} \\ & \mathcal{S}(1)\cap\mathcal{S}(2) = \emptyset, \quad \mathcal{S}(1),\mathcal{S}(2)\subseteq\mathcal{S}, \end{array} \tag{5.32}$$

with variables $\{\mathbf{m}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ and $\{\mathcal{S}(n)\}_{n\in\mathcal{N}}$. Set $\mathbf{m}_{nl,s}(t) = \mathbf{m}_{nl,s}$ and $\mathcal{S}(n,t) = \mathcal{S}(n)$ for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}(n)$.

5. Queue update: update $\{D_{nl}(t+1)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{Q_{nl}(t+1)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{X_n(t+1)\}_{n\in\mathcal{N}}$, and $\{Y_n(t+1)\}_{n\in\mathcal{N}}$ by using expressions (5.3), (5.4), (5.11), and (5.14). Set $t = t+1$ and go to step 1.

In step 1, the per-unit price of the spectrum is set for the operators, and is obtained by minimizing the righthand side of expression (5.28) over variables $\{q_n(t)\}_{n\in\mathcal{N}}$. A similar pricing strategy is obtained in [206], and this is known as bang-bang pricing. This pricing strategy alternates between periods of free service (i.e., a price set to zero) and periods where price is set to a pre-specified maximum value $q^{\max}$, according to the values of the virtual queues $\{W_n(t) = X_n(t)+Y_n(t)\}_{n\in\mathcal{N}}$. The value of virtual queue $X_n(t)$ can be interpreted as the utility gain that nth operator has yet to obtain in order to maximize its profit[42]. Thus, the bang-bang pricing strategy, in step 1, sets the price value to

[42]Similarly, the value of virtual queue $Y_n(t)$ can be interpreted as the utility gain that nth operator has yet to obtain so that its utility is greater than its disagreement point U_n^0.

a pre-specified value $q^{\max}$ [43] for an operator, whose utility is lagging behind than that of the other operator.

Observe that except step 4 of Algorithm 5.1, the problems in each step of the algorithm are decoupled into two subproblems, one for each operator. To solve step 1, the coordination between operators is required to exchange parameters $W_1(t)$ and $W_2(t)$ (i.e., two real scalers are needed to be exchanged between the operators). The signalling overhead required to solve step 4 depends on the, specific, strategy that is adopted to solve problem (5.32). In Section 5.3 and Section 5.4, we provide centralized and distributed algorithms to solve step 4 of Algorithm 5.1, respectively. This leads to the centralized and the distributed versions of Algorithm 5.1 (i.e., the centralized and the distributed dynamic control algorithms)[44].

The performance of Algorithm 5.1 can be evaluated by using Theorem 5.4 in [42]. By using Theorem 5.4 in [42], we can show that Algorithm 5.1 yields the objective value of problem (5.7) and the network backlog with a trade-off $[O(1/V), O(V)]$. That is the objective value of problem (5.7) is pushed within $O(1/V)$ of its maximum value, with an increase in the network backlog with V.

5.3 Resource allocation - centralized algorithm

In this section we focus on resource allocation problem (5.32). Problem (5.32) is a combinatorial problem, and it requires exponential complexity to find the global solution. Here, we derive a computationally efficient, but possibly suboptimal, algorithm for problem (5.32). The proposed algorithm is based on the sequential convex programming [35].

We start by introducing binary variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ in problem (5.32). Variable b_{ns} is set to one, if subchannel w_s is assigned to nth operator, otherwise it is set to zero. Hence, by using binary variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$, problem (5.32)

[43] In the bargaining framework, it is assumed that both operators have full knowledge of each other's preferences [195]. Thus, we assume that a value of $q^{\max}$ is set by the operators on mutual agreement. For example, a value of $q^{\max}$ can be related to the utilities that the operators can get by using their spectrum bands, i.e., we can set $q^{\max} = c \times g_{nl}(A^{\max})$, for some constant $c > 0$, (see step 2 of Algorithm 5.1 for a definition of $A^{\max}$).

[44] In the case when an operator deploys a heterogeneous system (for example, see [207]), the proposed Algorithm 5.1 can be extended by embedding the required inter-tier interference coordination strategy in the resource allocation subproblem (5.32).

can be equivalently written as

$$\begin{aligned}
\text{maximize} \quad & \sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\sum_{s\in\mathcal{S}} b_{ns}w_s\log_2\Bigg(1 \\
& + \frac{|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nl,s}|^2}{b_{ns}N_0w_s + \sum_{j\in\mathcal{L}(n),j\neq l}|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2}\Bigg) \\
& + \sum_{n\in\mathcal{N}} W_n(t)q_n(t)\big(\textstyle\sum_{s\in\mathcal{S}} b_{\underline{n}s}w_s - B\big)^+ \\
& - \sum_{n\in\mathcal{N}} W_n(t)q_{\underline{n}}(t)\big(\textstyle\sum_{s\in\mathcal{S}} b_{ns}w_s - B\big)^+ \\
\text{subject to} \quad & \textstyle\sum_{l\in\mathcal{L}(n)}\sum_{s\in\mathcal{S}}\|\mathbf{m}_{nl,s}\|_2^2 \leq p_n^{\max}, \quad n\in\mathcal{N} \qquad (5.33a)\\
& \textstyle\sum_{n\in\mathcal{N}} b_{ns} = 1, \quad s\in\mathcal{S} \qquad (5.33b)\\
& b_{ns} = \{0,1\}, \quad n\in\mathcal{N}, s\in\mathcal{S}, \qquad (5.33c)
\end{aligned}$$

with the optimization variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ and $\{\mathbf{m}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$; where $W_n(t) = X_n(t) + Y_n(t)$ for all $n\in\mathcal{N}$. Note that we have used expression (5.2) to write the objective function of problem (5.33), and variable w_s is replaced with $b_{ns}w_s$. In problem (5.33) constraints (5.33b) and (5.33c) ensures that a subchannel w_s is allocated to a single operator. Hence, the constraints associated with orthogonal subchannel allocation in problem (5.32) have been dropped out.

Now we relax hard binary constraint (5.33c), and employ a penalty function to promote binary values for variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$, leading to

$$\begin{aligned}
\text{maximize} \quad & \sum_{n\in\mathcal{N}}\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\sum_{s\in\mathcal{S}} b_{ns}w_s\log_2\Bigg(1 \\
& + \frac{|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nl,s}|^2}{b_{ns}N_0w_s + \sum_{j\in\mathcal{L}(n),j\neq l}|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2}\Bigg) \\
& + \sum_{n\in\mathcal{N}} W_n(t)q_n(t)\big(\textstyle\sum_{s\in\mathcal{S}} b_{\underline{n}s}w_s - B\big)^+ \\
& - \sum_{n\in\mathcal{N}} W_n(t)q_{\underline{n}}(t)\big(\textstyle\sum_{s\in\mathcal{S}} b_{ns}w_s - B\big)^+ + \delta\sum_{n\in\mathcal{N}}\sum_{s\in\mathcal{S}} b_{ns}\log(b_{ns}) \\
\text{subject to} \quad & \textstyle\sum_{l\in\mathcal{L}(n)}\sum_{s\in\mathcal{S}}\|\mathbf{m}_{nl,s}\|_2^2 \leq p_n^{\max}, \quad n\in\mathcal{N} \qquad (5.34a)\\
& \textstyle\sum_{n\in\mathcal{N}} b_{ns} = 1, \quad s\in\mathcal{S} \qquad (5.34b)\\
& 0 \leq b_{ns} \leq 1, \quad n\in\mathcal{N}, s\in\mathcal{S}, \qquad (5.34c)
\end{aligned}$$

with variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ and $\{\mathbf{m}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$; where $\delta > 0$ is a problem parameter. The penalty function $b_{ns}\log(b_{ns})$ is the negative entropy function,

and it has a maximum value at b_{ns} equal to zero or one[45]. Thus, there exists a value for parameter δ that can achieve binary values for variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$. It is worth noting that problem (5.34) is a non-combinatorial optimization problem, however, it is still a nonconvex problem. In fact, problem (5.34) is NP-hard [30].

Since the RA problem (5.34) is NP-hard, we use sequential convex programming to approximate its solution. In order to simplify the algorithm development, let us introduce variables $u_{nl,s}$ and $z_{nl,s}$ for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$ as

$$\begin{aligned} u_{nl,s} &= \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2 \\ z_{nl,s} &= \sum_{j\in\mathcal{L}(n),j\neq l} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2. \end{aligned} \tag{5.35}$$

Furthermore, for the sake of brevity, let us define following functions:

$$\psi_{nl}(\mathbf{u}_{nl},\mathbf{b}_n) = -\sum_{s\in\mathcal{S}} b_{ns}w_s \log_2\Big(N_0 w_s + \frac{u_{nl,s}}{b_{ns}}\Big), \tag{5.36}$$

$$\phi_{nl}(\mathbf{z}_{nl},\mathbf{b}_n) = -\sum_{s\in\mathcal{S}} b_{ns}w_s \log_2\Big(N_0 w_s + \frac{z_{nl,s}}{b_{ns}}\Big), \tag{5.37}$$

$$\chi_1(\mathbf{b}_1) = \big(W_1(t) - W_2(t)\big)q_2(t)\Big(\sum_{s\in\mathcal{S}} b_{1s}w_s - B\Big)^+, \tag{5.38}$$

$$\chi_2(\mathbf{b}_2) = \big(W_2(t) - W_1(t)\big)q_1(t)\Big(\sum_{s\in\mathcal{S}} b_{2s}w_s - B\Big)^+, \tag{5.39}$$

$$\zeta_n(\mathbf{b}_n) = \textstyle\sum_{s\in\mathcal{S}} b_{ns}\log(b_{ns}), \tag{5.40}$$

where $\mathbf{u}_{nl} = [u_{nl,1},\ldots,u_{nl,S}]^{\mathrm{T}}$, $\mathbf{z}_{nl} = [z_{nl,1},\ldots,z_{nl,S}]^{\mathrm{T}}$, and $\mathbf{b}_n = [b_{n1},\ldots,b_{nS}]^{\mathrm{T}}$. Then, by using expressions (5.35)-(5.40), and changing the sign of the objective function of problem (5.34), it can be equivalently expressed as the following

[45]Instead, a penalty function $b_{ns}(b_{ns}-1)$, for all $n \in \mathcal{N}$ and $s \in \mathcal{S}$, can also be used to promote binary value for variables $\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$.

minimization problem:

$$\begin{aligned}
\text{minimize} \quad & \sum_{n\in\mathcal{N}} \Big(\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\big(\psi_{nl}(\mathbf{u}_{nl},\mathbf{b}_n) - \phi_{nl}(\mathbf{z}_{nl},\mathbf{b}_n)\big) - \delta\zeta_n(\mathbf{b}_n) + \theta_n \Big) \\
\text{subject to} \quad & u_{nl,s} = \sum_{j\in\mathcal{L}(n)} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2, \quad n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S} \qquad (5.41a) \\
& z_{nl,s} = \sum_{j\in\mathcal{L}(n), j\neq l} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2, \quad n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S} \qquad (5.41b) \\
& \chi_n(\mathbf{b}_n) \le \theta_n, \quad n\in\mathcal{N} \qquad (5.41c) \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}} \|\mathbf{m}_{nl,s}\|_2^2 \le p_n^{\max}, \quad n\in\mathcal{N} \qquad (5.41d) \\
& \textstyle\sum_{n\in\mathcal{N}} b_{ns} = 1, \quad s\in\mathcal{S} \qquad (5.41e) \\
& 0 \le b_{ns} \le 1, \quad n\in\mathcal{N}, s\in\mathcal{S}, \qquad (5.41f)
\end{aligned}$$

with variables $\{\mathbf{u}_{nl},\mathbf{z}_{nl}\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n,\theta_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{m}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$. Note that we have used the relation $\log(A/B) = \log(A) - \log(B)$ to express the objective function in problem (5.41). In the sequel, we first approximate problem (5.41) and express it as a DC programming problem [36]. We then present an algorithm that finds a solution for DC problem (i.e., the solution for problem (5.41)) by solving a sequence of approximated convex problems.

Note that functions $\psi_{nl}(\mathbf{u}_{nl},\mathbf{b}_n)$, $\phi_{nl}(\mathbf{z}_{nl},\mathbf{b}_n)$, and $\zeta_n(\mathbf{b}_n)$ are convex [29, Sec. 3.1.5 and 3.2.6]. Thus, the objective function of problem (5.41) can be expressed as the difference of the following two convex functions:

$$\begin{aligned}
\breve{f}_0 &\triangleq \breve{f}_0(\{\mathbf{u}_{nl},\mathbf{b}_n,\theta_n\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}) \\
&= \textstyle\sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\psi_{nl}(\mathbf{u}_{nl},\mathbf{b}_n) + \theta_n, \qquad (5.42)
\end{aligned}$$

$$\begin{aligned}
\breve{g}_0 &\triangleq \breve{g}_0(\{\mathbf{z}_{nl},\mathbf{b}_n\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}) \\
&= \textstyle\sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\phi_{nl}(\mathbf{z}_{nl},\mathbf{b}_n) + \delta\zeta_n(\mathbf{b}_n), \qquad (5.43)
\end{aligned}$$

i.e., the objective function of problem (5.41) becomes $\breve{f}_0 - \breve{g}_0$.

We now turn to the inequality constraint (5.41c). We can easily see that functions $\chi_1(\mathbf{b}_1)$ and $\chi_2(\mathbf{b}_2)$ are convex functions if $W_1(t) \ge W_2(t)$ and $W_2(t) \ge W_1(t)$, respectively. Otherwise, both functions $\chi_1(\mathbf{b}_1)$ and $\chi_2(\mathbf{b}_2)$ are concave functions. Thus, we introduce the following approximations for functions $\chi_1(\mathbf{b}_1)$

and $\chi_2(\mathbf{b}_2)$:

$$\hat{\chi}_1(\mathbf{b}_1) = \begin{cases} \big(W_1(t) - W_2(t)\big) q_2(t) \Big(\sum_{s\in\mathcal{S}} b_{1s} w_s - B \Big)^+ & W_1(t) \geq W_2(t) \\ \big(W_1(t) - W_2(t)\big) q_2(t) \Big(\sum_{s\in\mathcal{S}} b_{1s} w_s - B \Big) & \text{otherwise,} \end{cases} \tag{5.44}$$

$$\hat{\chi}_2(\mathbf{b}_2) = \begin{cases} \big(W_2(t) - W_1(t)\big) q_1(t) \Big(\sum_{s\in\mathcal{S}} b_{2s} w_s - B \Big)^+ & W_2(t) \geq W_1(t) \\ \big(W_2(t) - W_1(t)\big) q_1(t) \Big(\sum_{s\in\mathcal{S}} b_{2s} w_s - B \Big) & \text{otherwise.} \end{cases} \tag{5.45}$$

Note that in expressions (5.44) and (5.45), we have used the upper bound functions of $\chi_1(\mathbf{b}_1)$ and $\chi_2(\mathbf{b}_2)$ if $W_1(t) < W_2(t)$ and $W_2(t) < W_1(t)$, respectively.

To approximate (5.41a) and (5.41b) with convex constraints, let us introduce the new variable $\mathbf{M}_{nl,s} = \mathbf{m}_{nl,s}\mathbf{m}_{nl,s}^{\mathrm{H}}$ such that $\mathrm{Rank}(\mathbf{M}_{nl,s}) = 1$ for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$. Then by applying a standard SDR [182] and using expressions (5.42)-(5.45), problem (5.41) can be approximated as the following DC program:

$$\begin{aligned}
\text{minimize} \quad & \breve{f}_0(\{\mathbf{u}_{nl}, \mathbf{b}_n, \theta_n\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}) - \breve{g}_0(\{\mathbf{z}_{nl}, \mathbf{b}_n\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}) \\
\text{subject to} \quad & u_{nl,s} = \sum_{j\in\mathcal{L}(n)} \mathbf{h}_{nl,s}^{\mathrm{H}}(t) \mathbf{M}_{nj,s} \mathbf{h}_{nl,s}(t), \quad n \in \mathcal{N}, l \in \mathcal{L}(n), s \in \mathcal{S} && \text{(5.46a)} \\
& z_{nl,s} = \sum_{j\in\mathcal{L}(n), j\neq l} \mathbf{h}_{nl,s}^{\mathrm{H}}(t) \mathbf{M}_{nj,s} \mathbf{h}_{nl,s}(t), \quad n \in \mathcal{N}, l \in \mathcal{L}(n), \\
& \qquad\qquad s \in \mathcal{S} && \text{(5.46b)} \\
& \hat{\chi}_n(\mathbf{b}_n) \leq \theta_n, \quad n \in \mathcal{N} && \text{(5.46c)} \\
& \textstyle\sum_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}} \mathrm{Trace}(\mathbf{M}_{nl,s}) \leq p_n^{\max}, \quad n \in \mathcal{N} && \text{(5.46d)} \\
& \textstyle\sum_{n\in\mathcal{N}} b_{ns} = 1, \quad s \in \mathcal{S} && \text{(5.46e)} \\
& 0 \leq b_{ns} \leq 1, \quad n \in \mathcal{N}, s \in \mathcal{S} && \text{(5.46f)} \\
& \mathbf{M}_{nl,s} \succeq 0, \quad n \in \mathcal{N}, l \in \mathcal{L}(n), s \in \mathcal{S}, && \text{(5.46g)}
\end{aligned}$$

with variables $\{\mathbf{u}_{nl}, \mathbf{z}_{nl}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n, \theta_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{M}_{nl,s}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S}}$. Note that in problem (5.46) we have removed $\mathrm{Rank}(\mathbf{M}_{nl,s}) = 1$ constraint for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$.

