



HAL
open science

Topological interference management in clustered ad hoc networks

David Jia

► **To cite this version:**

David Jia. Topological interference management in clustered ad hoc networks. Networking and Internet Architecture [cs.NI]. Université Paris-Saclay, 2023. English. NNT : 2023UPAST162 . tel-04639076

HAL Id: tel-04639076

<https://theses.hal.science/tel-04639076>

Submitted on 8 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Topological interference management in clustered ad hoc networks

*Gestion d'interférence topologique
dans les réseaux ad hoc clusterisés*

Thèse de doctorat de l'université Paris-Saclay

École doctorale n°580 Sciences et Technologies de l'Information et de la
Communication (STIC)

Spécialité de doctorat : Sciences des réseaux, de l'information et de la
communication

Graduate School : Sciences de l'ingénierie et des systèmes.

Référent : CentraleSupélec

Thèse préparée dans le Laboratoire des signaux et systèmes (Université Paris-Saclay,
CNRS, CentraleSupélec),

sous la direction de **Mohamad ASSAAD**, Professeur à CentraleSupélec,
la co-direction de **Christophe LE MARTRET**, Expert Thales, HDR,
le co-encadrement de **Xavier LETURC**, Ingénieur Docteur Thales

Thèse soutenue à Paris-Saclay, le 06 novembre 2023, par

David JIA

Composition du jury

Membres du jury avec voix délibérative

Luc VANDENDORPE

Professeur, Université catholique de Louvain

Xavier LAGRANGE

Professeur, IMT Atlantique

Samir PERLAZA

Chargé de recherche, HDR, INRIA

Marina PETROVA

Associate Professor, KTH Royal Institute of Technology

Président

Rapporteur & Examineur

Rapporteur & Examineur

Examinatrice

Titre : Gestion d'interférence topologique dans les réseaux ad hoc clusterisés

Mots clés : Gestion d'interférence, réseau ad hoc, ordonnancement, théorie de l'estimation apprentissage automatique, optimisation convexe et non-convexe.

Résumé : Cette thèse traite le problème de la gestion distribuée d'interférences dans les réseaux ad hoc clusterisés dans des canaux aléatoires. On suppose qu'un chef de cluster (CH) est sélectionné dans chaque groupe pour superviser les transmissions, l'allocation des ressources et des slots temporels.

Pour opérer un réseau ad hoc clusterisé, il faut attribuer une bande de fréquence (FB en anglais) à chaque cluster. Lorsque le nombre de FBs disponibles est supérieur ou égal au nombre de cluster, il est facile d'effectuer une attribution orthogonale de FB, même de manière distribuée. Toutefois, pour les déploiements importants dans la pratique, en particulier dans les environnements militaires, le nombre de clusters est souvent beaucoup plus élevé que le nombre de FBs disponibles. Bien que la réutilisation spatiale des FBs puisse être utilisée dans ce cas, si la topologie du réseau le permet, une telle solution peut ne pas fournir des performances satisfaisantes dans certains scénarios (par exemple, une forte densité de clusters), et d'autres mécanismes sont alors nécessaires.

Dans cette thèse, nous considérons que les clusters adjacents peuvent interférer entre eux et nous proposons d'utiliser une technique de gestion des interférences basée sur l'approche de la gestion des interférences topologiques (TIM en anglais). La méthode TIM classique nécessite la connaissance du graphe d'interférence du réseau, c'est-à-dire qu'il faut déterminer pour chaque récepteur si l'interférence induite par chaque autre nœud est faible ou forte dans des canaux déterministes. Des travaux antérieurs ont proposé d'étendre la méthode TIM aux canaux aléatoires en incorpo-

rant des informations statistiques sur l'état du canal (SCSI en anglais) dans l'algorithme.

Toutefois, ces solutions présentent des performances mauvaises car elles sont moins performantes en termes de débit global et d'équité que l'accès multiple par répartition temporelle (TDMA en anglais). Par conséquent, dans ce contexte, **la première contribution de cette thèse est d'améliorer l'état de l'art en proposant une autre extension du TIM, qui est appelée enhanced TIM (eTIM) dans cette thèse.** Pour ce faire, nous fournissons une nouvelle méthode de gestion des interférences qui combine l'eTIM avec les accès TDMA et améliore à la fois le débit global et l'équité des débits.

Pour appliquer cette solution, nous devons estimer le SCSI lié à chaque lien des grappes associées. Une solution triviale consiste à appliquer le TDMA, où chaque nœud transmet l'un après l'autre. Afin de réduire la durée de sensing, nous proposons deux nouvelles solutions d'ordonnancement qui utilisent des émissions simultanées et qui sont plus performantes que le TDMA. **Cela constitue la deuxième contribution de cette thèse.**

Après la phase d'estimation, chaque nœud connaît le SCSI lié aux liens de tous les autres nœuds. Pour mettre en œuvre l'eTIM, les SCSI doivent être partagés entre les deux clusters. Nous avons donc étudié plusieurs méthodes de réduction de dimension, telles que l'analyse en composantes principales (linéaire) et l'autoencodeur (non linéaire), appliquées à la matrice des SCSI afin de réduire la quantité d'échanges de signalisation entre les clusters. **Cela correspond à la troisième contribution de cette thèse.**

Title : Topological interference management in clustered ad hoc networks

Keywords : Interference management, ad hoc network, scheduling optimization, estimation theory, deep learning, convex and non-convex optimization.

Abstract : This thesis addresses the problem of distributed interference management in clustered ad hoc networks under random fading channels. We assume that a cluster head (CH) is selected in each cluster to oversee transmissions, resource, and time slot allocation.

To operate a clustered ad hoc network, one needs to allocate a frequency band (FB) to each cluster. When the number of available FBs is greater or equal to the number of clusters, we can easily perform an orthogonal FB allocation, even in a distributed manner. However, for large deployments in practice, especially in military environments, the number of clusters is often much larger than the number of available FBs. While spatial FB reuse can be used in this case, if the network topology allows it, such a solution may not provide satisfactory performance in various scenarios (e.g. high cluster density), and other mechanisms are necessary.

In this thesis, we consider that adjacent clusters can interfere with each other, and we propose to use an interference management technique based on the topological interference management (TIM) approach. The original TIM requires the knowledge of the interference graph of the network, i.e. determining for each receiver whether the interference caused by each other node is weak or strong in deterministic channels. Existing works have proposed to extend TIM in random channels by incorporating statistical channel state information (SCSI) in the algorithm.

However, these solutions lead to poor performance since they achieve lower performance in terms of both sum rate and fairness than time-division multiple access (TDMA).

Hence, in this context, the contribution of this thesis is threefold. **The first contribution of this thesis is to improve the state-of-the-art by proposing an extension of TIM, which is referred to as enhanced TIM (eTIM) in this thesis.** In this method, the precoders and decoders are designed by optimizing an approximated expression of the average sum rate that depends on the SCSI in the network. In addition, we provide a second extension of TIM by strategically combining eTIM with TDMA access in such a way to improve both the sum rate and the fairness among the users.

To apply this solution, we need beforehand to estimate the SCSI related to each link of the associated clusters. One trivial solution consists of applying TDMA, where each node transmits one after the other. To reduce the sensing duration, we propose two new scheduling solutions that use simultaneous transmissions and outperform TDMA. **This constitutes the second contribution of this thesis.**

After the estimation phase, each node knows the SCSI related to the links from all the other nodes. To implement eTIM, the SCSI need to be shared between the two clusters. Thus, we investigated several dimensionality reduction (DR) methods, such as principal component analysis (linear) and autoencoder (non-linear), applied to the SCSI matrix in order to reduce the amount of signaling exchanges between clusters. Both DR methods achieve a compression rate of up to 45-50%, while maintaining only a 1% loss in the sum rate and minimum rate per link compared to the solution without DR. **This corresponds to the third contribution of this thesis.**

Remerciements

Il est temps pour moi de clôturer cette aventure riche en émotion.

Je tiens avant tout à remercier chaleureusement mes directeurs de thèse Mohamad Assaad et Christophe Le Martret ainsi qu'à mon encadrant Xavier Leturc. Mohamad, merci pour ta pertinence scientifique et ta patience qui m'ont permis de me former tout au long de la thèse. Christophe, merci pour nos nombreuses discussions scientifiques très enrichissantes et de m'avoir poussé toujours plus loin pour avancer. Xavier, merci pour tes précieux conseils qui m'ont beaucoup apporté scientifiquement, mais aussi humainement. Tes encouragements ont été d'une grande importance pour moi.

Un grand merci à Xavier Lagrange et Samir Perlaza pour avoir accepté d'être rapporteurs de ma thèse, à Luc Vandendorpe pour avoir présidé le jury de thèse et à Marina Petrova pour avoir accepté de faire partie du jury. Vos retours et votre implication m'ont beaucoup aidé pour l'amélioration de mon manuscrit et je suis honoré de l'intérêt que vous avez porté pour mes travaux.

Je remercie Jean-Luc Peron pour m'avoir accueilli au sein du service WFD de Thales. De manière général, je remercie tous les collègues et ex-collègues de Thales pour avoir égaillé mes journées et avec qui j'ai passé beaucoup de moments à bien rigoler, notamment pendant les pauses-café. Merci à Adam, Anaël, Antonio, Baptiste, Benoit, Dorin, Ehsan, Elie, Eric, Fatya, Jean-Baptiste, Julien, Laetitia, Marie, Martin, Olivier, Philippe, Pierre, Raphaël, Romain, Serdar, Soriya, Sylvain, Titouan.

Je remercie par ailleurs Vania Conan pour l'accueil au sein du service CNS où je travaille actuellement et à tous les nouveaux collègues qui m'ont permis de finaliser ce travail dans de bonnes conditions.

Je tiens aussi à remercier mes amis grâce à qui j'ai toujours pu me changer les idées quand j'en avais besoin. Les potes de soirée, les potes de Télécom (mention spéciale à la binôme de Thessaloniki, Kamélia qui m'aura beaucoup soutenu pendant la thèse), la team de Londres, les copains de Stan, la famille de Noël, la Night Fever, les potes de l'impro, la team debating, les potes de colo, etc.

Mes derniers remerciements vont à ma famille. Merci à mes parents et à mon frère qui veillent sur moi et me soutiennent au quotidien. Je pense à vous constamment et vous aime très fort. Enfin, à ma chère et tendre compagne Clara, merci d'être présente dans ma vie depuis le début de cette thèse et de me rendre heureux.

Contents

List of Acronyms	vii
General Introduction	1
1 Linear interference management overview	5
1.1 Introduction	5
1.2 Interference alignment	6
1.3 Topological interference management	8
1.4 Hybrid interference management	14
1.5 Conclusion	14
2 Enhanced topological interference management	17
2.1 Introduction	17
2.2 Preliminaries	19
2.3 Enhanced topological interference management	22
2.4 Proposed method: eTIM-hybrid	34
2.5 Case two available frequency bands	55
2.6 Conclusion	61
3 Rayleigh channel statistics estimation	63
3.1 Introduction	63
3.2 Channel statistics estimation using SINR samples under single interference and Rayleigh fading	66
3.3 Scheduling optimization with simultaneous transmissions for channel statistics estimation	89
3.4 Conclusion	106
4 Channel statistics dimensionality reduction	109
4.1 Introduction	109
4.2 System model	110
4.3 Dimensionality reduction	113
4.4 Numerical results	121

4.5 Conclusion	129
Conclusions and Perspectives	131
Appendices	135
A Appendix related to Chapter 2	135
A.1 Derivative expressions in Section 2.3.2	135
A.2 Implementation of eTIM max sum rate with minimum rate constraint . . .	137
A.3 Scheduling optimization proofs in Section 2.4.2	140
B Appendix related to Chapter 3	153
B.1 Scheduling with two simultaneous transmissions	153
B.2 Scheduling optimization with simultaneous transmissions	166
Bibliography	172

List of Acronyms

AWGN additive white Gaussian noise

AO alternating optimization

CDMA code-division multiple access

CMSR-eTIM constrained MSR-eTIM

CRLB Cramer-Rao lower bound

CSI channel state information

CSIT CSI at the transmitter

CH cluster head

D2D device-to-device

DCT discrete cosine transform

DoF degree-of-freedom

DR dimensionality reduction

DWT discrete wavelet transform

eTIM enhanced TIM

FB frequency band

FDMA frequency-division multiple access

GKQ Gauss-Kronrod quadrature

GLQ Gauss-Laguerre quadrature

IA interference alignment

i.i.d independent and identically distributed

INR Interference-to-Noise ratio

JFA-eTIM-hy joint frequency allocation eTIM-hybrid

KPCA kernel PCA

LOTUS law of the unconscious statistician

LRMC low-rank matrix completion

LS least-square

LZW Lempel-Ziv-Welch

MIMO multi-input multi-output

MISE mean integrated squared error

MLE maximum likelihood estimator

MMSE minimum mean-square error

MoM method of moments

MSE mean square error

MSR max sum-rate

MSR-eTIM maximum sum-rate eTIM

MSR-IA maximum sum-rate IA

NOMA non-orthogonal multiple access

NMSE normalized MSE

OMA orthogonal multiple access

PCA principal components analysis

pdf probability density function

PF-eTIM proportional fairness eTIM

SCSI statistical CSI

SIC successive interference cancellation

SINR Signal-to-Interference-plus-Noise ratio

SISO single-input single-output

SOTA state-of-the-art

SNR Signal-to-Noise ratio

SVD singular value decomposition

TEL Trial and Error Learning

TIA topological IA

TIM topological interference management

TIN treating interference as noise

TDD time-division duplex

TDMA time-division multiple access

t-SNE t-distributed stochastic neighbor embedding

UMAP uniform manifold approximation and projection

V2V vehicular-to-vehicular

General Introduction

Problem statement/Context

In wireless communications, ad hoc networks have been a useful paradigm for a while for a wide range of applications such as disaster management, military operations, and commercial environments [75, Section 1.3]. Ad hoc networks present the advantages of being decentralized, self-organized and fast to deploy without any fixed infrastructure. In such networks, each node is mobile and capable of transmitting, receiving, and forwarding data packets without the need for a central controller, e.g. a base station, contrarily to cellular networks. The dynamic changes that occur in ad hoc networks due to the nodes' mobility and the channel time-varying fluctuations make tasks such as resource allocation and interference management challenging. To address these challenges, several works have studied clustering techniques, in which nodes are grouped statically or dynamically into clusters [81]. In each cluster, a cluster head (CH) is selected among the nodes to oversee transmissions, resources, and time slot allocation within the cluster. This approach can improve the network performance by reducing the signaling exchanges and reducing the energy consumption of the nodes compared to conventional ad hoc networks. Clustered ad hoc networks have become increasingly popular in recent years since their applications include 5G networks, vehicular networks, and tactical networks [42, 129].

For an ad hoc network to operate, each cluster needs to be allocated a frequency band (FB). When there are more FBs than clusters, a basic solution is to allocate the FBs orthogonally, which can efficiently be performed in a distributed way. However, in practical scenarios with large deployments, particularly in military environments, it is common for the number of clusters to exceed the number of available FBs. To deal with this issue, alternative methods allow for spatial reuse of the FBs and can allocate a lower number of FBs than the clusters', but it creates interference. Existing solutions that we call *coloring* allow us to allocate, if necessary, and if the network topology allows it, the same FBs to different clusters while ensuring that the interference created does not negatively impact the network's proper functioning. One such solution is the Trial and Error Learning (TEL) method, which involves allocating channels in a distributed and dynamic way based on a criterion of maximization of a utility function based on the received interference level [34, 98]. The TEL algorithm converges to solutions that enable

FBs reuse to "distant" clusters, thus minimizing the impact of endogenous interference. Nevertheless, relying solely on this solution may not provide satisfactory performance in various scenarios, such as high cluster density. In such cases, adjacent clusters may end up using the same FB, resulting in interference that can negatively affect the network's performance.

In this context, lowering the number of clusters, thus increasing the ratio between the number of FBs and the number of clusters, should make the spatial frequency reuse work. One approach could consist in performing a new clustering with a larger cluster size. However, in this work, we assume to keep the clusters fixed by configuration and thus, we sought another approach. We propose to that end to associate two adjacent clusters using the same FB while dealing with the interference between them.

One naive solution would consist of applying time-division multiple access (TDMA) between both clusters, whose drawback is to divide the data rate by two. Instead, we propose to manage the interference using the topological interference management (TIM) method [50]. The original TIM technique relies on knowing the interference graph of the network, which indicates whether the strength of interference from each node to each receiver in deterministic channels is weak or strong. However, as we will see in the following chapters, the TIM scheme has the drawback of providing poor performance in terms of both sum rate and fairness, compared to traditional schemes like TDMA in ad hoc networks. Existing works have proposed an extension of TIM for random channels by incorporating statistical channel state information (CSI) into the procedure, but it also presents worse performance than TDMA in terms of sum rate and fairness [47, 124]. **The first objective of this thesis is thus to improve the state-of-the-art (SOTA) by proposing a novel TIM solution that incorporates the statistical CSI (SCSI) and outperforms TDMA in terms of both sum rate and fairness in ad hoc networks.**

To apply this solution, we need to estimate the statistical SCSI related to each link of the associated clusters. CSIs have the drawback of having an overhead from the nodes to the CH which is slower than the channel dynamics. As a result, the CH may have outdated CSI samples for performing interference management. In contrast, SCSIs have a longer validity period and are better suited for the constraints of ad hoc networks, where a direct return path for signaling may not exist. To describe the random fluctuations of the propagation channel, we consider a frequency-flat Rayleigh fading channel which is parametrized by the average channel power and is used in the literature to model communications without line-of-sight between the transmitter and the receiver. To estimate the SCSI, one trivial solution consists of applying TDMA, where each node transmits a reference signal one after the other. To reduce the sensing duration, **the second objective of this thesis is to develop new methods for estimating the SCSI in a frequency-flat Rayleigh fading channel, which outperforms TDMA.**

Finally, after the estimation phase, each node obtains the SCSI regarding the links from all other nodes and send them to the CH of their cluster. To implement our enhanced

TIM solution, the **SCSI** need to be shared between the **CHs** of the two clusters. For the sake of efficiency, we aim to minimize the overhead communication between the clusters. Therefore, **the third contribution of this thesis is to develop methods to reduce the amount of signaling exchanges between the CHs of the clusters.**

Outlines and contributions

This thesis is composed of five chapters. We present different interference management approaches in Chapter 1, whereas the original contributions are gathered in Chapters 2, 3, and 4. Conclusion and perspective for future works are provided in Chapter 5.

In Chapter 1, we provide a brief overview of several interference management methods assuming a single antenna at both transmitters and receivers. To have practical implementations, we focus on solutions based on linear precoders and decoders, such as interference alignment (**IA**) in Section 1.2 and **TIM** in Section 1.3.

In Chapter 2, we investigate the application of **TIM** in two associated adjacent clusters operating under a frequency-flat Rayleigh fading channel and **SCSI** assumption. When only one frequency band is available, we propose a new centralized interference management scheme. For that, we provide an approximation of the expected sum rate that only depends on the **SCSI**. We then formulate an enhanced framework of **TIM**, called enhanced **TIM** (**eTIM**), where the precoders and decoders are obtained by maximizing the approximated sum rate. To further improve the sum rate performance and fairness among the users, we propose a novel scheme, called **eTIM-hybrid**, that combines **eTIM** with **TDMA** access. This scheme provides higher a sum rate while guaranteeing a minimum rate per link. Furthermore, when two frequency bands are available, we propose a centralized method called joint frequency allocation **eTIM-hybrid** (**JFA-eTIM-hy**) that combines a frequency band allocation technique with **eTIM-hybrid**. Through extensive simulations, we show that both proposed solutions outperform existing methods in terms of average sum rate while maintaining an average minimum user rate for each link in both clusters.

In Chapter 3, we propose two approaches to address the problem of channel statistics estimation under frequency-flat Rayleigh fading and time-division duplex (**TDD**) mode assumption. The first approach uses a scheduling solution allowing two simultaneous transmissions and introduces two new estimators based on the maximum likelihood estimator (**MLE**) and the method of moments (**MoM**) to estimate the direct and interfering channels' magnitude mean power at each receiver. We also derive the Cramer-Rao lower bound (**CRLB**) associated with this estimation problem and propose an efficient computation of it. The second approach describes another scheduling method using simultaneous transmissions and orthogonal sequences to remove interference. Both approaches outperform **TDMA** in terms of sensing duration, at the expense of energy consumption.

In Chapter 4, we consider the problem of signaling exchange reduction between the

clusters. We propose linear and non-linear dimensionality reduction (DR) methods to compress the channel statistics that are communicated between clusters. The suitability of the proposed DR methods for interference management is evaluated in terms of average sum rate and minimum user rate per link when using the previously proposed eTIM-hybrid.

In Chapter 5, we provide concluding remarks and outline some future directions of research.

Chapter 1

Linear interference management overview

This chapter aims to review the literature about existing interference management approaches in clustered ad hoc networks. In Section 1.1, we introduce the interference management context. In Section 1.2, we cover IA's usage in single-input single-output (SISO) and provide the relevant literature. In Section 1.3, we discuss the application of TIM in SISO and review the related state-of-the-art. In Section 1.4, we provide an overview of hybrid interference management techniques combining TIM and other approaches.

1.1 Introduction

Information theory tells us [114, Section 6.1] that orthogonal multiple access (OMA) schemes such as TDMA, frequency-division multiple access (FDMA), and code-division multiple access (CDMA), are less efficient than non-orthogonal multiple access (NOMA) ones with proper interference mitigation procedures.

Several interference management techniques have recently emerged and been studied to improve the data rate of a network [70]. Examples include treating interference as noise (TIN) [23, 35] and interference neutralization [88]. However, these methods have only been studied in small topologies and present poor performance in the presence of more and stronger interference. NOMA combined with successive interference cancellation (SIC) has presented promising results [16, 51, 127], but requires a high amount of signaling exchanges and CSI knowledge which is impractical for ad hoc networks.

Therefore, for practical implementations, linear interference management strategies are needed. In this regard, we describe different classes of such techniques, namely, IA and TIM.

1.2 Interference alignment

IA is an interference management technique that aims to align the received interference by projecting them in a separate space so that the desired signal can be decoded. To this end, the precoders and decoders used by the transmitters and receivers, respectively, are carefully designed to remove interference and increase the capacity of the network.

1.2.1 Interference alignment in SISO

In [17], Cadambe and Jafar introduced an **IA** technique that we refer to as *conventional IA*. This scheme can align a large number of interfering signals, which resulted in the conclusion that wireless networks are not always limited by interference. The authors have proved that in conventional **IA** it is theoretically possible to limit interference at each receiver to about half of the received signal space, leaving the other half free of the interference for the desired signal. In an interference channel with K users, each node with a single antenna can achieve a degree-of-freedom (**DoF**) of $1/2$, resulting in a total **DoF** of $K/2$, where the **DoF** represents the maximum number of independent data streams that can be transmitted over a wireless network. This is an improvement over orthogonal access schemes such as **TDMA**, where the **DoF** is limited to $1/K$ per node, resulting in a total **DoF** of 1.

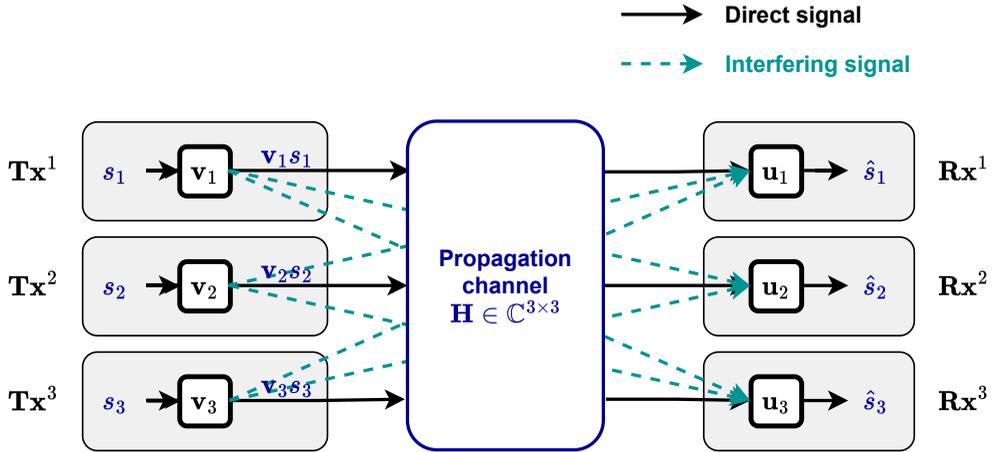


Figure 1.1: **IA** example for $N = 3$ links with a single antenna at each node.

To have a better understanding of **IA**, we consider a network composed of N links in **SISO** communicating on the same logical channel, where a link is referred to as a pair of transmitter-receivers. Each transmitter Tx^i uses a precoder $\mathbf{v}_i \in \mathbb{R}^{1 \times r}$ to transmit $s_i \mathbf{v}_i$, with s_i the information symbol, to their corresponding receiver Rx^i . This transmission can be seen as a temporal spreading of the symbol of r . A representation of this network for $N = 3$ links is illustrated in Fig. 1.1. When assuming that the propagation channel is constant over the transmissions of $s_i \mathbf{v}_i, i = 1, \dots, N$, i.e. over r channel uses, the received

signal at Rx^i can be written as:

$$\mathbf{y}_i = \sqrt{P_i}h_{ii}s_i\mathbf{v}_i^T + \sum_{\substack{j=1 \\ j \neq i}}^N \sqrt{P_j}h_{ij}s_j\mathbf{v}_j^T + \mathbf{n}_i, \quad (1.1)$$

where P_j is the transmit power of Tx^j , $h_{ij} \sim \text{CN}(0, \gamma_{ij})$ is the Rayleigh fading coefficient from transmitter Tx^j to receiver Rx^i , and $\text{CN}(\mu, \Gamma)$ stands for the complex Gaussian distribution with mean μ and variance Γ , γ_{ij} is the channel gain, $\mathbf{n}_i \sim \text{CN}(\mathbf{0}, P_n \mathbf{I}_r)$ is the additive white Gaussian noise (AWGN) where \mathbf{I}_n stands for the identity matrix of size n , $P_n = N_0 W_b$ is the noise power where N_0 is the thermal noise density and W_b is the signal bandwidth.

Each receiver Rx^i uses a decoder $\mathbf{u}_i \in \mathbb{R}^{1 \times r}$ to estimate the symbol which can be expressed as:

$$\hat{s}_i = \underbrace{\sqrt{P_i}h_{ii}s_i\mathbf{u}_i\mathbf{v}_i^T}_{\text{desired signal}} + \underbrace{\sum_{\substack{j=1 \\ j \neq i}}^N \sqrt{P_j}h_{ij}s_j\mathbf{u}_i\mathbf{v}_j^T}_{\text{interfering signals}} + \mathbf{u}_i\mathbf{n}_i. \quad (1.2)$$

To decode the intended signal s_i , conventional IA imposes the constraints that all the interference terms are removed and the desired signal is preserved over r channel uses. Therefore, the coders $\mathbf{u}_i, \mathbf{v}_i$ are designed for all links i with the following constraints:

$$\mathbf{u}_i\mathbf{v}_j^T = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise.} \end{cases} \quad (1.3)$$

Let us define the precoders and decoders matrices $\mathbf{U} := (\mathbf{u}_1, \dots, \mathbf{u}_N)^T$ and $\mathbf{V} := (\mathbf{v}_1, \dots, \mathbf{v}_N)^T$. The conventional IA aims at finding \mathbf{U}, \mathbf{V} with the following constraint:

$$\mathbf{U}\mathbf{V}^T = \mathbf{I}_N, \quad (1.4)$$

which provides a full rank matrix and can only be respected for $r = N$. One solution is to use the canonical basis, which is the same as performing TDMA.

Since we consider a single data stream and SISO systems, the DoF can be computed with the following [29, 47]:

$$\text{DoF} = \frac{1}{r}. \quad (1.5)$$

As a consequence, conventional IA which perfectly aligns the interference provides the same performance in terms of DoF as TDMA. To improve the DoF, multiple studies have proposed other types of IA schemes to design the coders \mathbf{U}, \mathbf{V} without perfect interference alignment, which decreases the coders' size r . In the following, we provide the state-of-the-art of such IA solutions.

1.2.2 Types of IA and related work

Several studies have emerged following the promising results of the DoF in IA techniques. Some works have focused on the theoretical aspect of IA by analyzing DoF theoretical bounds [15, 65, 96], while others have proposed various approaches to design IA precoders and decoders. Instead of completely removing the interference, multiple authors have proposed optimizing different objective functions such as interference leakage [40, 93], interference mean square error (MSE) [102], Signal-to-Interference-plus-Noise ratio (SINR) [24, 40], sum rate [48, 100, 113], or pairwise error probability [82]. All the previously mentioned IA methods require the knowledge of CSI. Some studies have explored IA schemes with more realistic settings such as imperfect CSI at the transmitter (CSIT) [5, 6, 65] or distributed IA using an autoencoder to design the coders [86]. While joint power control and IA solutions have been proposed using clustered IA in [131] and deep reinforcement learning in [118], a joint channel allocation and IA using graph theory is provided in [76].

Although the proposed IA schemes reduce the coders' size to increase the DoF, they require extensive CSI overhead. Such significant signaling exchanges overload the network, which is not desirable. To deal with this issue, we propose to use an interference management technique based on the topological interference management (TIM) approach, which we describe in the next section.

1.3 Topological interference management

TIM is a scheme for interference management that only considers the interference graph, i.e. determining for each receiver whether the interference caused by each other node is weak or strong. This approach can also be seen as a more general framework than traditional OMA techniques, which can be considered as special cases of TIM in some singular cases. By optimizing transmission scheduling based on topology, TIM has the potential to significantly improve the capacity and efficiency of wireless networks.

In this section, we outline the principles and limitations of conventional TIM. We then present eTIM, an enhanced version of TIM that assumes SCSIT knowledge to better manage the interference.

1.3.1 Conventional TIM

TIM [50], sometimes called topological IA (TIA) [47]¹, is a new approach that exploits IA principles under realistic assumptions on CSIT. Instead of starting with perfect CSIT knowledge like in IA and decreasing it incrementally, TIM starts with no CSIT assumption and gradually increases the available CSIT knowledge as illustrated in Fig 1.2. In this section, we study conventional TIM which only requires knowing the interference graph,

¹I am thankful to Babak Hassibi, Kishore Jaganathan, and Christos Thramboulidis for sharing the code related to this research project.



Figure 1.2: Motivation behind **TIM** from [4]. In the prior perspective, we consider **IA** with perfect **CSIT** and we incrementally decrease **CSIT** knowledge. In the new perspective, we assume no **CSIT** and we incrementally increase **CSIT** knowledge.

which is sometimes referred to in the literature as *network topology* [55] or *interference topology* [62].

1.3.1.1 **TIM** description

Ultimately, the objective behind **TIM** is to find a compromise between practical **CSIT** settings and network performance. In the **TIM** framework, the interference channel strength is represented with only one bit of **CSIT**, which significantly reduces the communication overhead. The interfering links are categorized into "weak" or "strong" based on criteria, which can depend on the received signal magnitude, pathloss, or link quality. Details on how to learn the interference graph are provided later in Section 1.3.1.3. In the following, we assume that the interference graph is already known whenever **TIM** is used.

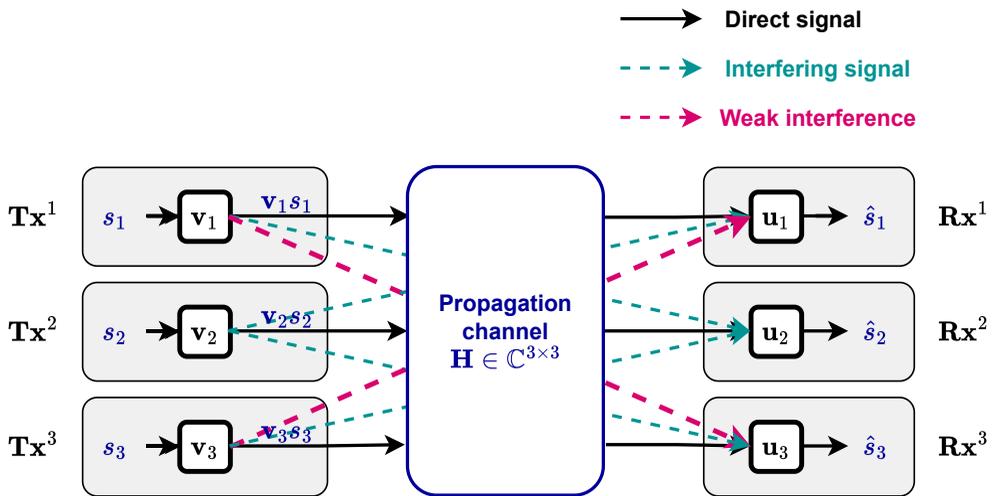


Figure 1.3: **TIM** example for $N = 3$ links with a single antenna at each node. The weak interference can be neglected.

Figs. 1.3 illustrates a **TIM** system with $N = 3$ links, where the links $\text{Tx}^1\text{-Rx}^3$ and $\text{Tx}^3\text{-Rx}^1$ are assumed to be weak interference. In the **TIM** framework, precoders and decoders are designed to remove strong interference while accepting weak interference from the

receivers. Formally, the coders for all links i are constructed to satisfy the following constraints:

$$\mathbf{u}_i \mathbf{v}_j^T = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } (i, j) \in \mathcal{S}, \\ \times & \text{otherwise,} \end{cases} \quad (1.6)$$

where \mathcal{S} is the set of strong interference links and " \times " represents an arbitrary value. The notation " \times " used in [26, 47] indicates that the interfering link (i, j) is weak and is not subjected to any constraint during the optimization of the coders \mathbf{U}, \mathbf{V} .

With this scheme, it becomes possible to construct coders \mathbf{U} and \mathbf{V} that satisfy the constraints (1.6) with smaller sizes, i.e., lower values of r . This results in a higher DoF compared to conventional IA with knowledge of the interference graph only. Following these results, many works in the literature have studied the DoF in TIM through a theoretical approach in various scenarios with partial connectivity such as the fast-fading scenarios [90], transmitter cooperation [126], alternating connectivity [7, 38] or under adversarial topology perturbation [74]. However, the works previously mentioned have not proposed schemes to design the coders \mathbf{U} and \mathbf{V} in TIM. One designed solution is to formulate a low-rank matrix completion (LRMC) problem from the constraints provided in (1.6).

1.3.1.2 TIM as a low-rank matrix completion problem

To maximize the DoF in TIM, the coders' length r needs to be minimized. One approach to achieve this is to set a matrix variable $\mathbf{X} = \mathbf{U}\mathbf{V}^T$ of rank r , namely TIM adjacency matrix, and then formulate an optimization problem minimizing the rank of \mathbf{X} while satisfying the constraints from (1.6). Thus, TIM can be viewed as an LRMC problem which can be written as follows:

Problem 1.1.

$$\mathbf{X}^* = \arg \min_{\mathbf{X} \in \mathbb{R}^{N \times N}} \text{rank}(\mathbf{X}) \quad (1.7)$$

$$\text{s.t.} \quad X_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } (i, j) \in \mathcal{S}, \end{cases} \quad (1.8)$$

where the optimal values \mathbf{U}, \mathbf{V} are then obtained from \mathbf{X}^* with matrix factorization techniques such as singular value decomposition (SVD) or QR decomposition.

Since the rank minimization problem is known to be NP-hard [18], several methods have been proposed in the literature to solve it. One such method involves minimizing the nuclear norm or trace instead of the rank function [19], but does not apply to TIM due to the structure of the adjacency matrix \mathbf{X} which requires ones on its diagonal and would return the identity matrix as the optimal solution. Other techniques to solve this

problem include alternating minimization [46, 55], Riemannian optimization [107, 123], semi-definite programming [27, 97], combining TIM with clustering [29, 56], and low-rank tensor completion in time-varying topology networks [57]. Another solution is to solve Problem 1.1 for a fixed value of r like in [47, 108, 123]. Therefore, Problem 1.1 can be formulated for $r \in \{2, \dots, N - 1\}$ as the following:

Problem 1.2.

$$\text{find } \mathbf{X} \in \mathbb{R}^{N \times N} \quad (1.9)$$

$$\text{s.t. } X_{ij} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } (i, j) \in \mathcal{S}, \end{cases} \quad (1.10)$$

$$\text{rank}(\mathbf{X}) = r. \quad (1.11)$$

Using the notations with \mathbf{U}, \mathbf{V} , the TIM problem for a given r can thus be written as:

Problem 1.3.

$$\mathbf{U}^*, \mathbf{V}^* = \arg \min_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \sum_{i=1}^N (\mathbf{u}_i \mathbf{v}_i^T - 1)^2 + \sum_{(i,j) \in \mathcal{S}} (\mathbf{u}_i \mathbf{v}_j^T)^2 \quad (1.12)$$

$$\text{s.t. } \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\}, \quad (1.13)$$

with $\|\cdot\|_2$ standing for the Euclidean norm. The norm constraints are added for power control. In practice, Problem 1.3 is solved iteratively for $r \in \{2, \dots, N - 1\}$ using techniques such as alternating minimization.

Now that we have formulated the TIM problem, we provide in the next section a new method to construct the interference graph.

1.3.1.3 New criteria for interference graph

In this section, we propose a new method to learn the interference graph, which distinguishes the strong and weak interference.

A common approach in the literature is to apply a threshold on the received power from the undesired links. In [50], the interfering links are considered weak if their collective contribution is below the noise power. However, in certain scenarios, such as high-density clusters or when the noise power is low, the influence of interference can be stronger than that of the noise, resulting in the classification of all interference as strong. This, in turn, reduces the DoF. Moreover, [50] did not propose an efficient implementation of the interference graph, which would involve comparing all possible combinations of interference sums with the noise power. Another limitation of this approach is that it classifies all the interference as weak or strong, regardless of the magnitude of the direct transmission.