We find a solution for problem (5.46) by solving a sequence of approximated convex problems [36]. The best convex approximation of problem (5.46) can be obtained by replacing $\breve{g}_0$ with its first order approximation (i.e., by replacing $\phi_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)$ and $\zeta_n(\mathbf{b}_n)$ with their first order approximations) [36]. The first

order approximation of $\phi_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)$ and $\zeta_n(\mathbf{b}_n)$, respectively, near an arbitrary positive point $(\hat{\mathbf{z}}_{nl}, \hat{\mathbf{b}}_n)$ can be expressed as

$$\begin{aligned}\hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n) = \phi_{nl}(\hat{\mathbf{z}}_{nl}, \hat{\mathbf{b}}_n) + \frac{1}{\log(2)} \sum_{s\in\mathcal{S}} \Bigg(- \frac{\hat{b}_{ns} w_s}{\hat{b}_{ns} N_0 w_s + \hat{z}_{nl,s}} \Bigg)(z_{nl,s} - \hat{z}_{nl,s}) \\ + \frac{1}{\log(2)} \sum_{s\in\mathcal{S}} w_s \Bigg(\frac{\hat{z}_{nl,s}}{\hat{b}_{ns} N_0 w_s + \hat{z}_{nl,s}} \\ - \log\Big(\frac{\hat{b}_{ns} N_0 w_s + \hat{z}_{nl,s}}{\hat{b}_{ns}} \Big) \Bigg)(b_{ns} - \hat{b}_{ns}) \quad (5.47)\end{aligned}$$

and

$$\hat{\zeta_n}(\mathbf{b}_n) = \zeta_n(\hat{\mathbf{b}}_n) + \sum_{s\in\mathcal{S}} \big(1 + \log(\hat{b}_{ns})\big)(b_{ns} - \hat{b}_{ns}). \quad (5.48)$$

Hence, by using the expression of $\breve{f}_0$ (see 5.42) and the first order approximation of $\breve{g}_0$ (obtained by substituting (5.47) and (5.48) in (5.43)), problem (5.46) near an arbitrary positive point $(\hat{\mathbf{z}}_{nl}, \hat{\mathbf{b}}_n)$ can be expressed as the following convex optimization problem:

$$\begin{aligned} &\text{minimize} && \sum_{n\in\mathcal{N}} \Big(\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\big(\psi_{nl}(\mathbf{u}_{nl}, \mathbf{b}_n) - \hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)\big) - \delta\hat{\zeta_n}(\mathbf{b}_n) + \theta_n \Big) \\ &\text{subject to} && \text{constraints } (5.46a) - (5.46g) \end{aligned} \quad (5.49)$$

with variables $\{\mathbf{u}_{nl}, \mathbf{z}_{nl}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n, \theta_n\}_{n\in\mathcal{N}}$, and $\{\mathbf{M}_{nl,s}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S}}$. Finally, we summarize the proposed algorithm based on sequential convex programming for the resource allocation problem (5.33) in Algorithm 5.2.

Algorithm 5.2. *Centralized algorithm for the resource allocation problem (5.33)*

1. Initialization: given initial feasible starting point $\{\mathbf{z}_{nl}^0, \mathbf{b}_n^0\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$ and parameter $\delta > 0$. Set iteration index $k = 0$.
2. Set $\hat{\mathbf{z}}_{nl} = \mathbf{z}_{nl}^k$ and $\hat{\mathbf{b}}_n = \mathbf{b}_n^k$, then form $\hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)$ and $\hat{\zeta_n}(\mathbf{b}_n)$ by using expressions (5.47) and (5.48), respectively, for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$.
3. Solve problem (5.49), and denote the solution by $\{\mathbf{u}_{nl}^\star, \mathbf{z}_{nl}^\star\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n^\star\}_{n\in\mathcal{N}}$, and $\{\mathbf{M}_{nl,s}^\star\}_{n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S}}$. Update $\mathbf{u}_{nl}^{k+1} = \mathbf{u}_{nl}^\star$, $\mathbf{z}_{nl}^{k+1} = \mathbf{z}_{nl}^\star$, $\mathbf{b}_n^{k+1} = \mathbf{b}_n^\star$, and $\mathbf{M}_{nl,s}^{k+1} = \mathbf{M}_{nl,s}^\star$ for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$.
4. Stopping criterion: if the stopping criterion is satisfied, go to step 5. Otherwise set $k = k + 1$, and go to step 2.

5. Obtain a rank one approximation of $\mathbf{M}^{\star}_{nl,s}$ and denote it by $\mathbf{m}^{\star}_{nl,s}$, for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$. Return solution $\{\mathbf{b}^{\star}_{n}, \mathbf{m}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$.

The first step initializes the algorithm. Step 2 performs a first order approximation of functions $\hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)$ and $\hat{\zeta_n}(\mathbf{b}_n)$ at the point $(\hat{\mathbf{z}}_{nl}, \hat{\mathbf{b}}_n)$ for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$. Then the approximated convex problem (5.49) is solved at step 3. Step 4 checks the stopping criteria[46]. Note that we have dropped the $\text{Rank}(\mathbf{M}_{nl,s}) = 1$ constraint for all $n \in \mathcal{N}$, $l \in \mathcal{L}(n)$, and $s \in \mathcal{S}$ to arrive at problem (5.49); thus a solution $\{\mathbf{M}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ obtained at step 3 may not be rank one in general. Hence, at step 5, we perform a rank one approximation of $\{\mathbf{M}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ to obtain the transmit beamformers $\{\mathbf{m}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ for problem (5.33). Specifically, we use the randomization technique (randA method) presented in [209, Sec. IV] to obtain a rank one solution. Then the power and direction of the transmit beamformer associated with lth user of BS n in subchannel s can be set to $p_{nl,s} = \|\mathbf{m}^{\star}_{nl,s}\|_2^2$ and $\mathbf{v}_{nl,s} = \mathbf{m}^{\star}_{nl,s}/\|\mathbf{m}^{\star}_{nl,s}\|_2$, respectively.

In the randA method [209, Sec. IV], we calculate the eigen-decomposition of $\mathbf{M}^{\star}_{nl,s}$ as $\mathbf{M}^{\star}_{nl,s} = \sum_{i=1}^{T} \lambda^{i}_{nl,s}\mathbf{v}^{i}_{nl,s}(\mathbf{v}^{i}_{nl,s})^{\text{H}}$; where $\lambda^{i}_{nl,s}$ is the eigenvalue and vector $\mathbf{v}^{i}_{nl,s}$ is the corresponding eigenvector. Then a rank one approximation of $\mathbf{M}^{\star}_{nl,s}$ is calculated as $\mathbf{m}^{\star}_{nl,s} = \sum_{i=1}^{T} c^{i}(\lambda^{i}_{nl,s})^{1/2}\mathbf{v}^{i}_{nl,s}$; where c^{i} is a random variable uniformly distributed on the unit circle in the complex plane, i.e., $c^{i} = e^{j\varepsilon_i}$ and ε_i is uniformly distributed on $[0, 2\pi)$.

5.3.1 Signaling overhead

Algorithm 5.2 is a centralized algorithm. Hence, it is required to collect the value of network layer queues $\{Q_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$ and CSI $\{\mathbf{h}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ of all the users of the operators in a central controller (CC). That is a total of $\check{L}_1 + \check{L}_2$ real scalars of parameters $\{Q_{nl}(t)\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, and a total of $2S(\check{L}_1 + \check{L}_2)$ real scalars of parameters for $\{\mathbf{h}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ has to be sent to the CC. Hence, the number of real scalars that has to be sent from the operators to the CC is $(2S + 1)(\check{L}_1 + \check{L}_2)$. Next, the CC has to send the computed beamformers

[46] The algorithm can be stopped either when a difference between the achieved objective value of problem (5.46) between two successive iterations is less than a given threshold, or it runs for a finite number of iterations [113, Sec. 3.2.2], [208, Sec. IV.B].

associated with the users to their respective operators. This requires $2S(\check{L}_1 + \check{L}_2)$ number of scalars to be sent to the operators. Thus, the total number of real scalars required to be exchanged between the operators and the CC in Algorithm 5.2 is $(4S + 1)(\check{L}_1 + \check{L}_2)$.

5.3.2 Convergence

Algorithm 5.2 solves the DC programming problem (5.46) by using an approach similar to that in [36]. Hence Algorithm 5.2 is a descent algorithm [36, Sec. 1.3]. The proof is identical to that provided in [36, Sec. 1.3], and it is omitted here for brevity.

5.4 Resource allocation - distributed algorithm

In this section we extend Algorithm 5.2 to derive a distributed algorithm for resource allocation problem (5.33). The distributed algorithm is derived by solving step 3 of Algorithm 5.2 (i.e, problem (5.49)) using the ADMM [113].

We start by introducing an auxiliary variable x_{ns} as a copy of b_{ns}, for all $n \in \mathcal{N}$ and $s \in \mathcal{S}$. Then problem (5.49) can be equivalently written as

$$\begin{aligned}
\text{minimize} \quad & \sum_{n\in\mathcal{N}} \Big(\sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\big(\psi_{nl}(\mathbf{u}_{nl}, \mathbf{b}_n) - \hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)\big) - \delta\hat{\zeta}_n(\mathbf{b}_n) + \theta_n \Big) \\
\text{subject to} \quad & \text{constraints } (5.46a) - (5.46d),\ (5.46f),\ (5.46g) && \text{(5.50a)} \\
& b_{ns} = x_{ns}, \quad n \in \mathcal{N}, s \in \mathcal{S} && \text{(5.50b)} \\
& \textstyle\sum_{n\in\mathcal{N}} x_{ns} = 1, \quad s \in \mathcal{S}, && \text{(5.50c)}
\end{aligned}$$

with variables $\{\mathbf{u}_{nl}, \mathbf{z}_{nl}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n, \theta_n\}_{n\in\mathcal{N}}$, $\{\mathbf{M}_{nl,s}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n), s\in\mathcal{S}}$, and $\{x_{ns}\}_{n\in\mathcal{N}, s\in\mathcal{S}}$. Observe that without constraint (5.50c), problem (5.50) can be easily decoupled into two subproblems, one for each operator.

We now express problem (5.50) more compactly. To do this, let us define the matrix $\mathbf{M}_{nl} = [\mathbf{M}_{nl,1}, \ldots, \mathbf{M}_{nl,S}]$, and the following set

$$\mathcal{O}_n = \left\{ \{\mathbf{u}_{nl}, \mathbf{z}_{nl}, \mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)}, \mathbf{b}_n, \theta_n \;\middle|\; \begin{array}{l} u_{nl,s} = \sum\limits_{j\in\mathcal{L}(n)} \mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{M}_{nj,s}\mathbf{h}_{nl,s}(t),\ l\in\mathcal{L}(n), s\in\mathcal{S} \\ z_{nl,s} = \sum\limits_{j\in\mathcal{L}(n), j\neq l} \mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{M}_{nj,s}\mathbf{h}_{nl,s}(t),\ l\in\mathcal{L}(n), s\in\mathcal{S} \\ \hat{\chi}_n(\mathbf{b}_n) \leq \theta_n \\ \sum\limits_{l\in\mathcal{L}(n)} \sum_{s\in\mathcal{S}} \mathrm{Trace}(\mathbf{M}_{nl,s}) \leq p_n^{\max} \\ 0 \leq b_{ns} \leq 1, \quad s\in\mathcal{S} \\ \mathbf{M}_{nl,s} \succeq 0, \quad l\in\mathcal{L}(n), s\in\mathcal{S} \end{array} \right\}. \tag{5.51}$$

Furthermore, for the sake of brevity, let us define the following functions

$$\Phi_n\big(\{\mathbf{u}_{nl}, \mathbf{z}_{nl}, \mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)}, \mathbf{b}_n, \theta_n\big) = \begin{cases} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\big(\psi_{nl}(\mathbf{u}_{nl}, \mathbf{b}_n) - \hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)\big) & \\ -\delta\hat{\zeta}_n(\mathbf{b}_n) + \theta_n & \{\mathbf{u}_{nl}, \mathbf{z}_{nl}, \mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)}, \mathbf{b}_n, \theta_n \in \mathcal{O}_n \\ \infty & \text{otherwise,} \end{cases} \tag{5.52}$$

and

$$I_n\big(\{x_{ns}\}_{n\in\mathcal{N}, s\in\mathcal{S}}\big) = \begin{cases} 0 & \sum_{n\in\mathcal{N}} x_{ns} = 1 \text{ for all } s\in\mathcal{S} \\ \infty & \text{otherwise} \end{cases}. \tag{5.53}$$

Then by using expressions (5.51)-(5.53), problem (5.50) can be written compactly as

$$\begin{array}{ll} \text{minimize} & \sum\limits_{n\in\mathcal{N}} \Phi_n\big(\{\mathbf{u}_{nl}, \mathbf{z}_{nl}, \mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)}, \mathbf{b}_n, \theta_n\big) + I_n\big(\{x_{ns}\}_{n\in\mathcal{N}, s\in\mathcal{S}}\big) \\ \text{subject to} & b_{ns} = x_{ns}, \quad n\in\mathcal{N}, s\in\mathcal{S}, \end{array} \tag{5.54}$$

with variables $\{\mathbf{u}_{nl}, \mathbf{z}_{nl}, \mathbf{M}_{nl}\}_{n\in\mathcal{N}, l\in\mathcal{L}(n)}$, $\{\mathbf{b}_n, \theta_n\}_{n\in\mathcal{N}}$, and $\{x_{ns}\}_{n\in\mathcal{N}, s\in\mathcal{S}}$.

5.4.1 Distributed algorithm via ADMM

To derive the ADMM algorithm for problem (5.54) we first form the augmented Lagrangian [113]. Let $\{\lambda_{ns}\}_{n\in\mathcal{N}, s\in\mathcal{S}}$ be the dual variables associated with the

equality constraints of problem (5.54). Then the augmented Lagrangian can be written as

$$L_\rho\big(\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl}\}_{n\in\mathcal{N},l\in\mathcal{L}(n)},\{\mathbf{b}_n,\theta_n\}_{n\in\mathcal{N}},\{x_{ns},\lambda_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}\big) = \sum_{n\in\mathcal{N}}\Phi_n\big(\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n,\theta_n\big) + I_n\big(\{x_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}\big) + \sum_{n\in\mathcal{N}}\sum_{s\in\mathcal{S}}\big(\lambda_{ns}(b_{ns}-x_{ns}) + (\rho/2)(b_{ns}-x_{ns})^2\big), \quad (5.55)$$

where $\rho>0$ is a penalty parameter that adds a quadratic penalty to the standard Lagrangian L_0 for the violation of the equality constraints of problem (5.54).

Each iteration of the ADMM algorithm consists of the following three steps [113]

$$\{\mathbf{u}_{nl}^{i+1},\mathbf{z}_{nl}^{i+1},\mathbf{M}_{nl}^{i+1}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n^{i+1},\theta_n^{i+1} = \underset{\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n,\theta_n\in\mathcal{O}_n}{\operatorname{argmin}} L_\rho\big(\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl},\mathbf{b}_n,\theta_n,x_{ns}^i,\lambda_{ns}^i\}_{l\in\mathcal{L}(n),s\in\mathcal{S}}\big), \quad n\in\mathcal{N} \quad (5.56)$$

$$\{x_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}} = \underset{\{x_{ns}\}_{n\in\mathcal{N},n\in\mathcal{S}}}{\operatorname{argmin}} L_\rho\big(\{\mathbf{u}_{nl}^{i+1},\mathbf{z}_{nl}^{i+1},\mathbf{M}_{nl}^{i+1},\mathbf{b}_n^{i+1},\theta_n^{i+1},x_{ns},\lambda_{ns}^i\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}\big) \quad (5.57)$$

$$\lambda_{ns}^{i+1} = \lambda_{ns}^i + \rho\left(b_{ns}^{i+1}-x_{ns}^{i+1}\right), \quad n\in\mathcal{N}, s\in\mathcal{S}, \quad (5.58)$$

where superscript i is the iteration counter. Note that steps (5.56) and (5.58) are completely decentralized, and hence, can be carried out independently in parallel by each operator. Step (5.57) requires to gather the updated local variables $\big\{\{\mathbf{u}_{nl}^{i+1},\mathbf{z}_{nl}^{i+1}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n^{i+1},\theta_n^{i+1}\big\}$ and the dual variables $\{\lambda_{ns}^i\}_{s\in\mathcal{S}}$ from both operators. In the sequel we first explain, in detail, how to solve the ADMM step (5.57), and then simplify the above ADMM update-steps (5.56)-(5.58) into two update-steps. Then we summarize the proposed ADMM based distributed algorithm.

The update $\{x_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ in (5.57) is a solution of the following optimization problem:

$$\text{minimize} \quad \sum_{n\in\mathcal{N}}\sum_{s\in\mathcal{S}}\big(\lambda_{ns}^i(b_{ns}^{i+1}-x_{ns}) + \frac{\rho}{2}(b_{ns}^{i+1}-x_{ns})^2\big) + I_n\big(\{x_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}\big), \quad (5.59)$$

with variable $\{x_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$. Let $v_{ns}=(1/\rho)\lambda_{ns}$ be a scaled dual variable. Then by using notation (5.53), problem (5.59) can be equivalently expressed as

$$\begin{array}{ll}\text{minimize} & \sum_{n\in\mathcal{N}}\sum_{s\in\mathcal{S}}(\rho/2)(x_{ns}-v_{ns}^i-b_{ns}^{i+1})^2 \\ \text{subject to} & \sum_{n\in\mathcal{N}}x_{ns}=1, \quad s\in\mathcal{S},\end{array} \quad (5.60)$$

with the optimization variable $\{x_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$. Note that in the objective function of problem (5.60) [47], we have dropped the constant term $(\rho/2)(v_{ns}^i)^2$ since it does not affect the solution of the problem. For the equality constrained convex optimization problem (5.60), we can easily find the optimal solution by solving KKT optimality conditions [29, Ch. 5.5.3]. By solving KKT optimality conditions of problem (5.60), the solution $\{x_{ns}^\star\}_{n\in\mathcal{N},s\in\mathcal{S}}$ can be expressed as

$$x_{ns}^\star = v_{ns}^i + b_{ns}^{i+1} - \tilde{v}_s^i - \tilde{b}_s^{i+1} + 1/N, \quad n\in\mathcal{N}, s\in\mathcal{S}, \tag{5.61}$$

where $N=|\mathcal{N}|$, $\tilde{v}_s^i = (1/N)\sum_{n\in\mathcal{N}} v_{ns}^i$ and $\tilde{b}_s^{i+1} = (1/N)\sum_{n\in\mathcal{N}} b_{ns}^{i+1}$. Therefore, the update x_{ns}^{i+1} is

$$x_{ns}^{i+1} = x_{ns}^\star, \quad n\in\mathcal{N}, s\in\mathcal{S}. \tag{5.62}$$

Now we substitute expression (5.61) for x_{ns}^{i+1} in the dual variable update step (5.58). Furthermore, by using a scaled dual variable $v_{ns} = (1/\rho)\lambda_{ns}$, step (5.58) can be simplified as

$$v_{ns}^{i+1} \quad = \quad \tilde{v}_s^i + \tilde{b}_s^{i+1} - 1/N, \quad n\in\mathcal{N}, s\in\mathcal{S}. \tag{5.63}$$

Expression (5.63) shows that the dual variables v_{ns} for all $n\in\mathcal{N}$ are equal. Hence, the dual variables $\{v_{ns}\}_{n\in\mathcal{N}}$ can be replaced with a single variable $v_s\in\mathbb{R}$ in the ADMM iteration (5.56)-(5.58). Finally, by substituting the expression for x_{ns}^i (that can be obtained from (5.62)) in (5.56), the ADMM iteration (5.56)-(5.58) can be simplified into the following two-step iterative algorithm:

$$\begin{aligned}\{\mathbf{u}_{nl}^{i+1}, \mathbf{z}_{nl}^{i+1}, \mathbf{M}_{nl}^{i+1}\}_{l\in\mathcal{L}(n)}, \mathbf{b}_n^{i+1}, \theta_n^{i+1} = \\ \underset{\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n\in\mathcal{O}_n}{\operatorname{argmin}} \Big(\Phi_n\big(\{\mathbf{u}_{nl},\mathbf{z}_{nl},\mathbf{M}_{nl}\}_{l\in\mathcal{L}(n)},\mathbf{b}_n,\theta_n\big) \\ + \sum_{s\in\mathcal{S}} (\rho/2)(b_{ns} - b_{ns}^i + \tilde{b}_s^i + v_s^i - 1/N)^2\Big), \quad n\in\mathcal{N}\end{aligned} \tag{5.64}$$

$$v_s^{i+1} = v_s^i + \tilde{b}_s^{i+1} - 1/N, \quad s\in\mathcal{S}. \tag{5.65}$$

Note that we have used expression (5.55) to arrive at (5.64). By using notation (5.52), the optimization problem to update variables $\{\mathbf{u}_{nl}^{i+1}, \mathbf{z}_{nl}^{i+1}, \mathbf{M}_{nl}^{i+1}\}_{l\in\mathcal{L}(n)}$,

[47] We can simplify $\lambda_{ns}^i(b_{ns}^{i+1}-x_{ns})+(\rho/2)(b_{ns}^{i+1}-x_{ns})^2 = (\rho/2)(x_{ns}-v_{ns}^i-b_{ns}^{i+1})^2-(\rho/2)(v_{ns}^i)^2$.

$\mathbf{b}_n^{i+1}$, θ_n^{i+1} in (5.64) can be expressed as

$$
\begin{aligned}
\text{minimize} \quad & \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)\big(\psi_{nl}(\mathbf{u}_{nl},\mathbf{b}_n)-\hat{\phi}_{nl}(\mathbf{z}_{nl},\mathbf{b}_n)\big)-\delta\hat{\zeta}_n(\mathbf{b}_n)+\theta_n \\
& +\sum_{s\in\mathcal{S}}(\rho/2)(b_{ns}-b_{ns}^i+\tilde{b}_s^i+v_s^i-1/N)^2 \\
\text{subject to} \quad & u_{nl,s}=\textstyle\sum_{j\in\mathcal{L}(n)}\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{M}_{nj,s}\mathbf{h}_{nl,s}(t), \quad l\in\mathcal{L}(n), s\in\mathcal{S} \\
& z_{nl,s}=\textstyle\sum_{j\in\mathcal{L}(n),j\neq l}\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{M}_{nj,s}\mathbf{h}_{nl,s}(t), \quad l\in\mathcal{L}(n), s\in\mathcal{S} \\
& \hat{\chi}_n(\mathbf{b}_n)\leq\theta_n \\
& \textstyle\sum_{l\in\mathcal{L}(n)}\sum_{s\in\mathcal{S}}\mathrm{Trace}(\mathbf{M}_{nl,s})\leq p_n^{\max} \\
& 0\leq b_{ns}\leq 1, \quad s\in\mathcal{S} \\
& \mathbf{M}_{nl,s}\succeq 0, \quad l\in\mathcal{L}(n), s\in\mathcal{S},
\end{aligned}
\tag{5.66}
$$

with variables $\{\mathbf{u}_{nl},\mathbf{z}_{nl}\}_{l\in\mathcal{L}(n)}$, $\mathbf{b}_n$, θ_n, and $\{\mathbf{M}_{nl,s}\}_{l\in\mathcal{L}(n),s\in\mathcal{S}}$. We now summarize the proposed distributed algorithm for problem (5.33) as follows:

Algorithm 5.3. *Distributed algorithm for problem (5.33)*

1. Initialization: given initial feasible starting points $\{\mathbf{z}_{nl}^0,\mathbf{b}_n^0\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}$, $\{v_s^0\}$, and parameters $\delta>0$ and $\rho>0$. Set iteration indices $i=0$ and $k=0$.
2. Set $\hat{\mathbf{z}}_{nl}=\mathbf{z}_{nl}^i$ and $\hat{\mathbf{b}}_n=\mathbf{b}_n^i$, then form $\hat{\phi}_{nl}(\mathbf{z}_{nl},\mathbf{b}_n)$ and $\hat{\zeta}_n(\mathbf{b}_n)$ by using expressions (5.47) and (5.48), respectively, for all $n\in\mathcal{N}$ and $l\in\mathcal{L}(n)$.
3. ADMM iteration:
   a) each operator $n\in\mathcal{N}$ updates the local variables $\{\mathbf{u}_{nl}^{i+1},\mathbf{z}_{nl}^{i+1},\mathbf{M}_{nl}^{i+1}\}_{l\in\mathcal{L}(n)}$, $\mathbf{b}_n^{i+1}$, and θ_n^{i+1} by solving (5.66).
   b) operators exchange their updated local variables $\{b_{ns}^{i+1}\}_{s\in\mathcal{S}}$ with each other.
   c) each operator $n\in\mathcal{N}$ updates the dual variables $\{v_s^{i+1}\}_{s\in\mathcal{S}}$ by solving (5.65).
   d) ADMM stopping criterion: if the stopping criterion is satisfied, go to step 4. Otherwise, set $i=i+1$, and go to step 3a.
4. Stopping criterion: if the stopping criterion is satisfied, go to step 5. Otherwise set $i=i+1$, $k=k+1$, and go to step 2.
5. Set $\mathbf{b}_n^\star=\mathbf{b}_n^{i+1}$, obtain a rank one approximation of $\mathbf{M}_{nl,s}^{i+1}$ and denote it by $\mathbf{m}_{nl,s}^\star$, for all $n\in\mathcal{N}$, $l\in\mathcal{L}(n)$, and $s\in\mathcal{S}$. Return $\{\mathbf{b}_n^\star,\mathbf{m}_{nl,s}^\star\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$.

The steps of Algorithm 5.3 are similar to those of the centralized Algorithm 5.2, except the step 3 of both algorithms. Step 3 of Algorithm 5.2 solves problem (5.49) in a central controller. However, in Algorithm 5.3 the same problem is solved by using ADMM (i.e., by performing iterations in steps (3.a)-(3.d)). Note that we have applied ADMM to a convex problem, and therefore the ADMM iterations converge to the global optimal value [193, Proposition 4.2], [113]. Thus, by following Section 5.3.2, we can see that a sequence of the objective value of program (5.46), that is produced upon ADMM convergence (i.e., after step 3 of Algorithm 5.3) is monotonic.

Step 3.d of Algorithm 5.3 checks the ADMM stopping criteria. In the ADMM algorithm, standard stopping criteria is to check the primal and dual residuals [113]. We refer to each execution of steps 2-4 as an outer iteration, and we use index k to count this. Step 4 checks the stopping criteria for the outer iteration[48].

Since ADMM usually produces acceptable results for practical use within only a few iterations, a predefined or fixed number of iterations can be used as stopping criterion for the ADMM iterations [113], in step 3.d. The transmit beamformers $\{\mathbf{m}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$ for problem (5.33) are computed at step 5, upon the convergence of the algorithm. At step 5, we can use the randA method[49] presented in [209, Sec. IV] to obtain $\{\mathbf{m}^{\star}_{nl,s}\}_{n\in\mathcal{N},l\in\mathcal{L}(n),s\in\mathcal{S}}$. Then the power and direction of the transmit beamformer associated with lth user of BS n in subchannel s can be set to $p_{nl,s} = \|\mathbf{m}^{\star}_{nl,s}\|_2^2$ and $\mathbf{v}_{nl,s} = \mathbf{m}^{\star}_{nl,s}/\|\mathbf{m}^{\star}_{nl,s}\|_2$, respectively.

5.4.2 Signalling overhead

It is worth pointing out that, in Algorithm 5.3, operators do not need to share their users' data and the channel state information with each other. Except step 3.b, all other steps of Algorithm 5.3 are decoupled over the operators. Step 3.b requires coordination between operators to exchange their updated value of local variables $\{b^{i+1}_{ns}\}_{s\in\mathcal{S}}$. Hence, at each iteration an operator requires

[48] The algorithm can be stopped either when a difference between the achieved objective value of problem (5.46) between two successive iterations is less than a given threshold, or when it has run for a finite number of iterations [113, Sec. 3.2.2], [208, Sec. IV.B].

[49] The dominant eigenvalue and corresponding eigenvector of $\mathbf{M}^{i+1}_{nl,s}$ can also be used to obtain a rank one approximation.

S number of real scalars to be sent to the other operator. Thus, at each iteration the total number of real scalars required to be exchanged between operators is $2S$.

5.4.3 Early termination of ADMM iteration

To speed up Algorithm 5.3 we can stop the ADMM iteration after a finite number of iterations before it converges. In this case the intermediate solutions $\{b_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ provided by the ADMM iteration do not necessarily result in a feasible solution for the original problem (5.33). In particular constraint (5.33b) may not hold $\big($i.e., $\sum_{n\in\mathcal{N}} b_{ns}^{i+1} \neq 1$ for some $s \in \mathcal{S}\big)$. Thus, we need to project $\{b_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ on the set $\mathcal{F}$, defined as $\mathcal{F} = \{\{b_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}} | \sum_{n\in\mathcal{N}} b_{ns} = 1, s \in \mathcal{S}\}$, to evaluate expressions $\hat{\phi}_{nl}(\mathbf{z}_{nl}, \mathbf{b}_n)$ and $\hat{\zeta_n}(\mathbf{b}_n)$ at step 2 of Algorithm 5.3. The projection of $\{b_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ on the set $\mathcal{F}$ can be obtained by solving problem (5.60), with v_{ns}^i set to zero for all $n \in \mathcal{N}$ and $s \in \mathcal{S}$. Hence the projection of $\{b_{ns}^{i+1}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ on the set $\mathcal{F}$ is $\{b_{ns}^{i+1} - \tilde{b}_s^{i+1} + 1/N\}_{n\in\mathcal{N},s\in\mathcal{S}}$, which is obtained by setting $v_{ns}^i = 0$ for all $n \in \mathcal{N}$ and $s \in \mathcal{S}$ in expression (5.61).

5.4.4 Complexity of the proposed algorithms

Algorithm 5.2 and Algorithm 5.3 are iterative[50] algorithms. Thus, we focus on characterizing their complexity per iteration. Note that both algorithms solve convex problems at each iteration (i.e., problem (5.49) is solved in Algorithm 5.2, and problem (5.66) is solved in Algorithm 5.3). Thus, these problems can be efficiently solved by using an interior-point method that relies on the Newton's method applied to a sequence of modified versions of the original problem [29, Ch. 11]. The complexity of solving a Newton step for problem (5.49) is $O((2LS(T^2+1) + NS + N + S)^3)$, where $L = \sum_{n\in\mathcal{N}} \check{L}_n$; and that of problem (5.66) is $O((2\check{L}_n S(T^2+1) + S + 1)^3)$ [29, Ch. 10.4] [51]. Note that, in general, a convex problem requires only a modest number of Newton steps to

[50] It is important to point out that, in practice, the quality of the solution achieved within the first few iterations are more important than the asymptotic results, as we usually have time to perform only small number of iterations.

[51] The complexity order is computed by relaxing the first and second equality constraints of problem (5.49) and (5.66), as they hold with equality at the optimal solution.

solve it with high accuracy (i.e., a number of Newton steps between $30-100$ are enough for most of the applications) [29, Ch. 11.3.2].

5.5 Numerical examples

We illustrate the performance of the proposed Algorithm 5.1, Algorithm 5.2, and Algorithm 5.3 by using the setup as shown in Fig. 5.2. The network consists of a cell with two coexisting BSs, belonging to two operators. The BSs are assumed to be installed in a same tower [210–212], and they are placed at different height levels of the tower[52]. Each BS consists of $T = 2$ transmit antennas. We assume a circular cell, with a radius R_{BS}. We assume that there are $\check{L}_1 = 3$ users associated with BS 1, and $\check{L}_2 = 6$ users associated with BS 2. The locations of the users associated with each BS are arbitrarily chosen as shown in Fig. 5.2. We assume that each operator shares $B = 2$ MHz spectrum band, and hence, the total spectrum band of 4 MHz is available for both operators. We split the spectrum band 4 MHz into $S = 4$ subchannels, and the bandwidth of each subchannel $w_s = 1$ MHz.

We assume an exponential path loss model, where the channel vector from nth BS to its lth user on subchannel s is modeled as

$$\mathbf{h}_{nl,s}(t) = \left(\frac{d_{nl}(t)}{d_0}\right)^{-\eta/2} \mathbf{c}_{nl,s}(t), \tag{5.67}$$

where $d_{nl}(t)$ is the distance from BS n to its lth user, d_0 is the *far field reference distance* [173], η is the path loss exponent, and $\mathbf{c}_{nl,s}(t) \in \mathbb{C}^T$ is arbitrarily chosen from the distribution $\mathcal{CN}(0, \mathbf{I})$ (i.e., frequency-flat fading channel with uncorrelated antennas). Note that the term $(d_{nl}(t)/d_0)^{-\eta/2}$ denotes large scale fading, and the term $\mathbf{c}_{nl,s}(t)$ denotes small scale fading. Here, we refer an arbitrarily generated set of fading coefficients $\check{\mathcal{C}}(t) = \{\mathbf{c}_{nl,s}(t), d_{nl}(t) | n \in \mathcal{N}, l \in \mathcal{L}(n), s \in \mathcal{S}\}$ as a *single fading realization*.

[52] This particular setup is chosen to illustrate that the operators are operating in a same geographical area. However, our problem formulation and the proposed algorithms are general, and they are applicable when the BSs of the operators are far apart and also to the case of correlated channels.

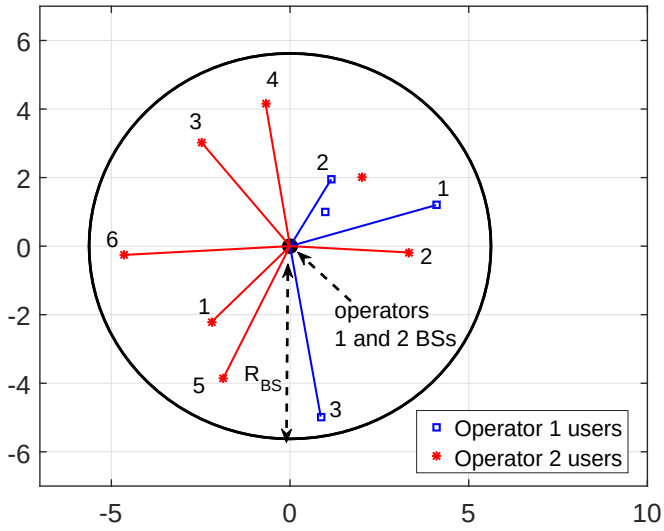


Fig. 5.2. A cell with two coexisting BSs belonging to different operators. BSs are placed on the same cell site. $\mathcal{N} = \{1, 2\}$**,** $\mathcal{L}(1) = \{1, 2, 3\}$**, and** $\mathcal{L}(2) = \{1, 2, 3, 4, 5, 6\}$**, [165] ©2017, IEEE.**

We assume that $p_n^{\max} = p_0^{\max}$ for all $n \in \mathcal{N}$. We define the SNR operating point at a distance r as

$$\mathrm{SNR}(r) = \left(\frac{r}{d_0}\right)^{-\eta} \frac{p_0^{\max}}{N_0 w_s}. \tag{5.68}$$

In the following simulations, we set $d_0 = 1$, $\eta = 4$, and the cell radius R_{BS} is fixed throughout the simulations so that $\mathrm{SNR}(R_{\mathrm{BS}}) = 10$ dB for $p_0^{\max}/(N_0 w_s) = 40$ dB.

To solve step 4 of Algorithm 5.1, we use either centralized Algorithm 5.2 or distributed Algorithm 5.3. Thus, we first present the performance of Algorithm 5.2 and Algorithm 5.3. Then we evaluate the performance of Algorithm 5.1. In Algorithm 5.2 and Algorithm 5.3, we set a penalty parameter $\delta = (0.1 w_s)k$. That is a varying penalty parameter δ is used so that more weight is given to the penalty function $\hat{\zeta}_n(\mathbf{b}_n)$ as the algorithm progresses. In Algorithm 5.3, for the ADMM iteration, we use the standard stopping criteria presented in [113, Sec. 3.3.1]. The stopping criteria in [113, Sec. 3.3.1] calculates the primal residual $r_{\mathrm{pri}}^i = (\sum_{n\in\mathcal{N}} \sum_{s\in\mathcal{S}} (b_{ns}^i - x_{ns}^i)^2)^{1/2}$ and the dual residual $r_{\mathrm{dual}}^i = (\rho \sum_{n\in\mathcal{N}} \sum_{s\in\mathcal{S}} (x_{ns}^i - x_{ns}^{i-1})^2)^{1/2}$, see (5.61) for the expression of x_{ns}^i. Then the ADMM iteration is stopped, if $r_{\mathrm{pri}}^i \leq \epsilon$ and $r_{\mathrm{dual}}^i \leq \epsilon$, where $\epsilon > 0$ is a

given tolerance. In the simulation, we set $\epsilon = 0.1$. Furthermore, we limit the ADMM iteration to a maximum of 10 iterations.

To evaluate the performance of Algorithm 5.2 and Algorithm 5.3, we consider a single fading realization. The weight $Q_{nl}(t)$ associated with each user is set to one for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$; the price per-unit spectrum $q_n(t)$ of each operator is set to zero for all $n \in \mathcal{N}$. As we set $q_n(t) = 0$ for all $n \in \mathcal{N}$, Algorithm 5.2 and Algorithm 5.3 solve a WSR maximization problem (see problem (5.33)), and that jointly allocates spectrum band to the operators and design transmit beamformers.

Fig. 5.3 shows the convergence behavior of the centralized Algorithm 5.2 for SNR = 5 dB and SNR = 10 dB. The WSR values of problem (5.33) are computed after step 3 of the algorithm. Results show that the proposed Algorithm 5.2 converges within the first few iterations.

Fig. 5.4 shows the convergence behavior of the distributed Algorithm 5.3 for SNR = 5 dB, along with the objective value obtained by the centralized Algorithm 5.2. We set the ADMM penalty parameter $\rho = 5w_s$ and $10w_s$. The WSR values of problem (5.33) are computed after step 3.c of Algorithm 5.3. The markers "circle" and "asterisk" in the figure for $\rho = 5w_s$ and $10w_s$, respectively, represent the start of the ADMM iteration for a new point $\big(\{\hat{\mathbf{z}}_{nl}\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}, \{\hat{\mathbf{b}}_n\}_{n\in\mathcal{N}}\big)$ that is set at step 2 of the algorithm. Results show that the proposed distributed algorithm converges to the centralized objective value for different values of ρ. Furthermore, results show that the number of iterations between two successive "circle" markers (and also between two successive "asterisk" markers) decreases as the algorithm progress. That is, a fewer number of ADMM iterations (i.e., step 3 of Algorithm 5.3) are required as the algorithm progresses.

In order to see the average behavior of Algorithm 5.2 and Algorithm 5.3, we next consider the fading case. For Algorithm 5.3, we set $\rho = w_s,\ 5w_s$, and $10w_s$. Let $\breve{h}_0^k$ denote the objective value of problem (5.46) obtained at kth iteration, i.e., $\breve{h}_0^k = \breve{f}_0(\{\mathbf{u}_{nl}^k, \mathbf{b}_n^k, \theta_n^k\}_{n\in\mathcal{N},l\in\mathcal{L}(n)}) - \breve{g}_0(\{\mathbf{z}_{nl}^k, \mathbf{b}_n^k\}_{n\in\mathcal{N},l\in\mathcal{L}(n)})$; we stop both algorithms when either[53] $|\breve{h}_0^{k+1} - \breve{h}_0^k|/|\breve{h}_0^k| \leq 0.001$, or the algorithms run for a maximum of 25 iterations (see step 4 of both algorithms). To the best of our knowledge there is no algorithm for joint subchannel allocation and beamforming design for problem (5.33) in the literature. Thus, as a benchmark, we consider

[53] We use relative stopping criteria because it is scale-independent [213, Sec. 8.2].