To have a better representation of the link quality, [62] has incorporated the Signal-to-Noise ratio (SNR) in the interference strength criteria. Other works have computed the instantaneous SINR to decide whether the induced interference is strong or weak [20, 21, 128]. Nevertheless, in random channels, the SNR and the SINR are both random variables since both the useful and interfering signals are subject to random fluctuations. In some contexts such as vehicular-to-vehicular (V2V) [73] or device-to-device (D2D) [119] networks, the SINR feedback from the receiver to the transmitter is slower than the channel dynamics, and thus, only outdated SINR values are available at the transmitter. Consequently, it seems more relevant to decide whether an interference is strong or weak based on a probabilistic criterion on the SINR. To do this, it is necessary to compute the probability density of the SINR, which requires estimating the statistical parameters of the channel of all pairs of links.

Since we consider a statistical approach, establishing the probabilistic criterion on the SINR mentioned above requires knowing the probability density of the SINR. To do so, we define a new probabilistic criterion of the interference graph based on the outage probability of the SINR to discriminate strong interference from weak interference. The main advantage is that for any transmission topology, the outage probability of the SINR can be calculated immediately from the channel statistics. The new interference graph is defined as follows:

- Edge: for a transmission $j \rightarrow i$, the interfering link $k \rightarrow i$ corresponds to strong interference if $P(\text{SINR}_{j \rightarrow i, k} > \Gamma_0) < \tau_{thres}$, otherwise the interference is weak and the link is not represented, where Γ_0 is a threshold for SINR, τ_{thres} a threshold for the probability and the SINR can be expressed as:

$$\text{SINR}_{j \rightarrow i, k} = \frac{P_j |h_{ij}|^2}{P_n + P_k |h_{ik}|^2}. \quad (1.14)$$

- Weight: for a transmission $j \rightarrow i$, the interfering link k to i has a weight defined as:

$$w_{ik} := P(\text{SINR}_{j \rightarrow i, k} < \Gamma_0) \quad (1.15)$$

Since we assume that the channel follows a Rayleigh distribution, it can be shown using the same calculations for the SINR probability density function (pdf) provided in [33] that the outage probability of SINR can be written after simplifications in our context as follows:

$$P(\text{SINR}_{j \rightarrow i, k} > \Gamma_0) = \frac{\exp\left(-\frac{\Gamma_0 P_n}{P_j \gamma_{ij}}\right)}{1 + \Gamma_0 \frac{P_k \gamma_{ik}}{P_j \gamma_{ij}}}. \quad (1.16)$$

1.3.1.4 TIM's limits

While all of the TIM approaches in Section 1.3.1.2 have been proposed to optimize the DoF in TIM, using DoF as a performance metric has some limitations. Optimizing DoF

involves removing strong interference while tolerating weak interference. However, this neglects the fact that weak interference may not be as insignificant as assumed. The interference graph proposed in the previous section assesses the strength of the interference by considering their individual contribution, whereas the collective contributions of weak interference can become strong. This can result in significant signal deterioration and thus, reduces the sum rate. Even though [50] has proposed a threshold criterion on the sum of interfering links power, its approach cannot be applied to high-density clusters, as we mentioned earlier.

While TIM may achieve high DoF, its performance in terms of the sum rate is often poor. In fact, according to [47], TIM in SISO yields lower sum rates compared to orthogonal schemes such as TDMA when applied to hexagonal cellular arrays and ad hoc networks. Therefore, relying on TIM with only knowledge of the interference graph does not provide satisfactory results in terms of sum rate. Remaining in the TIM philosophy, we assume that we have more information on the channel and we extend TIM by incorporating SCSi in the procedure.

1.3.2 Extension of TIM

In this section, we consider an extension of the TIM procedure by integrating SCSi, and we refer to this existing extension as TIM-SCSi in this thesis. SCSi is particularly useful in practical settings of ad hoc networks where instantaneous CSI may not be available. To address the weak interference issue discussed in the previous section, [47] and [124] proposed adding a regularization term to the weak interference. As a result, the TIM-SCSi optimization problem can be formulated for $r \in \{2, \dots, N-1\}$ as follows:

Problem 1.4.

$$\mathbf{U}^*, \mathbf{V}^* = \arg \min_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \sum_{i=1}^N (\mathbf{u}_i \mathbf{v}_i^T - 1)^2 + \sum_{(i,j) \in \mathcal{S}} (\mathbf{u}_i \mathbf{v}_j^T)^2 + \sum_{(i,j) \in \bar{\mathcal{S}}} \lambda_i (\mathbf{u}_i \mathbf{v}_j^T)^2 \quad (1.17)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\}, \quad (1.18)$$

with $\bar{\mathcal{S}}$ the set of weak interference links and $\lambda_i = \frac{1}{\max_{\{j|(i,j) \in \bar{\mathcal{S}}\}} \mathbb{E}[|h_{ij}|^2]}$. This optimization problem has been solved using alternating minimization in [47] and Riemannian optimization in [124].

However, the proposed TIM-SCSi in [47, 124] provides a sum rate similar to TDMA. Furthermore, the implementation of TIM-SCSi in ad hoc networks in [47] assumes that transmitters and receivers are close to each other (less than 1 km), which is not practical in our clustered ad hoc network context. In [28], another TIM-SCSi technique that uses semi-definite programming to minimize the rank of the TIM-SCSi adjacency matrix is proposed. Nevertheless, this approach has limitations since it optimizes the DoF metric, which does not represent well the actual sum rate performance, as explained earlier.

Since the above **TIM-SCSI** methods exhibit poor performance in terms of sum rate and fairness, we propose in the next section other schemes using on hybrid interference management.

1.4 Hybrid interference management

To improve network performance, several works have explored the use of hybrid interference management methods that combine multiple techniques, such as **TIM** with **TDMA**. The goal is to identify scenarios where the network can benefit from **TIM** and use **TDMA** in cases where it may not be effective, to achieve better performance than either method alone.

The authors in [36] proposed a hybrid **TIM-TIN** solution for a 5-user network with quantized channel strength knowledge. However, this approach has limitations. First, it does not consider fairness in terms of data rate. Second, although quantized **CSIT** is more practical than perfect **CSIT**, it is still difficult to implement due to channel fluctuations. In [63], a hybrid **TIM-NOMA** solution was proposed for a 5-user network to improve its performance. Nevertheless, it does not take fairness into account as well and requires perfect **CSIT** knowledge for **NOMA**, which may not always be available in practice. Moreover, the length of the **TIM** coders in this solution was equal to the number of users, which is the same as performing **TDMA**.

Therefore, a novel approach that optimizes link grouping while ensuring fairness in terms of data rate is needed, especially in the **SCSI** context. A detailed solution that combines **eTIM** and **TDMA** in the **SCSI** context is presented in Chapter 2 to address this challenge.

1.5 Conclusion

This chapter provides an overview of the linear interference management schemes **IA** and **TIM** and their limitations in ad hoc networks. The choice of these solutions is motivated by their practical implementation, contrary to other non-linear interference management methods. Several types of **IA** algorithms have been proposed in the literature to improve the sum rate compared to **TDMA**, but require extensive **CSI** overhead. To avoid significant signaling exchanges in the network, we have proposed the use of the **TIM** since it only assumes knowing the interference graph, which indicates whether the interference is weak or strong. Multiple works have formulated **TIM** as an **LRMC** problem and have shown improvements in terms of **DoF**. However, it has been shown that despite a reduced **DoF**, **TIM** presents a lower or similar sum rate as **TDMA** due to underestimation of the influence of the weak interference. Therefore, we assume that we have **SCSI** information and consider **eTIM** which is an extension of **TIM**. Since it also provides poor performance in terms of sum rate and fairness compared to **TDMA**, we examine hybrid interference

management schemes in this thesis. This study is performed on a system composed of two adjacent clusters and under the assumption of frequency-flat Rayleigh fading channel and TDD mode.

In Chapter 2, we describe the proposed hybrid interference management solution which combines eTIM with TDMA access in an ad hoc network in the SCSI context. In Chapter 3, we propose two new scheduling procedures to estimate the channel statistics. In Chapter 4, we present a DR solution that reduces the signaling exchanges between clusters to communicate the channel statistics between the CHs.

Chapter 2

Enhanced topological interference management

2.1 Introduction

2.1.1 Context and state-of-the-art

In this chapter, we address the first goal of this thesis, which is to design interference management solutions based on **TIM** in an ad hoc network in the **SCSI** context.

To avoid the idealistic assumption requiring perfect instantaneous channel coefficients, we choose to focus on a statistical approach that we will refer to as the **SCSI** context. In this approach, we assume that the transmitters have access to the statistics of the channel, which corresponds to the channel's average power. We recall that we will propose solutions in Chapter 3 to estimate these channel magnitudes under flat Rayleigh fading. Furthermore, the **SCSI** context has the benefit of finding a balance between i) the idealistic perfect **CSIT** knowledge assumption, and ii) the no **CSIT** and 1-bit **CSIT** assumptions which provide poor performance in terms of sum rate. **SCSI** is more suitable for practical settings such as in ad hoc networks where it may not be possible to have instantaneous **CSI**. In a **SCSI** context, only a few works have implemented **TIM** in small topologies [47, 124], but not in ad hoc networks. Moreover, these solutions have the drawbacks of providing a similar sum rate to **TDMA** and having fairness issues.

Therefore, new solutions using **TIM** in the **SCSI** context are required to improve the sum rate while dealing with the fairness issues. In the rest of this chapter, we propose new schemes using the **TIM** framework in the **SCSI** context, which we refer to as **eTIM**, to manage the interference between the clusters.

2.1.2 Contributions

The contributions of this chapter are provided as follows.

- We propose an eTIM solution that uses the maximum sum-rate IA (MSR-IA) algorithm [100], which typically requires perfect CSIT. However, we adapt this method by providing an approximation of the objective function that only requires SCSi to be optimized. To the best of our knowledge, this is the first work using this approach. In addition, since maximizing the total sum rate can suffer from a lack of fairness, we also implement two techniques from the literature to deal with fairness: i) the proportional fairness optimization and ii) by introducing a per link minimum rate constraint in the eTIM optimization problem. We provide simulation results to assess the performance of these methods and compare them with the TDMA.
- Due to the unsatisfactory performance of the previous fairness schemes in terms of sum rate, we propose a novel approach that combines TDMA and eTIM. The main idea is to partition the links into two groups: in the first group, we use eTIM whereas TDMA is used in the second group. Its implementation is divided into two main stages:
 1. Links partition: the links are separated into "good links" on which we apply eTIM and "bad links" on which TDMA is used. We propose an iterative algorithm to perform this step.
 2. Scheduling optimization: we optimize the time allocation for each link within a given frame. Our proposed approach involves allocating a portion of time to the "good links" using eTIM and to the "bad links" using TDMA.
- Finally, we consider the case when two frequency bands are available and develop a joint frequency band allocation and eTIM-hybrid scheme. Our proposed scheme consists of creating two groups of links from both clusters and allocating the FBs to them. We call this scheme JFA-eTIM-hy and show through numerical simulations that it outperforms the standard orthogonal frequency allocation that assigns a different frequency band to each cluster.

2.1.3 Chapter organization

The chapter is organized as follows. In Section 3.2.1, we describe the system model. In Section 2.3, we describe the eTIM methods using an approximation of the objective function in the SCSi context. Then, in Section 2.4, we present the new scheme we propose that we call eTIM-hybrid. Afterward, in Section 2.5, we provide the proposed interference management method JFA-eTIM-hy in the context of two available frequency bands. Finally, in Section 3.4, we draw some conclusions.

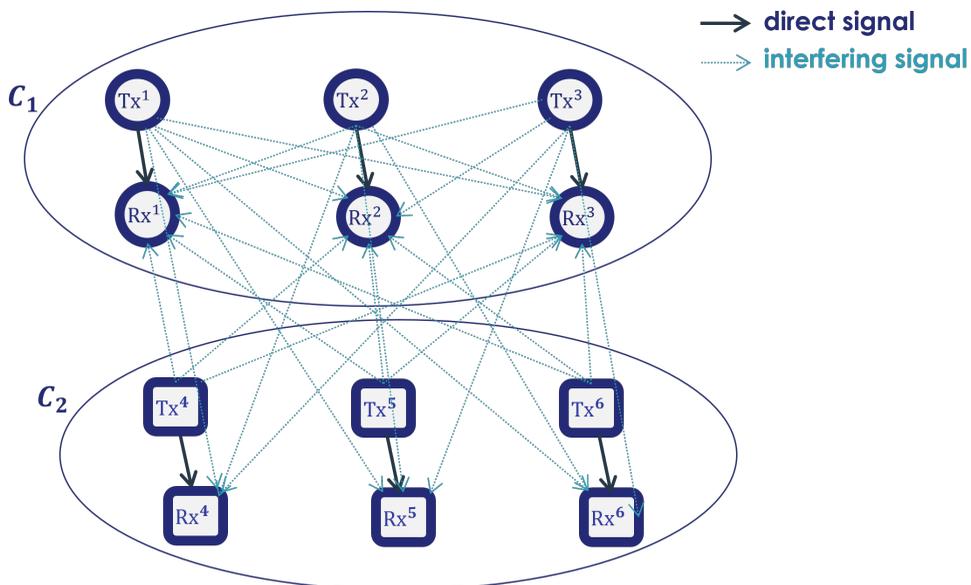


Figure 2.1: Representation of two associated clusters C_1 and C_2 with $n_{C_1} = n_{C_2} = 3$ links per cluster communicating on the same frequency band. Link i is represented as a pair $\text{Tx}^i\text{-Rx}^i$.

2.2 Preliminaries

In this section, we introduce the system model as well as the hypothesis and notations related to this chapter.

2.2.1 System model

We consider two adjacent clusters C_1 and C_2 communicating on the same frequency band, which creates interference. Each cluster C_k contains n_{C_k} links, where link i is defined as a pair of transmitter-receiver $\text{Tx}^i\text{-Rx}^i$. Thus, the whole network composed of both clusters contains $N := n_{C_1} + n_{C_2}$ links. In Fig. 2.1, we represent an example of two adjacent clusters with $n_{C_1} = n_{C_2} = 3$ links per cluster with the direct and interfering links.

Each received signal is corrupted by intra-cluster (from interfering transmitters in the same cluster) and inter-cluster (from interfering transmitters in the other cluster) interference. Like in Section 1.2.1, transmitter Tx^i uses a precoder $\mathbf{v}_i \in \mathbb{R}^{1 \times r}$ to send the information symbol s_i , from which we can derive the following received signal at Rx^i :

$$\mathbf{y}_i = \sum_{j=1}^N \mathbf{v}_j^T \sqrt{P_j} \gamma_{ij} \tilde{h}_{ij} s_j + \mathbf{n}_i, \quad (2.1)$$

where P_j is the transmit power of Tx^j , γ_{ij} is the channel gain, $\tilde{h}_{ij} \sim \mathcal{CN}(0, 1)$ is the Rayleigh fading coefficient from transmitter Tx^j to receiver Rx^i , $\mathbf{n}_i \sim \mathcal{CN}(\mathbf{0}, P_n \mathbf{I}_r)$ is the

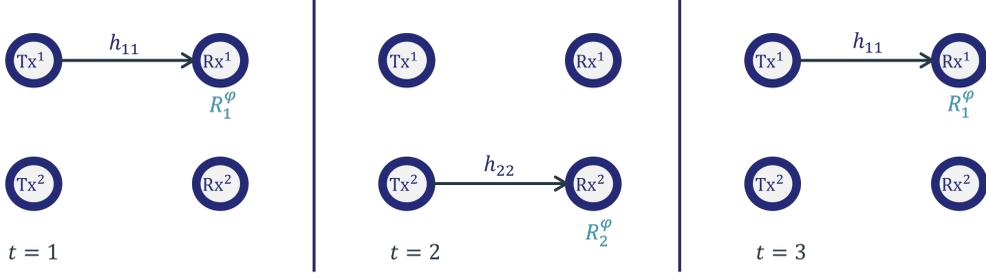


Figure 2.2: Representation of $N = 2$ links for $L = 3$ slots. The physical rates R_1^ϕ, R_2^ϕ are related to the physical layer. The user rates which are related to the protocol layer are provided as: $R_{\text{TDMA},1}^u = \frac{2}{3}R_1^\phi$ and $R_{\text{TDMA},2}^u = \frac{1}{3}R_2^\phi$.

AWGN, $P_n = N_0W_b$ is the noise power, N_0 is the thermal noise density and W_b is the signal bandwidth.

Then, after applying the decoder \mathbf{u}_i at receiver Rx^i , the estimated symbol can be expressed as:

$$\hat{s}_i = \mathbf{u}_i \mathbf{y}_i, \quad (2.2)$$

which can be rewritten by plugging (2.1) in (2.2) as:

$$\hat{s}_i = \mathbf{u}_i \mathbf{v}_i^T \sqrt{P_i \gamma_{ii}} \tilde{h}_{ii} s_i + \sum_{\substack{j=1 \\ j \neq i}}^N \mathbf{u}_i \mathbf{v}_j^T \sqrt{P_j \gamma_{ij}} \tilde{h}_{ij} s_j + \mathbf{u}_i \mathbf{n}_i. \quad (2.3)$$

In the **SCSI** context, we assume that the channel magnitude γ_{ij} is perfectly known. In addition, we also suppose that the noise power P_n and the transmit powers are known.

2.2.2 Rate expressions

In this section, we describe how to calculate the sum rate of the overall network composed of the links of both clusters C_1 and C_2 , to assess the performance of our interference management methods. In the following, we provide the rates expression when applying **TIM** and **TDMA** framework.

2.2.2.1 Rate expression for **TDMA** framework

We first consider the **TDMA** framework where transmissions are performed without interference. The physical rate is defined as the rate related to the physical layer, which can be written at receiver Rx^i as:

$$R_i^\phi := W_b \log_2(1 + \text{SNR}_i), \quad (2.4)$$

where the **SNR** at Rx^i can be expressed as:

$$\text{SNR}_i := \frac{P_i \gamma_{ii} |\tilde{h}_{ii}|^2}{P_n}, \quad (2.5)$$

with $:=$ is the notation for defining a variable. Let us define the user rate which is related to the protocol layer and is equal to the physical rate multiplied by the proportion of time allocated to each link for transmission. We consider a frame containing L time slots that we refer to as a "frame of length L ". The term *slot* represents the shortest time interval of the frame and we aim at optimizing its allocation to the links of the network. Then, we define n_i as the number of transmissions for each link $i \in \{1, \dots, N\}$ of the network where each transmission has a duration of one slot. We assume $L \geq N$ so that every link can be allocated at least one transmission. The user rate at receiver Rx^i on frame of length L can thus be written as:

$$R_{\text{TDMA},i}^u := \frac{n_i}{L} R_i^\varphi. \quad (2.6)$$

Fig. 2.2 represents an example of a communication on $L = 3$ slots and $N = 2$ links to distinguish physical and user rates.

In addition, the sum rate of the network is defined as the sum of the user rates of each receiver of the network. When applying the **TDMA** scheme, the sum rate can be written as:

$$R_{\text{sum}}^{\text{TDMA}} := \sum_{i=1}^N R_{\text{TDMA},i}^u. \quad (2.7)$$

Then, we respectively define the average physical/user/sum rate as:

$$\bar{R}_i^\varphi := \mathbb{E} \left[R_i^\varphi \right], \quad (2.8a)$$

$$\bar{R}_{\text{TDMA},i}^u := \mathbb{E} \left[R_{\text{TDMA},i}^u \right], \quad (2.8b)$$

$$\bar{R}_{\text{sum}}^{\text{TDMA}} := \mathbb{E} \left[R_{\text{sum}}^{\text{TDMA}} \right], \quad (2.8c)$$

where the expected value is estimated using Monte-Carlo simulations on several channel realizations. It is worth mentioning that what we call "average sum rate" is sometimes referred to as the "ergodic capacity" in the literature [79, 112].

Moreover, we consider in the **TDMA** framework a scheme using the round-robin algorithm which consists of allocating portions of time to each link in equal portions and circular order, i.e. the nodes transmit successively one after each other in a loop [67]. This method is referred to as "round-robin **TDMA**". When using round-robin **TDMA** on a long frame, i.e. when L is high, we can state the following approximation:

$$n_i \approx \frac{L}{N}, \quad \forall i \in \{1, \dots, N\}, \quad (2.9)$$

from which we can deduce the asymptotic user rate at receiver Rx^i for high values of L which can be written as:

$$R_{\text{RR-TDMA},i}^u := \frac{R_i^\varphi}{N}. \quad (2.10)$$

We define the average user rate when using round-robin **TDMA** which can be written as:

$$\bar{R}_{\text{RR-TDMA},i}^u := \frac{\bar{R}_i^\varphi}{N}, \quad (2.11)$$

with \bar{R}_i^φ defined in (2.8a).

2.2.3 Rate expression for TIM framework

When using the TIM scheme, simultaneous transmissions are performed for r slots which generates interference between nodes. The physical rate at receiver Rx^i can be written as:

$$R_{\text{TIM},i}^\varphi := W_b \log_2(1 + \text{SINR}_i), \quad (2.12)$$

where the SINR at Rx^i can be expressed as:

$$\text{SINR}_i = \frac{P_i \gamma_{ii} |\tilde{h}_{ii}|^2 |\mathbf{u}_i \mathbf{v}_i^T|^2}{P_n \|\mathbf{u}_i\|^2 + \sum_{j \neq i} P_j \gamma_{ij} |\tilde{h}_{ij}|^2 |\mathbf{u}_i \mathbf{v}_j^T|^2}. \quad (2.13)$$

The user rate is defined as the physical rate divided by the number of slots required to apply eTIM which is r . Formally, it can be written as:

$$R_{\text{TIM},i}^u := \frac{1}{r} R_{\text{TIM},i}^\varphi. \quad (2.14)$$

In addition, the sum rate of the network can be expressed as:

$$R_{\text{sum}}^{\text{TIM}} := \sum_{i=1}^N R_{\text{TIM},i}^u. \quad (2.15)$$

The average physical/user/sum rate can be written as:

$$\bar{R}_{\text{TIM},i}^\varphi := \mathbb{E} [R_{\text{TIM},i}^\varphi], \quad (2.16a)$$

$$\bar{R}_{\text{TIM},i}^u := \mathbb{E} [R_{\text{TIM},i}^u], \quad (2.16b)$$

$$\bar{R}_{\text{sum}}^{\text{TIM}} := \mathbb{E} [R_{\text{sum}}^{\text{TIM}}], \quad (2.16c)$$

which can be estimated using Monte-Carlo simulations.

2.3 Enhanced topological interference management

In this section, we propose several interference management methods to improve the network sum rate compared to TDMA in the SCSi context. To do so, we propose to combine the MSR-IA algorithm [100, 113] which aims at maximizing the sum rate of the network assuming perfect CSIT, and the "hardening bound" approximation [80] to lower the complexity of the optimization problem. This method will be referred to as maximum sum-rate eTIM (MSR-eTIM). Moreover, since maximum sum rate algorithms are known for lacking fairness between users, we extend the proposed eTIM solution to deal with fairness. To this end, we implement state-of-the-art algorithms and show the limit of these schemes through numerical simulations.

This section is organized as follows. First, we present the hardening bound approximation, before using it to formulate the MSR-eTIM optimization problem and detail how to solve it. Then, we describe the eTIM solutions dealing with fairness that we have implemented. Finally, numerical simulations are provided to assess the performance of each method.

2.3.1 Hardening bound approximation

We consider the [MSR-IA](#) algorithm by optimizing the ergodic capacity instead of the sum rate since we assume the [SCSI](#) context. For a given coders' length r , the optimization problem to solve can be written as:

Problem 2.1.

$$\mathbf{U}_{\text{eTIM}}^* \mathbf{V}_{\text{eTIM}}^* = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \bar{R}_{\text{sum}}^{\text{eTIM}} \quad (2.17)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\}, \quad (2.18)$$

where $\bar{R}_{\text{sum}}^{\text{eTIM}} := \sum_{i=1}^N \bar{R}_{\text{eTIM},i}^u$ and $\bar{R}_{\text{eTIM},i}^u$ is defined using (2.16b), $\mathbf{U} := (\mathbf{u}_1, \dots, \mathbf{u}_N)^T$, $\mathbf{V} := (\mathbf{v}_1, \dots, \mathbf{v}_N)^T$ and $\|\cdot\|_2$ corresponds to the Euclidean norm.

This optimization problem presents two complexity issues:

- A numerical complexity issue: in the objective function, because of the expected value term, the rates $\bar{R}_{\text{eTIM},i}^u$ are evaluated using a numerical procedure like with the Monte-Carlo method or a numerical integration scheme. This estimation is required at each iteration when using an iterative framework for Problem 2.1, leading to high numerical complexity and computational inefficiencies.
- An algorithmic complexity issue: although no formal proof is provided, it is stated in [100, 113] that Problem 2.1 is neither convex nor concave with respect to \mathbf{U} or \mathbf{V} .

To deal with the numerical complexity issue, we propose to use the "hardening bound" approximation from [80, Section 3.2]. Usually, this approximation is applied in massive multi-input multi-output ([MIMO](#)) systems in the presence of the phenomenon called channel hardening where the small-scale fading effects disappear for a large number of antennas. However, it has been shown in [85] that this approximation performs well even in the absence of channel hardening. Although this approximation is suboptimal when applied in [SISO](#), we will show through numerical simulations that it provides good results. Since the "hardening bound" approximation consists of neglecting the effects of small-scale fading, in the [SINR](#) expression in (2.13), we replace the terms $|\tilde{h}_{ij}|^2$ by its expected value which is equal to 1. In the objective function, the term $\bar{R}_{\text{eTIM},i}^u$ is thus approximated by the hardening bound which can be written as:

$$\bar{R}_{\text{eTIM},i}^{\text{hard}} := \log_2(1 + \text{SINR}_i^{\text{hard}}), \quad (2.19)$$

where the approximation $\text{SINR}_i^{\text{hard}}$ can be expressed as:

$$\text{SINR}_i^{\text{hard}} := \frac{P_i \gamma_{ii} |\mathbf{u}_i \mathbf{v}_i^T|^2}{P_n \|\mathbf{u}_i\|_2^2 + \sum_{\substack{j=1 \\ j \neq i}}^N P_j \gamma_{ij} |\mathbf{u}_i \mathbf{v}_j^T|^2}. \quad (2.20)$$

Thanks to the hardening bound approximation, we can avoid using Monte-Carlo methods to estimate (2.16b), which solves the numerical complexity issue. Using this approximation, in the next section, we propose a framework to deal with the algorithmic complexity issue.

2.3.2 Max sum rate eTIM

2.3.2.1 Optimization problem formulation

It is worth noticing that when using the hardening bound approximation described previously, the objective function to optimize is an approximation of the average sum rate instead of the average U sum rate itself. Yet, with an abuse of notations, we refer to this method as MSR-eTIM. The related optimization problem for a given coders' length r can be written as:

Problem 2.2.

$$\mathbf{U}_{\text{eTIM}}^* \mathbf{V}_{\text{eTIM}}^* = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \bar{R}_{\text{sum}}^{\text{hard}} \quad (2.21)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\}, \quad (2.22)$$

with $\bar{R}_{\text{sum}}^{\text{hard}} := \sum_{i=1}^N \bar{R}_{\text{eTIM},i}^{\text{hard}}$ and $\bar{R}_{\text{eTIM},i}^{\text{hard}}$ is defined in (2.19).

As reminded in Chapter 1, Problem 2.2 is solved for a given coders' length r . To find the optimal value of r , a brute-force approach has been implemented, which consists of solving Problem 2.2 for each $r = 2, \dots, N$, where the case $r = N$ leads to a suboptimal solution of round-robin TDMA. Then, we choose the coders' length r giving the highest average sum rate by using Monte-Carlo simulations. This procedure is described in Algorithm 2.1 for any optimization problem based on eTIM.

Algorithm 2.1: Procedure to find the best coders' length r^* with eTIM for N links

for $r = 2, \dots, N$ **do**

Find the solutions $\mathbf{U}_{\text{eTIM}}^*, \mathbf{V}_{\text{eTIM}}^* \in \mathbb{R}^{N \times r}$ of the eTIM problem, e.g. Problem 2.1 or 2.2.

Get $\bar{R}_{\text{sum}}^{\text{TIM}}(r)$ from (2.16) with N_{MC} Monte-Carlo simulations.

end

Return $r^* = \arg \max_{r \in \{2, \dots, N\}} \bar{R}_{\text{sum}}^{\text{TIM}}(r)$.

Like Problem 2.1, Problem 2.2 has a similar objective function to the one in [100, 113] which states that Problem 2.2 is neither convex nor concave with respect to \mathbf{U} or \mathbf{V} . Hence, to deal with this algorithmic complexity issue, we implement a suboptimal method combining alternating optimization (AO) and gradient descent like in [100, 113].

To simplify the notations, during the AO, the term $\bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{u}_1, \dots, \mathbf{u}_N, \mathbf{v}_1, \dots, \mathbf{v}_N)$ designates $\bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{U}, \mathbf{V})$.

2.3.2.2 Alternating optimization

AO is an iterative procedure that optimizes alternately between the optimization variables until convergence is reached, but cannot guarantee the optimality of the convergence point. Like in [113]'s AO implementation, we choose to alternate between the optimization variables $\mathbf{u}_1, \dots, \mathbf{u}_N, \mathbf{v}_1, \dots, \mathbf{v}_N$. To maximize $R_{\text{sum}}^{\text{eTIM}}$, at iteration t , we execute the following procedure:

We first formulate the optimization problem alternately between the decoders, which can be expressed for all $k \in \{1, \dots, N\}$ as:

Problem 2.3.

$$\mathbf{u}_k^{(t)} = \arg \max_{\mathbf{u}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{u}_1^{(t)}, \dots, \mathbf{u}_{k-1}^{(t)}, \mathbf{u}_k, \mathbf{u}_{k+1}^{(t-1)}, \dots, \mathbf{u}_N^{(t-1)}, \mathbf{v}_1^{(t-1)}, \dots, \mathbf{v}_N^{(t-1)}) \quad (2.23)$$

$$\text{s.t.} \quad \|\mathbf{u}_k\|_2^2 = 1, \quad (2.24)$$

where $\mathbf{u}_1^{(t)}, \dots, \mathbf{u}_{k-1}^{(t)}, \mathbf{u}_{k+1}^{(t-1)}, \dots, \mathbf{u}_N^{(t-1)}, \mathbf{v}_1^{(t-1)}, \dots, \mathbf{v}_N^{(t-1)}$ are fixed.

Then, we formulate the optimization problem alternately between the precoders, which can be written for all $k \in \{1, \dots, N\}$ as:

Problem 2.4.

$$\mathbf{v}_k^{(t)} = \arg \max_{\mathbf{v}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{u}_1^{(t)}, \dots, \mathbf{u}_N^{(t)}, \mathbf{v}_1^{(t)}, \dots, \mathbf{v}_{k-1}^{(t)}, \mathbf{v}_k, \mathbf{v}_{k+1}^{(t-1)}, \dots, \mathbf{v}_N^{(t-1)}) \quad (2.25)$$

$$\text{s.t.} \quad \|\mathbf{v}_k\|_2^2 = 1,$$

where $\mathbf{u}_1^{(t)}, \dots, \mathbf{u}_N^{(t)}, \mathbf{v}_1^{(t)}, \dots, \mathbf{v}_{k-1}^{(t)}, \mathbf{v}_{k+1}^{(t-1)}, \dots, \mathbf{v}_N^{(t-1)}$ are fixed.

2.3.2.3 Gradient descent methods

Now that we have formulated the proposed AO procedure, we propose to use a numerical procedure based on gradient descent to solve Problems 2.3 and 2.4. In [113], similar optimization problems were formulated to determine the linear precoders in the context of IA. In the following, we adapt these methods to obtain both the precoders and the decoders.

Let us first explain our optimization method for Problem 2.3.

2.3.2.3.1 Optimization method for Problem 2.3

Since applying gradient descent is easier on an unconstrained optimization problem, we remove the norm constraint in Problem 2.3 by performing the following change of variables: $\mathbf{u}_k = \frac{\tilde{\mathbf{u}}_k}{\|\tilde{\mathbf{u}}_k\|_2}$ for $k \in \{1, \dots, N\}$ in Problem 2.3.

After applying the change of variables at iteration t , Problem 2.3 can be equivalently rewritten as:

Problem 2.5.

$$\tilde{\mathbf{u}}_k^{(t)} = \arg \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum}, \mathbf{V}}^{\text{hard}}(\tilde{\mathbf{u}}_1^{(t)}, \dots, \tilde{\mathbf{u}}_{k-1}^{(t)}, \tilde{\mathbf{u}}_k, \tilde{\mathbf{u}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{u}}_N^{(t-1)}), \quad (2.26)$$

where $\bar{R}_{\text{sum}, \mathbf{V}}^{\text{hard}} := \sum_{i=1}^N \bar{R}_{\mathbf{V}, i}^{\text{hard}}$ and $\bar{R}_{\mathbf{V}, i}^{\text{hard}}$ is defined as:

$$\bar{R}_{\mathbf{V}, i}^{\text{hard}} := \log_2 \left(1 + \frac{\beta_{\mathbf{V}, ii} |\tilde{\mathbf{u}}_i \mathbf{V}_i^T|^2}{P_n + \sum_{\substack{j=1 \\ j \neq i}}^N \beta_{\mathbf{V}, ij} |\tilde{\mathbf{u}}_i \mathbf{V}_j^T|^2} \right), \quad (2.27)$$

with $\beta_{\mathbf{V}, ij} := \frac{P_j \gamma_{ij}}{\|\tilde{\mathbf{u}}_i\|_2^2}$ and $\tilde{\mathbf{U}} := (\tilde{\mathbf{u}}_1, \dots, \tilde{\mathbf{u}}_N)^T$.

Afterward, we apply a gradient descent algorithm to Problem 2.5 until convergence. To this end, after several calculations, we compute the partial derivative of $\bar{R}_{\text{sum}, \mathbf{V}}^{\text{eTIM}}$ with respect to $\tilde{\mathbf{u}}_k$ and provide its expression in Appendix A.6. The gradient descent algorithm is performed with a controlled step to obtain $\tilde{\mathbf{u}}_k^*$, the details of its implementation are provided in Algorithm 2.2. Finally, the solution \mathbf{u}_k^* of Problem 2.3 is obtained after applying the change of variables $\mathbf{u}_k^* = \frac{\tilde{\mathbf{u}}_k^*}{\|\tilde{\mathbf{u}}_k^*\|_2}$.

Algorithm 2.2: Gradient descent with step control to solve $\arg \max_{\mathbf{x} \in \mathbb{R}^r} f(\mathbf{x})$ [14]

Set $\epsilon_g > 0$.

Initialize $\mathbf{x}^{(0)}$.

while $\|\nabla f(\mathbf{x})\|_2^2 > \epsilon_g$ **do**

 Initialize $\alpha^{(p)} = 1$.

while $f(\mathbf{x}^{(p+1)}) \leq f(\mathbf{x}^{(p)} - \alpha^{(p)} \nabla f(\mathbf{x}^{(p)}))$ **do**

 Update $\mathbf{x}^{(p+1)} = \mathbf{x}^{(p)} - \alpha^{(p)} \nabla f(\mathbf{x}^{(p)})$.

 Update $\alpha^{(p)} = \alpha^{(p)} / 2$.

end

 Increment $p = p + 1$.

end

Return $\mathbf{x}^{(p+1)}$

Now, let us describe the optimization method for Problem 2.4.

2.3.2.3.2 Optimization method for Problem 2.4

Similarly, as the previous section, we perform the change of variables $\mathbf{v}_k = \frac{\tilde{\mathbf{v}}_k}{\|\tilde{\mathbf{v}}_k\|_2}$ for $k \in \{1, \dots, N\}$ in Problem 2.4 and formulate the equivalent unconstrained optimization problem:

Problem 2.6.

$$\tilde{\mathbf{v}}_k^{(t)} = \arg \max_{\tilde{\mathbf{v}}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum,U}}^{\text{hard}}(\tilde{\mathbf{v}}_1^{(t)}, \dots, \tilde{\mathbf{v}}_{k-1}^{(t)}, \tilde{\mathbf{v}}_k, \tilde{\mathbf{v}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{v}}_N^{(t-1)}), \quad (2.28)$$

with $\bar{R}_{\text{sum,U}}^{\text{hard}} := \sum_{i=1}^N \bar{R}_{\text{U},i}^{\text{hard}}$ and where $\bar{R}_{\text{U},i}^{\text{hard}}$ is defined as:

$$\bar{R}_{\text{U},i}^{\text{hard}} := \log_2 \left(1 + \frac{\beta_{\text{U},ii} |\mathbf{u}_i \tilde{\mathbf{v}}_i^T|^2}{P_n + \sum_{\substack{j=1 \\ j \neq i}}^N \beta_{\text{U},ij} |\mathbf{u}_i \tilde{\mathbf{v}}_j^T|^2} \right), \quad (2.29)$$

with $\beta_{\text{U},ij} := \frac{P_j \gamma_{ij}}{\|\tilde{\mathbf{v}}_j\|_2^2}$ and $\tilde{\mathbf{V}} := (\tilde{\mathbf{v}}_1, \dots, \tilde{\mathbf{v}}_N)^T$.

Afterward, the expression of the partial derivative of $\bar{R}_{\text{sum,U}}^{\text{hard}}$ with respect to $\tilde{\mathbf{v}}_k$ to compute the gradient descent algorithm is derived in Appendix 2.6. To implement it, we apply the gradient descent algorithm described in Algorithm 2.2 to Problem 2.6, and we get $\tilde{\mathbf{v}}_k^*$. Finally, the solution of Problem 2.4 is $\mathbf{v}_k^* = \frac{\tilde{\mathbf{v}}_k^*}{\|\tilde{\mathbf{v}}_k^*\|_2}$.

2.3.2.4 Analysis of AO's implementation

It is worth mentioning that we have initially implemented three procedures for Problem 2.2 and that we have only presented in the previous section the one providing the best performance in terms of sum rate and convergence speed. The two alternatives are described as follows:

- Use of a global gradient descent approach without any alternating optimization step. Although it provides a similar sum rate as the proposed AO implementation, we have observed through simulations that the implemented gradient algorithm was much slower in this case, especially in the last steps.
- Use of a gradient descent approach and optimize alternately between two terms: \mathbf{U} and \mathbf{V} . Nevertheless, we have observed through simulations that it would lead to poorer performance in terms of sum rate.

In the end, the details of our AO-based algorithm are summarized in Algorithm 2.3.