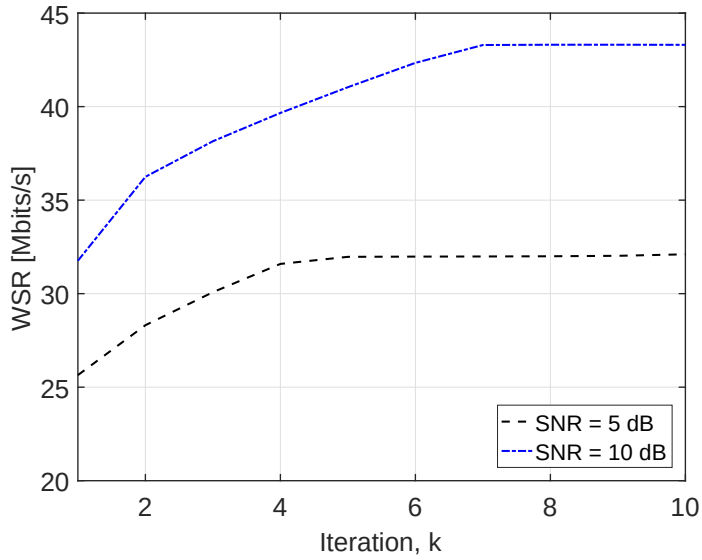


Fig. 5.3. WSR (Mbits/s) versus iteration for $\mathrm{SNR} = 5$ dB and 10 dB of Algorithm 5.2, [165] ©2017, IEEE.

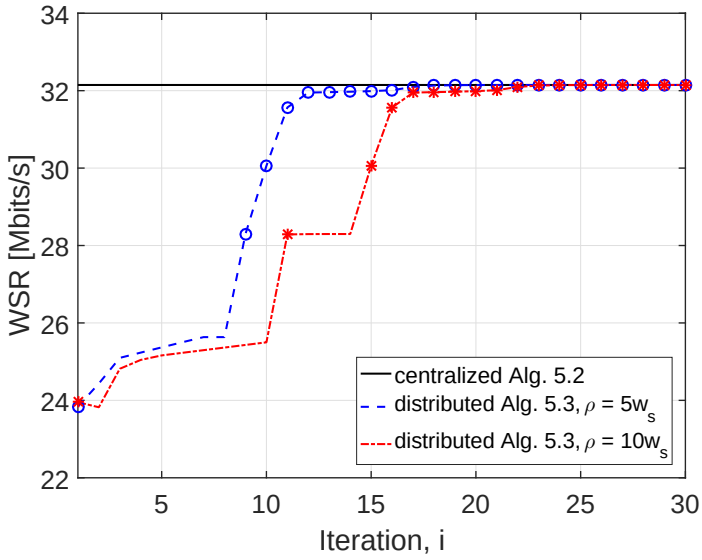


Fig. 5.4. WSR (Mbits/s) versus iteration for $\mathrm{SNR} = 5$ dB of Algorithm 5.3, [165] ©2017, IEEE.

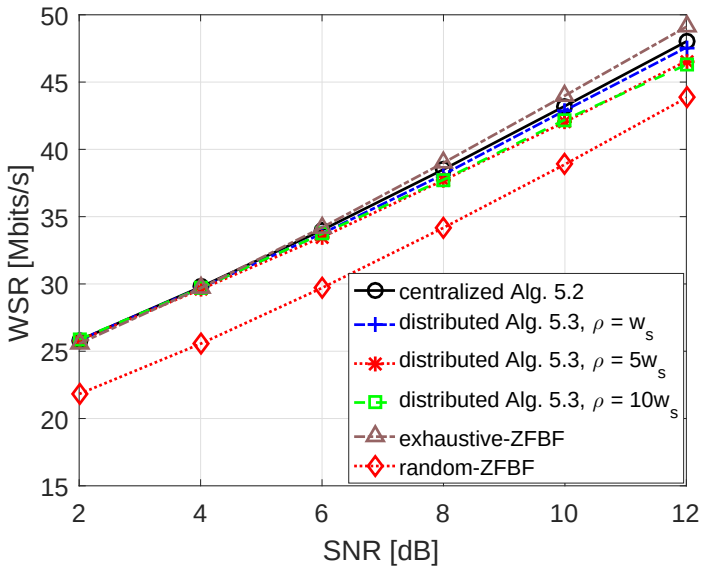


Fig. 5.5. Average WSR (Mbits/s) versus SNR, [165] ©2017, IEEE.

the zero-forcing beamforming (ZFBF) with user selection algorithm proposed in [214, Sec. VI.B], and we partition the S subchannels between the operators by exhaustive search and random allocation. In random subchannel allocations, the subchannels are partitioned between the operators so that each of them receive an equal number of subchannels. We refer to algorithm [214, Sec. VI.B] combined with an exhaustive search for partitioning subchannels as *exhaustive-ZFBF*; and the algorithm [214, Sec. VI.B] combined with a random channel allocation as *random-ZFBF*. We run all algorithms for 500 fading realizations.

Fig. 5.5 shows the average WSR (i.e., the average objective value of problem (5.33)), obtained by Algorithm 5.2 and Algorithm 5.3, versus SNR. In both algorithms if the solution $b^{\star}_{ns}$, for all $n \in \mathcal{N}$ and $s \in \mathcal{S}$, is not exactly binary, we round it to the nearest binary value to assign sth subchannel to a single operator. Results show that the proposed algorithms perform slightly better than *exhaustive-ZFBF* at low to medium SNR values, while *exhaustive-ZFBF* performs better at high SNR values. However, *exhaustive-ZFBF* partitions the S subchannels by exhaustive search, and its complexity is exponential in the number of subchannels. Thus, it quickly becomes intractable as the number of subchannels increases. On the other hand, both the proposed algorithms

outperform *random-ZFBF* for the entire range of SNR values. Furthermore, results show that the distributed Algorithm 5.3 achieves WSR values closer to the centralized Algorithm 5.2.

In order to see the effect of the penalty function $\zeta_n(\mathbf{b}_n)$, we also run Algorithm 5.2 by setting $\zeta_n(\mathbf{b}_n) = \sum_{s\in\mathcal{S}} b_{ns}(b_{ns}-1)$ for all $n \in \mathcal{N}$ [54]. Fig. 5.6(a) shows the average WSR obtained by Algorithm 5.2 with $\zeta_n(\mathbf{b}_n) = \sum_{s\in\mathcal{S}} \zeta_{ns}$, where ζ_{ns} is set to $b_{ns}\log(b_{ns})$ and $b_{ns}(b_{ns}-1)$. Fig. 5.6(b) shows the distribution of $\{b^{\star}_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ obtained by Algorithm 5.2 and Algorithm 5.3 for SNR = 10 dB. Fig. 5.6(a) shows that the average WSR obtained by both penalty functions are almost equal. However, the proposed algorithms with $\zeta_{ns} = b_{ns}\log(b_{ns})$ achieve the solution $\{b^{\star}_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ that is (almost) either near to zero or one, compared to that obtained by using $\zeta_{ns} = b_{ns}(b_{ns}-1)$, see Fig. 5.6(b).

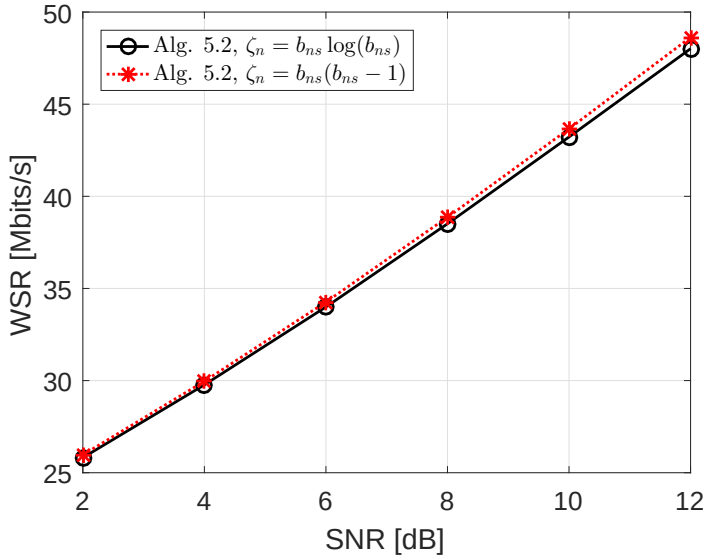
We now evaluate the performance of Algorithm 5.1. We suppose that the utility functions of users are given by $g_{nl}(a_{nl}) = \log(1+a_{nl})$ for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$. The parameter $A^{\max}$ is computed as described in [42, Sec. 4.2.1], and it is given by $A^{\max} = Sw_s\log_2(1+p_0^{\max}/N_0w_s)$. In fact, parameter $A^{\max}$ is an upper bound on the total transmission rate obtained by using S subchannels with the transmit power $p_0^{\max}$. The maximum per-unit price of the spectrum band is set to $q^{\max} = g_{nl}(A^{\max})$ [unit/MHz] [55]. The parameter $\mu^{\max}$ is set so that it contains the optimal value of $\overline{U}_n - U_n^0$ for all $n \in \mathcal{N}$ [43, Ch. 5]. Since $A^{\max}$ is an upper bound on the total transmission rate, we have $\overline{U}_n - U_n^0 \leq A^{\max} + q^{\max}B$ [56]. Thus, we set $\mu^{\max} = A^{\max} + q^{\max}B$. A value of disagreement point U_n^0 is obtained by solving a problem in Appendix 5, for all $n \in \mathcal{N}$. For simplicity, we assume that the transport layer storage reservoirs are saturated, i.e., there is always enough data waiting to be sent.

We run centralized and distributed versions of Algorithm 5.1 for $T^{\max} = 1000$ time slots (fading realizations). The centralized version of Algorithm 5.1 is obtained by solving step 4 of the algorithm by using Algorithm 5.2. The distributed version of Algorithm 5.1 is obtained by solving step 4 of the algorithm

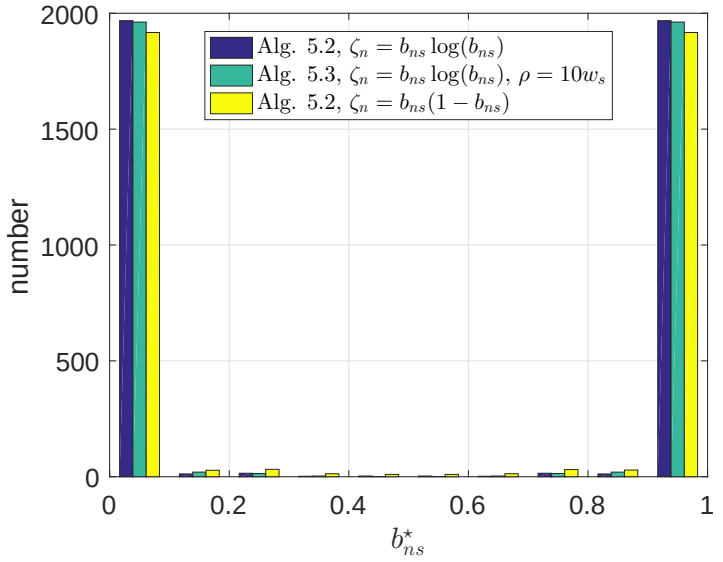
[54] The first order approximation of function $\zeta_n(\mathbf{b}_n) = \sum_{s\in\mathcal{S}} b_{ns}(b_{ns}-1)$ near an arbitrary positive point $\hat{\mathbf{b}}_n$ is $\hat{\zeta}_n(\mathbf{b}_n) = \sum_{s\in\mathcal{S}}(\hat{b}_{ns}^2 + 2\hat{b}_{ns}(b_{ns}-\hat{b}_{ns}) - b_{ns})$.

[55] The proposed dynamic algorithms are independent of the unit of price, and hence we work with a normalized unit. However, the results could be scaled with a proper price-unit value that operators mutually agree on.

[56] Under network stability $\sum_{l\in\mathcal{L}(n)} \overline{a}_{nl}(t) \leq A^{\max}$ for all $n \in \mathcal{N}$. Hence from expression (5.6), we have $\overline{U}_n \leq A^{\max} + q^{\max}B$ for all $n \in \mathcal{N}$.



(a)



(b)

Fig. 5.6. (a) Average WSR (Mbits/s) versus SNR for Algorithm 5.2; (b) Distribution of variables $\{b^\star_{ns}\}_{n\in\mathcal{N},s\in\mathcal{S}}$ for $\mathrm{SNR} = 10$ dB.

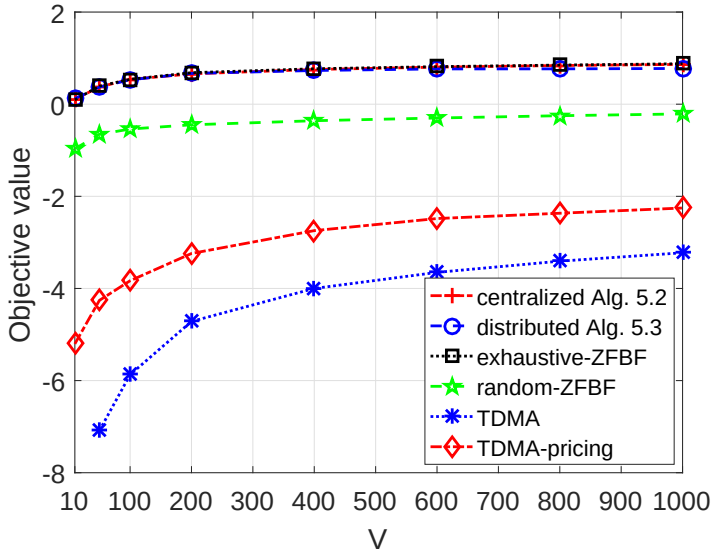
by using Algorithm 5.3. As a benchmark, we consider the *exhaustive-ZFBF* and *random-ZFBF* algorithms to solve step 4 of Algorithm 5.1. Furthermore, to evaluate the performance of the proposed algorithms, we also consider a time-division multiple access (TDMA) approach to solve step 4 of Algorithm 5.1; specifically, operators are allowed to fully access the common spectrum pool in alternating time slots. We run Algorithm 5.1 embedded with the TDMA approach, with and without considering the spectrum pricing strategy of Section 5.1.1, for $T^{\max} = 4000$ time slots[57]. In the simulations, we set SNR = 10 dB, and the ADMM penalty parameter $\rho = 10w_s$.

Fig. 5.7(a) shows the objective values of problem (5.7) versus parameter V. Results show that the objective value improves as V increases. Fig. 5.7(b) shows the time average network backlog, i.e., $1/T^{\max} \sum_{\tau=1}^{T^{\max}} \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(\tau)$ versus parameter V. Results show that the time average network backlog increases with the parameter V. For the TDMA based algorithm, the plots are drawn starting from $V = 50$, as below this value the algorithm is unable to obtain operators' utilities above the disagreement points[58]. Fig. 5.7(a) and Fig. 5.7(b) show that there is a trade-off between the achieved objective value of problem (5.7) and the network congestion. That is when the objective value increases, the network backlog is also increased. Furthermore, results show that both centralized and distributed versions of Algorithm 5.1 perform almost the same, and very close to those obtained by using the *exhaustive-ZFBF* based dynamic control algorithm. However, the proposed centralized and distributed versions of Algorithm 5.1 outperform control algorithms that are implemented using *random-ZFBF*, TDMA, and TDMA with pricing (TDMA-pricing) to solve step 4 of Algorithm 5.1. In addition, the TDMA based algorithms show that the pricing strategy of Section 5.1.1 improves the objective of problem (5.7) while increasing the queue backlog.

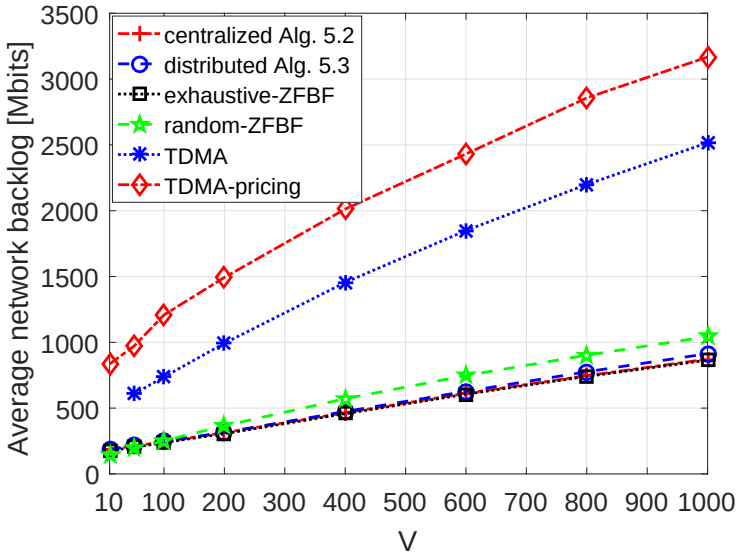
Fig. 5.8 shows the average profits of the operators $\{\overline{U}_n\}_{n\in\mathcal{N}}$ versus parameter V. Results show that the average profits of both the operators obtained by sharing their spectrum band with each other are greater than their disagreement points, i.e., $\overline{U}_n \geq U_n^0$ for all $n \in \mathcal{N}$. In other words, both operators gain in their

[57]We used larger averaging window as TDMA based algorithm requires slightly longer time to reach a near steady state [43].

[58]Recall that in Algorithm 5.1 the value of V puts emphasis on the objective of problem (5.7), see expression (5.19).



(a)



(b)

Fig. 5.7. (a) Objective values of problem (5.7) versus parameter V; (b) Average network backlog versus parameter V, [165] ©2017, IEEE.