2.3.3 Implementation of fair solutions

Since max sum-rate (MSR) algorithms are known to lack in terms of fairness, we extend the MSR-eTIM optimization problem to deal with fairness. To this end, we implement two state-of-the-art approaches which are often used in the literature:

- the proportional fairness optimization [44].
- the MSR under minimum rate constraints which consists of adding a minimum rate constraint per user in the MSR-eTIM optimization problem [61].

Algorithm 2.3: AO procedure for MSR-eTIM, i.e. for Problem 2.2

```

Set  $\epsilon_{AO} > 0, t = 1, \Delta_{AO} = \epsilon_{AO} + 1$ .
Randomly initialize  $\mathbf{U}^{\text{init}}, \mathbf{V}^{\text{init}} \in \mathbb{R}^{N \times r}$ .
Set for all  $i, \mathbf{u}^{(0)} = \mathbf{u}^{\text{init}} / \|\mathbf{u}^{\text{init}}\|_2$  and  $\mathbf{v}^{(0)} = \mathbf{v}^{\text{init}} / \|\mathbf{v}^{\text{init}}\|_2$ .
while  $\Delta_{AO} > \epsilon_{AO}$  do
    for  $k = 1, \dots, N$  do
        Apply Algorithm 2.2 to Problem 2.5 and get  $\tilde{\mathbf{u}}_k^{(t)}$ .
        Apply the change of variables  $\mathbf{u}_k^{(t)} = \frac{\tilde{\mathbf{u}}_k^{(t)}}{\|\tilde{\mathbf{u}}_k^{(t)}\|_2}$ .
    end
    for  $k = 1, \dots, N$  do
        Apply Algorithm 2.2 to Problem 2.6 and get  $\tilde{\mathbf{v}}_k^{(t)}$ .
        Apply the change of variables  $\mathbf{v}_k^{(t)} = \frac{\tilde{\mathbf{v}}_k^{(t)}}{\|\tilde{\mathbf{v}}_k^{(t)}\|_2}$ .
    end
    Compute  $\bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{U}^{(t)}, \mathbf{V}^{(t)}) = \sum_{i=1}^N \bar{R}_{\text{eTIM},i}^{\text{hard}}(\mathbf{U}^{(t)}, \mathbf{V}^{(t)})$  with  $\bar{R}_{\text{eTIM},i}^{\text{hard}}$  defined in (2.19).
    Set  $\Delta_{AO} = |\bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{U}^{(t)}, \mathbf{V}^{(t)}) - \bar{R}_{\text{sum}}^{\text{hard}}(\mathbf{U}^{(t-1)}, \mathbf{V}^{(t-1)})|$ .
    Increment  $t = t + 1$ .
end
return  $\mathbf{U}^{(t+1)}, \mathbf{V}^{(t+1)}$ 

```

2.3.3.1 Proportional fairness

The proportional fairness problem consists in maximizing the product of the rates, instead of the sum of the rates like in Problem 2.2. With this approach, the algorithm avoids allocating rates close to 0 since the product of the rates strongly decreases otherwise.

Since the logarithm function is strictly increasing, the proportional fairness optimization problem can be written as:

Problem 2.7.

$$\mathbf{U}_{\text{PF eTIM}}^* \mathbf{V}_{\text{PF eTIM}}^* = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \sum_{i=1}^N \log \bar{R}_{\text{eTIM},i}^{\text{hard}} \quad (2.30)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\}. \quad (2.31)$$

In addition, using the partial derivatives of $\log \bar{R}_{\text{eTIM},i}^{\text{eTIM}}$ whose expression can be written

as:

$$\begin{aligned}\frac{\partial \log \bar{R}_{\text{eTIM},i}^{\text{hard}}}{\partial \bar{\mathbf{u}}_k} &= \frac{1}{\bar{R}_{\text{eTIM},i}^{\text{hard}}} \frac{\partial \bar{R}_{\text{eTIM},i}^{\text{hard}}}{\partial \bar{\mathbf{u}}_k} \\ \frac{\partial \log \bar{R}_{\text{eTIM},i}^{\text{hard}}}{\partial \bar{\mathbf{v}}_k} &= \frac{1}{\bar{R}_{\text{eTIM},i}^{\text{hard}}} \frac{\partial \bar{R}_{\text{eTIM},i}^{\text{hard}}}{\partial \bar{\mathbf{v}}_k},\end{aligned}\tag{2.32}$$

we can perform an **AO** and gradient descent like in Section 2.3.2 to solve this Problem 2.7. This method is referred to as proportional fairness **eTIM** (**PF-eTIM**).

2.3.3.2 eTIM Max sum rate with minimum rate constraint

To overcome the fairness issue of **MSR-eTIM**, one conventional approach is to add a minimum user rate constraint for every link while maximizing the sum rate. Hence, in this section, we formulate the corresponding optimization problem and propose to solve it using the subgradient method.

After adding minimum rate constraints to Problem 2.2, the optimization problem is formulated as follows:

Problem 2.8.

$$\mathbf{U}_{\text{CeTIM}}^*, \mathbf{V}_{\text{CeTIM}}^* = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N \times r}} \bar{R}_{\text{sum}}^{\text{hard}}\tag{2.33}$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \{1, \dots, N\},\tag{2.34}$$

$$\frac{\bar{R}_{\text{eTIM},i}^{\text{hard}}}{r} \geq R_0^u, \quad \forall i \in \{1, \dots, N\},\tag{2.35}$$

where R_0^u is the minimum user rate constraint for every link that can be set by the system designer to meet users' rate needs.

To solve Problem 2.8, we develop a solution called constrained **MSR-eTIM** (**CMSR-eTIM**) which combines **AO** and the subgradient method. The details of its implementation are provided in Appendix A.2.

2.3.4 Numerical results

In this section, we present numerical simulations to assess the performance of the solutions described in Sections 2.3.2 and 2.3.3 in terms of average sum rate and minimum user rate per link. The goal is to compare the performance of these methods with **TDMA**.

2.3.4.1 Setup

We assume two square clusters of side L_1 separated by a distance δ and we define $d_{\text{size}} := \sqrt{2}L_1$ as the length of the diagonal of the cluster. In each cluster, we uniformly generate n_C transmitter nodes and n_C receiver nodes in the $L_1 \times L_1$ square. It is worth

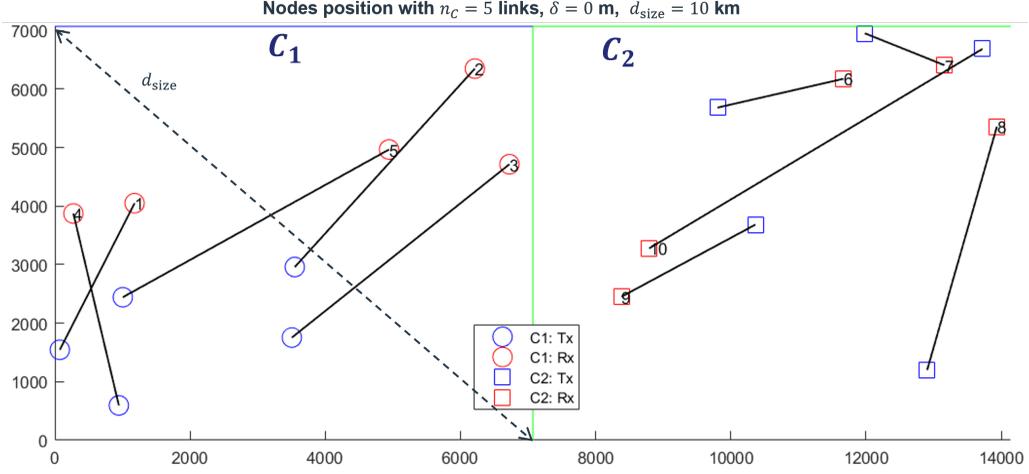


Figure 2.3: Wireless ad hoc clustered network with $K = 2$ clusters composed of $n_C = 5$ links per cluster separated, both clusters are separated from $\delta = 0$ m and with $d_{size} = 10$ km. Each node is uniformly distributed in an $L_1 \times L_1$ square.

mentioning that [24, 47, 130] work under the assumption the receivers are randomly placed close to the corresponding transmitter (≤ 1 km). This assumption is not realistic when implementing ad hoc networks since the nodes can be anywhere in the cluster. Moreover, the works [36, 40, 93, 100, 113] have only considered small topologies (less than 5 links), whereas we implement our solutions for $n_C = [5, 10, 15, 20]$ links per cluster which are more realistic for real-world communication scenarios, such as military or 5G networks. An example of a randomly generated link topology in the clusters is illustrated in Fig. 2.3. All transmit powers are equal, i.e. $\forall i, P_i = P_{Tx}$. Parameters are listed in Table 2.1.

Moreover, in the simulations, the solutions calculating precoders and decoders \mathbf{U}, \mathbf{V} are run 10 times with different random initializations and we keep the one providing the highest average sum rate. The average sum rates and minimum user rates are estimated with Monte-Carlo methods on 10^5 random channel realizations. In the figures, the sum rate and minimum use rate per link are averaged over 100 random topologies where in each simulation, the nodes' position is different.

2.3.4.1.1 Channel model

To model the propagation channel, we set a distance d_{range} such that a transmitter and a receiver separated by d_{range} provide a fixed given average SNR, denoted as SNR_{range} , that we refer to as the *target average SNR*. Moreover, in our simulations, the channel average gain γ_{ij} is computed with a three-slope model which can be written as:

$$\gamma_{ij} = A d_{ij}^{-2} \left(1 + \frac{d_{ij}}{100}\right)^{-1} \left(1 + \frac{d_{ij}}{1000}\right)^{-1}, \quad (2.36)$$

Table 2.1: Simulation parameters values

P_{Tx}	1 W	W_B	1 MHz	P_n	$4.05 \cdot 10^{-15}$	d_{range}	10 km
δ	0 m	k_B	$1.38 \cdot 10^{-23} \text{ WK}^{-1}\text{s}$	d_{size}	10 km		
L_1	7.07 km	T	293 K	$\text{SNR}_{\text{range}}$	10 dB		

where d_{ij} is the distance in meters between Tx^i and Rx^i , A is set such that the target $\text{SNR}_{\text{range}}$ is achieved when $d_{ij} = d_{\text{range}}$. The target average SNR is defined as:

$$\text{SNR}_{\text{range}} := \mathbb{E} \left[\frac{P_{\text{Tx}} \gamma_{\text{range}} |\tilde{h}|^2}{P_n} \right], \quad (2.37)$$

which is equal to:

$$\text{SNR}_{\text{range}} := \frac{P_{\text{Tx}} \gamma_{\text{range}}}{P_n}, \quad (2.38)$$

with γ_{range} computed using (2.36) for $d_{ij} = d_{\text{range}}$ and $\tilde{h} \sim \mathcal{CN}(0, 1)$.

2.3.4.1.2 Minimum user rate constraint

In this paragraph, we set the minimum user rate constraint for every link R_0^u to ensure that the problem is feasible, i.e. we can find TIM coders \mathbf{U}, \mathbf{V} guaranteeing this rate constraint. In our simulations, we choose to set R_0^u as the worst user rate receivers can get when applying round-robin TDMA defined in Section 2.2.2.1, i.e. without interference. In other words, R_0^u is equal to the average of the physical rate corresponding to a transmitter and receiver separated from a distance d_{size} divided by the total number of links N , which can be written as:

$$R_0^u := \frac{W_b}{N} \mathbb{E}[\log_2(1 + \text{SNR}_{\text{worst}})], \quad (2.39)$$

where $\text{SNR}_{\text{worst}}^u$ can be expressed as:

$$\text{SNR}_{\text{worst}} := \frac{P_{\text{Tx}} \gamma_{\text{worst}} |\tilde{h}|^2}{P_n}, \quad (2.40)$$

with γ_{worst} computed using (2.36) for $d_{ij} = d_{\text{size}}$ and $\tilde{h} \sim \mathcal{CN}(0, 1)$.

As a consequence, the following statement holds:

$$\bar{R}_{\text{RR-TDMA},i}^u \geq R_0^u, \quad \forall i \in \{1, \dots, N\}, \quad (2.41)$$

with $\bar{R}_{\text{RR-TDMA},i}^u$ defined in (2.11).

2.3.4.2 Performance analysis

In this section, we evaluate the performance of described methods through two criteria: i) the average sum rate, and ii) the average minimum user rate per link among all the links. The sum rate assesses the global network performance whereas the minimum user rate

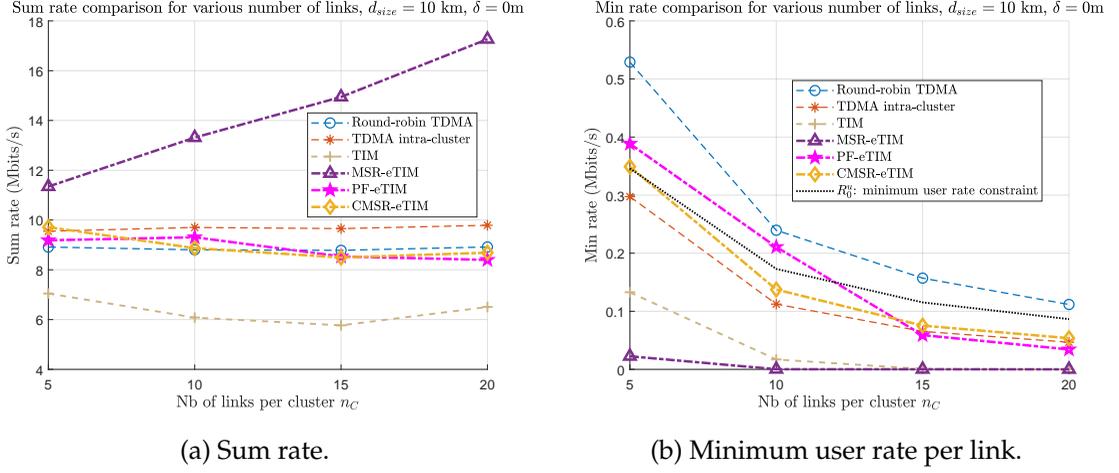


Figure 2.4: Comparison of interference management methods in SCSi context versus the number of links per cluster n_C for $d_{size} = 10$ km and $\delta = 0$ m.

per link is provided to assess the fairness of the solutions. The implemented methods are the following:

- The round-robin TDMA: round-robin TDMA is applied within each cluster and between the clusters. Moreover, the resulting rates can be approximated for long frames using (2.10).
- TDMA intra-cluster: each cluster applies round-robin TDMA independently from the other cluster. Thus, at each time, there is one transmission for each cluster, which means that every receiver is subjected to a single inter-cluster interference.
- TIM: we implement the conventional TIM described in the previous chapter which solves Problem 1.3 with an AO procedure and which uses the interference graph designed in Section 1.3.1.3.
- MSR-eTIM: described in Section 2.3.2.
- PF-eTIM: described in Section 2.3.3.1.
- CMSR-eTIM: described in Section 2.3.3.2.

We study the influence of the number of links per cluster n_C by plotting the average sum rate in Fig. 2.4a and the average minimum user rate per link in Fig. 2.4b. In terms of sum rate, we observe in Fig. 2.4a that for an increasing value of number of links per cluster, all the methods have a steady sum rate, except for MSR-eTIM's sum rate which increases. Moreover, we can observe that the fair solutions PF-eTIM and CMSR-eTIM provide a similar, if not worse sum rate than the TDMA-based solutions. CMSR-eTIM performance can be explained as follows: to respect the minimum user rate constraint,

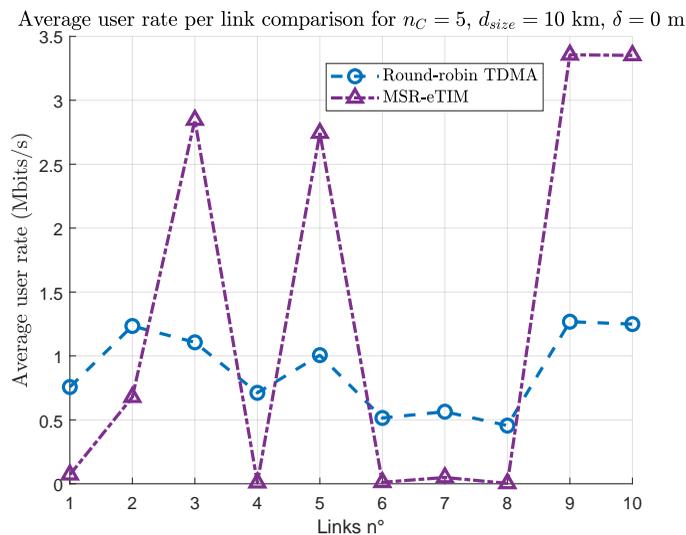


Figure 2.5: Comparison between **MSR-eTIM** and round-robin **TDMA** in terms of average user rate per link, versus the number of links per cluster n_C for $d_{size} = 10$ km and $\delta = 0$ m.

CMSR-eTIM needs to provide longer coders, i.e. r increases, which degrades the sum rate. Besides, among these solutions, **MSR-eTIM** provides higher average sum rate than all the others.

However, Fig. 2.4b shows that **MSR-eTIM** suffers from fairness issues since its average minimum user rate is close to 0. Round-robin **TDMA** shows a minimum user rate higher than the rate constraint R_0^u , which is due to the choice of R_0^u as round-robin **TDMA** worst case in terms of average user rate per link. Besides, among the other solutions, none of them has a minimum user rate higher than R_0^u at each value of n_C . The result for **CMSR-eTIM** is also surprising since the constraint in Problem 2.8 does not seem to be respected. This can be explained as follows: the use of hardening bound approximation has led our estimated rates to be slightly below the real ones, which explains why **CMSR-eTIM**'s average minimum user rate is slightly below R_0^u (around 0.04 Mbits/s difference for $n_C \geq 10$). Hence, implementing a solution based on **CMSR-eTIM** while respecting the constraint requires solving Problem 2.1 which is hardly treatable in terms of complexity.

To conclude, although the proposed state-of-the-art fair solutions show better fairness than **MSR-eTIM**, they do not perform well in terms of average sum rate compared to conventional solutions like **TDMA**. Since using standard approaches is not sufficient to provide a higher sum rate than **TDMA** while guaranteeing a minimum user rate per link, a new perspective is required to implement a method achieving these requirements.

2.3.5 Analysis of MSR-eTIM

Our simulations have confirmed that despite its high average sum rate, MSR-eTIM provides several links with an average user rate close to 0, which is also illustrated in Fig. 2.5 and confirms a significant flaw of MSR-eTIM in practice. Besides, Fig. 2.5 compares the average user rate per link when using TDMA and MSR-eTIM. This representation highlights an interesting property of MSR-eTIM: we can observe that for some links (3, 5, 9, 10), MSR-eTIM provides much higher average user rates than for round-robin TDMA (more than 2.5 Mbits/s). For other links (1, 4, 6, 7, 8), MSR-eTIM shows average user rates close to 0, while round-robin TDMA shows average user rates between 0.45 and 1.3 Mbits/s. Therefore, when using MSR-eTIM, it is possible to distinguish links having a high average user rate and links having a low one, compared to round-robin TDMA's. In the next section, we propose to exploit this property MSR-eTIM to develop our proposed solution called eTIM-hybrid.

2.4 Proposed method: eTIM-hybrid

This section presents one of the main contributions of this thesis. We have previously seen that in the SCSi context, multiple solutions based on eTIM are limited since they reveal either a lower average sum rate than TDMA-based ones or fairness issues. Hence, we propose a new method that we call eTIM-hybrid which enables us to achieve a higher average sum rate than TDMA while guaranteeing a chosen minimum user rate constraint. The hybrid procedure is composed of the following two phases:

1. We partition the links into two groups: i) a set of "good links" in which an eTIM-based method is applied, and ii) a set of "bad links" in which TDMA is used.
2. We optimize the scheduling between the links which consists in allocating the slots in the frame to the links using eTIM, and other slots to the links using TDMA, to maximize the sum rate.

This section is organized as follows. In Section 2.4.1, we describe the iterative algorithm to partition the links into two specific groups of links. In Section 2.4.2, we formulate and solve our scheduling optimization between the links of both partitions to maximize the sum rate of the network while guaranteeing a minimal user rate for each link. In Section 2.4.3, we summarize the implementation of the proposed solution eTIM-hybrid and explain how to apply it in real-traffic simulations. In Section 2.4.4, we provide numerical simulations to assess the performance of eTIM-hybrid in terms of average sum rate and minimum user rate per link.

2.4.1 Links partition

In this section, we explain how to partition the links into two groups. We define \mathcal{T}_e and \mathcal{T}_d as the sets of "good links" and "bad links", respectively (more technical details are provided in what follows). \mathcal{T}_e and \mathcal{T}_d are computed such that we have $\mathcal{T}_e \cup \mathcal{T}_d = \{1, \dots, N\}$. Let $N_{\mathcal{T}_e} := |\mathcal{T}_e|$ and $N_{\mathcal{T}_d} := |\mathcal{T}_d|$ be the number of links in \mathcal{T}_e and \mathcal{T}_d , respectively.

While \mathcal{T}_e and \mathcal{T}_d can be computed using various eTIM and TDMA techniques, in this chapter, we opt to compute them using MSR-eTIM and round-robin TDMA. This selection is motivated by the earlier analysis of MSR-eTIM, which demonstrated significant differences in user rates between the links compared to round-robin TDMA. Therefore, \mathcal{T}_e and \mathcal{T}_d respectively correspond to the set of links having higher and lower user rate at the receiver when applying MSR-eTIM compared to round-robin TDMA. Formally, their expression can be written as:

$$\mathcal{T}_e := \{i \mid \bar{R}_{eTIM,i}^u > \bar{R}_{TDMA,i}^u\}, \quad (2.42a)$$

$$\mathcal{T}_d := \{i \mid \bar{R}_{eTIM,i}^u \leq \bar{R}_{TDMA,i}^u\}. \quad (2.42b)$$

Now, let us present the different steps of the proposed links partition solution. This procedure is illustrated in Fig. 2.6 and is later described in Algorithm 2.4.

1. We initialize $\mathcal{T}_e = \{1, \dots, N\}$ and $\mathcal{T}_d = \emptyset$.
2. We apply MSR-eTIM restricted on \mathcal{T}_e , which corresponds to solving for a given $r = 2, \dots, M$, the following optimization problem:

Problem 2.9.

$$\mathbf{U}_{eTIM}^{\mathcal{T}_e}, \mathbf{V}_{eTIM}^{\mathcal{T}_e} = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{M \times r}} \bar{R}_{\text{sum}, \mathcal{T}_e}^{\text{hard}} \quad (2.43)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2^2 = \|\mathbf{v}_i\|_2^2 = 1, \quad \forall i \in \mathcal{T}_e, \quad (2.44)$$

with $\bar{R}_{\text{sum}, \mathcal{T}_e}^{\text{hard}} := \sum_{i \in \mathcal{T}_e} \bar{R}_{eTIM,i}^{\text{hard}, \mathcal{T}_e}$ and where $\bar{R}_{eTIM,i}^{\text{hard}, \mathcal{T}_e}$ is computed using (2.19). Algorithms 2.3 is used on Problem 2.9 to compute $\mathbf{U}_{eTIM}^{\mathcal{T}_e}, \mathbf{V}_{eTIM}^{\mathcal{T}_e}$ and Algorithm 2.1 is performed to find its corresponding optimal coders' length denoted as $r^{\mathcal{T}_e}$.

3. Since applying MSR-eTIM on \mathcal{T}_e could lead to new "bad links" in \mathcal{T}_e , we propose to move them from \mathcal{T}_e to \mathcal{T}_d . The set of new "bad links" $\mathcal{T}_{e,\text{bad}}$ can be computed as:

$$\mathcal{T}_{e,\text{bad}} := \{i \mid \bar{R}_{eTIM,i}^{u, \mathcal{T}_e} \leq \bar{R}_{TDMA,i}^u\}, \quad (2.45)$$

with $\bar{R}_{eTIM,i}^{u, \mathcal{T}_e}$ computed by injecting $\mathbf{U}_{eTIM}^{\mathcal{T}_e}, \mathbf{V}_{eTIM}^{\mathcal{T}_e}$ into (2.16b) and with Monte-Carlo simulations. Then, we update \mathcal{T}_e and \mathcal{T}_d as the following:

$$\begin{aligned} \mathcal{T}_e &= \mathcal{T}_e \setminus \mathcal{T}_{e,\text{bad}} \\ \mathcal{T}_d &= \mathcal{T}_d \cup \mathcal{T}_{e,\text{bad}}. \end{aligned} \quad (2.46)$$

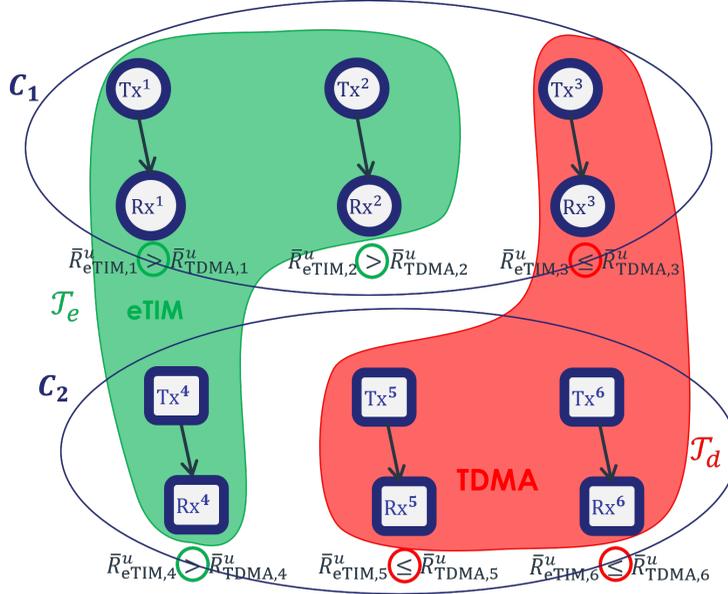


Figure 2.6: Links partition using Algorithm 2.4. Interference is dealt with using eTIM-based method, such as MSR-eTIM in our implementation, on the links in \mathcal{T}_e and a TDMA-based technique on the links in \mathcal{T}_d . The interference links have been omitted for clarity.

4. We continue Steps 2 and 3 until \mathcal{T}_e and \mathcal{T}_d converge.

In Step 4, applying MSR-eTIM on \mathcal{T}_e can lead to "bad links" in \mathcal{T}_e since the MSR algorithm could help increase the rate of certain links at the cost of others to increase the sum rate in \mathcal{T}_e . In our study, we propose two solutions to deal with these "bad links". The first solution is an iterative solution described in the above procedure and consists of removing the "bad links" from \mathcal{T}_e to \mathcal{T}_d as formulated in (2.46). In practice, we iterate Steps 2 and 3 until \mathcal{T}_e and \mathcal{T}_d converge. The second solution involves adding a minimum user rate constraint for each link in Problem 2.9 by solving the following optimization problem:

Problem 2.10.

$$\mathbf{U}_{\text{CeTIM}}^{\mathcal{T}_e} \mathbf{V}_{\text{CeTIM}}^{\mathcal{T}_e} = \arg \max_{\mathbf{U}, \mathbf{V} \in \mathbb{R}^{M \times r}} \bar{R}_{\text{sum}, \mathcal{T}_e}^{\text{eTIM}} \quad (2.47a)$$

$$\text{s.t.} \quad \|\mathbf{u}_i\|_2 = \|\mathbf{v}_i\|_2 = 1, \quad \forall i \in \{1, \dots, N\}, \quad (2.47b)$$

$$\bar{R}_{eTIM,i}^{u, \mathcal{T}_e} \geq \bar{R}_{TDMA,i'}^u \quad \forall i \in \{1, \dots, N\}, \quad (2.47c)$$

where $\bar{R}_{\text{sum}, \mathcal{T}_e}^{\text{eTIM}} := \sum_{i \in \mathcal{T}_e} \bar{R}_{eTIM,i}^{u, \mathcal{T}_e}$ and $\bar{R}_{eTIM,i}^{u, \mathcal{T}_e}$ is computed using (2.16b).

Contrarily to CMSR-eTIM, we do not use the hardening bound approximation in the constraint (2.47c) since we have concluded in Section 2.3.4.2 that it could not respect the user rate constraint. Therefore, this constraint is computed with the average user rate $\bar{R}_{eTIM,i'}^{u, \mathcal{T}_e}$ which requires Monte-Carlo simulations to evaluate it at each iteration of the optimization problem and thus, makes Problem 2.10 become numerically complex.

As a consequence, we use the first solution to deal with the "bad links" in the proposed procedure and we show in the following that it is guaranteed to converge.

Result 2.1. *The iterative procedure to partition the links described in Section 2.4.1 converges. In the end, we have either $\mathcal{T}_{e,bad} = \emptyset$ or $N_{\mathcal{T}_e} = 1$.*

Proof. The proposed iterative algorithm moves bad links from \mathcal{T}_e to \mathcal{T}_d until one of the following statement is reached:

- There is no bad link anymore, i.e. $\mathcal{T}_{e,bad} = \emptyset$, so \mathcal{T}_e and \mathcal{T}_d converge since they stop being updated.
- Every link has moved from \mathcal{T}_e to \mathcal{T}_d until there is only a single element left in \mathcal{T}_e , i.e. $N_{\mathcal{T}_e} = 1$. Hence, the transmission in \mathcal{T}_e is not subjected to any interference. This partition is considered the worst case and is the same as applying TDMA.

□

The links partition algorithm is detailed in Algorithm 2.4. Once the links have been

Algorithm 2.4: Links partition algorithm for eTIM-hybrid

Set channel statistics γ_{ij}, N_{MC} .

Compute $\mathbf{U}(r), \mathbf{V}(r)$ by solving Problem 2.2 using Algorithm 2.3 with r computed using Algorithm 2.1.

For all $i \in \{1, \dots, N\}$, compute $\bar{R}_{eTIM,i}^u$ using $\mathbf{U}(r), \mathbf{V}(r)$ in (2.16) and $\bar{R}_{TDMA,i}^u$ using (2.8) with N_{MC} Monte-Carlo simulations.

Compute \mathcal{T}_e and \mathcal{T}_d using (2.42).

if $M = 1$ **then**

 | Set stay_loop = false.

else

 | Set stay_loop = true.

while stay_loop **do**

 | Compute $\mathbf{U}^{\mathcal{T}_e}(r_{\mathcal{T}_e}), \mathbf{V}^{\mathcal{T}_e}(r_{\mathcal{T}_e})$ by solving Problem 2.9 with Algorithm 2.3 and compute $r^{\mathcal{T}_e}$ using Algorithm 2.1.

 | For all $i \in \mathcal{T}_e$, compute $\bar{R}_{eTIM,i}^{u,\mathcal{T}_e}$ using $\mathbf{U}^{\mathcal{T}_e}(r^{\mathcal{T}_e}), \mathbf{V}^{\mathcal{T}_e}(r^{\mathcal{T}_e})$ in (2.16) with N_{MC} Monte-Carlo simulations.

 | **if** $\forall i \in \mathcal{T}_e, \bar{R}_{eTIM,i}^{u,\mathcal{T}_e} > \bar{R}_{TDMA,i}^u$ **or** $M = 1$ **then**

 | Set stay_loop = false.

 | **else**

 | Update \mathcal{T}_e and \mathcal{T}_d using (2.46).

end

Return $\mathcal{T}_e, \mathcal{T}_d, \mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}$

partitioned, the next phase to implement eTIM-hybrid is to optimize the slots in the frame to allocate to the links using MSR-eTIM and to the links using TDMA.

2.4.2 Scheduling optimization

In this section, we expose the second phase of eTIM-hybrid procedure, which consists of finding the best portions of time to allocate to each link. We consider a frame of length L as defined in Section 2.2.2.1, and the goal of this section is to optimize the number of slots to allocate to each link to maximize the sum rate while guaranteeing a minimum user rate for each link of the network. This task will be referred to as "scheduling optimization". Furthermore, since the average sum rate of the network is also influenced by the frame length L , the optimization of this parameter will be considered as a second step in the scheduling optimization.

This section is organized as follows. In Section 2.4.2.1, we formulate the scheduling optimization problem for a given frame of length L and provide closed form expressions to assess its feasibility and to solve it optimally. Then, in Section 2.4.2.2, we formulate the joint scheduling and frame length optimization problem and provide its optimal solution.

2.4.2.1 Optimization for given frame length

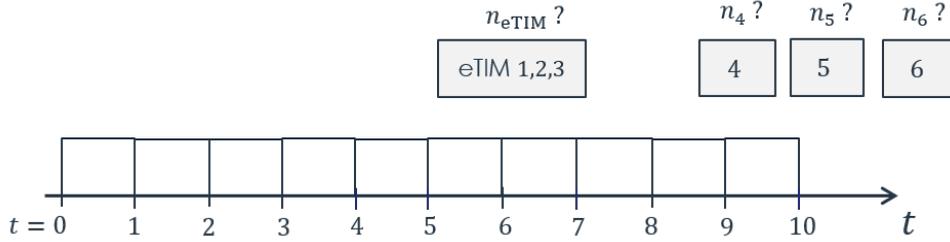


Figure 2.7: Example of scheduling optimization problem on a frame of length $L = 10$, $N = 6$ links, $\mathcal{T}_e = \{1, 2, 3\}$, $\mathcal{T}_d = \{4, 5, 6\}$ and $r = 2$ slots.

Let us define n_k as the number of transmissions for each link $k \in \mathcal{T}_d$ and n_{eTIM} as the number of transmissions of MSR-eTIM during the frame of length L . In this part, we formulate the scheduling optimization problem, which consists in optimizing $n_k, k \in \mathcal{T}_d$ and n_{eTIM} to maximize the average sum rate. In Fig. 2.7, the scheduling optimization problem is illustrated for a frame of length $L = 10$ and where eTIM is applied for $r = 2$ slots. Since L is the total number of slots to allocate for each link, n_k and n_{eTIM} are constrained by the following equality:

$$\sum_{k \in \mathcal{T}_d} n_k + n_{eTIM} r = L. \quad (2.48)$$

It is worth mentioning that the constraint in (2.48) is equality since the strict inequality $\sum_{k \in \mathcal{T}_d} n_k + n_{eTIM} r < L$ corresponds to the presence of empty slots in the frame, i.e. slots where there is not any transmission, which reduces the average sum rate. Besides, when using eTIM-hybrid framework, the average sum rate on a frame of length L is defined as:

$$\bar{R}_{\text{sum}}^{\text{eTIM-hy}} := \sum_{j \in \mathcal{T}_e} \bar{R}_{\text{eTIM},j}^{u,\mathcal{T}_e} + \sum_{k \in \mathcal{T}_d} \bar{R}_{\text{TDMA},k'}^u \quad (2.49)$$

where the average user rate is defined as follows:

$$\bar{R}_{\text{eTIM},j}^{u,\mathcal{T}_e} := \frac{n_{\text{eTIM}}}{L} \bar{R}_{\text{eTIM},j}^{\varphi} \quad (2.50)$$

$$\bar{R}_{\text{TDMA},k}^u := \frac{n_k}{L} \bar{R}_k^{\varphi}. \quad (2.51)$$

\bar{R}_k^{φ} can be computed using (2.8a) with Monte-Carlo simulations, whereas $\bar{R}_{\text{eTIM},j}^{\varphi}$ is computed by injecting $\mathbf{U}^{\mathcal{T}_e}(r^{\mathcal{T}_e}), \mathbf{V}^{\mathcal{T}_e}(r^{\mathcal{T}_e})$ into (2.16a) with Monte-Carlo simulations, where $\mathbf{U}^{\mathcal{T}_e}(r^{\mathcal{T}_e}), \mathbf{V}^{\mathcal{T}_e}(r^{\mathcal{T}_e})$ are computed using Algorithm 2.4.

Since our objective is to maximize the average sum rate while respecting the minimum average user rate constraints, the optimization problem we need to solve can be formulated as:

Problem 2.11.

$$(n_{k_1}^*, \dots, n_{k_Q}^*, n_{\text{eTIM}}^*) = \arg \max_{(n_{k_1}, \dots, n_{k_Q}, n_{\text{eTIM}}) \in \mathbb{N}^*} \bar{R}_{\text{sum}}^{\text{eTIM-hy}} \quad (2.52a)$$

$$\text{s.t.} \quad \bar{R}_{\text{TDMA},k}^u \geq R_0^u, \quad \forall k \in \mathcal{T}_d, \quad (2.52b)$$

$$\bar{R}_{\text{eTIM},\min}^u \geq R_0^u, \quad (2.52c)$$

$$\sum_{k \in \mathcal{T}_d} n_k + n_{\text{eTIM}} = L. \quad (2.52d)$$

with n_{k_m} the number of transmissions of link $k_m \in \mathcal{T}_d$ and eTIM's average rate defined as follows:

$$\bar{R}_{\text{eTIM},\min}^u := \frac{n_{\text{eTIM}}}{L} \bar{R}_{\text{eTIM},\min}^{\varphi} \quad (2.53)$$

$$\bar{R}_{\text{eTIM},\min}^{\varphi} := \min \left\{ \bar{R}_{\text{eTIM},j}^{\varphi} \right\}_{j \in \mathcal{T}_e}. \quad (2.54)$$

Constraints (2.52b) represent the minimum user rate constraints for all the links in \mathcal{T}_d , i.e. when using TDMA, whereas constraint (2.52c) corresponds to the minimum user rate constraint for the links in \mathcal{T}_e , i.e. for the links using MSR-eTIM. Since the number of transmissions n_{eTIM} for the links in \mathcal{T}_e is the same for every link, we can only focus on the link with the lowest user rate $\bar{R}_{\text{eTIM},\min}^{\varphi}$, which guarantees that all user rate constraints in \mathcal{T}_e are satisfied. Constraint (2.52d) is the same as (2.48).

2.4.2.1.1 Optimization problem feasibility definition

Before solving Problem 2.11, we provide a few definitions of optimization problems by considering the following problem:

Problem 2.12.

$$\min_{\mathbf{x}} f(\mathbf{x}) \quad (2.55a)$$

$$\text{s.t.} \quad g_k(\mathbf{x}) \leq 0, \quad k = 1, \dots, m, \quad (2.55b)$$

$$h_k(\mathbf{x}) = 0, \quad k = 1, \dots, m, \quad (2.55c)$$

where $f : \mathbb{R}^p \rightarrow \mathbb{R}$ is the objective function of Problem 2.12 and for all $k = 1, \dots, m$, $g_k : \mathbb{R}^p \rightarrow \mathbb{R}$ and $h_k : \mathbb{R}^p \rightarrow \mathbb{R}$ are respectively the inequality and equality constraints functions.

A feasible point of Problem 2.12 is defined as a point satisfying Problem 2.12's constraints. The feasible set \mathcal{F} of Problem 2.12 is defined as the set of all feasible points. Formally, it can be defined as:

$$\mathcal{F} := \{\mathbf{x} \in \mathbb{R}^p \text{ such that: } \forall k \in \{1, \dots, m\}, \quad g_k(\mathbf{x}) \leq 0\}. \quad (2.56)$$

Problem 2.12 is said to be feasible if there exists at least one feasible point, i.e. \mathcal{F} is not empty.

2.4.2.1.2 Closed form solution of the scheduling optimization problem

In this section, we solve Problem 2.11 for a given frame length L by stating two results. In Result 2.2, whose proof is provided in Appendix A.3.1, we provide a condition to assess the feasibility of Problem 2.11. In Result 2.3, whose proof is provided in Appendix A.3.2, we provide the optimal solution of Problem 2.11 in closed form.

Result 2.2. *Problem 2.11 is feasible if and only if the following condition is fulfilled:*

$$\sum_{k \in \mathcal{T}_d} n_k^{\min} + n_{e\text{TIM}}^{\min} r \leq L, \quad (2.57)$$

where n_k^{\min} and $n_{e\text{TIM}}^{\min}$ are defined as:

$$\left\{ n_k^{\min} := \left\lceil \frac{LR_0^u}{\bar{R}_k^\varphi} \right\rceil, \quad \forall k \in \mathcal{T}_d, \quad (2.58a) \right.$$

$$\left. n_{e\text{TIM}}^{\min} := \left\lceil \frac{LR_0^u}{\bar{R}_{e\text{TIM},\min}^\varphi} \right\rceil, \quad (2.58b) \right.$$

where $\lceil \cdot \rceil$ represents the ceiling function which maps to the nearest integer up.

n_k^{\min} and $n_{e\text{TIM}}$ are the minimum number of transmissions to allocate to the links in \mathcal{T}_d and \mathcal{T}_e , respectively, to satisfy the constraints (2.52c) and (2.52d) from Problem 2.11. The proof of Result 2.2 is obtained by rewriting the constraints equations (2.52c) and (2.52d). Moreover, this result provides a closed form expression to immediately assess the feasibility of Problem 2.11.

Now, before stating the next result, we define k_{\max} as the link index corresponding to the highest average physical rate in \mathcal{T}_d , i.e. $k_{\max} := \arg \max_{k \in \mathcal{T}_d} \{\bar{R}_k^\varphi\}$. Moreover, we define $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi$ as the sum of the average physical rates corresponding to the links in \mathcal{T}_e , which can be written as:

$$\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi := \sum_{j \in \mathcal{T}_e} \bar{R}_{e\text{TIM},j}^\varphi. \quad (2.59)$$

Moreover, let us define L_{res} as the number of the remaining slots to allocate in the frame of length L after allocating the minimal number of time slots to each link that ensures feasibility. Formally, L_{res} can be expressed as:

$$L_{res} := L - \sum_{k \in \mathcal{T}_d} n_k^{min} - n_{eTIM}^{min} r, \quad (2.60)$$

which is positive if Problem 2.11 is feasible according to Result 2.2.

The following result, whose proof is provided in Appendix A.3.2, gives the closed form optimal solution of Problem 2.11.

Result 2.3. *The optimal solution of Problem 2.11 depends on the difference between $\bar{R}_{sum, \mathcal{T}_e}^\varphi$ and $r\bar{R}_{k_{max}}^\varphi$, which can be written as follows:*

If $\bar{R}_{sum, \mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{max}}^\varphi$, the optimal solution of Problem 2.11 can be written as:

$$\begin{cases} n_k^* = n_k^{min}, & \forall k \in \mathcal{T}_d \setminus \{k_{max}\}, \\ n_{eTIM}^* = n_{eTIM}^{min} \\ n_{k_{max}}^* = n_{k_{max}}^{min} + L_{res} \end{cases} \quad (2.61)$$

If $\bar{R}_{sum, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{max}}^\varphi$, the optimal transmissions allocation per link is

$$\begin{cases} n_k^* = n_k^{min}, & \forall k \in \mathcal{T}_d \setminus \{k_{max}\}, \\ n_{eTIM}^* = n_{eTIM}^{min} + \left\lfloor \frac{L_{res}}{r} \right\rfloor \\ n_{k_{max}}^* = n_{k_{max}}^{min} + L_{res} - \left\lfloor \frac{L_{res}}{r} \right\rfloor r, \end{cases} \quad (2.62)$$

with $n_{eTIM}^{min}, n_{k_{max}}^{min}$ defined in (2.58) and $\lfloor \cdot \rfloor$ represents the floor function which maps to the nearest integer down.

The optimal allocation provided by Result 2.3 consists in first allocating the required slots $n_k^{min}, n_{eTIM}^{min}$ to satisfy Problem 2.11's constraints. Then, it involves performing a greedy allocation, i.e. we allocate the L_{res} remaining slots to the links providing the highest average rates. The two possible allocations are to allocate the L_{res} slots to either the link in \mathcal{T}_d with the highest rate, which is k_{max} , or to the links in \mathcal{T}_e . To choose the option providing the highest average sum rate, we compare over r slots the average sum rate in \mathcal{T}_e , which is $\bar{R}_{sum, \mathcal{T}_e}^\varphi$, and the average sum rate in \mathcal{T}_d with all the slots allocated to k_{max} , which is $r\bar{R}_{k_{max}}^\varphi$. Then, the allocation is proceeded as the following:

- If $\bar{R}_{sum, \mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{max}}^\varphi$, we allocate the L_{res} slots to k_{max} .
- If $\bar{R}_{sum, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{max}}^\varphi$, we allocate the L_{res} slots to the links in \mathcal{T}_e . If L_{res} is not a multiple of r , there will remain $L_{res} - \left\lfloor \frac{L_{res}}{r} \right\rfloor r$ slots to allocate to k_{max} .

2.4.2.1.3 Influence of the frame length

In the previous section, we have derived the condition and optimal solutions of the scheduling optimization Problem 2.11 for a fixed frame length L . In this section, we illustrate the impact of the parameter L with a numerical example, where the feasibility and the optimal solution of Problem 2.11 vary depending on L . Indeed, a bad choice of L can lead to lower performance in terms of average sum rate, and may even render Problem 2.11 infeasible.

Example 1. We consider a network with the following parameters: $N = 6$ links, $\mathcal{T}_e = \{1, 2, 3\}$, $\mathcal{T}_d = \{4, 5, 6\}$, $r = 2$, and the average physical rates are provided in Table 2.2. On \mathcal{T}_e , we perform the *MSR-eTIM* method with an optimal coders' length as $r = 2$ and the minimum rate constraint for every link is $R_0^u = 1$ Mbits/s. Table 2.3 shows the feasibility and the performance of the *eTIM-hybrid* average sum rate for various values of L . We can see that Problem 2.11 is feasible for $L = 8, 10, 11$ since $L_{\text{res}} \geq 0$ whereas for $L = 9$, the problem is not feasible since $L_{\text{res}} < 0$. Moreover, after computing the average sum rate $\bar{R}_{\text{sum}}^{\text{eTIM-hy}}$ with the optimal allocation from Result 2.3, we can see that its value varies depending on L and that among the feasible solutions, the maximum average sum rate is obtained for $L = 10$. Therefore, finding the optimal value of L which both maximizes the average sum rate and verifies Problem 2.11 feasibility is of particular interest.

Table 2.2: Example of average physical rates for $N = 6$, $\mathcal{T}_e = \{1, 2, 3\}$, $\mathcal{T}_d = \{4, 5, 6\}$ and $r = 2$.

$\bar{R}_{\text{eTIM},1}^\varphi$	$\bar{R}_{\text{eTIM},2}^\varphi$	$\bar{R}_{\text{eTIM},3}^\varphi$	\bar{R}_4^φ	\bar{R}_5^φ	\bar{R}_6^φ
10	9	8	8	7	6

Table 2.3: Feasibility and average sum rate study of Problem 2.11 for different values of L . The associated average physical rates are provided in Table 2.2. If $L_{\text{res}} < 0$, the problem is not feasible like for $L = 9$.

L	n_{eTIM}^{\min}	n_4^{\min}	n_5^{\min}	n_6^{\min}	L_{res}	$\bar{R}_{\text{sum}}^{\text{eTIM-hy}}$
8	1	1	2	2	1	8.625
9	2	2	2	2	-1	10.67
10	2	2	2	2	0	9.6
11	2	2	2	2	1	9.455

Since frame length L has an impact on the optimization problem, we propose in the next section to jointly optimize frame length L with the scheduling parameters n_{eTIM} and n_k for $k \in \mathcal{T}_d$.

2.4.2.2 Joint scheduling and frame length optimization

In the previous section, we have shown that the frame length influences both the average sum rate and the feasibility of the scheduling problem and can thus be optimized. Moreover, it is worth mentioning that reducing frame length is also of interest since it allows low-latency communications. Multiple works have searched for ways to determine the shortest length frame satisfying given traffic demands [10, 60] in the TDMA framework. Thus, if multiple values of L are optimal, we prefer choosing the lowest possible value.

The joint scheduling and frame length optimization problem can be formulated as:

Problem 2.13.

$$(n_{k_1}^*, \dots, n_{k_Q}^*, n_{eTIM}^*, L^*) = \arg \max_{(n_{k_1}, \dots, n_{k_Q}, n_{eTIM}, L) \in \mathbb{N}^*} \bar{R}_{sum}^{eTIM-hy} \quad (2.63a)$$

$$\text{s.t.} \quad \bar{R}_{TDMA,k}^u \geq R_0^u, \quad \forall k \in \mathcal{T}_d, \quad (2.63b)$$

$$\bar{R}_{eTIM,min}^u \geq R_0^u, \quad (2.63c)$$

$$\sum_{k \in \mathcal{T}_d} n_k + n_{eTIM} r = L. \quad (2.63d)$$

Since Result 2.3 provides the optimal values n_k^*, n_{eTIM}^* for all $L \in \mathbb{N}^*$, Problem 2.13 can be formulated as an optimization problem which only depends on the variable L :

Problem 2.14.

$$L^* = \arg \max_{L \in \mathbb{N}^*} \bar{R}_{sum,opt}^{eTIM-hy}(L) \quad (2.64a)$$

$$\text{s.t.} \quad \bar{R}_{TDMA,k}^u \geq R_0^u, \quad \forall k \in \mathcal{T}_d, \quad (2.64b)$$

$$\bar{R}_{eTIM,min}^u \geq R_0^u, \quad (2.64c)$$

$$\sum_{k \in \mathcal{T}_d} n_k^* + n_{eTIM}^* r = L. \quad (2.64d)$$

with n_k^*, n_{eTIM}^* respectively defined in (2.61), (2.62) and where $\bar{R}_{sum,opt}^{eTIM-hy}$ expression is obtained by injecting (2.61) and (2.62) into (2.49). After several derivations provided in Appendix A.3.3, $\bar{R}_{sum,opt}^{eTIM-hy}$ can be expressed as:

$$R_{sum,opt}^{eTIM-hy}(L) = \begin{cases} \frac{1}{L} \left(n_{eTIM}^{\min} (\bar{R}_{sum,\mathcal{T}_e}^\varphi - r \bar{R}_{k_{max}}^\varphi) + \sum_{k \in \mathcal{T}_d} n_k^{\min} (\bar{R}_k^\varphi - \bar{R}_{k_{max}}^\varphi) \right) + \bar{R}_{k_{max}}^\varphi & \text{if } \bar{R}_{sum,\mathcal{T}_e}^\varphi \leq r \bar{R}_{k_{max}}^\varphi \\ \frac{1}{L} \left(\left(n_{eTIM}^{\min} + \left\lfloor \frac{L_{res}}{r} \right\rfloor \right) (\bar{R}_{sum,\mathcal{T}_e}^\varphi - r \bar{R}_{k_{max}}^\varphi) + \sum_{k \in \mathcal{T}_d} n_k^{\min} (\bar{R}_k^\varphi - \bar{R}_{k_{max}}^\varphi) \right) + \bar{R}_{k_{max}}^\varphi & \text{if } \bar{R}_{sum,\mathcal{T}_e}^\varphi > r \bar{R}_{k_{max}}^\varphi. \end{cases} \quad (2.65)$$

2.4.2.2.1 Scheduling optimization problem: results on its feasibility

Let us define the positive values $x_k := R_0^u / \bar{R}_k^\varphi, \forall k \in \mathcal{T}_d$ and $x_{eTIM} := R_0^u / \bar{R}_{eTIM, \min}^\varphi$. Using these notations, the feasibility equation (2.57) from Result 2.2 can be expressed as:

$$\frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{eTIM} \rceil \right) \leq 1. \quad (2.66)$$

In the following, we provide two results: one result to assess Problem 2.14's feasibility independently from L , and another one to solve it analytically. We first deliver two useful lemmas for proving the first result, namely Result 2.4, on Problem 2.13's feasibility before presenting it. In Lemma 2.1, we find a condition where Problem 2.13 is feasible and we express the closed form expression of its feasible set. Lemma 2.2 provides conditions to find feasible solutions to Problem 2.13 and shows that its feasible set is infinite. Their proof is respectively given in Appendix A.3.4 and A.3.5, respectively.

Lemma 2.1. *If $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} = 1$ and if x_{eTIM} and x_k are rational positive values, then there exist multiple feasible solutions $L \in \mathbb{N}_+^*$ to Problem 2.14.*

Lemma 2.2. *If $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} < 1$, then there exists $L_0 \in \mathbb{N}_+^*$ such that for all $L \geq L_0$, L is a feasible solution to Problem (2.14).*

Lemma 2.1 shows that if we have $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} = 1$, then the feasible set of Problem (2.14) can be formulated in closed form using (A.49), assuming x_{eTIM} and x_k are rational positive values. When assuming that we have $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} = 1$, Lemma 2.2 proves that any L high enough is in the feasible set of Problem (2.14). Moreover, with the same assumptions as Lemma 2.1, a corollary characterizing the feasible set of Problem (2.14) is provided in Appendix A.3.4.

Using Lemmas 2.1 and 2.2, we can state the following result which provides the closed form expression assessing the feasibility of Problem 2.14. Result 2.4's proof is provided in Appendix A.3.6 and can be stated as:

Result 2.4. *Problem 2.14 is feasible if and only if one of the following statements is true:*

1. $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} < 1$
2. $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} = 1$ and $(x_k)_{k \in \mathcal{T}_d}, x_{eTIM}$ are positive rational numbers.

The implication of Result 2.4 is proved using the ceiling function bounding properties, whereas the reciprocal is proved using Lemmas 2.1 and 2.2. Result 2.4 is useful in practice since it provides a closed form expression to assess the feasibility of Problem 2.14 before running any optimization method. Moreover, the rationality requirements are easily met since the rates are usually rational or can be approximated with a close rational number.

2.4.2.2.2 Scheduling optimization problem: results on its optimal solution

Now, we first deliver a useful lemma for proving Result 2.5, which gives the analytical expression of an upper bound of the average sum rate, which does not depend on L . Then, we provide the optimal solution of Problem 2.14 in Result 2.5. Lemma 2.3 and Result 2.5's proof are given in Appendices A.3.7 and Appendix A.3.8, respectively.

Lemma 2.3. $\bar{R}_{sum,opt}^{eTIM-hy}(L)$'s upper bound can be expressed as:

$$\forall L \in \mathbb{N}^*, \quad \bar{R}_{sum,opt}^{eTIM-hy}(L) \leq R_{sup}, \quad (2.67)$$

with R_{sup} defined as:

$$R_{sup} := \begin{cases} x_{eTIM} \left(\bar{R}_{sum,\mathcal{T}_e}^\varphi - r \bar{R}_{k_{max}}^\varphi \right) + N_{\mathcal{T}_d} R_0^u + \bar{R}_{k_{max}}^\varphi \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right), \\ \quad \text{if } \bar{R}_{sum,\mathcal{T}_e}^\varphi \leq r \bar{R}_{k_{max}}^\varphi, \\ \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right) \frac{\left(\bar{R}_{sum,\mathcal{T}_e}^\varphi - r \bar{R}_{k_{max}}^\varphi \right) + N_{\mathcal{T}_d} R_0^u + \bar{R}_{k_{max}}^\varphi \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right)}{r}, \\ \quad \text{if } \bar{R}_{sum,\mathcal{T}_e}^\varphi > r \bar{R}_{k_{max}}^\varphi. \end{cases} \quad (2.68)$$

In Lemma 2.3, we prove by using the flooring and ceiling function inequalities that the average sum rate $\bar{R}_{sum,opt}^{eTIM-hy}(L)$ is upper bounded by a value which does not depend on L . Furthermore, a corollary stating that this upper bound is also its limit when $L \rightarrow +\infty$ is given in Appendix A.3.7. Then, using this inequality, we express in the following result the set of the optimal solutions of Problem 2.14 which corresponds to the set of $L \in \mathbb{N}^*$ where $\bar{R}_{sum,opt}^{eTIM-hy}(L)$ is equal to its upper bound R_{sup} .

Result 2.5. We suppose that for all $k \in \mathcal{T}_d$, the physical rates \bar{R}_k^φ , $\bar{R}_{eTIM,min}^\varphi$ and R_0^u are positive rational numbers. The following statement holds:

$$\exists L \in \mathbb{N}^* : \quad \bar{R}_{sum,opt}^{eTIM-hy}(L) = R_{sup}, \quad (2.69)$$

with R_{sup} defined in (2.68).

Moreover, using rational numbers definition, we can write the following:

$$\begin{aligned} \forall k \in \mathcal{T}_d, \quad \exists p_k, q_k \in \mathbb{N}^* : x_k &= \frac{p_k}{q_k} \\ \exists p_{eTIM}, q_{eTIM} \in \mathbb{N}^* : x_{eTIM} &= \frac{p_{eTIM}}{q_{eTIM}}, \end{aligned} \quad (2.70)$$

with $\gcd(p_{eTIM}, q_{eTIM}) = 1$ and $\forall k \in \mathcal{T}_d, \gcd(p_k, q_k) = 1$, where $\gcd(a, b)$ stands for greatest common divisor between a and b . q_k and q_{eTIM} are respectively referred to as the smallest denominators of x_k and x_{eTIM} .

Then, we provide the following statements:

- If $\bar{R}_{sum,\mathcal{T}_e}^\varphi \leq r \bar{R}_{k_{max}}^\varphi$, the solutions of the equation in (2.69) are the set of $L \in \mathbb{N}^*$ such that the following conditions are satisfied:

1. L is a multiplier of q_k for all $k \in \mathcal{T}_d \setminus \mathcal{K}_{max}$.
2. If $\bar{R}_{sum, \mathcal{T}_e}^\varphi \neq r\bar{R}_{k_{max}}^\varphi$, L is a multiplier of q_{eTIM} . Otherwise, only the first condition is required.

with $\mathcal{K}_{max} := \{k | \bar{R}_k^\varphi = \bar{R}_{k_{max}}^\varphi\}$. Formally, the set of these solutions can be written as:

$$\mathcal{L}_{opt} := \left\{ L \in \mathbb{N}^* \mid \text{such that } \begin{cases} (\exists m_0 \in \mathbb{N}^* : L = m_0 q_{eTIM}) \vee (\bar{R}_{sum, \mathcal{T}_e}^\varphi = r\bar{R}_{k_{max}}^\varphi) \\ \exists m_k \in \mathbb{N}^* : L = m_k q_k, \quad \forall k \in \mathcal{T}_d \setminus \mathcal{K}_{max}. \end{cases} \right\}. \quad (2.71)$$

- If $\bar{R}_{sum, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{max}}^\varphi$, we first define (p_0, q_0) as follows:

$$\exists p_0, q_0 \in \mathbb{N}^* : \frac{1 - \sum_{k \in \mathcal{T}_d} x_k}{r} = \frac{p_0}{q_0}, \quad (2.72)$$

with $\gcd(p_0, q_0) = 1$. Then, we state that the solutions of the equation in (2.69) are the set of $L \in \mathbb{N}^*$ such that the following conditions are satisfied:

1. $\forall k \in \mathcal{T}_d$, L is a multiplier of q_k .
2. L is a multiplier of q_0 .

Formally, the set of these solutions can be written as:

$$\mathcal{L}_{opt} := \left\{ L \in \mathbb{N}^* \mid \text{such that } \begin{cases} \exists m_0 \in \mathbb{N}^* : L = m_0 q_0 \\ \exists m_k \in \mathbb{N}^* : L = m_k q_k, \quad \forall k \in \mathcal{T}_d. \end{cases} \right\}. \quad (2.73)$$

Corollary 2.4. The optimal solution of Problem 2.14 is the set \mathcal{L}_{opt} defined in (2.71) and (2.73).

In Result 2.5, we solve the equation defined in (2.69) and deduce that the set of its solutions provided in (2.71) and (2.73) is infinite. Therefore, using Corollary 2.4, we can deduce that Problem 2.14 admits an infinite number of optimal solutions. Among them, we choose to keep the smallest value L in our implementations since it reduces the latency of the communications and we refer to this value as L_{opt} .

In Figs. 2.8a et 2.8b, we represent the average sum rate $\bar{R}_{sum, opt}^{eTIM-hy}$ versus frame length L for the case $\bar{R}_{sum, \mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{max}}^\varphi$ with the physical rates provided in Table 2.4 whereas the case $\bar{R}_{sum, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{max}}^\varphi$ with the physical rates given in Table 2.5 is provided in Figs. 2.9a and 2.9b. Figs. 2.8a, 2.8b 2.9a and 2.9b also highlight the multiplicity of optimal solutions L of Problem 2.14 and the chosen value L_{opt} in the end, is in green. Finally, we can observe that when $L \rightarrow +\infty$, $\bar{R}_{sum, opt}^{eTIM-hy}$ tends to its upper bound value as expected according to Corollary A.2.

Finally, a summary of the results and lemmas, regarding the feasibility and the optimal solution of the scheduling optimization problem in eTIM-hybrid, are given in Table 2.6.

Table 2.4: Example of average physical rates for $N = 6$, $\mathcal{T}_e = \{1, 2, 3\}$, $\mathcal{T}_d = \{4, 5, 6\}$, $r = 2$. We have $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi \leq r \bar{R}_{k_{\max}}^\varphi$ and $\bar{R}_{\text{sum}, \text{opt}}^{\text{eTIM-hy}}(L)$ is represented in Fig. 2.8.

$\bar{R}_{\text{eTIM},1}^\varphi$	$\bar{R}_{\text{eTIM},2}^\varphi$	$\bar{R}_{\text{eTIM},3}^\varphi$	\bar{R}_4^φ	\bar{R}_5^φ	\bar{R}_6^φ	$r\bar{R}_4^\varphi$	$\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi$
7	6	5	9.5	7	6	19	18

Table 2.5: Example of average physical rates for $N = 6$, $\mathcal{T}_e = \{1, 2, 3\}$, $\mathcal{T}_d = \{4, 5, 6\}$ and $r = 2$. We have $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi > r \bar{R}_{k_{\max}}^\varphi$ and $\bar{R}_{\text{sum}, \text{opt}}^{\text{eTIM-hy}}(L)$ is represented in Fig. 2.9.

$\bar{R}_{\text{eTIM},1}^\varphi$	$\bar{R}_{\text{eTIM},2}^\varphi$	$\bar{R}_{\text{eTIM},3}^\varphi$	\bar{R}_4^φ	\bar{R}_5^φ	\bar{R}_6^φ	$r\bar{R}_4^\varphi$	$\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi$
7	6	5	8	7	5	16	18

2.4.2.3 Frame length choice in practice

Unfortunately, although the proposed solution provides the optimal value of frame length, we have observed that in practice, L_{opt} 's value can be very high and is thus not desired since it increases the latency. For example, in Figs 2.9a and 2.9b, we have $L_{\text{opt}} = 560$, which is acceptable since it corresponds to 0.56 seconds with a sampling frequency of 1000 Hz. However, we have found cases by simulations where L_{opt} was higher than 10^{10} . To overcome this issue, we propose to set a maximum value L_{\max} for L and we numerically solve the following optimization problem:

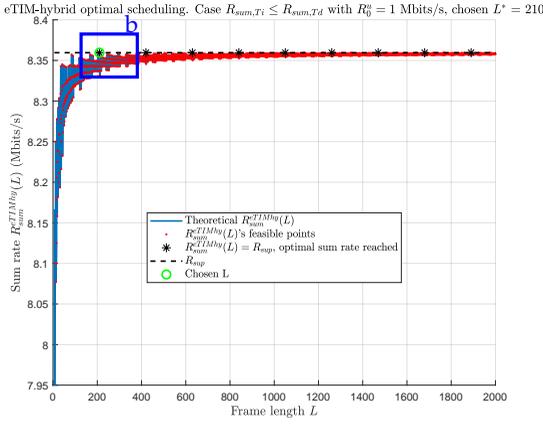
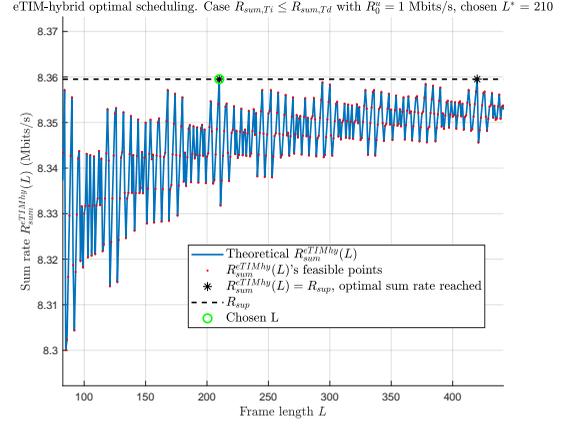
Problem 2.15.

$$\begin{aligned}
 L^* = \arg \max_{L \in \{1, \dots, L_{\max}\}} & \bar{R}_{\text{sum}, \text{opt}}^{\text{eTIM-hy}}(L) \\
 \text{s.t.} & \begin{cases} \forall k \in \mathcal{T}_d, \bar{R}_{\text{TDMA}, k}^u \geq R_0^u \\ \bar{R}_{\text{eTIM}, \min}^u \geq R_0^u \\ \sum_{k \in \mathcal{T}_d} n_k + n_{\text{eTIM}} r = L, \end{cases} \quad (2.74)
 \end{aligned}$$

Before describing the procedure to solve Problem 2.14, we set a few notations: $\text{lcm}(a, b, c, \dots)$ stands for least common multiple between a, b, c, \dots , $\text{lcm}(\mathcal{A})$ is referred to as the least common multiple between all the elements in the set \mathcal{A} , and finally, we define the following

Table 2.6: Summary of the results and the lemmas, regarding the feasibility and the optimal solution of the scheduling optimization problem in eTIM-hybrid

Results and lemmas	Feasibility	Optimal solution
Given frame length L	Result 2.2	Result 2.3
Optimization of L	Lemmas 2.1, 2.2, Result 2.4	Lemma 2.3, Result 2.5

(a) Convergence of $\bar{R}_{\text{sum,opt}}^{\text{eTIM-hy}}$ to R_{sup} .

(b) Zoom of Fig. 2.8a.

Figure 2.8: Example of eTIM-hybrid average sum rate defined in (2.49) for case $\bar{R}_{\text{sum},\mathcal{T}_e}^{\varphi} \leq r\bar{R}_{k_{\max}}^{\varphi}$, versus frame length L . We have $R_0^u = 1$ Mbits/s and the other parameters are in Table 2.5.

sets:

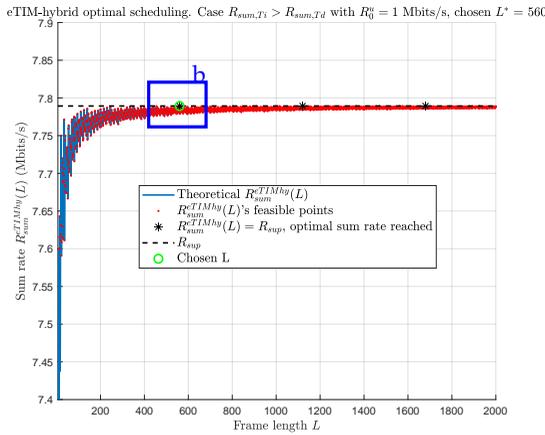
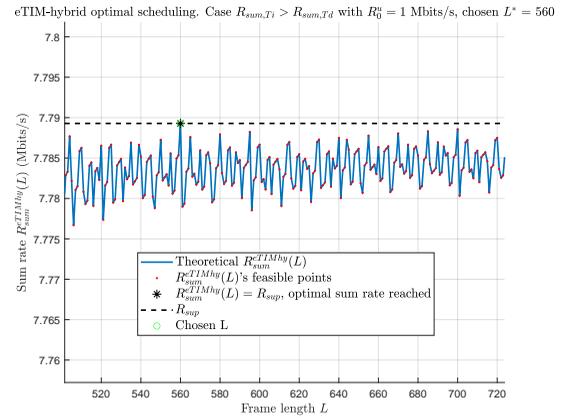
$$\mathcal{Q}_1 := \{q_k \mid k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}\} \cup \{q_{\text{eTIM}}\}, \quad (2.75a)$$

$$\mathcal{Q}_2 := \{q_k \mid k \in \mathcal{T}_d\} \cup \{q_0\}. \quad (2.75b)$$

We propose a procedure provided in Algorithm 2.5 to optimally solve Problem 2.15. We first assess Problem 2.15's feasibility using Result 2.4. Then, after computing and comparing $\bar{R}_{\text{sum},\mathcal{T}_e}^{\varphi}$ and $r\bar{R}_{k_{\max}}^{\varphi}$, we find the multiplicity conditions on L using Result 2.5 to find its optimal values (since there are several of them). Then, using the rationality properties of the input parameters, we find the required smallest denominators for the multiplicity requirements as described in Result 2.5. Since there are several optimal solutions respecting Result 2.5's requirements, we choose the smallest value L among them by performing a lcm operation. Finally, we ensure that the optimal frame length value L_{opt} is not higher than the upper bounding value L_{max} that we set. If it is, Problem 2.14 is numerically solved. It is worth mentioning that the smallest admissible solution of L is higher than L_{max} . In that case, Problem 2.15 is not feasible and we can either reduce the rate constraint R_0^u or increase L_{max} to solve it.

2.4.2.4 Special case: optimized TDMA

We describe the joint frame length scheduling optimization applied to the special case where the links partition iterative algorithm from Section 2.4.1 has led to the following links partition: \mathcal{T}_e is a singleton, i.e. $N_{\mathcal{T}_e} = 1$, and \mathcal{T}_d has the rest of the elements, i.e. $N_{\mathcal{T}_d} = N - 1$. The problem is thus reduced to a frame length scheduling optimization

(a) Convergence of $\bar{R}_{\text{sum,opt}}^{\text{eTIM-hy}}$ to R_{sup} .

(b) Zoom of Fig. 2.9a

Figure 2.9: Example of eTIM-hybrid average sum rate defined in (2.49) for case $\bar{R}_{\text{sum},\mathcal{T}_e}^{\varphi} > r\bar{R}_{k_{\max}}^{\varphi}$, versus frame length L . We have $R_0^u = 1$ Mbits/s and the other parameters are in Table 2.5.

in TDMA framework. Keeping the same notations as in Section 2.4.2, the scheduling optimization problem to solve for optimized TDMA can be formulated as:

Problem 2.16.

$$\begin{aligned} (n_1^*, \dots, n_N^*, L^*) &= \arg \max_{(n_1, \dots, n_N, L) \in \mathbb{N}^*} \bar{R}_{\text{sum}}^{\text{TDMA-opt}} \\ \text{s.t. } &\begin{cases} \forall k \in \{1, \dots, N\}, \bar{R}_{\text{TDMA},k}^u \geq R_0^u \\ \sum_{k=1}^N n_k = L. \end{cases} \end{aligned} \quad (2.76)$$

Since this problem is a special case of Problem 2.13, this problem can be solved using Algorithm 2.5. In the rest of this chapter, this method will be referred to as "optimized TDMA".

Remark 2.1. Like the methods based on eTIM, optimized TDMA also requires SCS to calculate its average rates to perform the scheduling optimization, contrarily to round-robin TDMA.

2.4.3 eTIM-hybrid implementation

In this section, we provide a summary of eTIM-hybrid implementation and describe it in Algorithm 2.6. Then, we explain how to implement it in real-traffic simulations.

2.4.3.1 eTIM-hybrid algorithm

We first apply the links partition algorithm to compute the sets $\mathcal{T}_e, \mathcal{T}_d$, the users' indices who are in \mathcal{T}_e and the coders $\mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}$ on \mathcal{T}_e . Then, after computing the physical rates in

Algorithm 2.5: Scheduling optimization for eTIM-hybrid

Set $\forall j \in \mathcal{T}_e, \bar{R}_{e\text{TIM},j}^\varphi$ and $\forall k \in \mathcal{T}_d, \bar{R}_k^\varphi, L_{\max}$.
 Assert Problem 2.14 feasibility using Result 2.4.
 Compute $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi$ using (2.59) and $k_{\max} = \arg \max_{k \in \mathcal{T}_d} \{\bar{R}_k^\varphi\}$.
if $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$ **then**
 Set $\mathcal{K}_{\max} = \{k | \bar{R}_k^\varphi = \bar{R}_{k_{\max}}^\varphi\}$.
 Find (p_k, q_k) for all $k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}$ and $(p_{e\text{TIM}}, q_{e\text{TIM}})$ defined in Lemma 2.1.
 $L_{\text{opt}} = \text{lcm}(\mathcal{Q}_1)$ with \mathcal{Q}_1 defined in (2.75a).
else
 Find (p_k, q_k) for all $k \in \mathcal{T}_d$ defined in Lemma 2.1 and (p_0, q_0) defined in (2.72)
 from Result 2.5.
 $L_{\text{opt}} = \text{lcm}(\mathcal{Q}_2)$ with \mathcal{Q}_2 defined in (2.75b).
if $L_{\text{opt}} > L_{\max}$ **then**
 Numerically solve Problem 2.15 and compute the new L_{opt} .
 Compute $n_{e\text{TIM}}$ and n_k for all $k \in \mathcal{T}_d$ using Result 2.3.
 Return $L_{\text{opt}}, n_{e\text{TIM}}, (n_k)_{k \in \mathcal{T}_d}$

\mathcal{T}_e and \mathcal{T}_d , we use the scheduling optimization Algorithm 2.5 to get the slots allocation $n_{e\text{TIM}}, (n_k)_{k \in \mathcal{T}_d}$ and the frame length L_{opt} . It is worth reminding that in the case where \mathcal{T}_e is a singleton, optimized TDMA will be used as the scheduling optimization phase.

Algorithm 2.6: eTIM-hybrid implementation

Set N_{MC} .
 Compute $\mathcal{T}_e, \mathcal{T}_d, \mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}, \bar{R}_k^\varphi$ for $k \in \{1, \dots, N\}$ defined in (2.8) and $\bar{R}_{e\text{TIM},j}^\varphi$ for $j \in \mathcal{T}_e$
 defined in (2.16) with N_{MC} Monte-Carlo simulations, using Algorithm 2.4.
 Compute $L_{\text{opt}}, n_{e\text{TIM}}, (n_k)_{k \in \mathcal{T}_d}$ using Algorithm 2.5.
 Return $\mathcal{T}_e, \mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}, L_{\text{opt}}, n_{e\text{TIM}}, (n_k)_{k \in \mathcal{T}_d}$

Remark 2.2. *The implementation of the proposed eTIM-hybrid aims at increasing the sum rate of the network. Although we have not implemented it in our work, it is worth mentioning that the proposed scheme can easily be adapted if the main goal is to maximize the weighted sum rate which is useful for prioritizing different user rates. Indeed, first, the adapted MSR-eTIM optimization problem with the weighted sum rate would be similarly handled since the gradient expressions for gradient descent are similar by the linearity of the derivative. Then, after performing the links partition phase, the scheduling optimization would also be dealt with similarly by performing the following change of variables: $\bar{R}_{e\text{TIM},w_j}^\varphi = w_j \bar{R}_{e\text{TIM},j}^\varphi$ for $j \in \mathcal{T}_e$ and $\bar{R}_{w_k}^\varphi = w_k \bar{R}_k^\varphi$ for $k \in \mathcal{T}_d$, where w_1, \dots, w_N are the weight coefficients of the weighted sum rate optimization.*

2.4.3.2 Real-traffic simulations

eTIM-hybrid algorithm computes the following parameters: $\mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e} \in \mathbb{R}^{M \times r^{\mathcal{T}_e}}$ for the links in \mathcal{T}_e , the number of slots to allocate to each transmission $(n_k)_{k \in \mathcal{T}_d}, n_{\text{eTIM}}$ and the optimized frame length L_{opt} . The transmissions take place as follows. On the one hand, each link in \mathcal{T}_d transmits its symbol using TDMA, that is in an interference-free communication. On a frame of length L_{opt} , the number of slots allocated to each link k in \mathcal{T}_d is $(n_k)_{k \in \mathcal{T}_d}$. On the other hand, the links in \mathcal{T}_e communicate using eTIM, i.e. before sending their respective symbol, the transmitters in \mathcal{T}_e perform a temporal spread of $r^{\mathcal{T}_e}$ slots using their respective precoders \mathbf{v}_i . Then, all the links in \mathcal{T}_e simultaneously transmit their spread symbol to their corresponding receiver. On a frame of length L_{opt} , the number of slots allocated to the links in \mathcal{T}_e is $n_{\text{eTIM}} r^{\mathcal{T}_e}$ since each repetition of eTIM lasts for $r^{\mathcal{T}_e}$ slots. Furthermore, we remind that when implementing an eTIM-based solution in a real-traffic simulation, the channel is assumed to be constant over r slots.