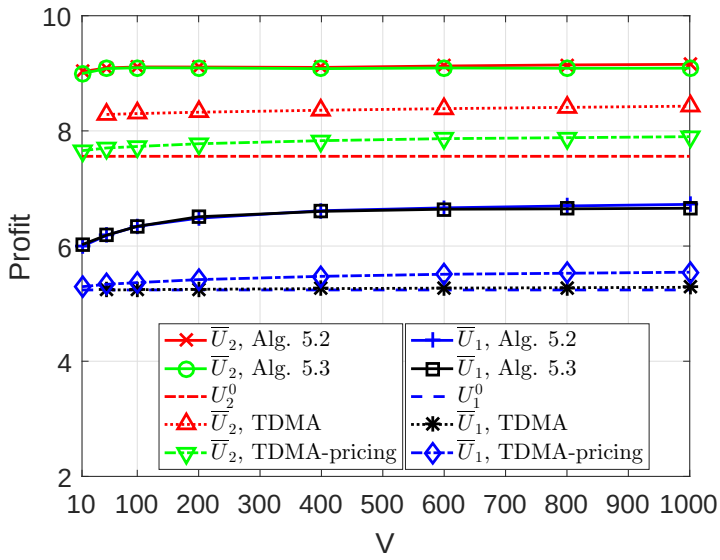


Fig. 5.8. Profits of the operators versus parameter V, [165] ©2017, IEEE.

profits by sharing their licensed spectrum band with each other, rather than using them exclusively. Furthermore, results show that the proposed algorithms outperform the control algorithms that are implemented by solving step 4 of Algorithm 5.1 with the TDMA approach. Moreover, results show that when Algorithm 5.1 is embedded with a TDMA approach, the operator with low spectrum demand (see operator 1) increases its profit by using the pricing strategy of Section 5.1.1. That is the operator with low spectrum demand gets paid for leasing its spectrum. Figure shows that the profit of each operator converges to its maximum value as V increases.

Fig. 5.9 shows the (sample path) evolution of the network queue backlog defined as $Q_{\text{sum}}(t) = \sum_{n\in\mathcal{N}} \sum_{l\in\mathcal{L}(n)} Q_{nl}(t)$. Here, step 4 of Algorithm 5.1 is solved by using Algorithm 5.2. The algorithm is run for $V = 10$ and 1000. Results show that the network queue backlog $Q_{\text{sum}}(t)$ increases until it reaches a certain value (e.g., for $V = 1000$ around 1400 Mbits), and then it oscillates. This is because of the negative drift property of the Lyapunov function [42, Ch. 4.4], and it ensures all queues are bounded and the network is stable.

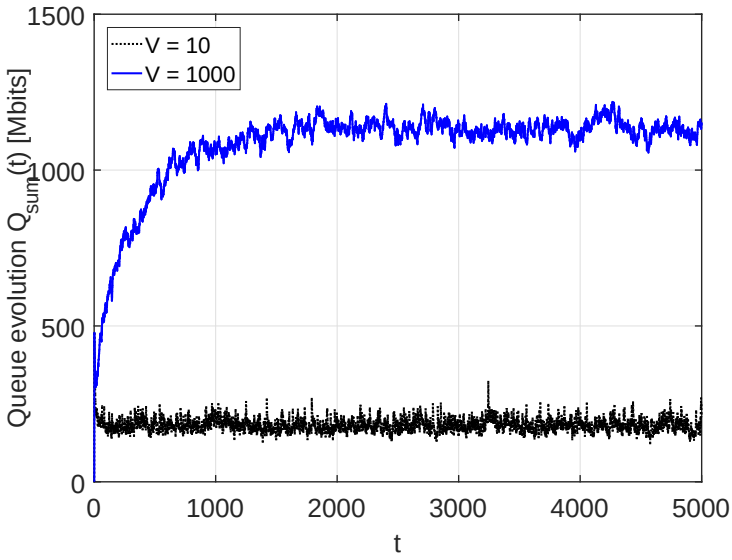


Fig. 5.9. Queue evolution $Q_{\mathrm{sum}}(t)$ for $V = 10$ and 1000, [165] ©2017, IEEE.

5.6 Summary and discussion

The problem of spectrum sharing between two operators in a *dynamic* network has been considered. We have allowed both operators to share (a fraction of) their licensed spectrum band with each other by forming a common spectrum band. The objective was to maximize the gain in profits of both operators by sharing their licensed spectrum bands rather than using them exclusively, while considering the fairness between the operators. This is modelled as a two-person bargaining problem, and cast as a stochastic optimization. To solve this problem, we have proposed centralized and distributed dynamic control algorithms by using Lyapunov optimization. At each time slot, the proposed algorithms perform the following tasks: 1) determine the spectrum price for the operators, 2) make flow control decisions for users' data, and 3) jointly allocate spectrum band to the operators and design transmit beamformers, which is known as RA. Since the RA problem is NP-hard, we have to rely on sequential convex programming to approximate its solution. To derive a distributed algorithm, we have used ADMM for solving the RA problem. Numerically, we have shown that the proposed distributed algorithm achieves almost the same performance as a

centralized one. Furthermore, it has been shown that operators gain in their profits by sharing their licensed spectrum band with each other, rather than using them exclusively; and there is a trade-off between the achieved profits of the operators and network congestion.

The proposed centralized algorithm can act as a tool to assess the gains achievable by sharing the spectrum between the operators, via offline simulations. The proposed distributed algorithm (or, approximate versions of it derived, e.g., by performing a finite number of iterations, exchanging quantized information, etc.) is more suitable for sharing the spectrum between the operators, as it requires a lower signaling overhead, compared with centralized one. In the proposed dynamic algorithm, at each time slot, the spectrum price for the operators is determined, and then a common spectrum pool is partitioned between the operators. That is, spectrum trading is performed at each time slot. It would be interesting to investigate the performance of the proposed algorithms by performing spectrum trading on a slower time scale (i. e., determining the spectrum price and portion of spectrum with long-term channel statistics). This would reduce required signaling overhead between the operators.

6 Conclusions and future work

6.1 Conclusions

The main focus of this thesis was to study radio resource management techniques for multicell MISO downlink networks. The first chapter included the motivation and the relevant literature associated with the scope of this thesis. In Chapter 2, a beamforming technique for multicell MISO downlink networks based on WSRMax criteria was studied. The WSRMax is central to many network optimization methods, and it is known to be an NP-hard problem. For this problem, a global solution method based on the BB technique was proposed. The proposed BB based algorithm computes a sequence of asymptotically tight upper and lower bounds. Novel bounding techniques via conic optimization were introduced. The proposed algorithm can be used to provide a performance benchmark to evaluate the performance loss of suboptimal algorithm. The proposed algorithm is not limited to only the WSRMax problem, it can be easily extended to maximize any system performance metric that can be expressed as a Lipschitz continuous and increasing function of an SINR.

In Chapter 3 robust resource allocation algorithms for multicell MISO downlink networks in the presence of imperfect CSI at BSs were studied. A bounded ellipsoidal model was used to model the CSI errors. Specifically, the problem of worst-case WSRMax in the presence of CSI errors was considered, and an optimal solution method was proposed by extending the BB based algorithm proposed in Chapter 2. The proposed optimal algorithm is a very useful tool for providing a performance benchmark for suboptimal algorithm that solves the worst-case WSRMax problem with CSI errors. As the convergence speed of the BB method can be slow for large networks, a fast but possibly suboptimal algorithm was also proposed using alternating optimization technique and sequential convex programming. The suboptimal algorithm involves solving SOCP and SDP at each iteration. In practice, the channel estimation errors may have a statistical distribution. Numerically, we have also shown how the proposed algorithms can be applied to a scenario with statistical channel errors.

In Chapter 4 distributed resource allocation algorithms for multicell MISO downlink networks were studied. The optimization criteria used were: P1)

minimization of the total transmission power subject to minimum SINR constraints of each user, and P2) SINR balancing subject to total transmit power constraints of BSs. The notion of consensus optimization was adopted, and decentralized algorithms for these problems were derived by using the ADMM. Problem P1 is not convex as such, however it can be reformulated so that it is convex. The proposed decentralized method for problem P1 converges to a global optimal value. Problem P2 is a quasiconvex problem. To the best of our knowledge there is no convergence theory to the ADMM method for a quasiconvex problem. Thus, the proposed decentralized method for problem P2 is not provably optimal. Numerical results have been provided to demonstrate the performance of the proposed distributed algorithms in comparison to the optimal centralized solutions

In Chapter 5 a resource allocation problem between two wireless operators in a dynamic network was studied. The objective was to maximize the gain in profits of both operators by sharing their licensed spectrum bands rather than using them exclusively, while considering the fairness between the operators. The notion of a two-person bargaining problem was adopted. Then the spectrum sharing problem was cast as a stochastic optimization problem, in order to capture the dynamics of the network. Lyapunov drift theory was used to solve the problem, and both centralized and distributed dynamic resource allocation algorithms have been proposed. Dynamic algorithms perform the following tasks at each time slot: 1) determine the spectrum price for the operators, 2) make flow control decisions for the users' data, and 3) jointly allocate the spectrum band to the operators and design the transmit beamformers, which is known as an RA problem. The RA problem leads to a general WSRMax problem, with an additional cost function on the spectrum use. The RA problem is NP-hard. A centralized algorithm for RA was derived using sequential convex programming. ADMM in conjunction with sequential convex programming was adopted to derive a distributed algorithm for RA. Numerically, it has been shown that with the proposed dynamic algorithms, the operators gain in their profits by sharing their licensed spectrum band with each other, rather than using them exclusively.

6.2 Future work

There are several possible research directions to extend the work carried out in this thesis such that it can be applied to the design and development of future-generation wireless networks. One important direction of interest is the development of radio resource management strategies for new mobile broadband millimeter wave (mm-wave) communications [215]. Due to the scarcity of the radio spectrum in the sub-6 GHz spectrum band (i.e., below 6 GHz band), recently, there has been growing interest in the exploration of the underutilized mm-wave frequency spectrum (30-300 GHz) for future-generation wireless networks. The millimeter wave frequency band can offer multi-GHz bandwidth, hence this increases the capacity of communications networks. However, in terms of propagation characteristics there are fundamental differences between mm-wave and sub-6GHz bands. For example, the mm-wave band suffers from high attenuation, and hence highly directional antennas are required between the transmitter and receiver. Therefore, by exploiting the intrinsic characteristics of mm-wave propagation, it would be interesting to investigate efficient resource management solutions, extending the methods presented in this thesis (e.g., optimal method in Chapter 2, the robust methods presented in Chapter 3, etc.).

In Chapters 4 and 5 we have used the ADMM to derive distributed algorithms. It is assumed that the steps of the distributed algorithm are synchronized, i.e., the algorithm progress only when all the BSs solve their respective subproblems. In practice, the processing speed of the BSs and the communication delay between them can be different, and these facts affect the overall speed of such a synchronous distributed algorithm. It is envisioned that the future-generation wireless networks will consist of densely deployed BSs. Thus, it is of great interest to design distributed algorithms that do not have to wait for the latest updates from all the BSs, and can even work with outdated information by using methods such as asynchronous ADMM [192], Fast-Lipschitz optimization [216], etc.

It is well known that the performance of an ADMM based algorithm depends on the choice of the penalty parameter [113]. For the problem P1 in Chapter 4, we have proposed a heuristic method to set the penalty parameter, and numerical results show that with this heuristic value the algorithm converges with high accuracy on the optimal value within first few iterations. For the problem

P2 in Chapter 4 and the problem in Chapter 5, we have set the ADMM penalty parameters with arbitrary values. The performance of these distributed algorithms could be improved, further, with careful selection of these parameters. Hence, it would be interesting to investigate a method to select penalty parameters that yield high performance for the ADMM based distributed algorithms.

The extension of the dynamic algorithms presented in Chapter 5 for more than two operators, including micro operators, is also another possible future direction for research. In Chapter 5, we used a cooperative game theoretic approach (i.e., it was assumed that both operators have full knowledge of each others' preferences). However, an operator can be greedy, thus it would also be interesting to investigate non-cooperative dynamic network control mechanisms; or, methods that require negligible signaling overhead and simplify the implementation of spectrum sharing [217, 218]. Further extensions may also include incorporating operators' spectrum demand learning and infrastructure sharing between operators.

References

1. GSMA (2017) The mobile economy 2017. Technical report, GSM Association. URI: http://www.gsma.com/mobileeconomy/.
2. Cisco (2017). Cisco visual networking index: Global mobile data traffic forecast update, 2016-2021.
3. ITU-T (2012) Impact of M2M communications and non-M2M mobile data applications on mobile networks. Technical report, International Telecommunication Union.
4. Demestichas P, Georgakopoulos A, Karvounas D, Tsagkaris K, Stavroulaki V, Lu J, Xiong C & Yao J (2013) 5G on the horizon: Key challenges for the radio-access network. IEEE Trans. Veh. Technol. 8(3): 47–53.
5. Ericsson (2016) 5G radio access. Technical report, Ericsson White Paper Uen 284 23-3204 Rev C.
6. Gupta A & Jha RK (2015) A survey of 5G network: Architecture and emerging technologies. IEEE Access 3: 1206–1232.
7. Dahlman E, Mildh G, Parkvall S, Peisa J, Sachs J & Selen Y (2014) 5G radio access. Ericsson Review 6: 2–6.
8. Cisco (2016). Radio resource management white paper. URI: http://www.cisco.com/c/en/us/td/docs/wireless/controller/technotes/8-3/b_RRM_White_Paper.pdf.
9. Tse D & Viswanath P (2005) Fundamentals of Wireless Communication. Cambridge University Press, Cambridge, UK.
10. Kim J & Lee I (2015) 802.11 WLAN: history and new enabling MIMO techniques for next generation standards. IEEE Communications Magazine 53(3): 134–140.
11. Li Q, Li G, Lee W, i Lee M, Mazzarese D, Clerckx B & Li Z (2010) MIMO techniques in WiMAX and LTE: a feature overview. IEEE Commun. Mag. 48(5): 86–92.
12. 3rd Generation Partnership Project (3GPP); Technical Specification Group Radio Access Network (2010) Requirements for evolved UTRA (E-UTRA) and evolved UTRAN (E-UTRAN) (TR 25.913 version 9.0.0 (release 9)). Technical report, 3rd Generation Partnership Project (3GPP).
13. Goldsmith A (2005) Wireless Communications. Cambridge University Press, New York, USA.
14. Winters J (1987) On the capacity of radio communication systems with diversity in a Rayleigh fading environment. IEEE J. Select. Areas Commun. 5(5): 871–878.
15. Foschini GJ (1996) Layered space–time architecture for wireless communication in a fading environment when using multi-element antennas. Bell Labs Tech. J. 1(2): 41–59.
16. Telatar E (1999) Capacity of multi-antenna Gaussian channels. European Trans. Telecommun. 10(6): 585–595.
17. Foschini GJ & Gans MJ (1998) On limits of wireless communications in a fading environment when using multiple antennas. Wireless Pers. Commun., Kluwer 6(3): 311–335.

18. Liu YF, Dai YH & Luo ZQ (2011) Coordinated beamforming for MISO interference channel: Complexity analysis and efficient algorithms. IEEE Trans. Signal Processing 59(3): 1142–1157.
19. Biguesh M & Gershman AB (2006) Training-based MIMO channel estimation: A study of estimator tradeoffs and optimal training signals. IEEE Trans. Signal Processing 54(3): 884–893.
20. Li Y, Cimini LJ & Sollenberger NR (1998) Robust channel estimation for OFDM systems with rapid dispersive fading channels. IEEE Trans. Commun. 46(7): 902–915.
21. Biguesh M & Gershman AB (2004) Downlink channel estimation in cellular systems with antenna arrays at base stations using channel probing with feedback. EURASIP J. Advances Signal Processing 2004(9): 1330–1339.
22. Sun Q, Cox DC, Huang HC & Lozano A (2002) Estimation of continuous flat fading MIMO channels. IEEE Trans. Wireless Commun. 1(4): 549–553.
23. Guillaud M, Slock DTM & Knopp R (2005) A practical method for wireless channel reciprocity exploitation through relative calibration. In: Proc. Int. Symp. Signal Processing and its Applications, volume 1, pp. 403–406.
24. FCC (2015). Amendment of the commission's rules with regard to commercial operations in the 3550-3650 MHz band, report and order and second further notice of proposed rulemaking. GN Docket No. 12-354.
25. METIS (2013). Intermediate description of the spectrum needs and usage principles. Doc. No. ICT-317669-METIS/D5.1.
26. Jorswieck EA *et al* (2010) Resource sharing in wireless networks: The SAPHYRE approach. In: Proc. Fut. Net. Mob. Summit, pp. 1–8.
27. FCC (2003). Notice for proposed rulemaking: Facilitating opportunities for flexible, efficient, and reliable spectrum use employing cognitive radio technologies. ET Docket No. 03-108.
28. Akyildiz IF, Lee WY, Vuran MC & Mohanty S (2008) A survey on spectrum management in cognitive radio networks. IEEE Commun. Mag. 46(4): 40–48.
29. Boyd S & Vandenberghe L (2004) Convex Optimization. Cambridge University Press, Cambridge, UK.
30. Luo Z & Zhang S (2008) Dynamic spectrum management: Complexity and duality. IEEE J. Select. Topics Signal Processing 2(1): 57–73.
31. Cendrillon R, Yu W, Moonen M, Verlinden J & Bostoen T (2006) Optimal multiuser spectrum balancing for digital subscriber lines. IEEE Trans. Commun. 54(5): 922–933.
32. Horst R, Pardolos P & Thoai N (2000) Introduction to Global Optimization, volume 48. Kluwer Academic Publishers, Dordrecht, Boston, London, second edition.
33. Audet C, Hansen P & Savard G (2005) Essays and Surveys in Global Optimization. Springer Science + Business Media, Inc, 233 Spring street, NY 10013, USA.
34. Niesen U, Shah D & Wornell GW (2009) Adaptive alternating minimization algorithms. IEEE Trans. Inform. Theory 55(3): 1423–1429.
35. Boyd S (2007) Sequential convex programming. [Online]. Available: `http://www.stanford.edu/class/ee364b/lectures/seq_slides.pdf`.

36. Lipp T & Boyd S (2016) Variations and extension of the convex-concave procedure. J. Optim. Eng., Springer 17(2): 263–287.
37. Weeraddana PC, Codreanu M, Latva-aho M, Ephremides A & Fischione C (2012) Weighted sum-rate maximization in wireless networks: A review. Found. Trends Net. 6: 1–163.
38. Chiang M & Bell J (2004) Balancing supply and demand of bandwidth in wireless cellular networks: Utility maximization over powers and rates. In: Proc. IEEE Int. Conf. on Comp. Commun., volume 4, pp. 2800–2811. Hong Kong.
39. Chiang M (2005) Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control. IEEE J. Select. Areas Commun. 23(1): 104–116.
40. Johansson B, Soldati P & Johansson M (2006) Mathematical decomposition techniques for distributed cross-layer optimization of data networks. IEEE J. Select. Areas Commun. 24(8): 1535–1547.
41. Palomar DP & Chiang M (2007) Alternative distributed algorithms for network utility maximization: Framework and applications. IEEE Trans. Automat. Contr. 52(12): 2254–2269.
42. Georgiadis L, Neely MJ & Tassiulas L (2006) Resource allocation and cross-layer control in wireless networks. Found. Trends Net. 1(1): 1–144.
43. Neely MJ (2010) Stochastic Network Optimization with Application to Communication and Queueing Systems, volume 7 of *Synthesis Lectures on Communication Networks*. Morgan & Claypool, San Rafael, CA.
44. Lin X & Shroff NB (2004) Joint rate control and scheduling in multihop wireless networks. In: Proc. Int. Conf. Dec. Contr., volume 5, pp. 1484–1489. Atlantis, Paradise Island, Bahamas.
45. Lin X & Shroff NB (2005) The impact of imperfect scheduling on cross-layer rate control in wireless networks. In: Proc. IEEE Int. Conf. on Comp. Commun., volume 3, pp. 1804–1814. Miami, USA.
46. Neely MJ (2006) Super-fast delay tradeoffs for utility optimal fair scheduling in wireless networks. IEEE J. Select. Areas Commun. 24(8): 1–12.
47. Yu W, Lui R & Cendrillon R (2004) Dual optimization methods for multiuser orthogonal frequency division multiplex systems. In: Proc. IEEE Global Telecom. Conf., volume 1, pp. 225–229.
48. Chan VMK & Yu W (2005) Joint multiuser detection and optimal spectrum balancing for digital subscriber lines. In: Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, volume 3, pp. 333–336.
49. Cendrillon R, Huang J, Chiang M & Moonen M (2007) Autonomous spectrum balancing for digital subscriber lines. IEEE Trans. Signal Processing 55(8): 4241–4257.
50. Lui R & Yu W (2005) Low-complexity near-optimal spectrum balancing for digital subscriber lines. In: Proc. IEEE Int. Conf. Commun., pp. 1947–1951. Seoul, Korea.
51. Hoo LMC, Halder B, Tellado J & Cioffi M (2004) Multiuser transmit optimization for multiuser broadcast channels: Asymptotic FDMA capacity region and algorithms. IEEE Trans. Commun. 52(6): 922–930.

52. Yu W & Cioffi M (2002) FDMA capacity of gaussian multiple-access channels with ISI. IEEE Trans. Commun. 50(1): 102–111.
53. Shen Z, Andrews JG & Evans BL (2003) Optimal power allocation in multiuser OFDM systems. In: Proc. IEEE Global Telecom. Conf., volume 1, pp. 337–341.
54. Xiao JJ & Luo ZQ (2007) Multiterminal source-channel communication over an orthogonal multiple-access channel. IEEE Trans. Inform. Theory 53(9): 3255–3264.
55. Xu Y, Le-Ngoc T & Panigrahi S (2008) Global concave minimization for optimal spectrum balancing in multi-user DSL networks. IEEE Trans. Signal Processing 56(7): 2875–2885.
56. Al-Shatri H & Weber T (2010) Optimizing power allocation in interference channels using D.C. programming. In: Proc. IEEE Int. Symp. Modeling Optim. Mobile, Ad Hoc Wireless Networks, pp. 360–366.
57. Tsiaflakis P, Vangorp J, Moonen M & Verlinden J (2007) A low complexity optimal spectrum balancing algorithm for digital subscriber lines. Signal Processing, Elsevier 87(7): 1735–1753.
58. Weeraddana PC, Codreanu M, Latva-aho M & Ephremides A (2010) Weighted sum-rate maximization for a set of interfering links via branch and bound. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1896–1900. Pacific Grove, CA, USA.
59. Weeraddana PC, Codreanu M, Latva-aho M & Ephremides A (2011) Weighted sum-rate maximization for a set of interfering links via branch and bound. IEEE Trans. Signal Processing 59(8): 3977–3996.
60. Qian L, Zhang YJA & Huang J (2009) MAPEL: Achieving global optimality for a non-convex wireless power control problem. IEEE Trans. Wireless Commun. 8(3): 1553–1563.
61. Tuy H (1986) A general deterministic approach to global optimization VIA d.c. programming. In: Hiriart-Urruty JB (ed.) Fermat Days 85: Mathematics for Optimization, volume 129 of *North-Holland Mathematics Studies*, pp. 273–303. North-Holland.
62. Phuong NTH & Tuy H (2003) A unified monotonic approach to generalized linear fractional programming. J. Global Optim., Springer pp. 229–259. Kluwer Academic Publishers.
63. Rubinov A, Tuy H & Mays H (2001) An algorithm for monotonic global optimization problems. Optimization 49(3): 205–221.
64. Björnson E & Jorswieck E (2013) Optimal resource allocation in coordinated multi-cell systems. Found. Trends Commun. Inform. Th. 9(2-3): 111–381.
65. Jorswieck EA & Larsson EG (2010) Monotonic optimization framework for the two-user MISO interference channel. IEEE Trans. Commun. 58(7): 2159–2168.
66. Jorswieck EA, Larsson EG & Danev D (2008) Complete characterization of the pareto boundary for the MISO interference channel. IEEE Trans. Signal Processing 56(10): 5292–5296.
67. Liu L, Zhang R & Chua KC (2012) Achieving global optimality for weighted sum-rate maximization in the K-User gaussian interference channel with multiple antennas. IEEE Trans. Wireless Commun. 11(5): 1933–1945.
68. Zhang R & Cui S (2010) Cooperative interference management with MISO beamforming. IEEE Trans. Signal Processing 58(10): 5450–5458.

69. Mohseni M, Zhang R & Cioffi JM (2006) Optimized transmission for fading multiple-access and broadcast channels with multiple antennas. IEEE J. Select. Areas Commun. 24(8): 1627–1639.
70. Zhang R, Liang YC, Chai CC & Cui S (2009) Optimal beamforming for two-way multi-antenna relay channel with analogue network coding. IEEE J. Select. Areas Commun. 27(5): 699–712.
71. Brehmer J & Utschick W (2009) Utility maximization in the multi-user MISO downlink with linear precoding. In: Proc. IEEE Int. Conf. Commun.
72. Brehmer J & Utschick W (2009) Nonconcave utility maximisation in the MIMO broadcast channel. EURASIP J. Advances Signal Processing 2009, Article ID 645041: 1–13.
73. Utschick W & Brehmer J (2012) Monotonic optimization framework for coordinated beamforming in multicell networks. IEEE Trans. Signal Processing 60(4): 1899–1909.
74. Tuy H, Al-Khayyal F & Thach P (2005) Monotonic optimization: Branch and cut methods. In: Audet C, Hansen P & Savard G (eds.) Essays and Surveys in Global Optimization, pp. 39–78. Springer US, Boston, MA.
75. Björnson E, Zheng G, Bengtsson M & Ottersten B (2012) Robust monotonic optimization framework for multicell MISO systems. IEEE Trans. Signal Processing 60(5): 2508–2523.
76. Balakrishnan V, Boyd S & Balemi S (1991) Branch and bound algorithm for computing the minimum stability degree of parameter-dependent linear systems. Int. J. Robust Nonlinear Contr. 1(4): 295–317.
77. Tan CW, Chiang M & Srikant R (2013) Fast algorithms and performance bounds for sum rate maximization in wireless networks. IEEE/ACM Trans. Networking 21(3): 706–719.
78. Julian D, Chiang M, O'Neil D, & Boyd S (2002) QoS and fairness constrained convex optimization of resource allocation for wireless cellular and ad-hoc networks. In: Proc. IEEE Int. Conf. on Comp. Commun., volume 2, pp. 477–486. New York, USA.
79. Chiang M, Tan CW, Palomar DP, O'Neill D & Julian D (2007) Power control by geometric programming. IEEE Trans. Wireless Commun. 6(7): 2640–2651.
80. Boyd S, Kim SJ, Vandenberghe L & Hassibi A (2007) A tutorial on geometric programming. Optimization and Engineering 8(1): 67–127.
81. Chiang M (2005) Geometric programming for communication systems. Found. Trends Commun. Inform. Th. 2(1-2): 1–154.
82. Venturino L, Prasad N & Wang X (2009) Coordinated scheduling and power allocation in downlink multicell OFDMA networks. IEEE Trans. Veh. Technol. 58(6): 2835–2848.
83. Codreanu M, Tölli A, Juntti M & Latva-aho M (2007) Joint design of Tx-Rx beamformers in MIMO downlink channel. IEEE Trans. Signal Processing 55(9): 4639–4655.
84. Viswanath P & Tse DNC (2003) Sum capacity of the vector Gaussian broadcast channel and uplink-downlink duality. IEEE Trans. Inform. Theory 49(8): 1912–1921.

85. Tölli A, Codreanu M & Juntti M (2008) Cooperative MIMO-OFDM cellular system with soft handover between distributed base station antennas. IEEE Trans. Wireless Commun. 7(4): 1428–1440.
86. Palomar DP (2003) A unified framework for communications through MIMO channels. Ph.D. thesis, Department of Signal Theory and Communications, Technical University of Catalonia, Barcelona, Spain.
87. Lobo MS, Vandenberghe L, Boyd S & Lebret H (1998) Applications of second-order cone programming. Linear Algebra and Applications 284: 193–228.
88. Tran LN, Hanif MF, Tölli A & Juntti M (2012) Fast converging algorithm for weighted sum rate maximization in multicell MISO downlink. IEEE Signal Processing Lett. 19(12): 872–875.
89. Christensen SS, Agarwal R, Carvalho E & Cioffi J (2008) Weighted sum-rate maximization using weighted MMSE for MIMO-BC beamforming design. IEEE Trans. Wireless Commun. 7(12): 4792–4799.
90. Shenouda M & Davidson T (2008) On the design of linear transceivers for multiuser systems with channel uncertainty. IEEE J. Select. Areas Commun. 26(6): 1015–1024.
91. Zhang X, Palomar DP & Ottersten B (2008) Statistically robust design of linear MIMO transceivers. IEEE Trans. Signal Processing 56(8): 3678–3689.
92. Lorenz RG & Boyd SP (2005) Robust minimum variance beamforming. IEEE Trans. Signal Processing 53(5): 1684–1696.
93. Shenouda M & Davidson TN (2007) Convex conic formulations of robust downlink precoder designs with quality of service constraints. IEEE J. Select. Topics Signal Processing 1(4): 714–724.
94. Yang K, Wu Y, Huang J, Wang X & Verdu S (2008) Distributed robust optimization for communication networks. In: Proc. IEEE Int. Conf. on Comp. Commun., volume 1. Phoenix, AZ, USA.
95. Vorobyov SA, Gershman AB & Luo ZQ (2003) Robust adaptive beamforming using worst-case performance optimization: A solution to the signal mismatch problem. IEEE Trans. Signal Processing 51(2): 313–324.
96. Mutapcic A, Kim SJ & Boyd S (2007) A tractable method for robust downlink beamforming in wireless communications. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1224–1228.
97. Zheng G, Wong KK & Ng TS (2008) Robust linear MIMO in the downlink: A worst-case optimization with ellipsoidal uncertainty regions. EURASIP J. Advances Signal Processing 2008: 1–15.
98. Shen C, Chang T, Wang K, Qiu Z & Chi C (2012) Distributed robust multicell coordinated beamforming with imperfect CSI: An ADMM approach. IEEE Trans. Signal Processing 60(6): 2988–3003.
99. Yang K, Huang J, Wu Y, Wang X & Chiang M (2014) Distributed robust optimization (DRO), part I: framework and example. J. Optim. Eng., Springer 15(1): 35–67.
100. Lee HH, Ko YC & Yang HC (2013) On robust weighted sum rate maximization for MIMO interfering broadcast channels with imperfect channel knowledge. IEEE Commun. Lett. 17(6): 1156–1159.

101. Jose J, Prasad N, Khojastepour M & Rangarajan S (2011) On robust weighted-sum rate maximization in MIMO interference networks. In: Proc. IEEE Int. Conf. Commun., pp. 1–6.
102. Tajer A, Prasad N & Wang X (2011) Robust linear precoder design for multi-cell downlink transmission. IEEE Trans. Signal Processing 59(1): 235–251.
103. Hanif MF, Tran LN, Tölli A, Juntti M & Glisic S (2013) Efficient solutions for weighted sum rate maximization in multicellular networks with channel uncertainties. IEEE Trans. Signal Processing 61(22): 5659–5674.
104. Wajid I, Pesavento M, Eldar YC & Ciochina D (2013) Robust downlink beamforming with partial channel state information for conventional and cognitive radio networks. IEEE Trans. Signal Processing 61(14): 3656–3670.
105. Vucic N & Boche H (2009) Robust QoS-constrained optimization of downlink multiuser MISO systems. IEEE Trans. Signal Processing 57(2): 714–725.
106. Vucic N, Boche H & Shi S (2009) Robust transceiver optimization in downlink multiuser MIMO systems. IEEE Trans. Signal Processing 57(9): 3576–3587.
107. Wang J & Palomar DP (2010) Robust MMSE precoding in MIMO channels with pre-fixed receivers. IEEE Trans. Signal Processing 58(11): 5802–5818.
108. Wijewardhana UL, Codreanu M, Latva-aho M & Ephremides A (2014) A robust beamformer design for underlay cognitive radio networks using worst case optimization. EURASIP J. Wireless Comm. and Netw. 2014(37).
109. Kim SJ, Magnani A, Mutapcic A, Boyd SP & Luo ZQ (2008) Robust beamforming via worst-case SINR maximization. IEEE Trans. Signal Processing 56(4): 1539–1547.
110. Kim SJ & Giannakis GB (2011) Optimal resource allocation for MIMO ad hoc cognitive radio networks. IEEE Trans. Inform. Theory 57(5): 3117–3131.
111. Ding L, Melodia T, Batalama SN & Matyjas JD (2015) Distributed resource allocation in cognitive and cooperative ad hoc networks through joint routing, relay selection and spectrum allocation. Computer Networks, Elsevier 83: 315–331.
112. Iltis RA, Kim SJ & Hoang DA (2006) Noncooperative iterative MMSE beamforming algorithms for ad hoc networks. IEEE Trans. Commun. 54(4): 748–759.
113. Boyd S, Parikh N, Chu E, Peleato B & Eckstein J (2011) Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends Mach. Learn. 3(1): 1–122.
114. Boyd S, Xiao L, Mutapcic A & Mattingley J (2008) Notes on decomposition methods: course reader for convex optimization II, Stanford. [Online]. Available: `http://web.stanford.edu/class/ee364b/lectures/decomposition_notes.pdf`.
115. Pischella M & Belfiore JC (2008) Distributed weighted sum throughput maximization in multi-cell wireless networks. In: Proc. IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun., pp. 1–5.
116. Park SH, Park H & Lee I (2010) Distributed beamforming techniques for weighted sum-rate maximization in MISO interference channels. IEEE Commun. Lett. 14(12): 1131–1133.
117. Rossi M, Tulino AM, Simeone O & Haimovich AM (2011) Non-convex utility maximization in gaussian MISO broadcast and interference channels. In: Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, pp. 2960–2963.

118. Choi HJ, Park SH, Lee SR & Lee I (2012) Distributed beamforming techniques for weighted sum-rate maximization in MISO interfering broadcast channels. IEEE Trans. Wireless Commun. 11(4): 1314–1320.
119. Venturino L, Prasad N & Wang X (2010) Coordinated linear beamforming in downlink multi-cell wireless networks. IEEE Trans. Wireless Commun. 9(4): 1451–1461.
120. Shi Q, Razaviyayn M, Luo ZQ & He C (2011) An iteratively weighted MMSE approach to distributed sum-utility maximization for a MIMO interfering broadcast channel. IEEE Trans. Signal Processing 59(9): 4331–4340.
121. Yoo T & Goldsmith A (2005) Optimality of zero-forcing beamforming with multiuser diversity. In: Proc. IEEE Int. Conf. Commun., volume 1, pp. 542–546.
122. Wiesel A, Eldar YC & Shamai S (2008) Zero-forcing precoding and generalized inverses. IEEE Trans. Signal Processing 56(9): 4409–4418.
123. Bengtsson M & Ottersten B (1999) Optimal downlink beamforming using semidefinite optimization. In: Proc. Annual Allerton Conf. Commun., Contr., Computing, pp. 987–996. Urbana-Champaign, IL.
124. Rashid-Farrokhi F, Liu KJR & Tassiulas L (1998) Transmit beamforming and power control for cellular wireless systems. IEEE J. Select. Areas Commun. 16(8): 1437–1450.
125. Visotsky E & Madhow U (1999) Optimum beamforming using transmit antenna arrays. In: Proc. IEEE Veh. Technol. Conf., volume 1, pp. 851 – 856. Houston, TX.
126. Shi Q, Razaviyayn M, Hong M & Luo ZQ (2012) SINR constrained beamforming for a MIMO multi-user downlink system. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1991–1995. Pacific Grove, California.
127. Wong KK, Zheng G & Ng TS (2005) Convergence analysis of downlink MIMO antenna systems using second-order cone programming. In: Proc. IEEE Veh. Technol. Conf., volume 1, pp. 492–496.
128. Pennanen H, Tölli A & Latva-aho M (2011) Decentralized coordinated downlink beamforming via primal decomposition. IEEE Signal Processing Lett. 18(11): 647–650.
129. Tölli A, Pennanen H & Komulainen P (2011) Decentralized minimum power multi-cell beamforming with limited backhaul signaling. IEEE Trans. Wireless Commun. 10(2): 570–580.
130. Boyd S (2007) Subgradient methods. [Online]. Available: `http://www.stanford.edu/class/ee364b/lectures/subgrad_method_slides.pdf`.
131. Dahroujand H & Yu W (2010) Coordinated beamforming for the multicell multi-antenna wireless system. IEEE Trans. Wireless Commun. 9(5): 1748–1759.
132. Wiesel A, Eldar YC & Shamai S (2006) Linear precoding via conic optimization for fixed MIMO receivers. IEEE Trans. Signal Processing 54(1): 161–176.
133. Nguyen DHN & Le-Ngoc T (2011) Multiuser downlink beamforming in multicell wireless systems: A game theoretical approach. IEEE Trans. Signal Processing 59(7): 3326–3338.
134. Schubert M & Boche H (2004) Solution of the multiuser downlink beamforming problem with individual SINR constraints. IEEE Trans. Veh. Technol. 53(1): 18–28.