It is worth noticing that the formulated scheduling optimization only provides the number of slots per transmission, without any slots arrangement requirement. As a consequence, the centralized entity performing the whole interference management process plans the scheduling arrangement taking the slot allocation into account, and then, sends it to the links of both clusters. Furthermore, since we consider two associated clusters in an ad hoc network, we can consider that eTIM-hybrid and the scheduling arrangement are computed by the cluster heads and are then sent to the links of their respective cluster.

Remark 2.3. *While the scheduling optimization resolution uses closed form expressions, the links partition algorithm performs a gradient descent method, which makes it the most complex step of eTIM-hybrid implementation.*

2.4.4 Numerical results

In this section, we provide numerical results to assess the performance of the newly proposed solution eTIM-hybrid. To this end, we use the same setup as in Section 2.3.4 and we study the influence of three parameters by varying them: the number of links per cluster n_C , the distance between the clusters δ , and the clusters' size d_{size} . A topology of the ad hoc clustered network used in the simulations after applying the links grouping algorithm is illustrated in Fig. 2.10. An example of the extreme cases $\delta = 0, -5000, 5000$ m for is represented in Fig. 2.11.

In this part, we compare the sum rate and the minimum rate per link of our proposed approach eTIM-hybrid with similar methods as in Section 2.3.4, that is round-robin TDMA, TDMA-intra, MSR-eTIM. We also implement optimized TDMA (TDMA-opt) described in Section 2.4.2.4 to have a fairer comparison with our solution eTIM-hybrid.

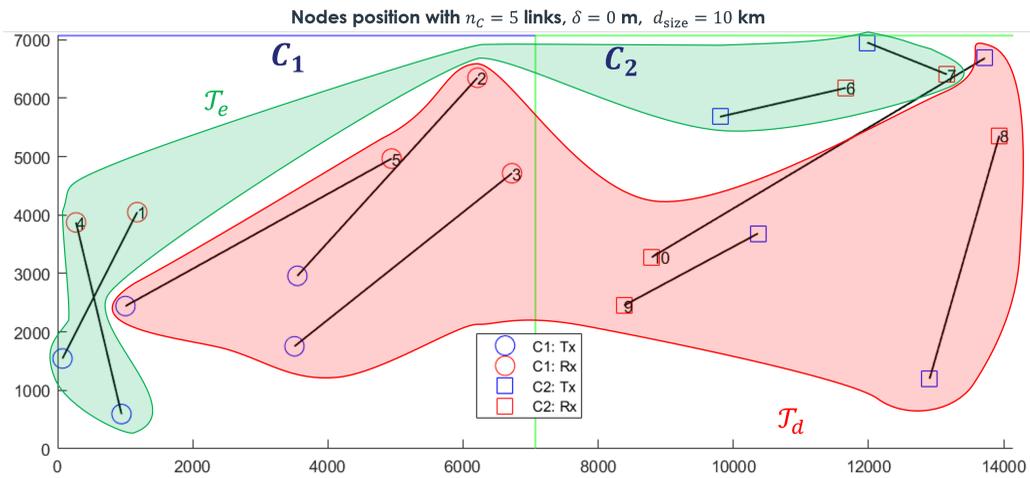


Figure 2.10: Links partition with Algorithm 2.4 applied to the links in Fig. 2.3 with 2 clusters and $n_C = 5$ links per cluster. The links in \mathcal{T}_e are in green whereas the links in \mathcal{T}_d are in red. The interference links have been omitted for clarity.

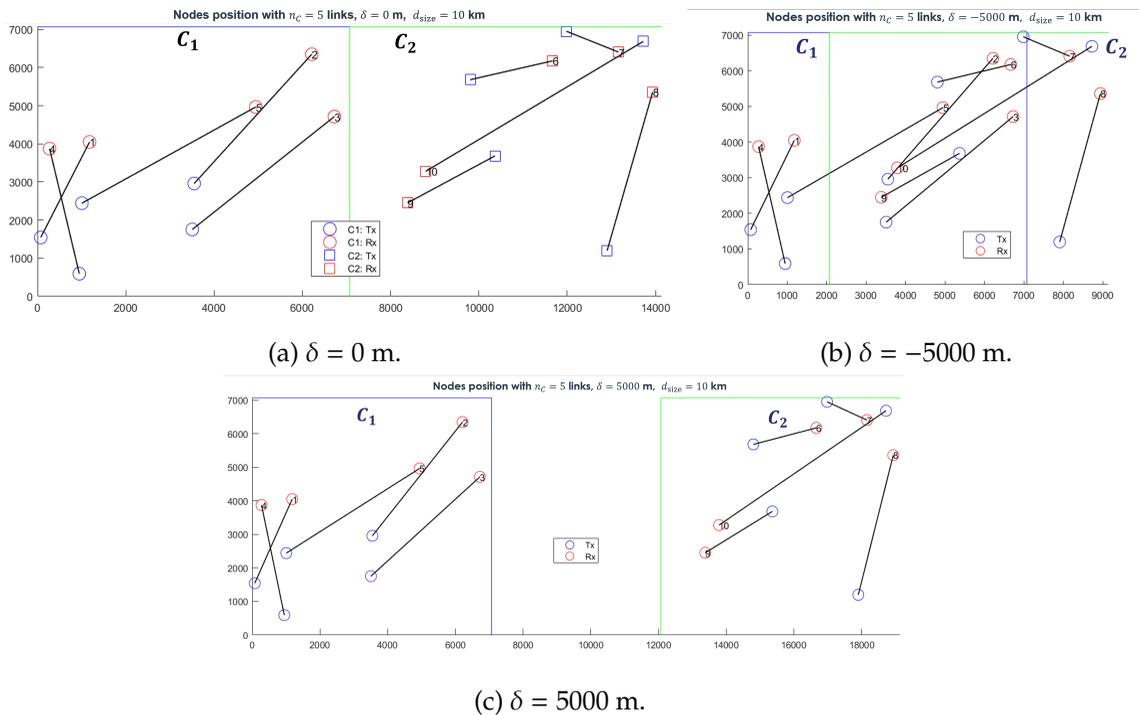


Figure 2.11: Wireless ad hoc clustered network with $K = 2$ clusters composed of $n_C = 5$ links per cluster, $d_{\text{size}} = 10$ km and $\delta = 0, -5000, 5000$ m. The interference links have been omitted for clarity.

2.4.4.1 Performance analysis

In this part, we evaluate the performance of the eTIM-hybrid method through the average sum rate and the average minimum user rate per link, like in the simulations from Section 2.3.4.

In Fig. 2.12, we study the influence of the number of links per cluster n_C by plotting the average sum rate in Fig. 2.12a and the average minimum rate per link in Fig. 2.12b. In terms of sum rate, we observe in Fig. 2.12a that for increasing values of the number of links per cluster, eTIM-hybrid, MSR-eTIM, and optimized TDMA's sum rate increase, whereas the other methods' sum rates are steady. Moreover, eTIM-hybrid and MSR-eTIM show similar sum rates which are both higher average sum rates than the other for $n_C > 10$ links per cluster. In addition, in terms of minimum rate per link, Fig. 2.12a shows that eTIM-hybrid and optimized TDMA's average minimum user rate per link is the same as the rate constraint R_0^u , which is expected due to the greedy behavior of our scheduling optimization (detailed in Section 2.4.2) which will maximize the sum rate after strictly respecting the average minimum user rate constraint. As shown previously, MSR-eTIM's minimum rate is close to 0.

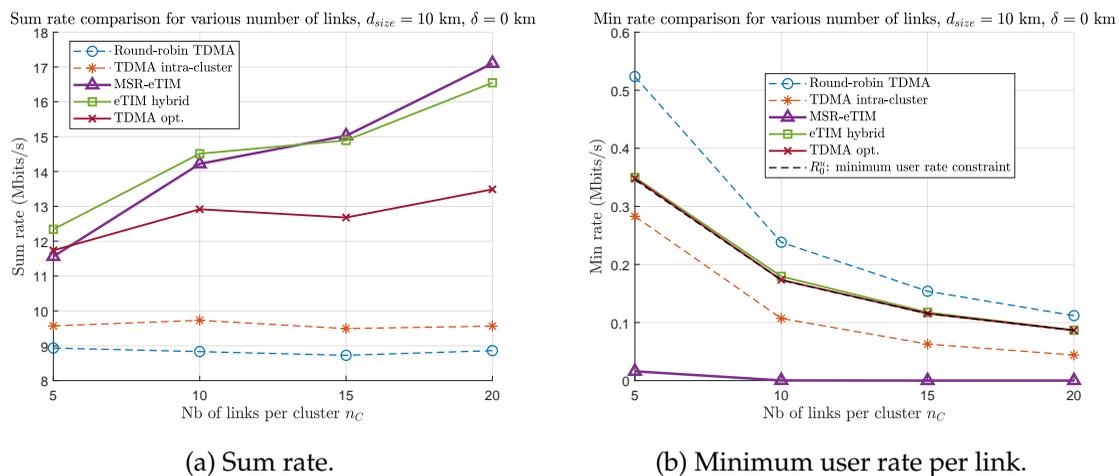


Figure 2.12: Performance comparison of eTIM-hybrid with other implemented methods versus the number of links per cluster for $d_{size} = 10$ km and $\delta = 0$ m.

In Fig. 2.13, we study the influence of the distance δ between the clusters by plotting the average sum rate in Fig. 2.13a and the average minimum user rate per link in Fig. 2.13b. Negative values of δ correspond to the situation where both clusters are intertwined. In terms of average sum rate, we observe that eTIM-hybrid, MSR-eTIM and TDMA-intra have increasing average sum rate for increasing value of δ since inter-cluster interference decreases. Moreover, we can see that round-robin TDMA and optimized TDMA provide steady performance since these methods avoid inter-cluster interference, which makes them independent from δ in terms of sum rate and minimum rate. Furthermore, we

observe that for $\delta \geq 0$ m, **eTIM**-hybrid and **MSR-eTIM** give higher average sum rates. This result shows our solution **eTIM**-hybrid is not restrained to adjacent clusters and can be applied to more distant clusters. However, when δ is negative, inter-cluster interference is increased and degrades **eTIM** methods performance since they will require longer codes, i.e. r increases, thus the average sum rate decreases. When interference becomes too strong, dealing with interference with **eTIM** gives lower performance than using **TDMA**, which explains why optimized **TDMA** has a higher average sum rate than **eTIM**-hybrid for negative δ . In addition, in terms of the average minimum user rate per link, **MSR-eTIM** shows an average minimum user rate close to 0, which is the same conclusion as we said previously. Likewise, **eTIM**-hybrid and optimized **TDMA**'s average minimum user rate per link is the same as the rate constraint R_0^u . Besides, for high values of δ , **TDMA**-intra shows better performance than optimized **TDMA** in terms of average sum rate while respecting the minimum rate constraint. Indeed, when δ becomes high, both clusters are far enough to neglect inter-cluster interference.

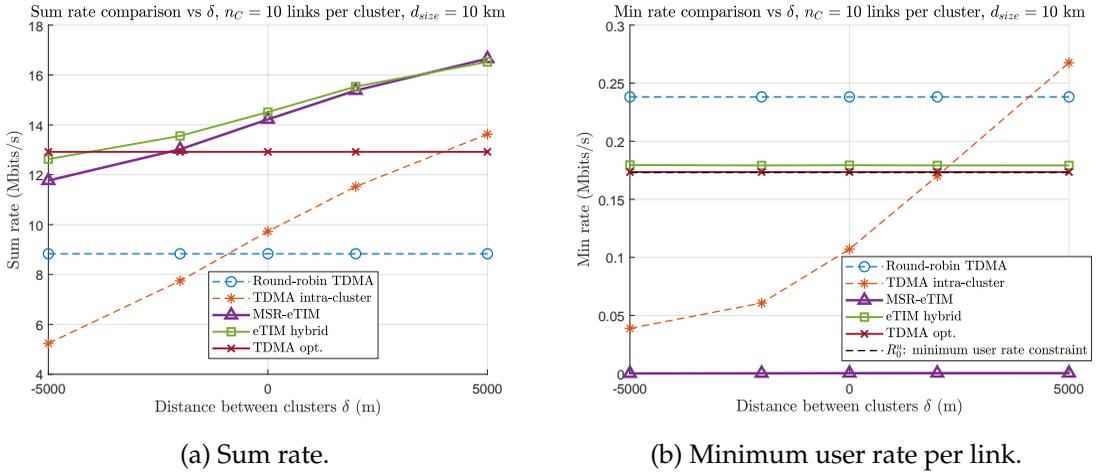


Figure 2.13: Performance comparison of **eTIM**-hybrid with other implemented methods versus the distance between the clusters δ for $n_C = 10$ links per cluster and $d_{\text{size}} = 10$ km.

In Fig. 2.14, we study the influence of the size of the clusters d_{size} by plotting the average sum rate in Fig. 2.14a and the average minimum user rate per link in Fig. 2.14b. In terms of average sum rate, we observe that for increasing values of d_{size} , all the implemented methods have decreasing average sum rate and average minimum user rate. Indeed, the influence of cluster size can be seen as the invert of the influence of transmit power (supposing the same power for every link). Smaller clusters correspond to increasing transmit power, and vice versa. In addition, we observe that our proposed method **eTIM**-hybrid shows the highest sum rate when d_{size} is higher than 10 km. When we have $d_{\text{size}} \leq 10$ km, optimized **TDMA**'s average sum rate becomes higher. Indeed, when clusters are too small, inter-cluster interference becomes strong and **eTIM** requires

longer coders to deal with them. In addition, in terms of the minimum rate per link, **MSR-eTIM** has a minimum rate close to 0, which is the same conclusion as we said previously. Likewise, **eTIM**-hybrid and optimized **TDMA**'s average minimum user rate per link is the same as the rate constraint R_0^u .

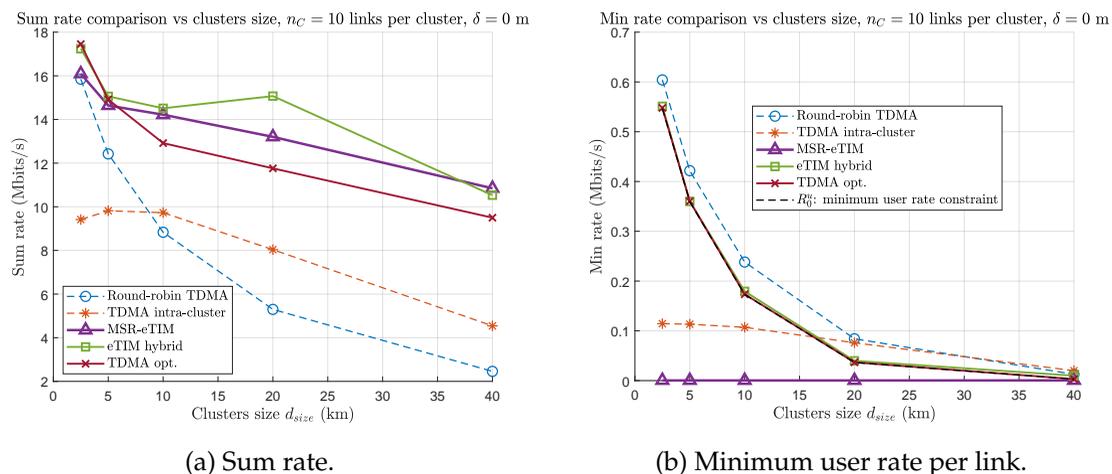


Figure 2.14: Performance comparison of **eTIM**-hybrid with other implemented methods versus clusters size d_{size} for $n_C = 10$ links per cluster and $\delta = 0$ m.

Remark 2.4. In this section, we have proposed our solution **eTIM**-hybrid when considering two adjacent clusters. Although we have not performed simulations to display it, this scheme can be extended to more than two clusters. This would result in a partition of the links of all the clusters into two groups \mathcal{T}_e and \mathcal{T}_d . Then, the scheduling optimization algorithm would be applied to these sets of links.

2.5 Case two available frequency bands

In the work done so far, we have always assumed that two adjacent clusters have only access to a single frequency band which might be due to spectrum scarcity. In this section, we consider the case where two adjacent clusters have access to two frequency bands, instead of sharing the same frequency band. In this context, we develop a novel joint frequency and **eTIM**-hybrid scheme that consists of associating two clusters and allocating the frequency bands to both clusters in a non-orthogonal way. To be more precise, the channels are now allocated to the set of links \mathcal{T}_e and \mathcal{T}_d , which have been described in the previous section, and both sets can contain links that can belong to either of the two clusters. This method is called joint frequency allocation **eTIM**-hybrid (**JFA-eTIM-hy**). In the following, we first describe its implementation. Then, we show through numerical simulations that the proposed scheme **JFA-eTIM-hy** outperforms the

standard orthogonal frequency allocation that assigns a different frequency band to each cluster while respecting the minimum user rate constraint.

2.5.1 Proposed interference management method with two frequency bands

When two frequency bands are available, one conventional approach consists of orthogonally allocating the frequency bands to both clusters and applying interference management methods in each cluster independently. For example, the links in cluster C_1 are allocated the frequency band f_1 and use an eTIM-hybrid to deal with interference. Similarly, the links in C_2 communicate on f_2 and use eTIM-hybrid, independently from C_1 .

However, there is a better way to allocate the frequency bands by considering the problem globally. One could allocate the first frequency band f_1 to some links which can be from both clusters, and the second frequency band f_2 to the other links. In this section, we present the new solution JFA-eTIM-hy which combines a new frequency band allocation and eTIM-hybrid solution. The main idea is to allocate the first frequency band f_1 to some links that use eTIM, and the second one f_2 to the other links that use TDMA. To this end, we consider the links of both clusters and we partition them by applying the same links partition in eTIM-hybrid using Algorithm 2.4. The three phases to implement JFA-eTIM-hy are described as follows:

1. We compute the links partition algorithm from Section 2.4 described in Algorithm 2.4. This results in getting the group of "good links" \mathcal{T}_e and the group of "bad links" \mathcal{T}_d .
2. We allocate the frequency band f_1 to the links in \mathcal{T}_e and the frequency band f_2 to the links in \mathcal{T}_d .
3. On f_1 , we perform MSR-eTIM on the links in \mathcal{T}_e . On f_2 , we apply optimized TDMA on the links in \mathcal{T}_d .

The proposed solution is inspired by eTIM-hybrid since it combines eTIM and TDMA in specific partition links: MSR-eTIM is used on \mathcal{T}_e to improve the global average sum rate and optimized TDMA is applied on \mathcal{T}_d to both improve the global average sum rate and guarantee the minimum user rate per link in \mathcal{T}_d . Moreover, the user rates in \mathcal{T}_e when using MSR-eTIM are greater or equal to the user rates when using round-robin TDMA since we use the links partition algorithm from Algorithm 2.4. A comparison of the methods employed in one or two available frequency bands is illustrated in Figs. 2.15 and 2.16 in terms of slot allocation. Finally, an implementation of JFA-eTIM-hy is depicted in Algorithm 2.7.

Remark 2.5. *In practice, we can imagine a scenario where for both of our clusters, the number of available frequency bands alternate between one and two. For example, the adjacent clusters*

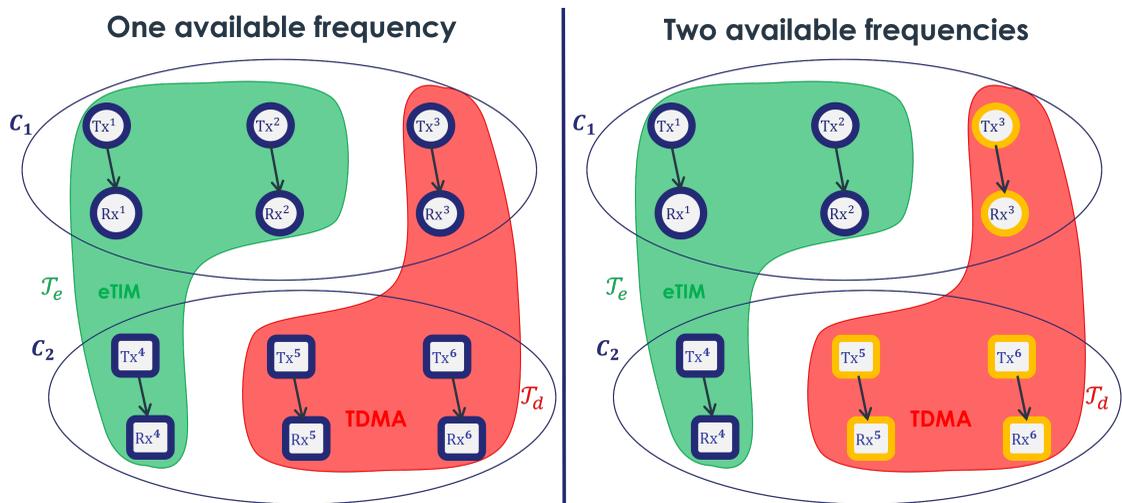


Figure 2.15: Use of links partition method from Algorithm 2.4 in the cases of one and two available frequency bands. We allocate band f_1 to the links in blue and frequency band f_2 to the links in yellow. The interference links have been omitted for clarity.

could first have access to a single frequency band. Then, after a certain amount of time, a second frequency band would be available, before becoming unavailable again after a certain amount of time etc.

It is worth noticing that one main advantage of our schemes *eTIM-hybrid* and *JFA-eTIM-hy* is that they require the same links partition \mathcal{T}_e and \mathcal{T}_d . Therefore, when alternating between both methods according to the availability of the frequency bands, we will only be required to compute \mathcal{T}_e and \mathcal{T}_d once, which corresponds to the most complex step.

2.5.2 Numerical results

2.5.2.1 Simulations parameters and implemented methods

In this part, we provide numerical results to assess the performance of the proposed solution *JFA-eTIM-hy* in terms of average sum rate and minimum user rate per link. To this end, we use the same setup as in Sections 2.3.4 and 2.4.4 and we study the influence of three parameters by varying them: the number of links per cluster n_C , the distance between the clusters δ , and the clusters size d_{size} .

For clarity, we refer to allocating a frequency band f_1 to the links in cluster C_1 and the other frequency band f_2 to the links in cluster C_2 as orthogonal frequency band allocation. Moreover, we describe the implemented interference management methods in the following:

- **OFA-TDMA**: orthogonal channel allocation with round-robin **TDMA** in cluster C_1 on frequency band f_1 and round-robin **TDMA** in cluster C_2 on frequency band f_2 .
- **JFA-TDMA-opt**: joint channel allocation and optimized **TDMA** scheme where the

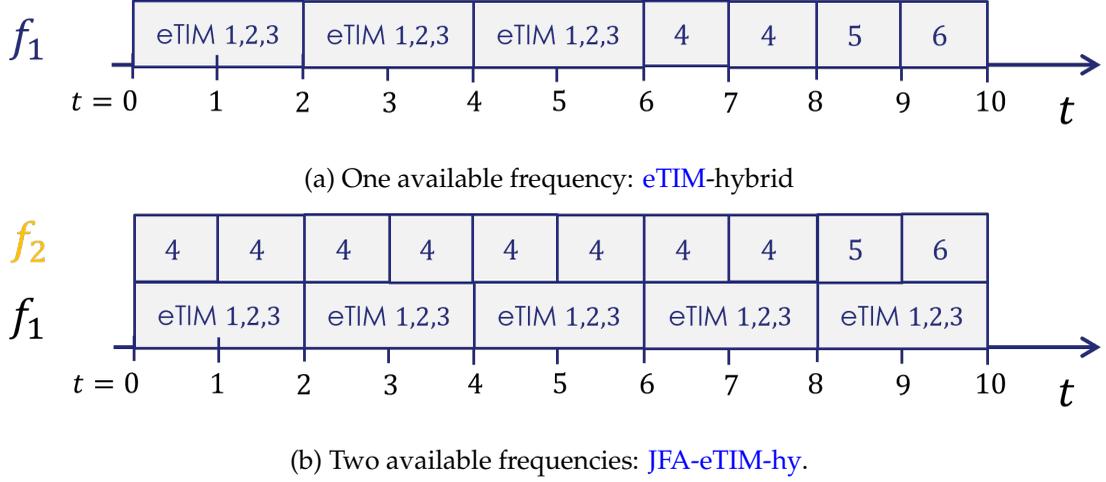


Figure 2.16: Comparison of JFA-eTIM-hy with other implemented methods versus the number of links per cluster for $d_{\text{size}} = 10$ km and $\delta = 0$ m. JFA-eTIM-hy requires clusters association and shows higher performance in terms of average sum rate.

performed channel allocation is the same as in JFA-eTIM-hy, and where optimized TDMA is applied to both the links in \mathcal{T}_e on f_1 and the links in \mathcal{T}_d on f_2 , independently.

- OFA-MSR-eTIM: orthogonal channel allocation with MSR-eTIM applied in C_1 on f_1 and MSR-eTIM applied in C_2 on f_2 .
- OFA-eTIM-hybrid: orthogonal channel allocation with eTIM-hybrid used in C_1 on f_1 and eTIM-hybrid used in C_2 on f_2 .
- JFA-eTIM-hy: the proposed solution.

2.5.2.2 Simulation results

In Fig. 2.17, we study the influence of the number of links per cluster by plotting the average sum rate in Fig. 2.17a and the average minimum user rate per link in Fig. 2.17b when two frequency bands are available. In terms of average sum rate, Fig. 2.17a shows that for increasing values of the number of links per cluster, OFA-TDMA and OFA-MSR-eTIM have a steady average sum rate while the other methods have an increasing average sum rate. We can also observe that our proposed method JFA-eTIM-hy has a higher average sum rate than the other implemented solutions, including OFA-eTIM-hybrid. This result highlights the benefit of performing clusters association when two frequency bands are available. Moreover, this gap increases with higher values of the number of links per cluster: around 0.4 Mbits/s for $n_C = 5$, 1.5 Mbits/s for $n_C = 10$, around 3 – 4 Mbits/s for $n_C = 15, 20$ links per cluster. Moreover, Fig. 2.17b shows that OFA-MSR-eTIM has an average minimum user rate close to 0 for more than 10 links per

Algorithm 2.7: JFA-eTIM-hy implementation

Set channel statistics γ_{ij} , transmit powers P_i , noise power P_n and N_{MC} .
 Compute $\mathcal{T}_e, \mathcal{T}_d, \mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}$ using Algorithm 2.4, with $M = |\mathcal{T}_e|$.
 For $k \in \{1, \dots, N\}$, compute \bar{R}_k^φ defined in (2.8) using N_{MC} Monte-Carlo simulations.
 For $j \in \mathcal{T}_e$, compute $\bar{R}_{eTIM,j}^\varphi$ defined in (2.16) using $\mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}$ and N_{MC} Monte-Carlo simulations.
 Compute $L_{opt,\mathcal{T}_d'}(n_k)_{k \in \mathcal{T}_d}$ using optimized TDMA on \mathcal{T}_d .
if $\sum_{j \in \mathcal{T}_e} \bar{R}_{eTIM,j}^\varphi \leq \sum_{j \in \mathcal{T}_e} \bar{R}_j^\varphi$ **then**
 | Return $\mathcal{T}_e, \mathbf{U}^{\mathcal{T}_e}, \mathbf{V}^{\mathcal{T}_e}, L_{opt,\mathcal{T}_d'}(n_k)_{k \in \mathcal{T}_d}$.
else
 | Compute $L_{opt,\mathcal{T}_e}(n_j)_{j \in \mathcal{T}_e}$ using optimized TDMA on \mathcal{T}_e .
 | Return $\mathcal{T}_e, L_{opt,\mathcal{T}_e}, L_{opt,\mathcal{T}_d'}, n_1, \dots, n_N$.

cluster, which confirms the fairness issue of this method. On the other hand, all the other implemented methods respect the minimum user rate constraint. Furthermore, we can observe that JFA-eTIM-hy and JFA-TDMA-opt present the same average minimum user rate per link as the user rate constraint, which is expected since the use of optimized TDMA includes a scheduling optimization algorithm that will set multiple user rates at the minimum user rate constraint.

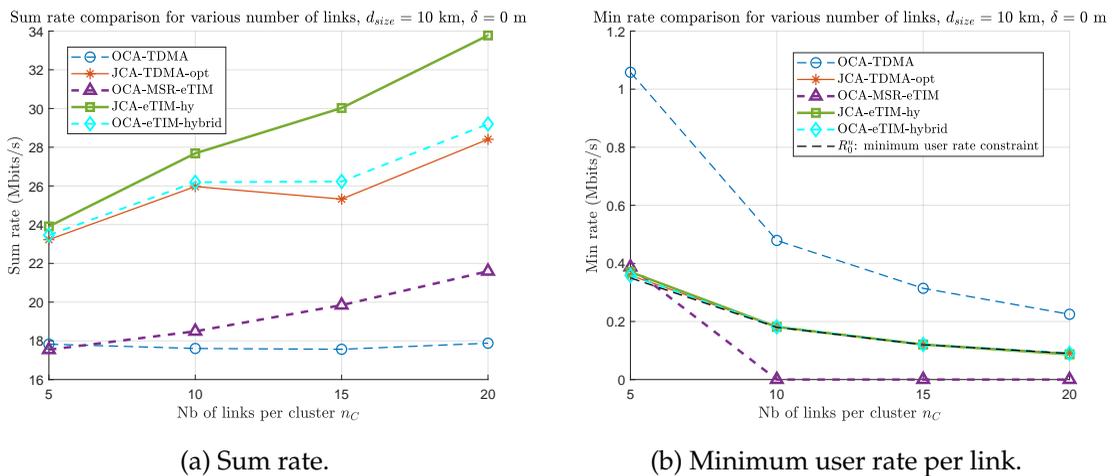


Figure 2.17: Comparison of JFA-eTIM-hy with other implemented methods versus the number of links per cluster for $d_{size} = 10$ km and $\delta = 0$ m. JFA-eTIM-hy requires clusters association and shows higher performance in terms of average sum rate.

In Fig. 2.18, we study the influence of the distance δ between the clusters by plotting the average sum rate in Fig. 2.18a and the average minimum user rate per link in Fig. 2.18b when two frequency bands are available. In terms of the average sum rate, we observe in Fig. 2.18a that JFA-eTIM-hy has an increasing value of the average sum rate when δ

increases which is due to the decrease of inter-cluster interference. On the contrary, when δ increases, JFA-TDMA-opt's average sum rate slightly decreases. Indeed, the decreasing behavior of JFA-TDMA-opt's average sum rate when δ increases can be explained by the following steps: i) for low δ , inter-cluster interference becomes stronger, ii) the performance of MSR-eTIM on the links of both clusters to partition the links into \mathcal{T}_e and \mathcal{T}_d decrease compared to round-robin TDMA, which leads to iii) a decreasing number of "good links" M in \mathcal{T}_e and finally, iv) when M decreases, the M best links in \mathcal{T}_e will transmit more often which will increase the overall network's average sum rate. Furthermore, we observe that our proposed solution JFA-eTIM-hy shows a higher average sum rate than the other methods for $\delta \geq 0$ m. The difference with OFA-eTIM-hybrid is 1.5 Mbits/s for $\delta = 0$, 2.9 Mbits/s for $\delta = 2000$ m and keeps increasing for increasing value of δ . However, for $\delta < 0$, the intertwining of both clusters creates stronger inter-cluster interference which makes eTIM solutions have trouble dealing with this strong interference, compared to TDMA techniques. In addition, Fig. 2.18a shows that the methods based on orthogonal frequency band allocation in dotted line have steady performance in terms of average sum rate and average minimum user rate per link, which is expected since interference management is performed independently for both clusters. Fig. 2.18b confirms that each method respects the average minimum user rate constraint, except OFA-MSR-eTIM as previously said.

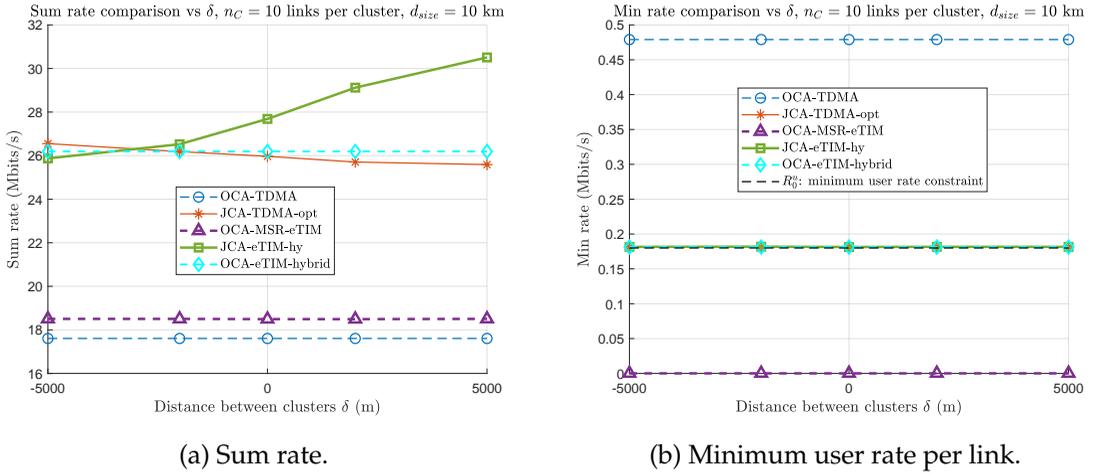


Figure 2.18: Comparison of JFA-eTIM-hy with other implemented methods versus δ for $n_C = 10$ links per cluster and for $d_{\text{size}} = 10$ km. JFA-eTIM-hy requires clusters association and shows higher performance in terms of average sum rate.

In Fig. 2.19, we study the influence of the size of the clusters d_{size} by plotting the average sum rate in Fig. 2.19a and the average minimum user rate per link in Fig. 2.19b. We observe that the implemented methods have a decreasing average sum rate and minimum user rate with for increasing d_{size} . This can be explained by the fact that with smaller clusters,

i.e. decreasing d_{size} , the proximity between the transmitter and receiver of the direct links increases. Besides, Fig. 2.19a shows that our proposed method **JFA-eTIM-hy** presents the highest average sum rate for $d_{\text{size}} \geq 10$ km. The difference between **OFA-eTIM-hybrid** is 1.5 Mbits/s for $d_{\text{size}} = 10$ km and 2.6 Mbits/s for $d_{\text{size}} = 20$ km. However, when we have $d_{\text{size}} \leq 5$ km, both **JFA-eTIM-hy** and **OFA-eTIM-hybrid** average sum rate become similar as **JFA-TDMA-opt**. Indeed, as said previously, when clusters are too small, inter-cluster interference becomes stronger and using **eTIM** provides similar if not lower performance than with round-robin **TDMA** in terms of sum rate. Fig. 2.19b confirms that each method respects the minimum rate constraint, except orthogonal **eTIM** as previously said.

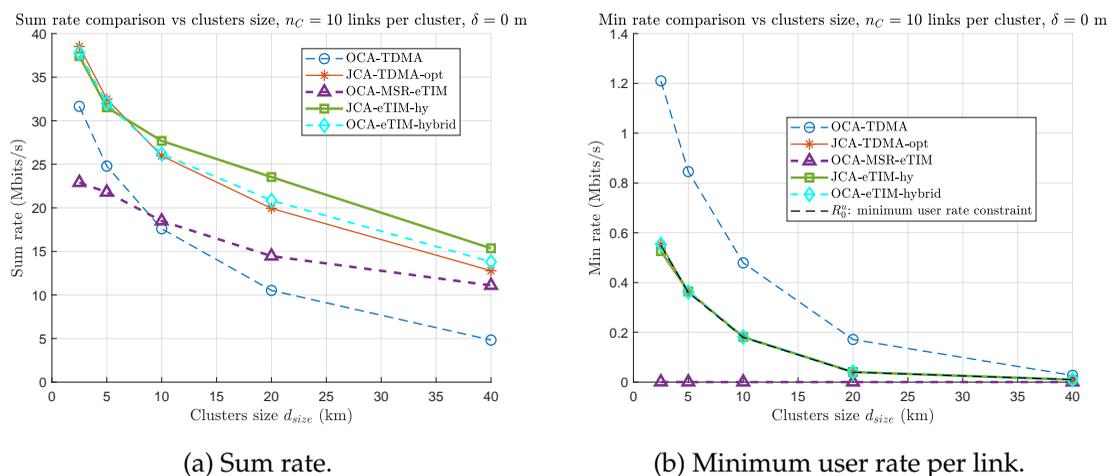


Figure 2.19: Comparison of **JFA-eTIM-hy** with other implemented methods versus δ for $n_C = 10$ links per cluster and $\delta = 0$ m. **JFA-eTIM-hy** requires clusters association and shows higher performance in terms of average sum rate.

2.6 Conclusion

In this chapter, we have addressed the **TIM** problem under frequency-flat Rayleigh fading channel with **SCSI** assumption. When a single frequency band is available, we have implemented several approaches that presented issues in either average sum rate or minimum user rate. We have proposed a new interference management scheme, called **eTIM-hybrid**, which combines **eTIM** with **TDMA** access. The scheme consists of two steps. First, the links are divided into two groups, with either **eTIM** or **TDMA** applied to each group. To do so, an iterative procedure based on rate criteria is performed. Second, we allocate the best slots to each link in a given frame. To this end, we formulate the scheduling problem and provide an optimal closed form solution. Through numerical simulations, we have shown that **eTIM-hybrid** outperforms the other solutions in most scenarios in terms of average sum rate while guaranteeing an average minimum user rate

for every link of both clusters. The comparison of our implemented solutions in terms of average sum rate and minimum user rate is summarized in Table 2.7.

Table 2.7: Comparison of the proposed solutions for one available frequency band.

Implemented solutions	Average sum rate	Average minimum user rate
MSR-eTIM Section 2.3.2	Better than TDMA	Some average user rates close to 0.
CMSR-eTIM Section 2.3.3.2	Similar to TDMA	Min. user rate guaranteed
PF-eTIM Section 2.3.3.1	Similar to TDMA	Min. user rate guaranteed
eTIM-hybrid	Better than TDMA	Min. user rate guaranteed

Furthermore, we have also considered another context when two frequency bands are available. One naive approach is to perform an orthogonal allocation of the frequency bands to both clusters and apply interference management methods. In this chapter, we have proposed a new scheme called [JFA-eTIM-hy](#) which combines a new frequency band allocation and [eTIM-hybrid](#) using the same links partition algorithm for [eTIM-hybrid](#). We show through numerical simulations that [JFA-eTIM-hy](#) outperforms the standard orthogonal frequency allocation solutions, including when using [eTIM-hybrid](#), in terms of average sum rate while guaranteeing the minimum user rate constraint. Our implemented solutions comparison when two frequency bands is summarized in Table 2.8.

Table 2.8: Comparison of the proposed solutions for two available frequency bands, all of their average sum rate are higher than [TDMA](#)'s

Implemented solutions	Average sum rate	Average minimum user rate
OFA-MSR-eTIM	4th	Some average user rates close to 0
OFA-eTIM-hy	2nd	Min. user rate guaranteed
JFA-TDMA-opt	3rd	Min. user rate guaranteed
JFA-eTIM-hy	1st	Min. user rate guaranteed

Results from this chapter have been published in [53].

Chapter 3

Rayleigh channel statistics estimation

3.1 Introduction

3.1.1 Context and state-of-the-art

In this chapter, we address the estimation of the propagation channel statistics under frequency-flat Rayleigh fading in an ad hoc network. This estimation is motivated by the implementation of the interference management methods presented in Chapter 2. We remind that these solutions require the use of channel statistics to avoid the idealistic assumption requiring perfect instantaneous channel coefficients. Therefore, we choose to focus on the [SCSI](#) context like in the previous chapter.

When considering Rayleigh fading, the statistical parameters of the channel are reduced to their variance and can therefore be estimated by taking the empirical variance of the estimated samples of the channel. These samples can be estimated using channel estimation techniques such as least-square ([LS](#)) or minimum mean-square error ([MMSE](#)) estimators [[101](#)]. For the Rice channel, a method to estimate its statistical parameters from noisy samples of the channel is proposed in [[71](#)] and from noisy samples and under Nakagami- m shadowing in [[72](#)]. For the Nakagami- m fading channel, an implementation to estimate its statistical parameters is proposed in [[68](#)]. However, these methods do not provide procedures to estimate the channel statistics of all pairs of nodes of the network.

In this chapter, we consider a network that can for example correspond to the nodes of a cluster or a group of clusters as mentioned in Chapter 1. The goal of this chapter is to estimate the channel statistics of all pairs of nodes of the network assuming [TDD](#) systems. To this end, one solution is to use [TDMA](#) where each node successively transmits to the other nodes until the channel statistics of all pairs of nodes of the network that we consider have been estimated. With this method, the receivers can perform channel statistics estimation methods without any interference. An alternative solution is to allow the nodes to transmit simultaneously which decreases the sensing duration compared to [TDMA](#). Such an approach requires the use of scheduling schemes to estimate the channel

statistics of all pairs of links because of the **TDD** mode assumption. An alternative solution is to allow the nodes to transmit simultaneously which decreases the sensing duration compared to **TDMA**. Because of the **TDD** mode assumption, such an approach requires combining simultaneous transmissions with the use of scheduling schemes to estimate the channel statistics of all pairs of links. Multiple works have aimed at optimizing the scheduling in a multi-user network. In [58, 69], the scheduling is optimized under the **TDD** assumption to satisfy certain traffic demands. [69] also aims at minimizing the length of the scheduling to decrease the duration of the transmissions. Nevertheless, these solutions describe scheduling methods assuming the nodes send different transmissions for each receiver. Since in our context, each transmitter sends the same sequences to the other nodes, the scheduling solutions proposed in [58, 69] are not adapted to our systems and would even present lower performance than **TDMA**.

Another issue with simultaneous transmissions is the occurrence of interference which can degrade the channel estimation performance. To deal with that, we propose to reduce the sensing duration of the channel statistics estimation of the network pairs of nodes, while dealing with the resulting interference. To this end, we describe two scheduling approaches which are described as follows:

- The first approach consists of allowing two nodes to transmit simultaneously which decreases the number of steps required to estimate the channel statistics of all pairs of nodes of the network, and thus decreases the sensing duration compared to **TDMA**. Since this method results in the occurrence of a single interference, the channel statistics estimation problem is equivalent for each receiver to estimate two scalar parameters: the direct and the interfering channels' magnitude mean power. To that purpose, the estimation is performed in a frequency-flat Rayleigh channel in the presence of a single interferer assuming that we collect a set of **SINR** values from the receiver.
- In the second approach, we propose a scheme allowing simultaneous transmissions from multiple nodes, i.e. more than two nodes. To deal with the interference created by several transmitters, we use a procedure using orthogonal sequences to remove that. Under the assumption of **TDD** mode systems, we perform a transmission scheduling optimization to outperform **TDMA** in terms of sensing duration while estimating the channel statistics of all pairs of nodes of the network.

We show in the following that the second scheduling approach outperforms the first one in terms of sensing duration. The first scheduling method can be performed if a procedure with orthogonal sequences cannot be used and under the assumption that receivers have only access to a set of **SINR** samples.

3.1.2 Contributions

The contributions are summarized as follows.

3.1.2.1 Rayleigh channel statistics estimation using SINR samples under single interference

The first approach for the channel statistics estimation of the pairs of nodes of the network consists of allowing two nodes to transmit simultaneously. The estimation problem is thus equivalent to estimating the direct and interfering channels' magnitude average power under single interference. The contributions to this problem are described in the following:

- The first contribution is a scheduling method to reach all users within less amount of time than TDMA.
- Assuming a set of SINR samples is available at the receiver and under flat Rayleigh fading, the second contribution is the derivation of two estimators: the first one based on MLE, and the other one based on the MoM. The performance of these estimators is compared in terms of bias and variance through numerical simulations.
- The third contribution of this approach is the derivation of the CRLB of the estimation problem in closed form. It is then numerically computed and compared with the other estimators' variance.

3.1.2.2 Scheduling optimization with simultaneous transmissions for channel statistics estimation

In the second approach to estimate the channel statistics, simultaneous transmissions from multiple nodes are allowed. The contribution is stated as follows: assuming TDD systems, we propose a scheduling solution using orthogonal sequences with a lower sensing duration for the channel statistics estimation of all pairs of nodes of the network. The complexity and performance of the proposed scheduling are analyzed through numerical simulations.

3.1.3 Chapter organization

The rest of the chapter is organized as follows. In Section 3.2, we describe the scheduling approach with two simultaneous transmissions to estimate the channel statistics of all the links of the network and formulate the equivalent estimation problem. Assuming the use of SINR samples under single interference, two estimators and the CRLB of the estimation problem are derived. In Section 3.3, we provide the second scheduling approach which allows simultaneous transmissions from multiple nodes to estimate the channel statistics of the links of the network. A novel optimized scheduling using orthogonal sequences is derived and we show that it outperforms TDMA in terms of sensing duration. Section 3.4 concludes the chapter.

3.2 Channel statistics estimation using SINR samples under single interference and Rayleigh fading

In this section, we describe the first approach to estimate the channel statistics of all pairs of nodes of the network with simultaneous transmissions. To decrease the sensing duration compared to TDMA, two nodes are allowed to simultaneously transmit to the others, which reduces the time to take the measure of all pairs of nodes. Since this scheduling results in the occurrence of a single interference, the channel statistics estimation problem amounts for each receiver to estimate two scalar parameters: the direct and the interfering channels' magnitude mean power.

Moreover, we also assume that the receiver considered in this work gives only access to a set of estimated SINR samples. The goal of this section is thus to estimate the direct and interfering statistical parameters from the available SINR samples. The authors of [89] have proposed a maximum likelihood estimator (MLE) of the SINR statistical parameters in the context of shadowing effects, by approximating the distribution of the SINR samples using a log-normal distribution. Unfortunately, such an approximation should present a lower performance than an estimator assuming the true distribution of the SINR. Notice that other works use the SINR samples by directly feeding them into a neural network to predict the SINR distribution [21]. However, our solution is performed without using the SINR statistical parameters, which are the values we want to estimate.

This approach aims to investigate at each receiver the estimation of the direct and interfering channels statistics without approximating the SINR distribution using a log-normal distribution as in [89]. We tackle this problem under flat Rayleigh fading with a single interferer assuming that we collect a set of SINRs' values from the receiver. To do so, we propose two new estimators:

- the MLE which is more accurate but requires careful initialization.
- the MoM estimator which is less complex but also less accurate.

For both estimators, we provide derivations to the schemes along with a numerical implementation based on Newton's algorithm. In addition, we derive the CRLB associated with this estimation problem and propose an efficient computation of it. We provide simulations to assess the performance and compare the complexity of the proposed estimators.

This section is organized as follows. Section 3.2.1 describes the system model and the proposed scheduling solution with two simultaneous transmissions, then formulates the equivalent estimation problem for the channel statistics. Section 3.2.2 shows how to derive the proposed estimators and their computation. Section 3.2.3 is dedicated to the derivation of the CRLB. Section 3.2.4 provides numerical results and performance comparison of the proposed estimators.

3.2.1 System model and scheduling solutions

In this chapter, we consider a network composed of N nodes which can correspond to the nodes of a cluster or a group of clusters. Note that this differs from Chapters 2 and 4 where we consider a network composed of N links. The systems are assumed to be in TDD mode, i.e. each node can either operate as a transmitter or a receiver. Moreover, we refer to a *link* as a pair of transmitter-receiver, i.e. the link is directed, as opposed to a *pair of nodes* that is not directed. In this context, the channel statistics γ_{ij} of all pairs of nodes (i, j) are supposed unknown and constant, and our goal is to estimate them. It is worth noting that unlike in Chapters 2 and 4, the notations γ_{ij} correspond to the channel statistics from node j to node i . With these notations, the channel average power symmetry leads to the equality $\gamma_{ij} = \gamma_{ji}$ and γ_{ii} is not defined.

In the following, we introduce a few definitions and describe the scheduling with TDMA.

3.2.1.1 Definitions and scheduling with TDMA

We first introduce definitions to compare different transmission scheduling solutions for channel statistics estimation.

Definition 3.1. We define a *cycle* as the total time required to estimate the channel statistics of all pairs of nodes of the network.

Definition 3.2. We define a *step* as the time during which the nodes transmit and the receivers estimate the channel statistics from the received signals.

Definition 3.3. We define the *length* of a scheduling as the total number of steps of the scheduling.

Definition 3.4. We define a *successful scheduling* as a scheduling leading to the estimation of the channel statistics of all pairs of nodes of the network.

To estimate the channel statistics of all pairs of nodes of the network, one solution is to use TDMA, i.e. the nodes transmit successively one after each other, and when a node is transmitting, all other nodes will be in receiver mode. This continues until all pairs of nodes are reached. This method is illustrated in Fig. 3.1 where node 1 first sends transmission to the other nodes, then node 2, and so on. Using these definitions, TDMA is considered a successful scheduling with a length of $N - 1$ steps since it allows to estimate γ_{ij} for all pairs of nodes (i, j) of the network, using the symmetry $\gamma_{ij} = \gamma_{ji}$.

We propose an alternative solution that allows two nodes to transmit simultaneously to decrease the sensing duration. This approach results in two issues:

- Because of the TDD assumption, it requires the use of scheduling techniques to both estimate the channel statistics of all pairs of nodes and perform better than TDMA in terms of the number of steps.

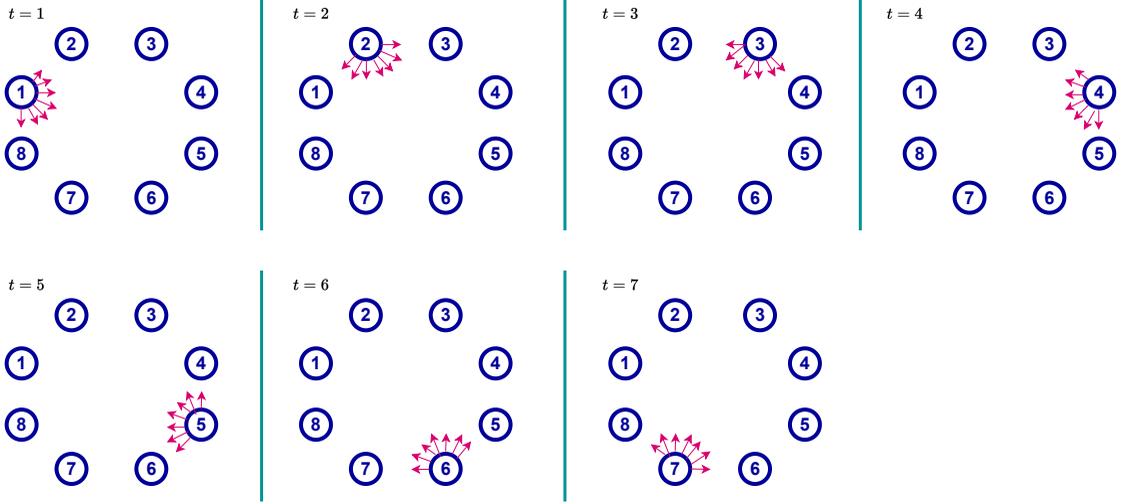


Figure 3.1: Example of scheduling with TDMA for $N = 8$ nodes. The channel statistics of all links are estimated in 7 steps.

- It leads to the occurrence of a single interference at receivers which can decrease the performance of the channel statistics estimation.

In the next section, we provide the proposed scheduling with two simultaneous transmissions.

3.2.1.2 Proposed scheduling solution

In this section, we describe the proposed scheduling solution with two nodes simultaneously transmitting at each step until all pairs of nodes of the network are reached. Its procedure is composed of the following phases:

1. Two adjacent nodes successively transmit until the $(N - 2)$ th node is reached. In other words, nodes 1, 2 transmit at step 1, nodes 3, 4 at step 2, etc., until node $(N - 2)$ transmits.
2. Two adjacent odd numbers successively transmit until the remaining channel statistics are estimated. In other words, nodes 1, 3 transmit at step $(L_{2S}^{(1)} + 1)$, nodes 5, 7 at step $(L_{2S}^{(1)} + 2)$, etc., until all the γ_{ij} are estimated.

An example of this scheduling method applied to a network composed of $N = 8$ nodes is represented in Fig. 3.2, where the scheduling is successful in 5 steps. The first phase of the scheduling takes around half the time of TDMA since two nodes successively transmit. However, because of the TDD mode assumption, some channel statistics are not estimated in the first phase. Indeed, when two nodes i and j transmit at the same step, γ_{ij} is not estimated and either node i or j is required to transmit again. Therefore, in the second phase, only one out of two nodes transmit a second time.

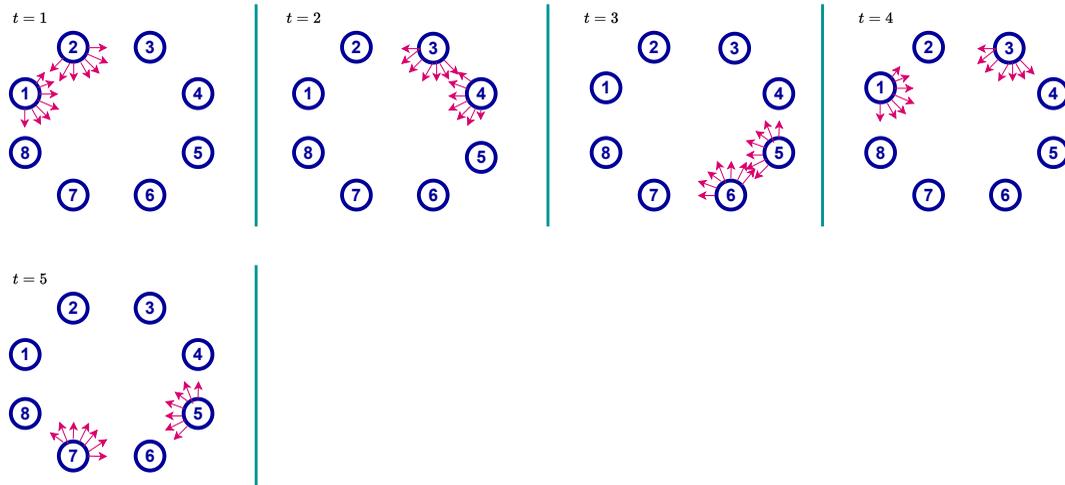


Figure 3.2: Example of scheduling with two simultaneous transmissions for $N = 8$ nodes performed in 5 steps.

Without providing any analytical expression, we can conjecture that the proposed scheduling with two simultaneous transmissions seems to outperform TDMA in terms of scheduling length, and thus, in terms of sensing duration.

In the next section, to handle the resulting interference, we formulate at each receiver the estimation problem for the channel statistics with a single interferer. This procedure is applied at each receiver to estimate the channel statistics of all pairs of nodes. The estimation is performed under flat Rayleigh fading assuming that we collect a set of SINR values from the receiver.

3.2.1.3 Formulation of the equivalent estimation problem

We have previously considered a procedure with two simultaneous transmissions for channel statistics estimation. In this section, we considered this estimation problem at each receiver. To that purpose, we consider a network illustrated in Fig. 3.3 and composed of three nodes: a transmitter Tx^1 communicating with its receiver Rx^1 and an interfering transmitter Tx^2 communicating on the same channel. As a consequence, Tx^2 creates interference at the receiver Rx^1 . In this case, the received signal at Rx^1 can be written as:

$$y_1 = \sqrt{P_1\gamma_1} \tilde{h}_1 s_1 + \sqrt{P_2\gamma_2} \tilde{h}_2 s_2 + n_1, \quad (3.1)$$

where s_i is the transmitted symbol from Tx^i , P_i is the transmit power of Tx^i , $\tilde{h}_j \sim CN(0,1)$, $j = 1,2$ are the Rayleigh flat fading coefficients from transmitter Tx^j to receiver Rx^1 , γ_1 and γ_2 are the direct and interfering channels' magnitude respectively and correspond to pathloss values, and $n_1 \sim CN(0, P_n)$ is the AWGN at receiver Rx^1 . In this chapter, we assume that the noise power P_n is perfectly known.

The goal of this section is to estimate γ_1 and γ_2 which are supposed unknown and constant. As in [20, 21, 43, 89], we assume that receiver Rx^1 delivers instantaneous SINR

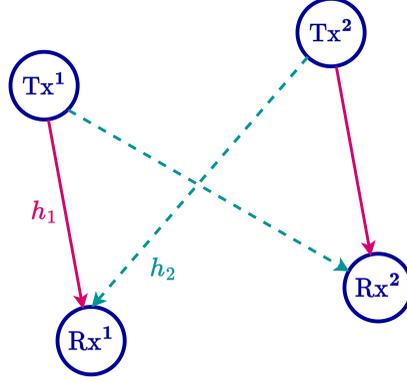


Figure 3.3: Representation of a transmission from Tx^1 to Rx^1 with the interferer Tx^2 .

estimated values. The instantaneous SINR is defined as $\mathcal{S} := P_1\gamma_1|\tilde{h}_{11}|^2/(P_n + P_2\gamma_2|\tilde{h}_{12}|^2)$, which can be reexpressed as:

$$\mathcal{S} = \frac{a\gamma_1\alpha_1}{1 + b\gamma_2\alpha_2}, \quad (3.2)$$

with $a := P_1/P_n$, $b := P_2/P_n$, $\alpha_1 := |\tilde{h}_{11}|^2$, $\alpha_2 := |\tilde{h}_{12}|^2$.

With these notations, we define the instantaneous SNR as $\text{SNR}_{\text{inst}} := a\gamma_1\alpha_1$ and the instantaneous Interference-to-Noise ratio (INR) as $\text{INR}_{\text{inst}} := b\gamma_2\alpha_2$. In the following, for the sake of clarity, the terms "SNR" and "INR" will refer to the expected value of SNR_{inst} and INR_{inst} , which are equal to $a\gamma_1$ and $b\gamma_2$ respectively.

Since we assume that $\tilde{h}_i \sim \mathcal{CN}(0, 1)$, $i = 1, 2$, the fading coefficients α_1 and α_2 follow an exponential distribution with parameter $\lambda = 1$ [94, Section 2.3]. In [32], the pdf of \mathcal{S} is derived and can be written as follows

$$p_{\mathcal{S}}(x; \gamma) = \exp\left(-\frac{x}{a\gamma_1}\right) \frac{a\gamma_1 + b\gamma_2x + a\gamma_1b\gamma_2}{(a\gamma_1 + b\gamma_2x)^2}, \quad (3.3)$$

with $\gamma := (\gamma_1, \gamma_2)$.

In the following, we assume that we have access to N_S independent and identically distributed (i.i.d.) samples of SINR denoted by $\mathbf{S} := (\mathcal{S}_1, \dots, \mathcal{S}_{N_S})$ to perform the estimation of γ_1 and γ_2 .

3.2.2 Proposed estimators

In this section, we derive two different algorithms to estimate γ_1 and γ_2 from \mathbf{S} . We remind that this estimation procedure can be performed at each receiver of the network considered in Section 3.2.1 to estimate the channel statistics of all pairs of nodes. The proposed estimation solutions are the following: one is based on the maximum likelihood approach and the other one is on the MoM. For both estimators, we provide derivation of the estimators along with a numerical implementation based on Newton's method. The MoM estimator is less complex but also less accurate, whereas the MLE is more accurate

but requires careful initialization. Thus, we also propose to use the MoM's estimates to initialize the MLE algorithm, which results in an MLE with lower complexity.

3.2.2.1 Maximum likelihood estimator

The MLE is one of the most popular estimators since it has the asymptotic properties of being i) consistent, i.e. it converges in probability to the true parameters, and ii) efficient, that is when $N_S \rightarrow +\infty$, it achieves the CRLB [64, Chapter 7]. In our problem, the MLE of γ is obtained by maximizing the log-likelihood function of \mathbf{S} denoted by $\ell_S(\gamma)$.

Since the elements of \mathbf{S} are independent and identically distributed (i.i.d), the log-likelihood $\ell_S(\gamma)$ can be expressed as:

$$\ell_S(\gamma) = \log \left(\prod_{i=1}^{N_S} p_S(\mathcal{S}_i; \gamma) \right). \quad (3.4)$$

Plugging (3.3) into (3.4) yields to the following expression for the log-likelihood:

$$\ell_S(\gamma) = \sum_{i=1}^{N_S} \left[-\frac{\mathcal{S}_i}{a\gamma_1} + \log(a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2) - 2 \log(a\gamma_1 + b\gamma_2\mathcal{S}_i) \right]. \quad (3.5)$$

One possibility to maximize $\ell(\mathbf{S}; \gamma)$ with respect to γ is to perform a grid search over a finite interval with a small grid step to have a good precision of the maximum likelihood estimates. However, its implementation is complex because it requires the evaluation of the log-likelihood on each point of a grid which must be accurate enough to ensure a good estimation precision.

Another possibility to maximize $\ell_S(\gamma)$ with respect to γ is to find its stationary points, i.e. to find γ_{MLE} such that

Result 3.1 (Maximum likelihood estimator). *The MLE estimates are provided by solving the following equation:*

$$\nabla \ell_S(\gamma_{MLE}) = [0, 0]^T, \quad (3.6)$$

where $\nabla \ell_S(\gamma) := [\frac{\partial}{\partial \gamma_1} \ell_S(\gamma), \frac{\partial}{\partial \gamma_2} \ell_S(\gamma)]^T$ is the gradient vector of $\ell_S(\gamma)$ with respect to γ and $\frac{\partial}{\partial \gamma_i} \ell_S(\gamma)$ is the partial derivative of $\ell_S(\gamma)$ with respect to γ_i , and to select the one maximizing $\ell_S(\gamma)$ among the stationary points.

After some algebra, we obtain the log-likelihood partial derivatives with respect to γ_1 and γ_2 :

$$\begin{aligned} \frac{\partial}{\partial \gamma_1} \ell_S(\gamma) &= \sum_{i=1}^{N_S} \left(\frac{\mathcal{S}_i}{a\gamma_1^2} + \frac{a(1+b\gamma_2)}{a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2} - \frac{2a}{a\gamma_1 + b\gamma_2\mathcal{S}_i} \right) \\ \frac{\partial}{\partial \gamma_2} \ell_S(\gamma) &= \sum_{i=1}^{N_S} \left(\frac{b(\mathcal{S}_i + a\gamma_1)}{a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2} - \frac{2b\mathcal{S}_i}{a\gamma_1 + b\gamma_2\mathcal{S}_i} \right) \end{aligned} \quad (3.7)$$

Without being able to prove it formally, we provide after a numerical study of $\ell_S(\gamma)$ the following conjecture.

Conjecture 1. *The function $\ell_S(\gamma)$ has the following two properties:*

1. *it has a unique stationary point.*
2. *it is quasi-concave.*

We deduce from 1. that the identification of a stationary point of $\ell_S(\gamma)$ by any optimization method is the global optimum, whereas property 2. indicates that the initialization of the optimization method must be done carefully to guarantee the convergence of the used optimization algorithm.

We thus propose to use a descent method based on Newton's algorithm [31] to solve $\nabla \ell_S(\gamma_{\text{MLE}}) = [0, 0]^T$. The descent vector is $-\mathbf{H}_\ell^{-1} \nabla \ell_S(\gamma)$ where \mathbf{H}_ℓ is the Hessian matrix of $\ell_S(\gamma)$ whose expression is detailed in Appendix B.1.1 and the Newton's iteration step can be expressed as:

$$\gamma^{(t+1)} = \gamma^{(t)} - \mathbf{H}^{-1} \nabla \ell(\mathbf{x}, \gamma^{(t)}). \quad (3.8)$$

It is worth noticing that since \mathbf{H}_ℓ is a 2×2 matrix, we can find the closed-form expressions of its inverse \mathbf{H}_ℓ^{-1} . Moreover, it is known that the condition to guarantee the convergence of

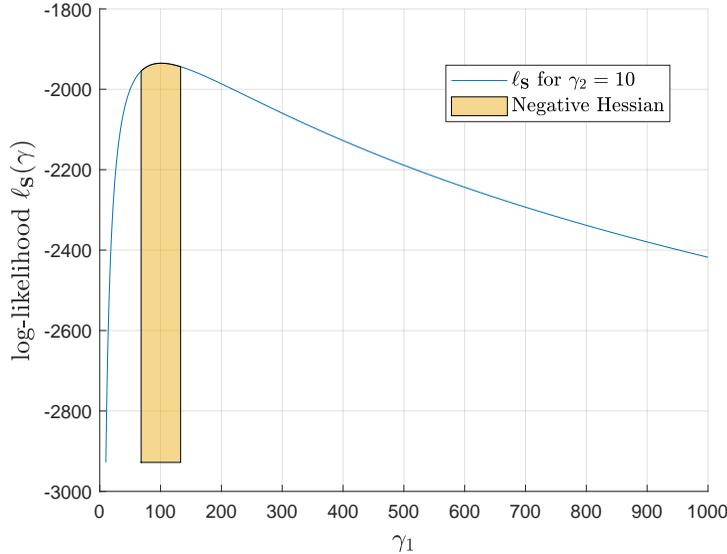


Figure 3.4: Log-likelihood $\ell_S(\gamma)$ for $\gamma_2 = 10$, versus γ_1 . The interval in yellow corresponds to negative definite Hessian \mathbf{H}_ℓ . Hence, the initialization of γ_1 for Newton's method for $\gamma_2 = 10$ requires to be in this interval.

Newton's algorithm is that \mathbf{H}_ℓ is negative definite [14]. Therefore, a careful initialization point respecting this condition is required. A representation of $\ell_S(\gamma)$ with the region where \mathbf{H}_ℓ is negative definite is illustrated in Fig. 3.4.

We propose two alternatives for the initialization of the Newton's method.

- i) An initialization based on a random trial and error search is described in the rest of this section.

ii) An initialization using MoM estimates which are provided in Section 3.2.2.2.

The first possibility is described in Algorithm 3.1 and consists in initializing Newton's method through a random trial and error search until we find a point such that \mathbf{H}_ℓ is negative definite. To this end, we define a region set $\Theta_{\gamma_1, \gamma_2}$ from which we randomly draw our initial points $\gamma^{(0)}$. This region can be written as:

$$\Theta_{\gamma_1, \gamma_2} := \{(\gamma_1, \gamma_2) \mid \gamma_1 \in [\gamma_{1,\min}, \gamma_{1,\max}] \text{ and } \gamma_2 \in [\gamma_{2,\min}, \gamma_{2,\max}]\}, \quad (3.9)$$

where $\gamma_{i,\min}, \gamma_{i,\max}$ are γ_i minimal and maximum values to set, with $i = 1, 2$.

When γ_2 is close to 0, its influence is smaller when facing a stronger noise power and its estimation becomes more difficult. In those cases, we struggle to find a suitable initialization point for Newton's method, i.e. such that the Hessian matrix is negative. Therefore, in practice, we set a maximum number of random initializations $N_{init,MLE}$. If the $N_{init,MLE}$ initializations are reached without a negative Hessian matrix, we set $\gamma_{2,MLE} = 0$. Given the properties of $\ell_S(\gamma)$, this condition guarantees the convergence of the Newton's

Algorithm 3.1: Newton's method implementation for MLE

```

Set  $\mathbf{S}, \epsilon_g, \epsilon_\alpha, \Theta_{\gamma_1, \gamma_2}, \gamma_{MoM}$ .
Randomly initialize  $\gamma^{(0)}$  in  $\Theta_{\gamma_1, \gamma_2}$ .
while  $\mathbf{H}_\ell(\gamma^{(0)}) \not\leq 0$  do
    | Randomly initialize  $\gamma^{(0)}$  in  $\Theta_{\gamma_1, \gamma_2}$ .
end
while  $\|\nabla \ell_S(\gamma^{(n)})\| > \epsilon_g$  or  $n < N_{max}$  do
    | Initialize  $\alpha = 1$ .
    | Update  $\gamma_\alpha^{(n+1)} = \gamma^{(n)} - \alpha \mathbf{H}_\ell^{-1}(\gamma^{(n)}) \mathbf{g}_\ell(\gamma^{(n)})$ .
    | while  $(\ell(\gamma_\alpha^{(n+1)}) < \ell(\gamma^{(n)})$  or  $\mathbf{H}_\ell(\gamma_\alpha^{(n+1)}) \not\leq 0$ ) and  $\alpha > \epsilon_\alpha$  do
    | | Update  $\alpha = \alpha/2$  and set  $\gamma_\alpha^{(n+1)}$ .
    | end
    | Set  $\gamma^{(n+1)} = \gamma_\alpha^{(n+1)}$ .
    | Increment  $n = n + 1$ .
end
Return  $\gamma^{(n+1)}$ 
    
```

algorithm with a controlled step. The step is initialized to 1 and successively divided by 2 until the following two conditions are met:

i) $\ell_S(\gamma)$ increases.

ii) \mathbf{H}_ℓ is negative definite.

We have observed that the trial-and-error step is numerically complex since i) it requires multiple trials, and ii) each trial requires to compute \mathbf{H}_ℓ whose components are computationally high. Therefore, we propose a second possibility which consists of

initializing Newton's method with the MoM estimates. With these estimates, if \mathbf{H}_ℓ is not negative definite, Newton's method is initialized through the random trial and error search as described above. This procedure is illustrated in the block diagram in Fig. 3.5. We provide the MoM estimates in the next section.

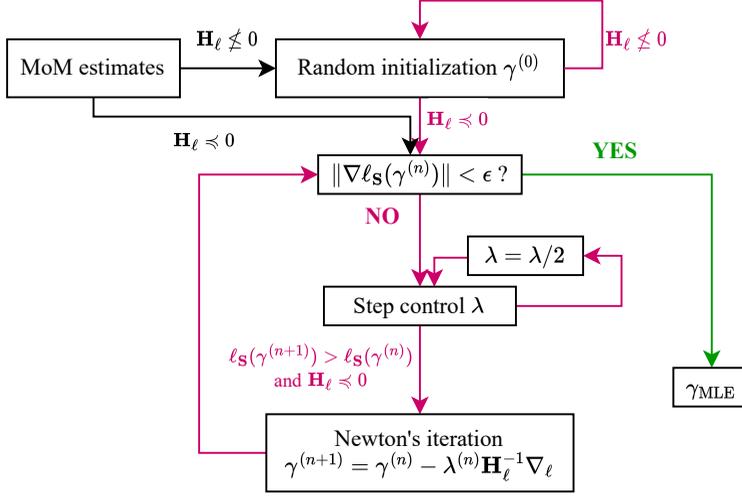


Figure 3.5: Illustration of MLE implementation initialized with MoM estimates.

3.2.2.2 Method of moments estimator

In this section, we derive an alternative estimator to the MLE one based on the MoM. The MoM is an estimator based on the solution of the equations matching the theoretical and empirical moments. It has the advantage of being asymptotically consistent [64, Chapter 9], but contrarily to the MLE, its asymptotic efficiency is not guaranteed. The MoM estimator is a conventional alternative to the MLE estimator when the later is complex to implement. In this chapter, we will show through a complexity analysis that the complexity of the MoM implementation is much lower than the one of the MLE estimator. In addition, if its performance is not satisfying enough, the MoM estimator can be used to initialize the MLE through Newton's method. In the next section, we derive a MoM estimator of the SINR distribution parameters γ_1 and γ_2 from \mathbf{S} .

3.2.2.2.1 Estimator derivation

Since there are two parameters to estimate and because the variance of the MoM estimator generally increases with the order [64], i.e. the number of moments, we choose to use the first and second moments. To match the first and second theoretical and empirical moments, let $M_1 := (1/N) \sum_{n=1}^N \mathcal{S}_n$ and $M_2 := (1/N) \sum_{n=1}^N \mathcal{S}_n^2$ be the first and second-order empirical moments of the received samples, and let $\phi_1(\gamma) := \mathbb{E}[\mathcal{S}]$, $\phi_2(\gamma) := \mathbb{E}[\mathcal{S}^2]$ be the first and second-order theoretical moments. Thus, the MoM estimates $\gamma_{\text{MoM}} =$

$(\gamma_{1,MoM}, \gamma_{2,MoM})$ are the solutions of the following system:

$$\begin{cases} \phi_1(\gamma_{MoM}) = M_1 \\ \phi_2(\gamma_{MoM}) = M_2. \end{cases} \quad (3.10)$$

Even though the theoretical moments $\phi_1(\gamma)$ and $\phi_2(\gamma)$ have been computed in [109], for the sake of completeness, we detail their derivations in Appendix B.1.2.1 using the notations of this thesis and provide their expression in the following:

$$\begin{cases} \phi_1(\gamma) = a\gamma_1\omega e^\omega E_1(\omega) \\ \phi_2(\gamma) = 2(a\gamma_1)^2\omega(1 - \omega e^\omega E_1(\omega)), \end{cases} \quad (3.11)$$

with $\omega := 1/(b\gamma_2)$ and $E_1(z) = \int_z^{+\infty} (e^{-t}/t)dt$ the exponential integral function. For the sake of completeness, we provide the derivations of $\phi_1(\gamma)$ and $\phi_2(\gamma)$ in Appendix B.1.2.1 using the notations of this thesis.

Defining

$$\rho := \frac{M_2}{(M_1)^2}, \quad (3.12)$$

we derive in Appendix B.1.2.2 the following result

Result 3.2 (Method of moment estimators). *The MoM estimators $\gamma_{1,MoM}$ and $\gamma_{2,MoM}$ are the solutions of the system*

$$\begin{cases} f_{MoM}(\gamma_{2,MoM}) = 0 \\ \gamma_{1,MoM} = \frac{M_1}{2a} \left(1 + \sqrt{1 + 2b\rho\gamma_{2,MoM}}\right), \end{cases} \quad (3.13a)$$

$$(3.13b)$$

where f_{MoM} is defined as

$$f_{MoM}(z) := \frac{1 + \sqrt{1 + 2b\rho z}}{2bz} \exp(1/bz) E_1(1/bz) - 1. \quad (3.14)$$

The solution of the system provided in (3.13) is done by first computing $\gamma_{2,MoM}$ by solving (3.13a), and then by inserting this value in (3.13b) to deduce $\gamma_{1,MoM}$. To characterize the solutions of (3.13a), we need to study the properties of the function $f_{MoM}(z)$, which is not straightforward.

3.2.2.2.2 Function f_{MoM} behavior study

This section aims to study the function f_{MoM} to find an adequate method to find its roots, i.e. the solutions of (3.13a). To this end, we study the intervals where f_{MoM} is increasing or decreasing by computing the derivative of f_{MoM} denoted as f'_{MoM} and by calculating f_{MoM} 's limits. After some derivations provided in Appendix B.1.2.3, the expression of f'_{MoM} can be expressed as follows:

$$f'_{MoM}(z) = \frac{1}{2bz} \left(g'_2(z)g_{34}(z) + \frac{g_2(z)}{z} \left(1 - \frac{g_{34}(z)}{bz}(1 + bz)\right) \right), \quad (3.15)$$

with $g_2(z) := 1 + \sqrt{1 + 2b\rho z}$, $g_{34}(z) := \exp(1/bz)E_1(1/bz)$ and $g'_2(z) = b\rho(1 + 2b\rho z)^{-1/2}$.

Since we were not able to find f'_{MoM} 's sign on \mathbb{R}^+ because of its complex expression, we were able to compute its limit in 0^+ . Let us define $f(0^+) := \lim_{z \rightarrow 0^+} f(z)$ and $f'(0^+) := \lim_{z \rightarrow 0^+} f'(z)$ when the limits exist. We were able to find the following limits, whose proofs are provided in Appendix B.1.2.4.

Result 3.3. Let f_{MoM} be defined in (3.14) and f'_{MoM} defined in (3.15). The following properties hold:

$$p_1) f_{\text{MoM}}(0^+) = 0,$$

$$p_2) \lim_{z \rightarrow +\infty} f_{\text{MoM}}(z) = -1,$$

$$p_3) f'_{\text{MoM}}(0^+) = b(\rho/2 - 1).$$

From p_3 in Result 3.3, we can deduce the following corollary which provides a condition on the sign of $f'_{\text{MoM}}(0^+)$ that only depends on the empirical moment ratio ρ .