135. Boche H & Schubert M (2003) Optimal multi-user interference balancing using transmit beamforming. Wireless Pers. Commun., Kluwer 26(4): 305–324.
136. Schubert M & Boche H (2004) Comparison of ℓ_∞-norm and ℓ_1-norm optimization criteria for SIR-balanced multi-user beamforming. Signal Processing, Elsevier 84(2): 367–378.
137. Tölli A, Codreanu M & Juntti M (2007) Minimum SINR maximization for multiuser MIMO downlink with per BS power constraints. In: Proc. IEEE Wireless Commun. and Networking Conf., pp. 1144–1149. Hong Kong.
138. Huang Y, Zheng G, Bengtsson M, Wong KK, Yang L & Ottersten B (2011) Distributed multicell beamforming with limited intercell coordination. IEEE Trans. Signal Processing 59(2): 728–738.
139. Jorswieck EA, Badia L, Fahldieck T, Karipidis E & Luo J (2014) Spectrum sharing improves the network efficiency for cellular operators. IEEE Commun. Mag. 52(3): 129–136.
140. Lindblom J & Larsson EG (2012) Does non-orthogonal spectrum sharing in the same cell improve the sum-rate of wireless operators? In: Proc. IEEE Works. on Sign. Proc. Adv. in Wirel. Comms., pp. 6–10.
141. Gangula R, Gesbert D, Lindblom J & Larsson EG (2013) On the value of spectrum sharing among operators in multicell networks. In: Proc. IEEE Veh. Technol. Conf., pp. 1–5.
142. Middleton G, Hooli K, Tölli A & Lilleberg J (2006) Inter-operator spectrum sharing in a broadband cellular network. In: Proc. IEEE Int. Symp. Spread Spectrum Techniques and Applications, pp. 376–380.
143. Bennis M & Lilleberg J (2007) Inter base station resource sharing and improving the overall efficiency of B3G systems. In: Proc. IEEE Veh. Technol. Conf., pp. 1494–1498.
144. Kamal H, Coupechoux M & Godlewski P (2009) Inter-operator spectrum sharing for cellular networks using game theory. In: Proc. IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun., pp. 425–429.
145. Anchora L, Badia L, Karipidis E & Zorzi M (2012) Capacity gains due to orthogonal spectrum sharing in multi-operator LTE cellular networks. In: Proc. Int. Symp. Wireless Commun. Systems, pp. 286–290.
146. Suris JE, DaSilva LA, Han Z & MacKenzie AB (2007) Cooperative game theory for distributed spectrum sharing. In: Proc. IEEE Int. Conf. Commun., pp. 5282–5287.
147. Luoto P, Pirinen P, Bennis M, Samarakoon S, Scott S & Latva-aho M (2015) Co-primary multi-operator resource sharing for small cell networks. IEEE Trans. Wireless Commun. 14(6): 3120–3130.
148. Weiss TA & Jondral FK (2004) Spectrum pooling: An innovative strategy for the enhancement of spectrum efficiency. IEEE Commun. Mag. 42(3): S8–14.
149. Si P, Ji H, Yu FR & Leung VCM (2010) Optimal cooperative internetwork spectrum sharing for cognitive radio systems with spectrum pooling. IEEE Trans. Veh. Technol. 59(4): 1760–1768.
150. Osborne MJ & Rubinstein A (1994) A course in game theory. MIT Press.
151. Han Z, Ji Z & Liu KJR (2005) Fair multiuser channel allocation for OFDMA networks using Nash bargaining solutions and coalitions. IEEE Trans. Commun.

53(8): 1366–1376.
152. Leshem A & Zehavi E (2006) Bargaining over the interference channel. In: Proc. IEEE Int. Symp. Inform. Theory, pp. 2225–2229.
153. Ka Z, Ho M & Gesbert D (2008) Spectrum sharing in multiple-antenna channels: A distributed cooperative game theoretic approach. In: Proc. IEEE Int. Symp. Pers., Indoor, Mobile Radio Commun.
154. Larsson EG & Jorswieck EA (2008) Competition versus cooperation on the MISO interference channel. IEEE J. Select. Areas Commun. 26(7): 1059–1069.
155. Luo J *et al* (2012) Transmit beamforming for inter-operator spectrum sharing: From theory to practice. In: Proc. Int. Symp. Wireless Commun. Systems, pp. 291–295.
156. Karipidis E *et al* (2011) Transmit beamforming for inter-operator spectrum sharing. In: Future Network Mobile Summit (FutureNetw), 2011, pp. 1–8.
157. Joshi SK, Weeraddana PC, Codreanu M & Latva-aho M (2012) Weighted sum-rate maximization for MISO downlink cellular networks via branch and bound. IEEE Trans. Signal Processing 60(4): 2090–2095.
158. Joshi S, Weeraddana PC, Codreanu M & Latva-Aho M (2011) Weighted sum-rate maximization for MISO downlink cellular networks via branch and bound. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1569–1573. Pacific Grove, CA, USA.
159. Joshi SK, Wijewardhana UL, Codreanu M & Latva-aho M (2015) Maximization of worst-case weighted sum-rate for MISO downlink systems with imperfect channel knowledge. IEEE Trans. Commun. 63(10): 3671–3685.
160. Joshi S, Wijewardhana UL, Codreanu M & Latva-aho M (2015) Maximization of worst-case weighted sum-rate for MISO downlink systems with channel uncertainty. In: Proc. IEEE Int. Conf. Commun., pp. 2289–2294.
161. Wijewardhana UL, Joshi S, Codreanu M & Latva-aho M (2013) Worst-case weighted sum-rate maximization for MISO downlink systems with imperfect channel knowledge. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1–5. Pacific Grove, USA.
162. Joshi SK, Codreanu M & Latva-aho M (2014) Distributed resource allocation for MISO downlink systems via the alternating direction method of multipliers. EURASIP J. Wireless Comm. and Netw. 2014(1): 1–19.
163. Joshi S, Codreanu M & Latva-Aho M (2012) Distributed resource allocation for MISO downlink systems via the alternating direction method of multipliers. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 488–493. Pacific Grove, CA, USA.
164. Joshi S, Codreanu M & Latva-Aho M (2013) Distributed SINR balancing for MISO downlink systems via the alternating direction method of multipliers. In: Proc. IEEE Int. Symp. Modeling Optim. Mobile, Ad Hoc Wireless Networks, pp. 318–325. Tsukuba, Japan.
165. Joshi SK, Manosha KBS, Codreanu M & Latva-aho M (2017) Dynamic inter-operator spectrum sharing via Lyapunov optimization. IEEE Trans. Wireless Commun. 16(10): 6365–6381.
166. Joshi S, Manosha K, Codreanu M & Latva-aho M (2017) Inter-operator dynamic spectrum sharing: A stochastic optimization approach. In: Proc. IEEE Wireless

On-demand Net. sys. and Serv.
167. Joshi S, Manosha KBS, Jokinen M, Hanninen T, Pirinen P, Posti H & Latva-aho M (2016) ESC sensor nodes placement and location for moving incumbent protection in CBRS. In: Proc. Wireless Innov. forum on Wireless Commun. Tech. Software Defined Radio.
168. Manosha KBS, Joshi SK, Codreanu M & Latva-aho M (2017) Admission control algorithms for QoS-constrained multicell MISO downlink systems. IEEE Trans. Wireless Commun., conditionally accepted .
169. Manosha KBS, Joshi S, Rajatheva N & Latva-aho M (2012) Energy efficient power control and beamforming in multi-antenna enabled femtocells. In: Proc. IEEE Global Telecom. Conf., pp. 3472–3477.
170. Manosha KBS, Joshi S, Codreanu M, Rajatheva N & Latva-aho M (2013) Power-throughput tradeoff in MIMO heterogeneous networks. In: Proc. Annual Asilomar Conf. Signals, Syst., Comp., pp. 1228–1232.
171. Boyd S (2007) Branch-and-bound methods. [Online]. Available: `http://www.stanford.edu/class/ee364b/lectures/bb_slides.pdf`.
172. Grant M & Boyd S (2011) Cvx: Matlab software for disciplined convex programming. [Online]. Available: `http://www.stanford.edu/~boyd/cvx/`.
173. Kumar A, Manjunath D & Kuri J (2008) Wireless Networking. ELSEVIER Inc., Burlington, MA, USA.
174. Shi C, Berry RA & Honig ML (2008) Distributed interference pricing with MISO channels. Proc. Annual Allerton Conf. Commun., Contr., Computing pp. 539–546.
175. Gomadam K, Cadambe VR & Jafar SA (2008) Approaching the capacity of wireless networks through distributed interference alignment. Proc. IEEE Global Telecom. Conf. pp. 1–6.
176. Codreanu M (2007) Multidimensional Adaptive Radio Links for Broadband Communications. Ph.D. thesis, Centre for Wireless Communications, University of Oulu. Acta Universitatis Ouluensis, Oulu, Finland. [Online]. Available: `http://herkules.oulu.fi/isbn9789514286223`.
177. Hong M, Razaviyayn M, Luo ZQ & Pang JS (2016) A unified algorithmic framework for block-structured optimization involving big data: With applications in machine learning and signal processing. IEEE Signal Processing Mag. 33(1): 57–77.
178. Rong Y, Vorobyov SA & Gershman AB (2006) Robust linear receivers for multiaccess space-time block-coded MIMO systems: A probabilistically constrained approach. IEEE J. Select. Areas Commun. 24(8): 1560–1570.
179. Vorobyov SA, Chen H & Gershman AB (2008) On the relationship between robust minimum variance beamformers with probabilistic and worst-case distortionless response constraints. IEEE Trans. Signal Processing 56(11): 5719–5724.
180. Pascual-Iserte A, Palomar DP, Perez-Neira AI & Lagunas MA (2006) A robust maximin approach for MIMO communications with imperfect channel state information based on convex optimization. IEEE Trans. Signal Processing 54(1): 346–360.
181. Jönsson UT (2001) A lecture on the s-procedure. [Online]. Available: `http://www.math.kth.se/~ulfj/5B5746/Lecture.ps`.

182. Luo ZQ, Ma WK, So AMC, Ye Y & Zhang S (2010) Semidefinite relaxation of quadratic optimization problems. IEEE Signal Processing Mag. 27(3): 20–34.
183. Song E, Shi Q, Sanjabi M, Sun R & Luo ZQ (2011) Robust SINR-constrained MISO downlink beamforming: When is semidefinite programming relaxation tight? In: Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, pp. 3096–3099.
184. Bertsekas DP (1999) Nonlinear Programming. Athena Scientific, Belmont, MA, 2nd edition.
185. Flury B (1997) A First Course in Multivariate Statistics. Springer, New York.
186. Boyd S (2008) Linear dynamical systems. [Online]. Available: `http://stanford.edu/class/ee363/lectures/estim.pdf`.
187. Mathews JH & Fink KK (2004) Numerical Methods Using Matlab, Fourth Edition. Prentice-Hall Inc.
188. Cheney W & Kincaid D (1998) Numerical Mathematics and Computing, Fourth Edition. International Thomson Publishing.
189. Schizas ID, Ribeiro A & Giannakis GB (2008) Consensus in ad hoc WSNs with noisy links-part i: Distributed estimation of deterministic signals. IEEE Trans. Signal Processing 56(1): 350–364.
190. Gupta P & Kumar PR (2000) The capacity of wireless networks. IEEE Trans. Inform. Theory 46(2): 388–404.
191. Bertsekas DP (1982) Constrained Optimization and Lagrange Multiplier Methods. Academic Press.
192. Chang TH, Hong M, Liao WC & Wang X (2016) Asynchronous distributed ADMM for large-scale optimization-part I: Algorithm and convergence analysis. IEEE Trans. Signal Processing 64(12): 3118–3130.
193. Bertsekas DP & Tsitsiklis JN (1997) Parallel and Distributed Computation: Numerical Methods. Athena Scientific, Belmont, Massachusetts, USA.
194. Teng Y, Wang Y & Horneman K (2014) Co-primary spectrum sharing for denser networks in local area. In: Proc. IEEE Int. Conf. Cognitive Radio Oriented Wireless Networks and Commun., pp. 120–124.
195. Nash J (1950) The bargaining problem. Econometrica 18(2): 155–162.
196. Gao L, Iosifidis G, Huang J, Tassiulas L & Li D (2014) Bargaining-based mobile data offloading. IEEE J. Select. Areas Commun. 32(6): 1114–1125.
197. Boche H & Schubert M (2011) A generalization of Nash bargaining and proportional fairness to log-convex utility sets with power constraints. IEEE Trans. Inform. Theory 57(6): 3390–3404.
198. Vazirani VV (2012) The notion of a rational convex program, and an algorithm for the Arrow-Debreu Nash bargaining game. In: Pro. ACM-SIAM Symp. Discrete Algorithms, pp. 973–992.
199. METIS (2015). Future spectrum system concept. Doc. No. ICT-317669-METIS/D5.4.
200. Neely MJ (2003) Dynamic power allocation and routing for satellite and wireless networks with time varying channels. Ph.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA.
201. Mao Y, Zhang J & Letaief KB (2015) A Lyapunov optimization approach for green cellular networks with hybrid energy supplies. IEEE J. Select. Areas Commun.

33(12): 2463–2477.
202. Shuminoski T & Janevski T (2016) Lyapunov optimization framework for 5G mobile nodes with multi-homing. IEEE Commun. Lett. 20(5): 1026–1029.
203. Li S, Huang J & Li SYR (2014) Dynamic profit maximization of cognitive mobile virtual network operator. IEEE Trans. Mobile Comput. 13(3): 526–540.
204. Lakshminarayana S, Quek TQS & Poor HV (2014) Cooperation and storage tradeoffs in power grids with renewable energy resources. IEEE J. Select. Areas Commun. 32(7): 1386–1397.
205. Li J, Wu J, Peng M & Zhang P (2016) Queue-aware energy-efficient joint remote radio head activation and beamforming in cloud radio access networks. IEEE Trans. Wireless Commun. 15(6): 3880–3894.
206. Neely MJ (2007) Optimal pricing in a free market wireless network. In: Proc. IEEE Int. Conf. on Comp. Commun., pp. 213–221.
207. Peng M, Xie X, Hu Q, Zhang J & Poor HV (2015) Contract-based interference coordination in heterogeneous cloud radio access networks. IEEE J. Select. Areas Commun. 33(6): 1140–1153.
208. Weeraddana PC, Codreanu M, Latva-aho M & Ephremides A (2013) Multicell MISO downlink weighted sum-rate maximization: A distributed approach. IEEE Trans. Signal Processing 61(3): 556–570.
209. Sidiropoulos ND, Davidson TN & Luo ZQ (2006) Transmit beamforming for physical-layer multicasting. IEEE Trans. Signal Processing 54(6): 2239–2251.
210. ITU (2011) Isolation between antennas of IMT base stations in the land mobile service, document ITU-R M.2244. Technical report, International Telecommunication Union.
211. GSMA Mobile infrastructure sharing. Technical report, GSMA Foundation. URI: `http://www.gsma.com/publicpolicy/wp-content/uploads/2012/09/Mobile-Infrastructure-sharing.pdf`.
212. Meddour DE, Rasheed T & Gourhant Y (2011) On the role of infrastructure sharing for mobile network operators in emerging markets. Computer Networks, Elsevier 55(7): 1576–1591.
213. Chong EKP & Zak SH (2001) An Introduction to Optimization, Second Edition. John Wiley and Sons, New York.
214. Yoo T & Goldsmith A (2006) On the optimality of multiantenna broadcast scheduling using zero-forcing beamforming. IEEE J. Select. Areas Commun. 24(3): 528–541.
215. Rappaport TS, Sun S, Mayzus R, Zhao H, Azar Y, Wang K, Wong GN, Schulz JK, Samimi M & Gutierrez F (2013) Millimeter wave mobile communications for 5G cellular: It will work! IEEE Access 1: 335–349.
216. Fischione C (2011) Fast-Lipschitz optimization with wireless sensor networks applications. IEEE Trans. Automat. Contr. 56(10): 2319–2331.
217. Shokri-Ghadikolaei H, Boccardi F, Fischione C, Fodor G & Zorzi M (2016) Spectrum sharing in mmWave cellular networks via cell association, coordination, and beamforming. IEEE J. Select. Areas Commun. 34(11): 2902–2917.
218. Boccardi F, Shokri-Ghadikolaei H, Fodor G, Erkip E, Fischione C, Kountouris M, Popovski P & Zorzi M (2016) Spectrum pooling in mmWave networks: Opportunities, challenges, and enablers. IEEE Commun. Mag. 54(11): 33–39.

Appendix 1 : SDP formulation of problem (3.24)

SDP formulation of problem (3.24) is based on the following $\mathcal{S}$-lemma [29, 181]:

Lemma 1.0.1 ($\mathcal{S}$-lemma)**.** *Let Ψ_i be a real valued function of an m-dimensional complex vector $\mathbf{y}$, defined as*

$$\Psi_i(\mathbf{y}) = \mathbf{y}^H \mathbf{A}_i \mathbf{y} + 2\mathrm{Re}(\mathbf{b}_i^H \mathbf{y}) + c_i, \tag{1.0.1}$$

where $\mathbf{A}_i \in \mathbb{H}^m$, $\mathbf{b}_i \in \mathbb{C}^m$, $c_i \in \mathbb{R}$, and $i = 0, 1$. Assume that there exists a vector $\hat{\mathbf{y}} \in \mathbb{C}^m$ such that $\Psi_1(\hat{y}) < 0$. Then the following conditions are equivalent:

S1 : $\Psi_0(\mathbf{y}) \geq 0$ for all $\mathbf{y} \in \mathbb{C}^m$ such that $\Psi_1(\mathbf{y}) \leq 0$.
S2 : There exists $\lambda \geq 0$ such that the following LMI is feasible:

$$\begin{bmatrix} \mathbf{A}_0 & \mathbf{b}_0 \\ \mathbf{b}_0^H & c_0 \end{bmatrix} + \lambda \begin{bmatrix} \mathbf{A}_1 & \mathbf{b}_1 \\ \mathbf{b}_1^H & c_1 \end{bmatrix} \succeq 0.$$

Now, let us consider constraints (3.24a) and (3.24b), and express them as

$$\mathbf{e}_{ll}^{\mathrm{H}} \mathbf{V}_l \mathbf{e}_{ll} + 2\mathrm{Re}\big((\mathbf{V}_l \hat{\mathbf{h}}_{ll})^{\mathrm{H}} \mathbf{e}_{ll}\big) + \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l \hat{\mathbf{h}}_{ll} - \sum_{n \in \mathcal{N} \setminus \{\mathrm{tran}(l)\}} \sum_{j \in \mathcal{L}(n)} I_{jl} + \sigma_n^2 \geq 0, \quad l \in \mathcal{L} \tag{1.0.2}$$

$$\mathbf{e}_{ll}^{\mathrm{H}} \mathbf{Q}_{ll} \mathbf{e}_{ll} - 1 \leq 0, \quad l \in \mathcal{L}, \tag{1.0.3}$$

where $\mathbf{V}_l = \tilde{\mathbf{M}}_l / \gamma_l - \sum_{j \in \mathcal{L}(\mathrm{tran}(l)), j \neq l} \tilde{\mathbf{M}}_j$. For $\mathbf{e}_{ll} = 0$, the inequality (1.0.3) satisfy strictly. Thus, we can consider the left hand side of (1.0.2) and (1.0.3) as $\Psi_0(\mathbf{e}_{nn})$ and $\Psi_1(\mathbf{e}_{nn})$ in Lemma 1.0.1. Then, Lemma 1.0.1 implies that the inequality (1.0.2) is satisfied for all channel errors $\mathbf{e}_{ll}$ that satisfy (1.0.3) if there exists $\mu_{ll} \geq 0$ and the following LMI is feasible

$$\begin{bmatrix} \mathbf{V}_l & \mathbf{V}_l \hat{\mathbf{h}}_{ll} \\ \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l & \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l \hat{\mathbf{h}}_{ll} - \sum_{n \in \mathcal{N} \setminus \{\mathrm{tran}(l)\}} \sum_{j \in \mathcal{L}(n)} I_{jl} - \sigma_n^2 \end{bmatrix} + \mu_{ll} \begin{bmatrix} \mathbf{Q}_{ll} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \quad l \in \mathcal{L} \tag{1.0.4}$$

Similarly, we can express inequalities (3.24c) and (3.24d) as

$$- \mathbf{e}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j \mathbf{e}_{jl} - 2\mathrm{Re}\big((\tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl})^{\mathrm{H}} \mathbf{e}_{jl}\big) + I_{jl} - \hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \geq 0, \\ l \in \mathcal{L}, n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}, j \in \mathcal{L}(n) \tag{1.0.5}$$

$$\mathbf{e}_{jl}^{\mathrm{H}} \mathbf{Q}_{jl} \mathbf{e}_{jl} - 1 \leq 0, \quad l \in \mathcal{L}, n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}, j \in \mathcal{L}(n). \tag{1.0.6}$$