Corollary 3.1. From p_3 , we deduce that the sign of $f'_{\text{MoM}}(0^+)$ only depends on the ratio ρ , and thus we have the following statement:

$$\begin{aligned} \forall \rho > 2, \quad f'_{\text{MoM}}(0^+) &> 0 \\ \forall \rho \leq 2, \quad f'_{\text{MoM}}(0^+) &\leq 0. \end{aligned} \tag{3.16}$$

Next, we have also observed numerically that the function in (3.14) has the following properties (which we conjecture):

Conjecture 2. We conjecture the following properties on f_{MoM} and f'_{MoM} :

$$p_4) f'_{\text{MoM}}(0^+) \leq 0 \Rightarrow \forall z > 0, f_{\text{MoM}}(z) < 0$$

$$p_5) f'_{\text{MoM}}(0^+) > 0 \Rightarrow \exists! z_1 > 0 \text{ such that } f_{\text{MoM}}(z_1) = 0.$$

$$p_6) f'_{\text{MoM}}(0^+) > 0 \Rightarrow \exists! z_2 > 0 \text{ such that } f'_{\text{MoM}}(z_2) = 0.$$

Using the properties from Result 3.3 and Conjecture 2, we can distinguish two behaviors of f_{MoM} as illustrated in Fig. 3.6. First, from p_1 , p_2 and p_4 , we can deduce that if $f'_{\text{MoM}}(0^+) \leq 0$, function $f_{\text{MoM}}(z)$ behaves as follows: it is always negative and goes to -1 when $z \rightarrow +\infty$. In this case, f_{MoM} does not have any root. Otherwise, if $f'_{\text{MoM}}(0^+) > 0$, then (3.14) has a unique strictly positive root z_1 according to property p_5 , and $\gamma_{2,\text{MoM}}$ is estimated by finding the unique non-zero root of f_{MoM} in (3.13a). Moreover, we deduce from properties p_1 , p_2 , p_5 and p_6 that function $f_{\text{MoM}}(z)$ behaves as illustrated in Fig. 3.6 when INR = 10 dB: it is first positive and increasing, passes through a maximum and then decreases. It intersects the z axis at $z = z_1$, then becomes negative and goes to -1 when $z \rightarrow +\infty$.

From Corollary 3.1 and Conjecture 2, we deduce the link between the value of ρ and the existence of f_{MoM} 's non-zero root through the following Corollary:

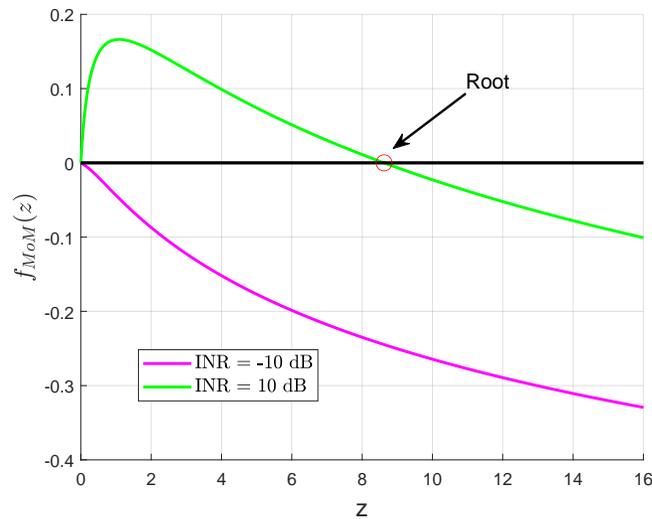


Figure 3.6: Examples of f_{MoM} defined in (3.14) with one non-zero root for $\text{INR}=10$ dB and without any non-zero root for $\text{INR}=-10$ dB.

Corollary 3.2. Let $\rho > 0$.

- If $\rho > 2$, f_{MoM} has a unique non-zero root.
- Else, if $\rho \leq 2$, f_{MoM} does not have any non-zero root.

3.2.2.3 Proposed implementation of the estimators

3.2.2.3.1 General approach

From Corollary 3.2, we propose to take into account ρ 's value in the estimators implementation detailed in the following result:

Result 3.4 (Estimators implementation).

- When $\rho > 2$, we estimate γ_2 and then γ_1 using the *MoM* estimators described in Result 3.2 and the *MLE* described in Result 3.1.
- When $\rho \leq 2$, since f_{MoM} does not have any non-zero root, we estimate γ_2 by setting $\gamma_{2,\text{MoM}} = \gamma_{2,\text{MLE}} = 0$. In this case, when $\gamma_{2,\text{MoM}} = 0$ or $\gamma_{2,\text{MLE}} = 0$, both γ_1 's estimators drop down to $\gamma_{1,\text{MoM}} = \gamma_{1,\text{MLE}} = \frac{M_1}{a}$.

This implementation choice is motivated by the following analysis.

1. We remind that $\text{INR} = b\gamma_2$ which, using the definition of b (provided below (3.2)), writes $\text{INR} = P_2 \frac{\gamma_2}{P_n}$. Thus, when the INR goes to 0, since P_2 and P_n are constant, γ_2 goes to 0. In this context, the interference is dominated by the noise power, i.e. $\gamma_2 \ll P_n$. Thus, it is very likely that γ_2 's estimation is inaccurate (and thus γ_1 as well

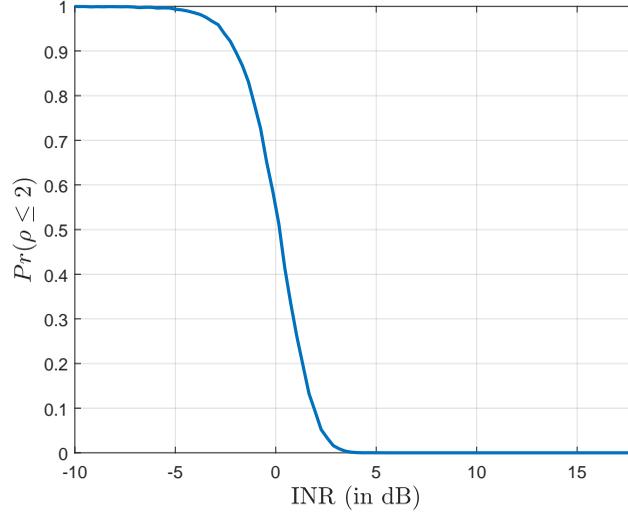


Figure 3.7: $\Pr(\rho \leq 2)$, versus INR for $N_S = 1000$ samples and SNR=10 dB.

due to the sequential estimation), hence the proposal to set the MoM and the MLE estimators as $\gamma_{2,MoM}, \gamma_{2,MLE} = 0$.

2. In order to apply the proposed estimators, one needs to know when the interference is dominated by the noise power. We have observed through extensive simulations that the probability of having $\rho \leq 2$ is a function of the INR, as illustrated in Fig. 3.7 for $N_S = 1000$ samples and SNR=10 dB. We can see that at low INR the value of $\Pr(\rho \leq 2)$ is close to 1 (and thus it is likely that f_{MoM} does not have any non-zero root), whereas at high INR, the value of $\Pr(\rho \leq 2)$ is close to 0 (and thus it is likely that f_{MoM} has a non-zero root). Hence the decision to base the proposed estimators depending on ρ 's value.

As we will see in Section 3.2.4, the proposed estimators implementation design is validated by the performance evaluation results achieving a lower variance than the CRLB.

In the next section, we propose a numerical procedure to determine (3.14)'s non-zero root when $\rho > 2$.

3.2.2.3.2 Root-finding algorithm implementation

We focus here on the case $\rho > 2$, and our purpose is to find the unique strictly positive solution of (3.13a). Since we were not able to provide an analytical solution for (3.13a), we propose an iterative algorithm to find it. The proposed algorithm is provided in Algorithm 3.2. We first assess whether we have $\rho > 2$ to assess if f_{MoM} admits a unique non-zero root. If yes, we propose to find it using a combination of the bisection method during its early iterations to properly initialize Newton's method, which fastens the

convergence.

The initialization of Newton's algorithm must be done carefully and close enough to the root $z = z_1$ to avoid the algorithm's divergence. We therefore propose in the first step to find a proper initialization for Newton's method using the bisection method, with the value smaller than the root $z_g < z_1$ verifying $f_{\text{MoM}}(z_g) > 0$ and the value greater than the root $z_d > z_1$ verifying $f_{\text{MoM}}(z_d) < 0$. One way to determine the first value of one of the bounds of the bisection is to set a value $z_0 = 1$ and then apply the following rule: if $f_{\text{MoM}}(z_0) < 0$, then we set $z_d = z_0$ and we determine z_g by successively dividing z_g by two until we find $f_{\text{MoM}}(z_g) > 0$. Otherwise, if $f_{\text{MoM}}(z_0) > 0$, then we set $z_g = z_0$ and we determine z_d by successively multiplying z_d by two until we find $f_{\text{MoM}}(z_d) < 0$. Once the initial values are found, we apply the bisection method iteratively until the root found is accurate enough to start Newton's algorithm. To this end, we propose to initialize Newton's method in the decreasing part of f_{MoM} . Moreover, we compute the first Newton's iterates if initialized by z_g and z_d , which can respectively be expressed as:

$$z^{(1)} = z - \frac{f_{\text{MoM}}(z)}{f'_{\text{MoM}}(z)}, \quad z = z_g, z_d, \quad (3.17)$$

are close enough. Therefore, we propose the following two conditions as stopping criteria for the bisection: 1) $f'_{\text{MoM}}(z_g) < 0$, and 2) the absolute difference between the first Newton's iterates initialized by z_g and z_d is less than a given threshold, which can be expressed as:

$$\left| z_g - \frac{f_{\text{MoM}}(z_g)}{f'_{\text{MoM}}(z_g)} - z_d + \frac{f_{\text{MoM}}(z_d)}{f'_{\text{MoM}}(z_d)} \right| < \varepsilon. \quad (3.18)$$

An illustration of f_{MoM} with high INR when the first and second conditions are not respected in Figs. 3.8a and 3.8b, respectively, and when both conditions are respected in Fig. 3.8c. When both conditions are verified, then Newton's algorithm is initialized to the value $(z_g + z_d)/2$.

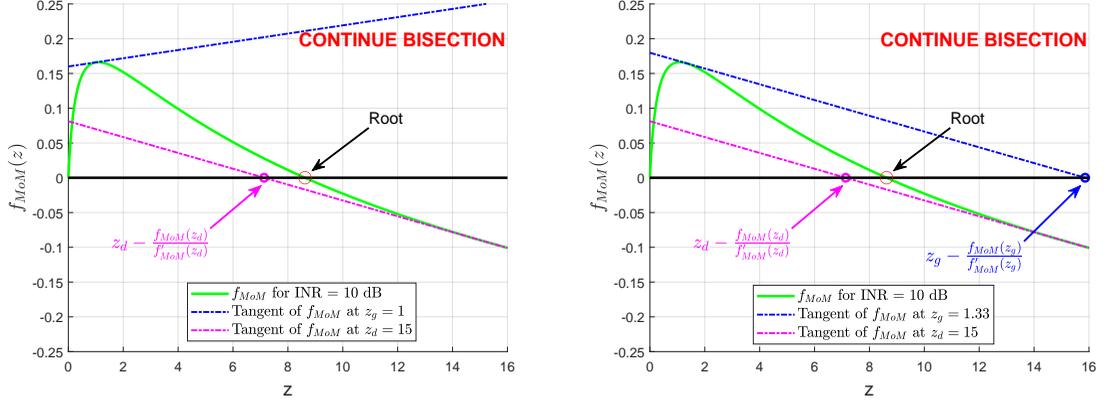
Although both MoM and MLE are obtained using iterative procedures, we provide a complexity analysis of these two estimators in Section 3.2.2.4 that shows that the complexity of MoM is much lower than the one of MLE.

3.2.2.4 Complexity analysis

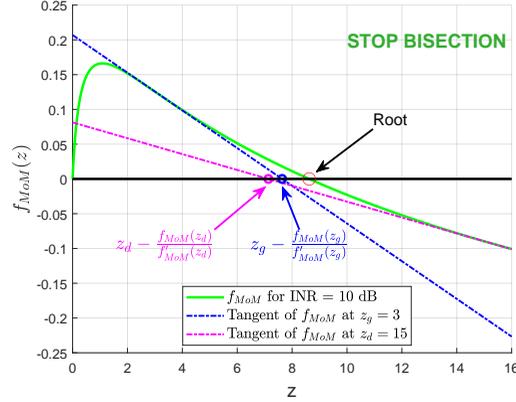
We study the complexity of the proposed estimators in terms of the number of elementary operations. Since the complexity of the estimators is dominated by the number of samples N_S , we formulate the complexity expression as a linear function $\alpha N_S + \beta$ where α and β are numerical values to compute.

Let us define N_{E_1} as the number of elementary operations to compute the exponential integral function $E_1(z)$. Since the complexity of the MoM is dominated by the computation of M_1 and M_2 , its complexity is given by

$$C^{\text{MoM}} N_S + O(N_{E_1}), \quad (3.19)$$



(a) Case $z_g = 1$ and $z_d = 15$ where $f'_{MoM}(z_g) < 0$. (b) Case $z_g = 1.33$ and $z_d = 15$ where the first Newton's iterates are too far.



(c) Case $z_g = 3$ and $z_d = 15$, the stopping criteria conditions are met.

Figure 3.8: Representation of f_{MoM} for INR = 10 dB for cases where the bisection conditions are met or not.

where $C^{MoM}N_S$ is the number of elementary operations to compute M_1 and M_2 and C^{MoM} is equal to 3. The complexity of the MLE is also linear with N_S , but is higher than the one of the MoM because Newton's method in the MLE requires to compute at each Newton's step the Hessian matrix whose components are computationally much more expensive than M_1 and M_2 . Let us define I_{rand} as the number of initializations performed by random search in the MLE initialized by trial-and-error search and I_{Ntn}^L as the number of iterations of the Newton's method procedure. The MLE initialized by trial-and-error search complexity is given by

$$C_{TE}^L N_S + \mathcal{O}(I_{rand} + I_{Ntn}^L), \quad (3.20)$$

where C_{TE}^L is the complexity term that depends on I_{rand} and I_{Ntn}^L . C_{TE}^L 's numerical value is provided in Table 3.1. For the MLE initialized with MoM estimates, it is worth mentioning

Algorithm 3.2: Root-finding algorithm for solving (3.13a).

```

Set  $\rho, \epsilon, \epsilon_{f'_{\text{MoM}}}, b$ .
if  $\rho \leq 2$  then
    |  $\gamma_{2,\text{MoM}} = 0$ 
else
    Initialize bisection method  $z_0 = 1$ .
    if  $f(z_0) < 0$  then
        |  $z_g, z_d = z_0$  and iterate  $z_g = z_g/2$  until  $f_{\text{MoM}}(z_g) > 0$ 
    else
        |  $z_g, z_d = z_0$  and iterate  $z_d = 2z_d$  until  $f(z_d) < 0$ 
    while  $|(z_g - f_{\text{MoM}}(z_g)/f'_{\text{MoM}}(z_g)) - (z_d - f_{\text{MoM}}(z_d)/f'_{\text{MoM}}(z_d))| < \epsilon$  and
         $f'_{\text{MoM}}(z_g), f'_{\text{MoM}}(z_d) < 0$  do
            | Apply the bisection method on  $f_{\text{MoM}}$  between  $z_g$  and  $z_d$ 
        end
    Initialize Newton's method with  $z^{(0)} = (z_g + z_d)/2$ .
    while  $|f'_{\text{MoM}}(z^{(n)})| < \epsilon_{f'_{\text{MoM}}}$  do
        |  $z^{(n+1)} = z^{(n)} - f_{\text{MoM}}(z^{(n)})/f'_{\text{MoM}}(z^{(n)})$ 
    end
    Set  $\gamma_{2,\text{MoM}} = z^{(n+1)}$ .
Return  $\gamma_{2,\text{MoM}}$ 
    
```

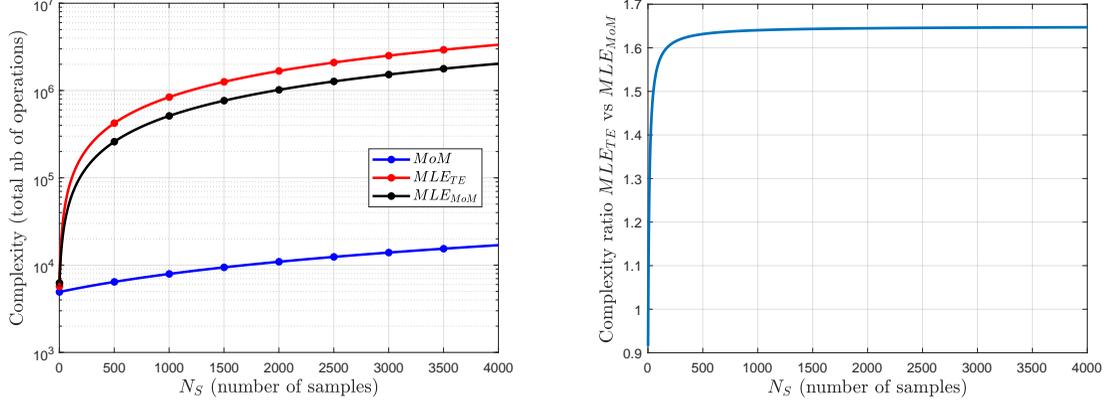
that I_{rand} is significantly lower than without MoM estimates since the trial-and-error search is only performed when the MoM estimate initialization provides a non-negative Hessian matrix. The MLE initialized with MoM estimates complexity is given by

$$C_{\text{MoM}}^L N_S + \mathcal{O}(I_{\text{rand}} + I_{\text{Ntn}}^L + N_{E_1}), \quad (3.21)$$

where C_{MoM}^L is the complexity term which depends on I_{rand} and I_{Ntn}^L . It is worth mentioning that the MLE initialized with MoM estimates complexity is computed by including the cases where MoM initializations fail and then trial-and-error initialization is performed, instead of only considering the cases with successful MoM initializations. Its numerical value is provided in Table 3.1. The numerical values of the terms involved in the complexity expressions are provided in Table 3.1, where each value is averaged through 10^6 Monte-Carlo simulations. We have observed through numerical simulations that the terms N_{E_1} , I_{rand} and I_{Ntn}^L do not depend on the number of samples N_S .

Table 3.1: Estimators' complexity (number of elementary operations)

	$C_{\text{MoM}}^{\text{MoM}}$	C_{TE}^L	C_{MoM}^L	N_{E_1}	I_{rand}	I_{Ntn}^L
MoM	3	-	-	333	-	-
MLE initialized by TE	-	835	-	-	143.74	7.83
MLE with MoM	-	-	506	-	1.01	5.70



(a) Estimators complexity in terms of number of elementary operations

(b) Ratio between the complexity of the MLE initialized with trial and error over the one initialized with MoM..

Figure 3.9: Estimators complexity comparison of the proposed estimators.

Fig. 3.9a represents the complexity in terms of the number of elementary operations, of the proposed estimators, versus the number of samples N_S . The complexity of the MoM estimator is much lower, i.e. more than 100 times less complex for $N_S \geq 100$ samples, than the MLE when it is initialized by trial-and-error or with MoM estimates. Moreover, Fig. 3.9b shows that the complexity of the MLE initialized with MoM estimates is 1.6 lower than the MLE initialized with trial-and-error for $N_S \geq 200$ samples. This result highlights the interest in initializing with MoM estimates to decrease the MLE implementation complexity.

3.2.3 Cramer-Rao Lower Bound

In this section, we derive the CRLB related to the estimation of the parameters $\gamma = (\gamma_1, \gamma_2)$ using SINR samples $\mathbf{S} = (S_1, \dots, S_{N_S})$. The CRLB is a lower bound on the variance of any unbiased estimator [64, Chapter 3]. In practice, it indicates if an estimator is optimal in terms of variance.

The CRLB is defined as

$$\text{CRLB}(\gamma) = \mathbf{I}^{-1}(\gamma), \quad (3.22)$$

where $\mathbf{I}(\gamma)$ is the Fischer information whose (i, j) -th entry can be expressed as

$$[\mathbf{I}(\gamma)]_{ij} := -\mathbb{E} \left[\frac{\partial^2 \ell_{\mathbf{S}}(\gamma)}{\partial \gamma_i \partial \gamma_j} \right], i, j = 1, 2, \quad (3.23)$$

where $\ell(\mathbf{S}, \gamma)$ is the joint log-likelihood defined in (3.5) and whose second-order partial derivatives expression are provided in (B.6).

After some derivations detailed in Appendix B.1.3.1, we obtain the CRLB provided in the following result.

Result 3.5. The CRLB related to the estimation of γ can be written as follows

$$\text{CRLB}(\gamma) = \frac{1}{N_S \theta} \begin{pmatrix} -\mathcal{I}_2 & \mathcal{I}_3 \\ \mathcal{I}_3 & -(A_1 + A_2 + \mathcal{I}_1) \end{pmatrix} \quad (3.24)$$

where $\theta = \mathcal{I}_2(A_1 + A_2 + \mathcal{I}_1) - (\mathcal{I}_3)^2$ with

$$\begin{aligned} A_1 &= -\frac{2\omega}{\gamma_1^2} e^\omega E_1(\omega), \\ A_2 &= \frac{2}{3\gamma_1^2} (\omega^3 e^\omega E_1(\omega) - \omega^2 + \omega + 1), \end{aligned} \quad (3.25)$$

and

$$\begin{aligned} \mathcal{I}_1 &= -\frac{\omega(\omega+1)^2}{\gamma_1^2} \int_0^{+\infty} \frac{e^{-t}}{(\omega+t)^2(\omega+t+1)} dt, \\ \mathcal{I}_2 &= b^2 \omega^3 \int_0^{+\infty} \frac{2t^2(\omega+t+1)^2 - (t+1)^2(\omega+t)^2}{(\omega+t)^4(\omega+t+1)} e^{-t} dt, \\ \mathcal{I}_3 &= \frac{b\omega^3}{\gamma_1} \int_0^{+\infty} \frac{(\omega+t+1)^2 + 2t + 2\omega + 1}{(\omega+t)^4(\omega+t+1)} t e^{-t} dt. \end{aligned} \quad (3.26)$$

The generalized exponential integral function involved in A_1 and A_2 can be easily computed using for instance [115]. The integrals involved in \mathcal{I}_1 , \mathcal{I}_2 and \mathcal{I}_3 may present some integrable singularities and thus require using numerical methods. Integral approximation methods based on Gaussian quadrature are provided in Appendix B.1.3.2 to implement CRLB's terms.

3.2.4 Numerical results

3.2.4.1 MLE-MoM estimator performance

In this section, we study the performance of the proposed MLE (initialized with the MoM) and of the MoM estimators in terms of both normalized bias magnitude and normalized MSE (NMSE), defined for a given estimator $\hat{\gamma}_j$ as:

$$\text{bias} = \mathbb{E} \left[\left| \frac{\hat{\gamma}_j - \gamma_j}{\gamma_j} \right| \right], \quad (3.27)$$

$$\text{NMSE} = \mathbb{E} \left[\left(\frac{\hat{\gamma}_j - \gamma_j}{\gamma_j} \right)^2 \right], \quad (3.28)$$

respectively, for $j = 1, 2$. In the following, for clarity, the normalized bias magnitude is referred to as the *bias magnitude*. We also study the performance of these estimators by computing the mean integrated squared error (MISE) which is defined as $\text{MISE} := \mathbb{E} \int |p(x) - \hat{p}(x)|^2 dx$, with $p := x \mapsto p_S(x; \gamma)$ and $\hat{p} := x \mapsto p_S(x; \hat{\gamma})$, where γ are the true parameters and $\hat{\gamma}$ a given estimator. We study the influence of the number of samples N_S and each result is averaged over 10^6 Monte-Carlo trials. In the following figures, the CRLB has been normalized by γ_j^2 .

The parameters are set as follows: $P_1 = P_2 = P_n = 1$ W, leading to $a = b = 1$, and $N_{init,MLE} = 10^6$. In the **MLE** initialized with the **MoM** implementation, we remind that if the **MoM** estimates do not provide a negative Hessian matrix, the trial-and-search method is performed to find an adequate initialization. In that case, the region set $\Theta_{\gamma_1, \gamma_2}$ is set such that $\gamma_{1,\min} = 0.1$, $\gamma_{1,\max} = 100$, $\gamma_{2,\min} = 0.1$ and $\gamma_{2,\max} = 100$, i.e. such that the **SNR** and **INR** are between -10 and 20 dB. In root-finding Algorithm 3.2, we set $\epsilon = 0.1$ and $\epsilon_{f'} = 10^{-8}$.

3.2.4.1.1 Influence of the number of samples N_S

In this section, we set $\gamma_1 = 100$ to obtain **SNR** = 20 dB and $\gamma_2 = 10$ to have **INR** = 10 dB. Fig. 3.10a and 3.10b represent the bias magnitude and **NMSE** of the proposed estimators, respectively, versus the number of samples N_S which varies between 50 and 4000 . Fig. 3.10b also shows the **CRLB** whose implementation is detailed in Section 3.2.3. We can draw the following observations: i) the bias magnitude of the **MLE** and the **MoM** estimators converge to zero as N_S increases, ii) the **NMSE** of the proposed estimators decrease with increasing N_S , iii) as expected, the **MLE** is optimal in terms of **NMSE** since it achieves the **CRLB** whereas the **NMSE** of the **MoM** is higher. Moreover, we can observe that while the **MLE** bias magnitude is similar to the **MoM** estimator's for γ_1 estimation, it is surprisingly a bit higher for γ_2 estimation.

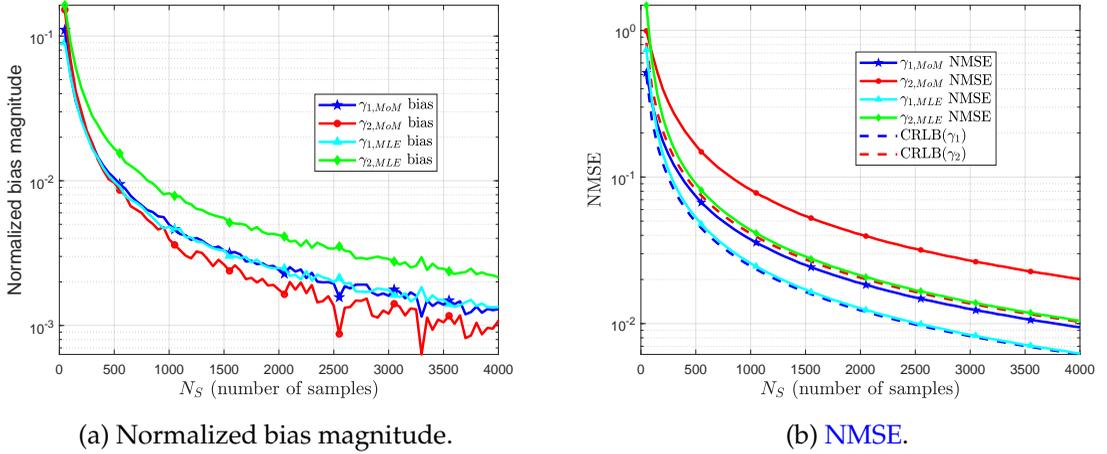


Figure 3.10: Performance of $\gamma_{1,MoM}$, $\gamma_{2,MoM}$ and $\gamma_{1,MLE}$, $\gamma_{2,MLE}$ and corresponding **CRLB** versus N_S , for **SNR** = 20 dB and **INR** = 10 dB. The biases of the **MLE** and the **MoM** estimators converge to zero as N_S increases. The **MLE** is close to optimal in terms of **NMSE** since it achieves the **CRLB**.

Fig. 3.11 represents the **MISE** between the **pdf** of the **SINR** using the true parameters γ , and the **pdf** of the **SINR** using the parameters γ_{MLE} and γ_{MoM} estimated using the **MLE** and the **MoM**, respectively. The **MISE** of both the **MLE** and the **MoM** estimators

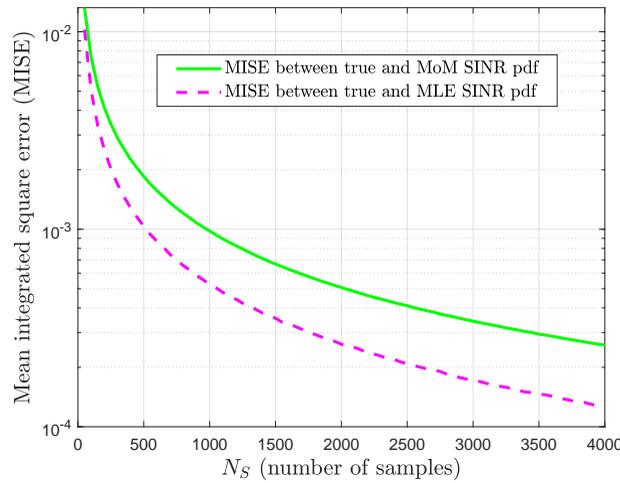


Figure 3.11: MISE versus N_S for MLE and MoM estimators, for SNR = 20 dB and INR = 10 dB.

decreases as N_S increases (both of them are below 10^{-2} for $N_S > 100$ samples). Besides, the MLE outperforms the MoM estimation in terms of MISE, which was expected thanks to the MLE optimality in terms of NMSE already observed in Fig. 3.10b.

3.2.4.1.2 Influence of SNR

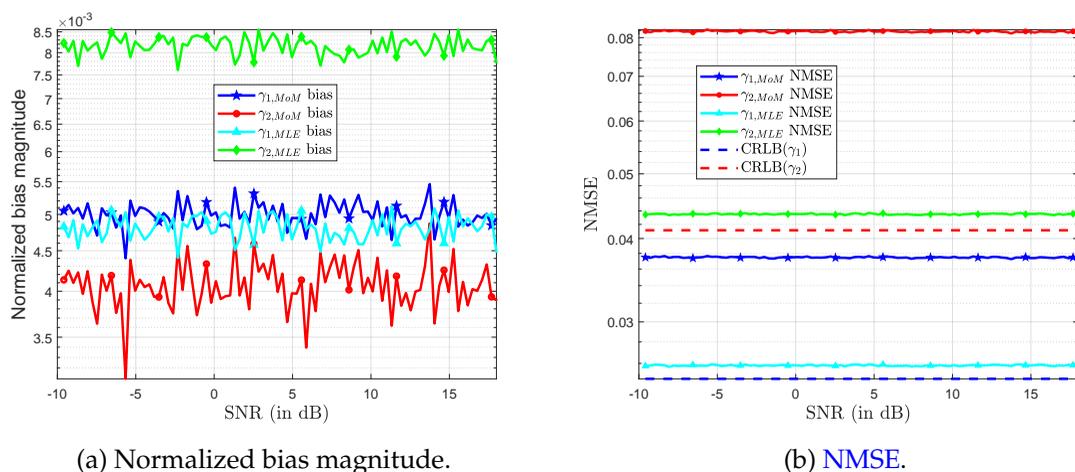


Figure 3.12: Performance of $\gamma_{1,MoM}$, $\gamma_{2,MoM}$ and $\gamma_{1,MLE}$, $\gamma_{2,MLE}$ and corresponding CRLB versus SNR, for $N_S = 1000$ samples and INR = 10 dB. The bias magnitudes and the NMSE of the MLE and the MoM estimators are steady for various SNRs.

In this section, we set $N_S = 1000$ samples and $\gamma_2 = 10$ to have INR = 10 dB. Fig. 3.12a represents the bias magnitude of both estimators for the two parameters γ_1, γ_2 , versus

the SNR which varies between -10 and 18 dB. Fig. 3.12b represents their NMSE and the CRLB, versus the SNR between -10 and 18 dB. We remind that the SNR expression is defined in Section 3.2.1 and that it is proportional to γ_1 . From these figures, we can observe that both the bias magnitude and the NMSE of the MLE and the MoM estimators are steady for various SNR. The NMSE of the MLE is lower than the MoM estimators'. Furthermore, the bias magnitude of both estimators is similar for γ_1 estimation whereas the MLE's is lower than the MoM estimator's for γ_2 estimation. Fig. 3.13 represents the

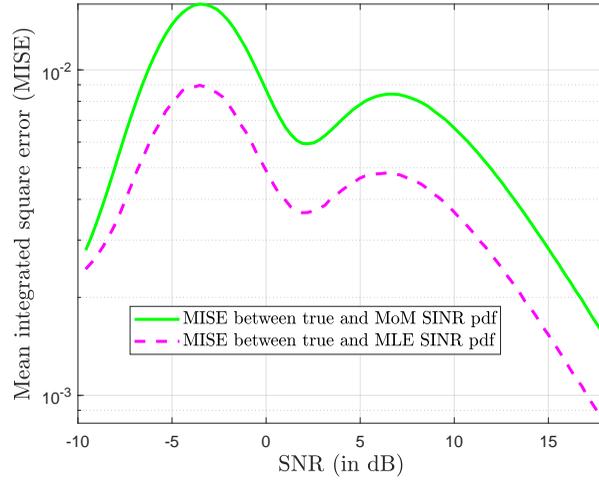


Figure 3.13: MISE versus SNR for MLE and MoM estimators, for $N_S = 1000$ samples and INR = 10 dB.

MISE between the pdf of the SINR using the true parameters γ , and the pdf of the SINR using γ_{MLE} and γ_{MoM} , respectively, for various SNR. As in Section 3.2.4.1.1, the MLE outperforms MoM estimation in terms of the MISE. While the bias magnitude and NMSE of both estimators for various SNR values, the MISE of both estimators has multiple increasing and decreasing variations for increasing values of SNR, which we were not able to analyze.

3.2.4.1.3 Influence of INR

In this section, we set $N_S = 1000$ samples and $\gamma_1 = 100$ to have SNR = 20 dB. Fig. 3.14a and 3.14b represent the bias magnitude and the NMSE of the estimators, respectively, versus the INR, with the CRLB illustrated in Fig. 3.14b. The INR varies between -5 and 18 dB in Fig. 3.14a and between -10 and 18 dB in Fig. 3.14b. We remind that the INR expression is defined in Section 3.2.1 and that it is proportional to γ_2 . For γ_2 estimation, when INR increases, the bias magnitudes of the MLE and the MoM estimator decrease for respectively INR between -10 and -7 , and between -10 and -2 . The case INR = -2 dB corresponds to $\gamma_2 \approx 0.63$, which is less than the noise power $P_n = 1$. This is due to the decreasing influence of γ_2 when facing a stronger noise power which makes

its estimation more difficult. We can observe that for γ_1 estimation, for increasing INR, the bias magnitude of the MLE decreases for INR between -5 and 3 dB, then increases for INR between 3 and 18 dB. For the MoM, the bias magnitude decreases for INR between -5 and 5 dB, then increases for INR between 5 and 18 dB. We remind that for lower values of the INR, we set $\gamma_{2,MLE} = 0$ and $\gamma_{2,MoM} = 0$ when no solution is found, some irregularities can appear in the bias magnitude which is illustrated by the presence of "peaks" for INR between -10 and 0 dB, which corresponds to the cases where γ_1 is lower than the noise power. This can also explain the peak for γ_2 estimation for both estimators for INR between -10 and -5 dB since γ_2 estimate depends on γ_1 estimate.

Concerning the NMSE of the MLE and the MoM estimator for INR < 5 dB which corresponds to $\gamma_2 \lesssim 3.16$.

Furthermore, we can observe in Fig. 3.14b that the NMSE of both estimators decreases for increasing INR between -10 and 5 dB for γ_1 estimation, which is due to a more difficult estimation of γ_1 for low INR. For γ_1 estimation, the NMSE of both estimators also decreases between -4 and 5 dB since γ_1 estimate depends on γ_2 . For INR between -10 and -3 dB, the NMSE of both γ_1 's estimators are similar, which is explained by the fact that for lower INR values, we set $\gamma_{2,MLE} = 0$ and $\gamma_{2,MoM} = 0$ when $\rho \leq 2$. Moreover, we observe that both the bias magnitude and the NMSE of both estimators increase for increasing values of the INR when INR > 10 dB.

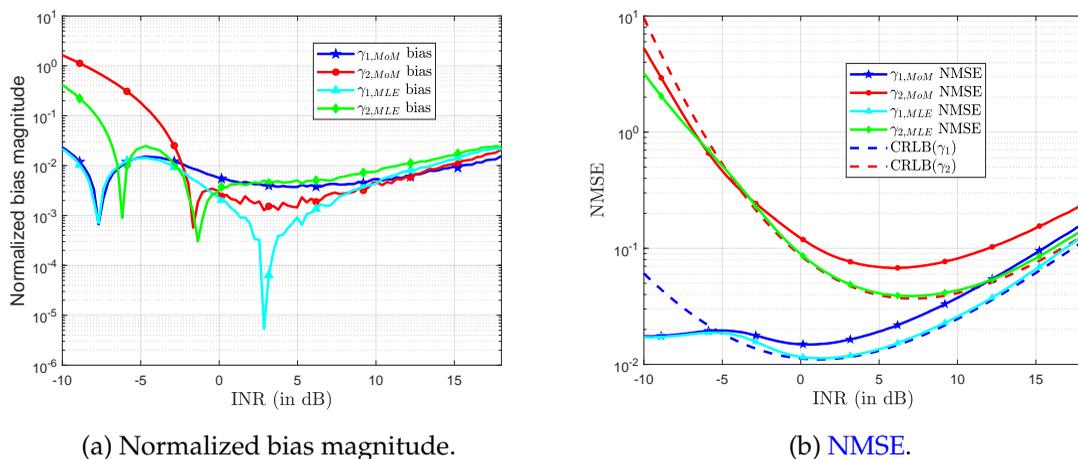


Figure 3.14: Performance of $\gamma_{1,MoM}$, $\gamma_{2,MoM}$ and $\gamma_{1,MLE}$, $\gamma_{2,MLE}$ and corresponding CRLB versus INR, for $N_S = 1000$ samples and SNR = 20 dB. The biases and the NMSE of the MLE and the MoM estimators decrease, and then increase.