Then, by following the same idea as in steps (1.0.2)-(1.0.4), we can express inequalities (1.0.5) and (1.0.6) equivalently as the following LMI

$$\begin{bmatrix} -\tilde{\mathbf{M}}_j & -\tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \\ -\hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j & I_{jl} - \hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \end{bmatrix} \\ + \mu_{jl} \begin{bmatrix} \mathbf{Q}_{jl} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \quad l \in \mathcal{L}, n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}, j \in \mathcal{L}(n) \tag{1.0.7}$$

for $\mu_{jl} \geq 0$. Hence, problem (3.24) as an SDP can be expressed as

$$\begin{aligned} \text{minimize} \quad & \textstyle\sum_{l \in \mathcal{L}} \mathrm{Trace}(\tilde{\mathbf{M}}_l) \\ \text{subject to} \quad & \begin{bmatrix} \mathbf{V}_l & \mathbf{V}_l \hat{\mathbf{h}}_{ll} \\ \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l & \hat{\mathbf{h}}_{ll}^{\mathrm{H}} \mathbf{V}_l \hat{\mathbf{h}}_{ll} - \sum\limits_{n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}} \sum\limits_{j \in \mathcal{L}(n)} I_{jl} - \sigma_n^2 \end{bmatrix} \\ & + \mu_{ll} \begin{bmatrix} \mathbf{Q}_{ll} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \quad l \in \mathcal{L} \\ & \begin{bmatrix} -\tilde{\mathbf{M}}_j & -\tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \\ -\hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j & I_{jl} - \hat{\mathbf{h}}_{jl}^{\mathrm{H}} \tilde{\mathbf{M}}_j \hat{\mathbf{h}}_{jl} \end{bmatrix} + \mu_{jl} \begin{bmatrix} \mathbf{Q}_{jl} & 0 \\ 0 & -1 \end{bmatrix} \succeq 0, \\ & \qquad\qquad l \in \mathcal{L}, n \in \mathcal{N} \backslash \{\mathrm{tran}(l)\}, j \in \mathcal{L}(n) \\ & \mu_{jl} \geq 0, \quad j, l \in \mathcal{L} \\ & \tilde{\mathbf{M}}_l \succeq 0, \quad l \in \mathcal{L} \\ & \textstyle\sum_{l \in \mathcal{L}(n)} \mathrm{Trace}(\tilde{\mathbf{M}}_l) \leq p_n^{\max}, \quad n \in \mathcal{N} \\ & \mathrm{Rank}(\tilde{\mathbf{M}}_l) = 1, \quad l \in \mathcal{L}, \end{aligned} \tag{1.0.8}$$

with variables $\tilde{\mathbf{M}}_l$, μ_{jl}, and I_{jl}, where $\mathbf{V}_l$ is defined as

$$\mathbf{V}_l = \frac{\tilde{\mathbf{M}}_l}{\gamma_l} - \sum_{j \in \mathcal{L}(\mathrm{tran}(l)), j \neq l} \tilde{\mathbf{M}}_j, \tag{1.0.9}$$

for all $l \in \mathcal{L}$.

Appendix 2

In this appendix we propose a bracketing method [187, 188] to solve problem (4.45). Let us start by combining the second (linear) and third (quadratic) terms of (4.46) as

$$p(\alpha_n) \quad = \quad \tilde{p}(\alpha_n) + \frac{\rho}{2}\left(\alpha_n - \gamma^i + \lambda_n^i - \frac{1}{\rho N}\right)^2 - \frac{1}{N}\left(\gamma^i - \lambda_n^i + \frac{1}{2}\right). \quad (2.0.1)$$

Let us drop the constant term of (2.0.1) and express it as

$$p(\alpha_n) \quad = \quad \tilde{p}(\alpha_n) + \frac{\rho}{2}(\alpha_n - \varpi)^2, \quad (2.0.2)$$

where $\varpi = \gamma^i - \lambda_n^i + \frac{1}{\rho N}$.

Note that the optimal value $\tilde{p}(\alpha_n)$ is a nondecreasing function of $\alpha_n \in [0, \alpha_n^{\max}]$ [59]. To examine that, let $\mathcal{P}_i$ and $\mathcal{P}_j$ be the feasible set of problem (4.47) for $\alpha_n = \alpha_n^i$ and $\alpha_n = \alpha_n^j$, respectively. If $\alpha_n^j \geq \alpha_n^i$, then it is easy to see that $\mathcal{P}_j \subseteq \mathcal{P}_i$. Hence, the optimal value $\tilde{p}(\alpha_n^j) \geq \tilde{p}(\alpha_n^i)$ for all $\alpha_n^j \geq \alpha_n^i$ and $\alpha_n^i, \alpha_n^j \in [0, \alpha_n^{\max}]$. Furthermore, there exists a partition of $[0, \alpha_n^{\max}]$ as $[0, \phi] \cup [\phi, \alpha_n^{\max}]$ such that

$$\tilde{p}(\alpha_n) \quad = \quad c, \quad \alpha_n \in [0, \phi], \quad (2.0.3)$$

where c is the optimal solution of problem (4.47) for $\alpha_n = 0$.

Next we propose to use bracketing method [187, 188] to find the infimum of function $p(\alpha_n)$ on the interval $\alpha_n \in [0, \alpha_n^{\max}]$. First, in Lemma 2.0.1, we show that the function $p(\alpha_n)$ is a unimodal function on the interval $\alpha_n \in [0, \alpha_n^{\max}]$ for the condition: C) $\varpi \leq \phi$.

Lemma 2.0.1. The function $p(\alpha_n)$,

$$p(\alpha_n) \quad = \quad \tilde{p}(\alpha_n) + \frac{\rho}{2}(\alpha_n - \varpi)^2, \quad (2.0.4)$$

is a unimodal function on the interval $\alpha_n \in [0, \alpha_n^{\max}]$ for the condition C.

Proof:

[59] The interval $[0, \alpha_n^{\max}]$ denotes the range of feasible α_n for problem (4.47).

1. For the case $\varpi \leq 0$, the proof is trivial, since $p(\alpha_n)$ is a sum of two increasing functions on the interval $\alpha_n \in [0, \alpha_n^{\max}]$.
2. For the case $\varpi > 0$, let us partition $[0, \alpha_n^{\max}]$ as $[0, \varpi] \cup [\varpi, \alpha_n^{\max}]$. On the interval $\alpha_n \in [0, \varpi]$, the function $\tilde{p}(\alpha_n)$ takes a constant value c. On the interval $\alpha_n \in [\varpi, \alpha_n^{\max}]$, the function $\tilde{p}(\alpha_n)$ is a nondecreasing function. Hence, the function $p(\alpha_n)$ is a sum of affine and convex functions on the interval $[0, \varpi]$, and a sum of nondecreasing and increasing functions on the interval $[\varpi, \alpha_n^{\max}]$. Thus, the function $p(\alpha_n)$ is a unimodal function. □

Lemma 2.0.1 implies that for the condition C (i.e., $\varpi \leq \phi$), the infimum of the function $p(\alpha_n)$ can be obtained optimally by using a bracketing method [187, 188].

For the case condition C is not satisfied (i.e., $\phi \leq \varpi$), let us partition $[0, \alpha_n^{\max}]$ as $[0, \phi] \cup [\phi, \varpi] \cup [\varpi, \alpha_n^{\max}]$. On the interval $\alpha_n \in [0, \phi]$, the function $p(\alpha_n)$ is a decreasing function (since $\tilde{p}(\alpha_n)$ takes a constant value c, and $(\alpha_n - \varpi)^2$ is a decreasing function). On the interval $\alpha_n \in [\varpi, \alpha_n^{\max}]$, the function $p(\alpha_n)$ is an increasing function (since $\tilde{p}(\alpha_n)$ is a nondecreasing function and $(\alpha_n - \varpi)^2$ is an increasing function). On the interval $\alpha_n \in [\phi, \varpi]$, analytically expressing the curvature of $p(\alpha_n)$ is difficult, since the curvature of function $\tilde{p}(\alpha_n)$ depends on the numerical parameters. This implies that for the case $\phi \leq \varpi$, the infimum of the function $p(\alpha_n)$ lies on the interval $[\phi, \varpi]$, i.e.,

$$\underset{\alpha_n \in [0, \alpha_n^{\max}]}{\operatorname{argmin}} \; p(\alpha_n) \in [\phi, \varpi]. \tag{2.0.5}$$

Thus in the case $\phi \leq \varpi$ (i.e., if condition C is not satisfied), the solution of problem (2.0.5) obtained by using bracketing method [187, 188] lies at most $(\varpi - \phi)$ away from the optimal solution. However, in all of our numerical simulations, we have always noted that the function $p(\alpha_n)$ is a unimodal function. In that case, problem (2.0.5) is solved optimally by the bracketing method [187, 188]. Moreover, the convergence of the proposed Algorithm 4.3 (see numerical example, Section 4.4) to a centralized solution shows that the bracketing method can be used to solve problem (4.45).

Appendix 3

In this appendix we discuss a method to set a value for penalty parameter ρ for Algorithm 4.1.

The ADMM method is guaranteed to converge for all values of its penalty parameter ρ [113]. However, the rate of convergence of the ADMM algorithm is sensitive to the choice of the penalty parameter ρ. In practice, the ADMM penalty parameter ρ is either tuned empirically for each specific application, or set equal to 1 by normalizing the problem data set [113, Ch. 11]. Note that in Algorithm 4.1, to solve the local variable update (4.23), we can normalize the problem data [i.e., sum-power $(\sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l\|_2^2)$] by normalizing factor $D_n>0$ and set $\rho=1$, which is equivalent to setting $\rho=D_n$ in Algorithm 4.1, if the problem data [i.e., sum-power $(\sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l\|_2^2)$] is not normalized. To elaborate further, let us express problem (4.23) as

$$\begin{aligned}
\text{minimize} \quad & \frac{1}{D_n}\left(\sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l\|_2^2\right)+\frac{1}{2}\|\mathbf{x}_n-\mathbf{z}_n^i+\mathbf{v}_n^i\|_2^2 \\
\text{subject to} \quad & \begin{bmatrix} \sqrt{1+\frac{1}{\gamma_l}}\mathbf{h}_{ll}^{\mathrm{H}}\mathbf{m}_l \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{ll} \\ \tilde{\mathbf{x}}_l \\ \sigma_l \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l\in\mathcal{L}(n) \\
& \begin{bmatrix} x_{n,nl} \\ \mathbf{M}_n^{\mathrm{H}}\mathbf{h}_{jl} \end{bmatrix} \succeq_{\mathrm{SOC}} 0, \quad l\in\mathcal{I}_{\mathrm{int}}(n)
\end{aligned} \tag{3.0.1}$$

with variables $\mathbf{M}_n=[\mathbf{m}_l]_{l\in\mathcal{L}(n)}$ and $\mathbf{x}_n$, where $D_n>0$ is the normalizing factor, $\tilde{\mathbf{x}}_l=(x_{n,bl})_{b\in\mathcal{N}_{\mathrm{int}}(l)}$ is a subset of $\mathbf{x}_n$ (see (4.12)). Note that we have set $\rho=D_n$ in problem (4.23) to get problem (3.0.1).

Let $\{\mathbf{m}_l^\star\}_{l\in\mathcal{L}(n)}$ denote the optimal solution of problem (3.0.1). Then the optimal value of D_n is $\sum_{l\in\mathcal{L}(n)}\|\mathbf{m}_l^\star\|_2^2$. However, before the convergence of Algorithm 4.1, we do not have optimal beamformers (i.e., $\mathbf{m}_l^\star$ for all $l\in\mathcal{L}(n)$). Thus, in our simulation, to estimate D_n, we ignore the interference and noise terms, and find beamforming vector $\tilde{\mathbf{m}}_l$ that achieves the required SINR threshold γ_l (in dB scale), for all $l\in\mathcal{L}$, which can be expressed as

$$\tilde{\mathbf{m}}_l=\sqrt{10^{0.1\times\gamma_l}}\mathbf{h}_{ll}/\|\mathbf{h}_{ll}\|_2^2, \quad l\in\mathcal{L}(n).$$

Hence, we approximate the factor D_n for problem (3.0.1) as

$$\begin{aligned} D_n &= \sum_{l \in \mathcal{L}(n)} \|\tilde{\mathbf{m}}_l\|_2^2 \\ &= \sum_{l \in \mathcal{L}(n)} (10^{0.1 \times \gamma_l})/\|\mathbf{h}_{ll}\|_2^2. \end{aligned}$$

Furthermore, we find $D = \max_{n \in \mathcal{N}} \{D_n\}$, and set $\rho = D$ for Algorithm 4.1.

Appendix 4

The stability of virtual queues $\{X_n(t)\}_{n\in\mathcal{N}}$ ensures constraint (5.9a). To show this, we start by noting the following inequality from expression (5.11):

$$X_n(t+1) \geq X_n(t) - x_n^{\text{out}}(t) + x_n^{\text{in}}(t). \tag{4.0.1}$$

Then, by summing the above expression (4.0.1) over $t \in \{0, 1, \ldots, \tau-1\}$, and dividing it by τ, we get

$$\tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} x_n^{\text{in}}(t) \leq \tfrac{1}{\tau}\sum_{t=0}^{\tau-1} x_n^{\text{out}}(t) + \tfrac{1}{\tau}\big(X_n(\tau) - X_n(0)\big). \tag{4.0.2}$$

We now substitute expressions (5.12) and (5.13) into (4.0.2), and re-arrange the resulting expression as

$$\begin{aligned}\tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} \mu_n(t) \leq \tfrac{1}{\tau}\sum_{t=0}^{\tau-1}\sum_{l\in\mathcal{L}(n)} g_{nl}(a_{nl}(t)) \\ + \tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n},t)} w_s - B\big)^+ \\ - \tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+ \\ - U_n^0 + \tfrac{1}{\tau}\big(X_n(\tau) - X_n(0)\big).\end{aligned} \tag{4.0.3}$$

By taking the expectations of both sides of expression (4.0.3) and $\liminf$ as $\tau \to \infty$, we get

$$\begin{aligned}\liminf_{\tau\to\infty} \tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} \mathrm{E}\{\mu_n(t)\} \leq \liminf_{\tau\to\infty} \Big(\tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1}\sum_{l\in\mathcal{L}(n)} \mathrm{E}\big\{g_{nl}(a_{nl}(t))\big\} \\ + \tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} \mathrm{E}\big\{q_n(t)\big(\sum_{s\in\mathcal{S}(\underline{n},t)} w_s - B\big)^+\big\} \\ - \tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} \mathrm{E}\big\{q_{\underline{n}}(t)\big(\sum_{s\in\mathcal{S}(n,t)} w_s - B\big)^+\big\}\Big) - U_n^0.\end{aligned} \tag{4.0.4}$$

Note that to write expression (4.0.4) from (4.0.3), we have used a property $\liminf_{\tau\to\infty} \mathrm{E}\{X_n(\tau)\}/\tau = 0$, which is satisfied if virtual queue $X_n(t)$ is strongly stable [42, Lem. 3.3]. Then by applying Jensen's inequality,

$$\tfrac{1}{\tau}\textstyle\sum_{t=0}^{\tau-1} \mathrm{E}\{g_{nl}(a_{nl}(t))\} \leq g_{nl}\big(\tfrac{1}{\tau}\sum_{t=0}^{\tau-1} \mathrm{E}\{a_{nl}(t)\}\big), \tag{4.0.5}$$

to the right hand side of (4.0.4), we get $\overline{\mu}_n \leq \overline{U}_n - U_n^0$ for all $n \in \mathcal{N}$, i.e., constraint (5.9a).

Appendix 5

In this appendix we discuss a method to set a value of disagreement point U_n^0 for problem (5.7) to numerically evaluate the performance of the proposed algorithms in Chapter 5.

We set U_n^0 to a value of utility that nth operator gains by using B Hz of spectrum band (i.e., without sharing its spectrum band with the other operator). The resource allocation problem for nth operator without sharing its licensed spectrum band can be obtained by modifying problem (5.32). Specifically, by dropping the constraint associated with the orthogonal subchannel allocation of problem (5.32), and the payment terms with the spectrum pricing in the objective function, the resource allocation problem for operator $n \in \mathcal{N}$ during time slot t can be expressed as

$$\begin{aligned} \text{maximize} \quad & \sum_{l \in \mathcal{L}(n)} Q_{nl}(t) \sum_{s \in \mathcal{S}(n,t)} w_s \log_2 \Bigg(1 \\ & + \frac{|\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nl,s}|^2}{N_0 w_s + \sum_{j \in \mathcal{L}(n), j \neq l} |\mathbf{h}_{nl,s}^{\mathrm{H}}(t)\mathbf{m}_{nj,s}|^2} \Bigg) \\ \text{subject to} \quad & \sum_{l \in \mathcal{L}(n)} \sum_{s \in \mathcal{S}(n,t)} \|\mathbf{m}_{nl,s}\|_2^2 \leq p_n^{\max}, \end{aligned} \tag{5.0.1}$$

with variables $\{\mathbf{m}_{nl,s}\}_{l \in \mathcal{L}(n), s \in \mathcal{S}(n,t)}$. Note that problem (5.0.1) can be solved with the approach presented in Section 5.3. Let us denote $\{\mathbf{m}_{nl,s}^{\star}(t)\}_{l \in \mathcal{L}(n), s \in \mathcal{S}(n,t)}$ as the solution of problem (5.0.1), and let the transmission rate be $r_{nl}^{\star}(t)$ of lth user of nth operator $\big($the transmission rate can be calculated by using expression (5.2)$\big)$. Let $\bar{r}_{nl}(t)$ denote the time average rate defined as $\bar{r}_{nl}(t) = \frac{1}{t} \sum_{\tau=0}^{t-1} r_{nl}^{\star}(\tau)$. Then the utility gain of lth user of nth operator based on its current data rate $\bar{r}_{nl}(t)$ is $g_{nl}(\bar{r}_{nl}(t))$.

To estimate a value of $\{U_n^0\}_{n \in \mathcal{N}}$, we solve problem (5.0.1) with $Q_{nl}(t) = 1$ for all $n \in \mathcal{N}$ and $l \in \mathcal{L}(n)$, and run simulation for 5000 fading realizations. Then a disagreement point for nth operator is set to $U_n^0 = \sum_{l \in \mathcal{L}(n)} g_{nl}(\bar{r}_{nl}(5000))$.

Book orders:
Granum: Virtual book store
http://granum.uta.fi/granum/

UNIVERSITY OF OULU P.O. Box 8000 FI-90014 UNIVERSITY OF OULU FINLAND

ACTA UNIVERSITATIS OULUENSIS

UNIVERSITY of OULU
OULUN YLIOPISTO

ISBN 978-952-62-1742-0 (Paperback)
ISBN 978-952-62-1743-7 (PDF)
ISSN 0355-3213 (Print)
ISSN 1796-2226 (Online)