Fig. 3.15 represents the MISE between the pdf of the SINR using the true parameters γ , and the pdf of the SINR using γ_{MLE} and γ_{MoM} , respectively, for various INR. We observe that MLE outperforms MoM estimation in terms of estimating the SINR pdf, which was expected thanks to the MLE lower NMSE already observed in Fig. 3.14b and that for

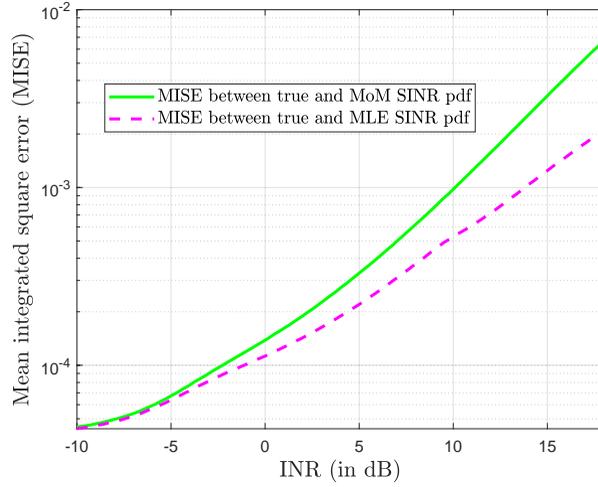


Figure 3.15: **MISE** versus N_S for **MLE** and **MoM** estimators, for SNR = 20 dB and INR = 10 dB.

decreasing values of the **INR**, **MLE** and **MoM** estimation provide similar performance in terms of **MISE**.

3.2.4.1.4 Analysis of the **CRLB** in Fig. 3.14b

In Fig. 3.14b, we can observe that for **INR** values between -10 and -3 dB, the **NMSE** of both γ_1 and γ_2 estimators become lower than the **CRLB**, which may look surprising. This can be explained as follows.

On the one hand, as discussed in Section 3.2.2.3, estimating γ_2 is difficult when the **INR** value is low, yielding a difficult estimation of γ_1 as well, which can be observed in Fig. 3.14b since the **CRLB** of both estimation variables increases for decreasing **INR** values, when $\text{INR} \leq 0$ dB.

On the other hand, the proposed estimators in Result 3.4 set γ_2 to 0 at low **INR** according to the distribution in Fig. 3.7. As a consequence, one can explain the fact that we achieve better performance than the **CRLB** by the following reasons: i) The implemented estimators are biased when $\rho \leq 2$, therefore the **CRLB** in Fig 3.14b does not apply to these estimators. ii) Setting $\gamma_2 = 0$ for low **INR** prevents from having noisy estimates of γ_2 , and consequently of γ_1 , leading to lower **NMSE** than the **CRLB**. This validates the proposed estimators design.

3.2.5 Conclusion of Section 3.2

To summarize our observations, we have provided a scheduling method to estimate the channel statistics of all pairs of nodes of a network. We have shown that this scheduling solution which allows two simultaneous transmissions outperforms **TDMA** in terms of scheduling length. Moreover, to deal with the resulting interference, we have formulated

an equivalent estimation problem for channel statistics at each receiver in a frequency-flat Rayleigh channel in the presence of a single interferer using a set of SINR samples. This problem consists of estimating for each receiver the direct and interfering channels' magnitude. To that purpose, we have introduced two new estimators based on MLE and MoM which both have the property of being consistent. We have implemented them using Newton's method and we have provided initialization procedures to ensure the convergence of both algorithms. We have evaluated the performance of these estimators through simulations and found that the variance of the MLE is close to the CRLB, while the variance of the MoM is slightly higher. However, the MoM estimator has the advantage of being less complex than the MLE. Furthermore, the MLE can be initialized with the MoM's estimates, which reduces its complexity compared to random initialization.

3.3 Scheduling optimization with simultaneous transmissions for channel statistics estimation

In this section, we consider a second scheduling approach with simultaneous transmissions to decrease the sensing duration for channel statistics estimation compared to TDMA. This scheduling solution differs from the previous one described in Section 3.2 for the following reasons:

- More than two nodes are allowed to transmit simultaneously at each step.
- The receivers have access to additional information beyond just SINR samples to estimate the channel statistics, such as measurements of the received signal magnitude or estimated channel samples that could help in the estimation process. For example, the channel statistics can be estimated by taking the variance of the estimated samples of the channel which can be obtained with channel estimation techniques such as LS or MMSE methods.

Since simultaneous transmissions result in the presence of interference from multiple transmitters, we propose to use orthogonal sequences to remove them before applying channel statistics estimation techniques. Hence, the remaining challenging issue is to find a suitable transmission scheduling with the following requirements:

- It asserts the estimation of the channel statistics of all pairs of links of the network. Although this verification is immediate when using TDMA, it is not so simple with a scheduling method allowing simultaneous transmissions in TDD mode.
- It performs better than TDMA in terms of the number of steps.

Therefore, we focus on deriving optimized scheduling solutions with the requirements we have just mentioned.

This section is organized as follows. In Section 3.3.1, we describe the system model and the criteria to choose the required orthogonal codes. In Section 3.3.2, we provide the proposed scheduling solutions for the channel statistics estimation and show that it outperforms TDMA in terms of scheduling length.

3.3.1 System model and orthogonal sequences

3.3.1.1 System model

We consider a similar system model as in Section 3.2, i.e. a network composed of N nodes in TDD mode with symmetric channel statistics $\gamma_{ij} = \gamma_{ji}$. We assume that a central entity (CE) computes the scheduling which is then transmitted to the individual nodes in the network. Moreover, we remind that the proposed scheduling solution allows simultaneous transmissions, which generates interference between nodes. To remove them, we propose to use orthogonal sequences which means in other words that at each step t , every transmitting node j sends a sequence $\mathbf{x}_j(t) \in \mathbb{R}^M$ to a receiving node k with the following conditions:

$$\begin{aligned} \mathbf{x}_k(t)^T \mathbf{x}_j(t) &= 1, & \text{if } j = k, \\ \mathbf{x}_k(t)^T \mathbf{x}_j(t) &= 0, & \text{otherwise.} \end{aligned} \quad (3.29)$$

The temporal dependency of the sequence $\mathbf{x}_j(t)$ indicates that different sequences can be used at each step. Indeed, the transmitting would rather send shorter sequences to decrease the sensing duration as long as all interference is removed.

Let j_1, \dots, j_m be the nodes that transmit at step t where each transmitting node j sends a sequence $\mathbf{x}_j(t) \in \mathbb{R}^M$. In this case, the received signal at node $k \notin \{j_1, \dots, j_m\}$ at step t can be written as:

$$\mathbf{y}_k(t) = \sum_{j \in \{j_1, \dots, j_m\}} \sqrt{P_j} h_{kj}(t) \mathbf{x}_j(t) + \mathbf{n}_k(t), \quad (3.30)$$

where P_j is the transmit power of node j , $h_{kj} \sim CN(0, \gamma_{kj})$ are the Rayleigh flat fading coefficients from transmitter j to receiver k , γ_{kj} is the pathloss values and corresponds to the channel statistics from j to k , $\mathbf{n}_k(t) \in \mathbb{R}^M$ is the AWGN at receiver k at step t where we have for each of its elements $n_{ki}(t) \sim CN(0, P_n)$ with P_n the noise power. We assume that $P_j, j = 1, \dots, N$, and P_n are perfectly known.

From the received signal expressed in (3.30), node k recovers the signal from node j by removing the interference which can be written as:

$$\mathbf{x}_j(t)^T \mathbf{y}_k(t) = \sqrt{P_j} h_{kj}(t) + \mathbf{x}_j(t)^T \mathbf{n}_k(t), \quad (3.31)$$

where $\mathbf{x}_j(t)^T \mathbf{n}_k(t)$ follows a complex Gaussian distribution by linearity of the Gaussian distribution.

When considering a transmission from node j to node k , it is necessary for node k to receive multiple packets from the sending node j to estimate the propagation channel

$h_{kj}(t)$ reliably enough. The channel samples can be estimated using channel estimation techniques on (3.31) such as LS or MMSE estimators [101]. Then, after repeating this operation n_{stat} times to get n_{stat} channel coefficients, an estimator of the channel statistics using the collected samples can be computed to obtain γ_{kj} .

In the following, we provide orthogonal sequences to remove the interference in our context.

3.3.1.2 Choice of the orthogonal sequences

One common choice for orthogonal codes in wireless communications is to use the Walsh-Hadamard codes. These codes can be implemented using N codes of size $N \times 1$ which follows the conditions in (3.29) [37]. However, this method has a limitation as it can only create N codes if N is a power of two. If N is not a power of two, this method may require a sequence size that is much larger than N to respect (3.29), which increases the sensing duration. An alternative approach is to use Zadoff-Chu sequences which have the useful property of having a null auto-correlation with its shifted version [30]. The rows of the following matrix can be used to create the orthogonal sequences for Zadoff-Chu sequences:

$$\mathbf{X}_{ZC} = \begin{pmatrix} x_u(1) & x_u(2) & \dots & x_u(N_{ZC}) \\ x_u(N_{ZC}) & x_u(1) & \dots & x_u(N_{ZC} - 1) \\ \vdots & \vdots & \ddots & \vdots \\ x_u(2) & \dots & x_u(N_{ZC}) & x_u(1) \end{pmatrix}, \quad (3.32)$$

with $x_u(n)$ expressed as:

$$x_u(n) = \frac{1}{\sqrt{N_{ZC}}} \exp\left(-j \frac{\pi u n(n+1)}{N_{ZC}}\right), \quad (3.33)$$

with N_{ZC} the length of the sequence, $0 < n \leq N_{ZC}$, $0 < u < N_{ZC}$ and $\text{gcd}(N_{ZC}, u) = 1$ with gcd standing for greatest common divisor operation.

In practice, each node receives the transmission scheduling that we describe later. Therefore, the number of transmissions at each step $m(t)$ is known and the Zadoff-Chu sequence can be computed at each step t by setting $N_{ZC} = m(t)$. It is worth mentioning that during the orthogonal sequences study, we did not take into account the synchronization issues.

In what follows, we focus on scheduling methods to estimate the channel statistics.

3.3.2 Scheduling methods for channel statistics estimation

In this section, we focus on scheduling methods using simultaneous transmissions to decrease the sensing duration of the channel statistics estimation. Before giving the proposed solutions, we provide a few definitions and assumptions and then, formulate the combinatorial optimization problem to solve.

3.3.2.1 Assumptions and definitions

Unlike the scheduling solutions described in Section 3.2, the scheduling methods proposed in this section can estimate the channel statistics of all pairs of links of the network which corresponds to all the γ_{ij} for $i, j = 1, \dots, N$. Because of the symmetry $\gamma_{ij} = \gamma_{ji}$, these scheduling techniques provide two measures of γ_{ij} estimation for $i, j = 1, \dots, N$. Therefore, to have a fair comparison, the scheduling solutions proposed in this section are compared with a scheduling method using TDMA which also estimates two measures of γ_{ij} . This comparison is different from the scheduling solutions in Section 3.2 since our proposed scheduling method could not estimate γ_{ij} for $i, j = 1, \dots, N$, and thus obtain two measures of γ_{ij} . It is worth mentioning that the proposed scheduling schemes described in the following can also be used in fully distributed systems where each receiver knows the channel gain of the other nodes.

In the following, for clarity, we refer to this scheduling scheme estimating γ_{ij} twice based on TDMA as **TDMA**. The number of steps of such a **TDMA** solution is thus N steps. Moreover, the definition of *successful scheduling* provided in Section 3.2.1.1 is modified in this section and refers to a scheduling leading to the estimation of the channel statistics of all pairs of links of the network, instead of the pairs of nodes previously. Before describing the proposed scheduling solutions, we provide the following definition:

Definition 3.5. We define an *optimal scheduling* as a successful scheduling whose length is optimal, i.e. the length of the optimal scheduling is the minimum among all the successful schedulings.

We represent in Fig 3.16 an example of optimal scheduling computed by exhaustive search where the channel statistics of all pairs of links have been estimated in 5 steps under **TDD** assumption, which reduces the probing time needed to estimate all the channel statistics.

Even though it is possible to provide better solutions by optimizing the scheduling method which estimates either γ_{ij} or γ_{ji} , in the following, we propose scheduling techniques that estimate both γ_{ij} and γ_{ji} . We show that even though the proposed scheduling methods are suboptimal, they outperform **TDMA** in terms of scheduling length.

Furthermore, although assessing that **TDMA** is a successful scheduling is simple, it is not obvious in the case of multiple transmissions. Moreover, it can be shown that in **TDD** mode, the transmission scheduling optimization reduces to a combinatorial optimization problem. Therefore, in the next section, we formalize the combinatorial optimization problem and provide a practical visualization method to verify whether a scheduling is successful or not.

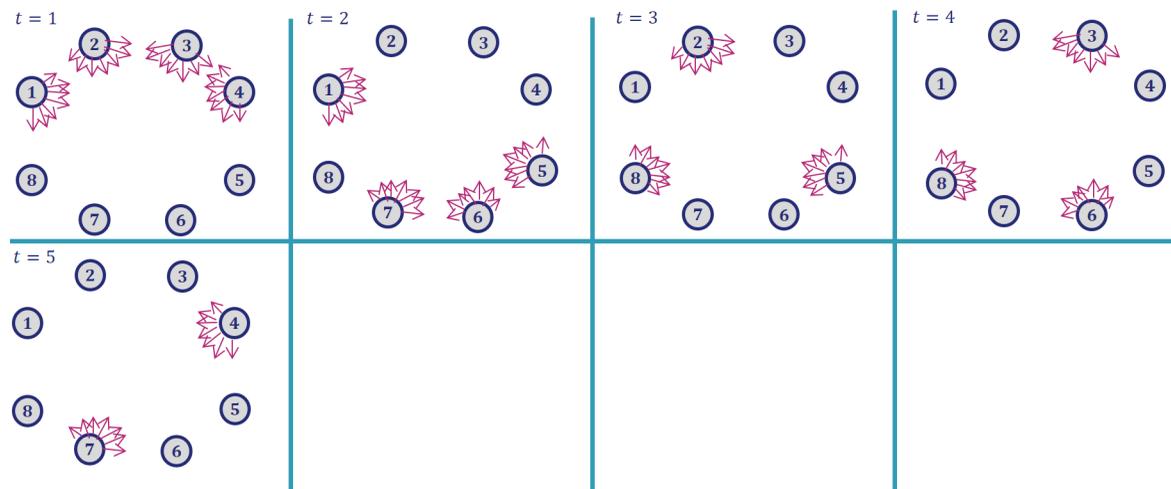


Figure 3.16: Example of an optimized scheduling requiring 5 steps applied to a network composed of $N = 8$ nodes.

3.3.2.2 Combinatorial optimization problem

3.3.2.2.1 Notations

Let $\mathcal{S} = (\alpha_1, \dots, \alpha_L)$ be a scheduling with $L := L_{\mathcal{S}}(N)$ and where $L_{\mathcal{S}}(N)$ is the length of the scheduling \mathcal{S} applied to a network composed of N nodes. Let $\alpha_t = (j_1, \dots, j_m)$ is the uplet composed of the index of the nodes j_1, \dots, j_m which transmit simultaneously at step t . For example, $\alpha_1 = (1, 3)$ means that at step $t = 1$, the nodes 1 and 3 transmit at the same time. Moreover, we represent the progress of the channel statistics estimation of a given scheduling with a matrix at each step t during a cycle. To that purpose, let $\mathbf{\Omega}(t) \in \{0, 1\}^{N \times N}$ be such a matrix where its (i, j) th element is denoted as $\Omega_{ij}(t)$ and is equal to 1 if the channel statistics γ_{ij} have been estimated at step t and to 0 otherwise. For each scheduling, this matrix is initialized as $\mathbf{\Omega}(0) = \mathbf{I}_N$, which is the identity matrix of size $N \times N$, since we do not estimate the $\gamma_{ii}, i = 1, \dots, N$.

Using these notations, when subjected to the transmissions $\alpha_t = (j_1, \dots, j_m)$ at step t , $\mathbf{\Omega}(t)$ updates as the following:

$$\forall j \in \{j_1, \dots, j_m\}, \quad \forall i \in \{1, \dots, N\} \setminus \{j_1, \dots, j_m\}, \quad \Omega_{ij}(t) = 1. \quad (3.34)$$

In other words, for such a transmission α_t , for each $j = j_1, \dots, j_m$, all the receivers i , with i different from j_1, \dots, j_m , have estimated γ_{ij} . For instance, for $N = 8$ and for a given scheduling with the transmissions $\alpha_1 = (1, 3)$, the progress matrix at step 1 is expressed

as follows:

$$\mathbf{\Omega}(1) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (3.35)$$

where the 1st and 3rd columns have been changed to 1, except for $\Omega_{13}(1)$ and $\Omega_{31}(1)$ because of the **TDD** assumption. With these notations, a scheduling \mathcal{S} of length L is successful if and only if we have $\mathbf{\Omega}(L) = \mathbf{1}_N$.

Let $\mathcal{S}_{\text{TDMA}}$ be the **TDMA** scheduling described above, which can be written as the following:

$$\mathcal{S}_{\text{TDMA}} = \{(1), (2), (3), (4), (5), (6), (7), (8)\}, \quad (3.36)$$

whose length is $L_{\mathcal{S}_{\text{TDMA}}}(8) = 8$. The optimal scheduling represented in Fig. 3.16 can be written as:

$$\mathcal{S}_{\text{opt}} = \{(1, 2, 3, 4), (1, 5, 6, 7), (2, 5, 8), (3, 6, 8), (4, 7)\}, \quad (3.37)$$

whose length is $L_{\mathcal{S}_{\text{opt}}}(8) = 5$.

Moreover, the scheduling optimization problem can be formulated as follows:

Problem 3.1.

$$\begin{aligned} & \arg \min_{\mathcal{S}} L_{\mathcal{S}}(N) \\ & s.t. \quad \mathbf{\Omega}(L_{\mathcal{S}}(N)) = \mathbf{1}_N. \end{aligned} \quad (3.38)$$

3.3.2.2 Graphical representation

In this section, we describe a visualization method that we refer to as a *grid of squares* to illustrate the combinatorial optimization problem and assess the success of a scheduling method. To that purpose, we model an $N \times N$ grid whose elements are squares that can be colored. This grid represents the progress of the channel statistics estimation of a scheduling at each step like the matrix $\mathbf{\Omega}(t)$. At step t , the (i, j) th square is white if $\Omega(t) = 0$, i.e. γ_{ij} has not been estimated during this cycle, and green otherwise. The diagonal of the grid is always black to illustrate that the values γ_{ii} do not need to be estimated. Like $\mathbf{\Omega}$'s initialization which is $\mathbf{\Omega}(0) = \mathbf{I}_N$, the grid is initialized with white squares only except for the diagonal in black. Then, during the scheduling, multiple transmissions occur, which colors the grid with green squares. When a transmission is only performed by node j , all the nodes $i \neq j$ are in receiver mode and estimate γ_{ij} , which then colors the j th column in green. In Fig. 3.17, when the nodes 1 and 3 transmit, the columns 1 and 3 become green except for the squares $(1, 3)$ and $(3, 1)$ because of the **TDD** mode assumption.

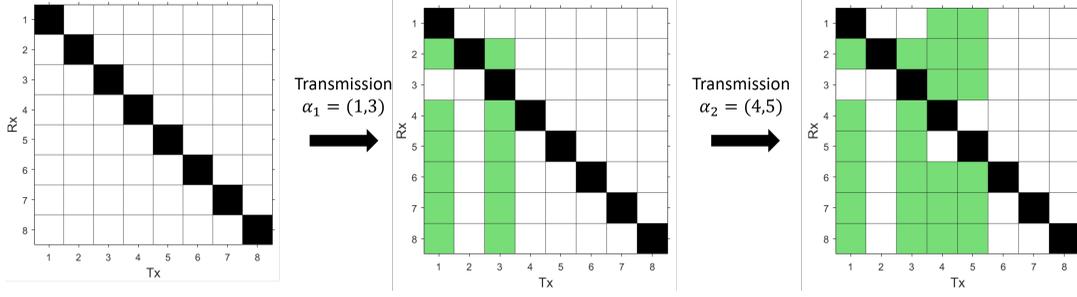


Figure 3.17: Representation with the grid of squares for $N = 8$ nodes of the scheduling $\mathcal{S} = \{\alpha_1, \alpha_2\}$ with $\alpha_1 = (1, 3)$ and $\alpha_2 = (4, 5)$. The channel statistic γ_{ij} has been estimated as the (i, j) th square is green. If not, the square is white.

3.3.2.2.3 Complexity of the combinatorial problem

In the scheduling problem, at every step t , each node has two possibilities of actions: to either transmit or receive. Hence, each step of a scheduling \mathcal{S} contains 2^N possibilities, and thus, for a given scheduling length L , the number of possible scheduling is 2^{LN} . For example, for $N = 8$ nodes, the number of possible scheduling in 5 steps is 2^{40} . The problem of finding the optimal scheduling is of great complexity, which makes it difficult to determine analytically the minimum number of steps for a scheduling to be successful.

We have only performed numerical simulations to find the optimal scheduling by exhaustive search up to $N = 8$ since for $N > 8$, the computation time is too high.

3.3.2.3 Proposed scheduling solution

In this section, we first propose two suboptimal scheduling solutions to Problem 3.1 and we theoretically prove that their length is lower than TDMA's. Then, we derive a third suboptimal scheduling solution which consists of combining those two scheduling strategies and we show that it outperforms the previous schemes.

3.3.2.3.1 Divide-in-2 scheduling

The first scheduling strategy to deal with Problem 3.1 is a greedy approach that we refer to as the divide-in-2 strategy. This method has been used in [106] in a different context with directional antenna systems which tend to be very inefficient for broadcasting information. The greedy strategy consists of estimating as many γ_{ij} as possible at each step, i.e. coloring as many squares in green as possible at each step. To this end, let $\mathcal{S}_{div} = \{\alpha_1, \dots, \alpha_{L_{S_{div}}(N)}\}$ be the divide-in-2 scheduling with $L_{S_{div}}$ the length of the scheduling and let $\mathbf{\Omega}_{div}$ be the corresponding progression matrix. In the following, we describe and analyze the first iterations of the divide-in-2 scheduling procedure.

After initializing the progression matrix as $\mathbf{\Omega}_{div}(0) = \mathbf{I}_N$, the transmissions $\alpha_1 = (1, \dots, \lceil N/2 \rceil)$ and $\alpha_2 = (\lceil N/2 \rceil + 1, \dots, N)$ are performed in steps 1 and 2, which updates $\mathbf{\Omega}_{div}$

as follows:

$$\mathbf{\Omega}_{div}(1) = \begin{pmatrix} \mathbf{I}_{\lceil N/2 \rceil} & \mathbf{0}_{\lceil N/2 \rceil, \lfloor N/2 \rfloor} \\ \mathbf{1}_{\lfloor N/2 \rfloor, \lceil N/2 \rceil} & \mathbf{I}_{\lfloor N/2 \rfloor} \end{pmatrix}, \quad (3.39)$$

$$\mathbf{\Omega}_{div}(2) = \begin{pmatrix} \mathbf{I}_{\lceil N/2 \rceil} & \mathbf{1}_{\lceil N/2 \rceil, \lfloor N/2 \rfloor} \\ \mathbf{1}_{\lfloor N/2 \rfloor, \lceil N/2 \rceil} & \mathbf{I}_{\lfloor N/2 \rfloor} \end{pmatrix}, \quad (3.40)$$

which can also be written as:

$$\mathbf{\Omega}_{div}(2) = \begin{pmatrix} \mathbf{\Omega}^{left}(0) & \mathbf{1}_{\lceil N/2 \rceil, \lfloor N/2 \rfloor} \\ \mathbf{1}_{\lfloor N/2 \rfloor, \lceil N/2 \rceil} & \mathbf{\Omega}^{right}(0) \end{pmatrix}, \quad (3.41)$$

with $\mathbf{\Omega}^{left}(0) = \mathbf{I}_{\lceil N/2 \rceil}$ and $\mathbf{\Omega}^{right}(0) = \mathbf{I}_{\lfloor N/2 \rfloor}$, and where $\mathbf{\Omega}^{left}$ and $\mathbf{\Omega}^{right}$ are the progression matrices of the subnetworks respectively composed of the nodes $1, \dots, \lceil N/2 \rceil$ and $\lfloor N/2 \rfloor, \dots, N$.

From this point, estimating the remaining γ_{ij} is equivalent to independently solving the scheduling problems for both subnetworks in parallel. Since the transmissions have been performed during two steps and we need to solve the scheduling optimization problem to a subnetwork composed of $\lceil N/2 \rceil$ nodes, we can deduce the following recurrence equation which can be expressed as:

$$L_{S_{div}}(N) = L_{div}(\lceil N/2 \rceil) + 2. \quad (3.42)$$

At steps 3 and 4, after using the divide-in-2 strategy to both subnetworks, the progression matrices update as follows:

$$\mathbf{\Omega}^{left}(2) = \begin{pmatrix} \mathbf{I}_{\lceil N/4 \rceil} & \mathbf{1}_{\lceil N/4 \rceil, \lfloor \lceil N/2 \rceil / 2 \rfloor} \\ \mathbf{1}_{\lfloor \lceil N/2 \rceil / 2 \rfloor, \lceil N/4 \rceil} & \mathbf{I}_{\lfloor \lceil N/2 \rceil / 2 \rfloor} \end{pmatrix}, \quad (3.43)$$

and

$$\mathbf{\Omega}^{right}(2) = \begin{pmatrix} \mathbf{I}_{\lfloor \lfloor N/2 \rfloor / 2 \rfloor} & \mathbf{1}_{\lfloor \lfloor N/2 \rfloor / 2 \rfloor, \lfloor N/4 \rfloor} \\ \mathbf{1}_{\lfloor N/4 \rfloor, \lfloor \lfloor N/2 \rfloor / 2 \rfloor} & \mathbf{I}_{\lfloor N/4 \rfloor} \end{pmatrix}, \quad (3.44)$$

from which we can update the progression matrix $\mathbf{\Omega}_{div}$ as the following:

$$\mathbf{\Omega}_{div}(4) = \left(\begin{array}{cc|cc} \mathbf{I}_{\lceil N/4 \rceil} & \mathbf{1}_{\lceil N/4 \rceil, \lfloor \lceil N/2 \rceil / 2 \rfloor} & & \\ \mathbf{1}_{\lfloor \lceil N/2 \rceil / 2 \rfloor, \lceil N/4 \rceil} & \mathbf{I}_{\lfloor \lceil N/2 \rceil / 2 \rfloor} & & \\ \hline & & \mathbf{1}_{\lceil N/2 \rceil, \lfloor N/2 \rfloor} & \\ \hline & & \mathbf{I}_{\lfloor \lfloor N/2 \rfloor / 2 \rfloor} & \mathbf{1}_{\lfloor \lfloor N/2 \rfloor / 2 \rfloor, \lfloor N/4 \rfloor} \\ \mathbf{1}_{\lfloor N/4 \rfloor, \lfloor \lfloor N/2 \rfloor / 2 \rfloor} & & & \mathbf{I}_{\lfloor N/4 \rfloor} \end{array} \right), \quad (3.45)$$

The divide-in-2 scheduling is iteratively applied to the resulting four subnetworks respectively composed of the following nodes:

- $1, \dots, \lceil N/4 \rceil$,
- $\lceil N/4 \rceil + 1, \dots, \lceil N/2 \rceil$,
- $\lfloor N/2 \rfloor + 1, \dots, \lfloor N/2 \rfloor + \lfloor N/4 \rfloor$

$$- \lceil N/2 \rceil + \lfloor N/4 \rfloor, \dots, N.$$

Finally, the divide-in-2 approach is graphically represented by the grid of squares in Fig. 3.18 for $N = 8$ nodes. In this figure, we can observe that after applying the trans-

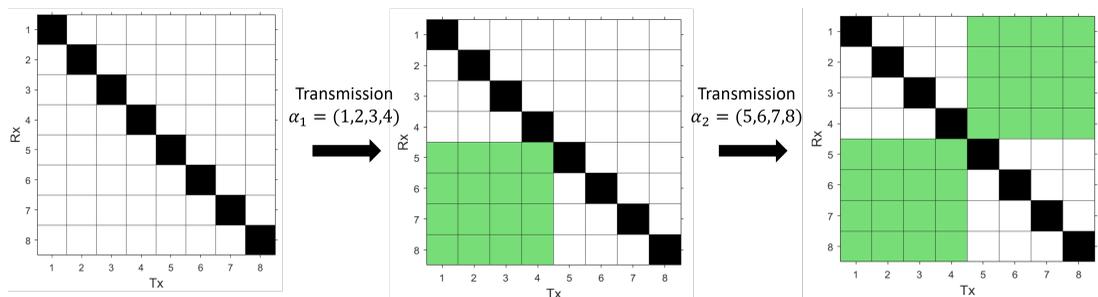


Figure 3.18: divide-in-2 scheduling strategy graphic illustrated with the grid of squares for $N = 8$ nodes.

missions α_1 and α_2 , the remaining squares to color are two 4×4 squares that can be treated independently and in parallel. This corresponds to estimate the remaining channel statistics in subnetworks composed of 4 nodes which can be solved independently and in parallel.

Furthermore, using the recurrence equation provided in (3.42), we can derive the theoretical length of the divide-in-2 scheduling. Before giving this result, we provide two useful lemmas.

Lemma 3.3. For $N \leq 4$, the length of the optimal scheduling \mathcal{S}_{opt} can be expressed as:

$$L_{\mathcal{S}_{opt}}(N) = N. \quad (3.46)$$

Proof. This lemma can be proved after performing an exhaustive search of all the possible schedulings for $N \leq 4$, which we have done numerically. \square

Lemma 3.3 implies that for $N \leq 4$, TDMA is an optimal scheduling. The following lemma provides equality of the length of divide-in-2 scheduling between an odd and even number of nodes N , and is proved in Appendix B.2.1.

Lemma 3.4.

$$\forall p \in \mathbb{N}^*, \quad L_{\mathcal{S}_{div}}(2p-1) = L_{\mathcal{S}_{div}}(2p). \quad (3.47)$$

The following result provides the theoretical number of steps of the divide-in-2 scheduling and its proof is provided in Appendix B.2.2

Result 3.6. Let $N_{min} \in \mathbb{N}^*$ such that $N_{min} \geq 4$. The following statement holds:

$$\forall N \geq N_{min}, \quad L_{\mathcal{S}_{div}}(N) = 2q + r, \quad (3.48)$$

with $q := \left\lceil \log_2 \left(\frac{N}{N_{min}} \right) \right\rceil$, $r := \left\lceil \frac{N}{2^q} \right\rceil$ and $r < N_{min}$.

In (3.48), q corresponds to the number of times we "divide the square in two" which corresponds for example to apply the transmissions α_1, α_2 in Fig. 3.18. N_{min} is the minimum number of nodes required to perform the divide-in-2 strategy, i.e. we stop "dividing the squares in two" for the subnetworks composed of several nodes smaller than N_{min} . After stopping the square dividing, we need to solve the scheduling optimization problems for the resulting subnetworks which are composed of r nodes at most. We propose to solve the remaining scheduling optimization problems with TDMA. Moreover, since TDMA is an optimal scheduling for $N \leq 4$ according to Lemma 3.3, we set $N_{min} = 5$ in practice. Therefore, to summarize, the divide-in-2 scheduling is performed as follows:

- For $N \geq N_{min}$, the divide-in-2 strategy, i.e. "divide the square in two", is applied until a resulting subnetwork is composed of strictly less than N_{min} nodes.
- For any network or subnetwork with less than N_{min} nodes, TDMA is used.

With this solution, the channel statistics of all pairs of links in a network of 8 nodes can be estimated in 6 steps which is lower than with TDMA.

In Algorithm 3.3, an implementation of this method is provided. The algorithm

Algorithm 3.3: divide-in-2 scheduling

Initialize $\mathcal{S}_{div} = \{\}$ and $n_{index} = 0$.

Update $\mathcal{S}_{div} = \text{divide}(\mathcal{S}_{div}, N, n_{index})$ with the function divide defined below.

Function $\text{divide}(\mathcal{S}_{div}, N, n_{index})$

if $N \leq 4$ **then**

$\mathcal{S}_{div} = \mathcal{S}_{div} \cup \{\alpha_i\}$ with $\alpha_i = (i + n_{index})$, for $i = 1, \dots, N_{min}$.

else

 Set $\alpha_1 = (1, \dots, \lceil N/2 \rceil) \oplus n_{index}$ and $\alpha_2 = (\lceil N/2 \rceil + 1, \dots, N) \oplus n_{index}$.

$\mathcal{S}_{div} = \mathcal{S}_{div} \cup \{\alpha_1, \alpha_2\}$.

$\mathcal{S}_{div} = \text{divide}(\mathcal{S}_{div}, \lceil N/2 \rceil, n_{index})$

$\mathcal{S}_{div} = \text{divide}(\mathcal{S}_{div}, \lfloor N/2 \rfloor, n_{index} + \lceil N/2 \rceil)$

Return \mathcal{S}_{div} .

initializes an empty scheduling $\mathcal{S}_{div} = \{\}$ and updates it using the recursive function divide. The function divide has two other input parameters N and n_{index} which means that the divide-in-2 method is applied to a network composed of the nodes $n_{index} + 1, \dots, n_{index} + N$. We also define the operator \oplus which performs the following operation: $(i, j) \oplus n = (i + n, j + n)$. For $N \leq 4$, TDMA is applied since its scheduling is optimal. For $N > 4$, the recursion is first applied on the first half of the transmissions, then on the second half.

This scheduling method implementation presents two advantages over TDMA for $N > 4$:

- its complexity of implementation is in $O(\log N)$ which is much lower than the exhaustive search approach. Indeed, this is due to the recursive function whose
-

recursion is applied to a network with a size twice smaller, i.e. $\lceil N/2 \rceil$ or $\lfloor N/2 \rfloor$ nodes.

- its scheduling length is in $O(\log N)$ according to Result 3.6 which is lower than TDMA.

In the next section, we present another scheduling solution whose length is also lower than TDMA to solve Problem 3.1.

3.3.2.3.2 2-transmission scheduling

In this section, we propose a novel scheduling method that we call *2-transmission* scheduling whose length is lower than TDMA. This scheduling method allows the estimation of the channel statistics γ_{ij} for all pairs of links (i, j) of the network. The idea of this technique is to make each node transmit at most twice during the cycle. This approach is motivated by the following:

- If simultaneous transmissions occur at each step of a scheduling method, each node is required to transmit exactly twice in a cycle to estimate γ_{ij} for all pairs (i, j) . Indeed, let us consider an example with two nodes i and j . When they transmit at the same time, the channel statistics γ_{ij} and γ_{ji} are not estimated because of the TDD mode constraint, which leads to the necessity of the nodes i and j to transmit again in the same cycle at different steps.
- By limiting the overall number of transmissions in the cycle, we hope to decrease the number of steps of this scheduling method.

This technique should not be mistaken with the scheduling solution proposed in Section 3.2 where two nodes transmit during the same step whereas in the *2-transmission* scheduling, each node transmits twice during a cycle.

Formally, let us define $\mathcal{S}_{2TX} = \{\beta_1, \dots, \beta_{L_{\mathcal{S}_{2TX}}}\}$ as the 2-transmission scheduling with β_t the transmissions at step t and $L_{\mathcal{S}_{2TX}}$ the length of the scheduling. Now, let us define $m < N$ as the maximum number of simultaneous transmissions during a step in the whole cycle. We will show later how to determine the value m . The 2-transmission scheduling is constructed as the following:

- We compute the transmissions from step 1 as follows: $\beta_1 = (1, \dots, m)$.
- Since nodes $1, \dots, m$ need to transmit again at different steps from each other, node 1 transmits again at step 2, node 2 at step 3, ..., and node m at step $m + 1$.
- At step 2, since there are $m - 1$ remaining slots, the nodes $m + 1, \dots, 2m - 1$ transmit at this step which leads to $\beta_2 = (1, m + 1, \dots, 2m - 1)$.

- Similarly, the nodes $m + 1, \dots, 2m - 1$ need to transmit again at different steps from each other, so node $m + 1$ transmits again at step 3, $m + 2$ at step 4, ..., $2m - 1$ at step $m + 1$.
- The previous steps are iteratively performed until each node has transmitted twice during the cycle.

An example of this procedure is illustrated in Fig. 3.19 for $N = 8$ nodes and $m = 4$. In this example, the 2-transmission method provides a scheduling method in 5 steps, which is lower than the divide-in-2 scheduling and TDMA. Moreover, its graphical representation with the grid of squares is provided in Fig. 3.20.

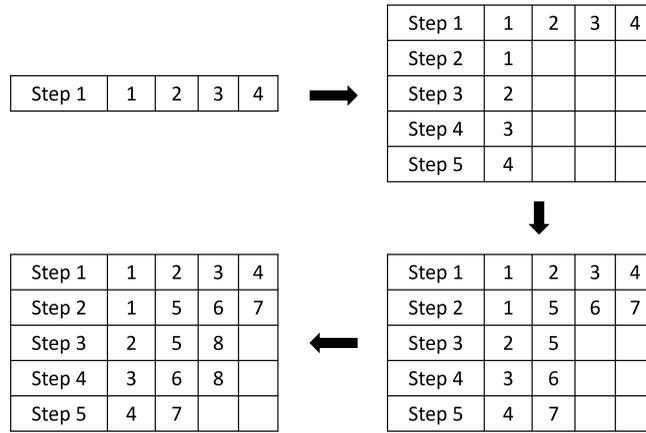


Figure 3.19: 2-transmission scheduling for $N = 8$ nodes and $m = 4$. At step 1, the transmissions $\beta_1 = (1, 2, 3, 4)$ are performed, hence the nodes 1, 2, 3, 4 need to transmit again at steps 2, 3, 4, 5, respectively. Then, at step 2, the transmissions $\beta_2 = (1, 5, 6, 7)$ are performed, so the nodes 5, 6, 7 transmit again at steps 3, 4, 5, and so on.

For this scheduling to be successful, m needs to be chosen high enough. Before deriving the optimal value of m , we provide the following properties.

Result 3.7. *Let m be the maximum number of simultaneous transmissions in a step. The following properties hold:*

$$p_1) L_{S_{2TX}} = m + 1.$$

$$p_2) m(m + 1) \geq 2N.$$

Proof. p_1 and p_2 can be proved as the following:

p_1 : After nodes $1, \dots, m$ have transmitted at step 1, these m nodes are required to transmit again at different steps from each other. Therefore, we have $L_{S_{2TX}} = m + 1$.

p_2 : On the one hand, the total number of transmissions from all the nodes in the cycle is $2N$ since each node transmits twice. On the other hand, since m is set as the

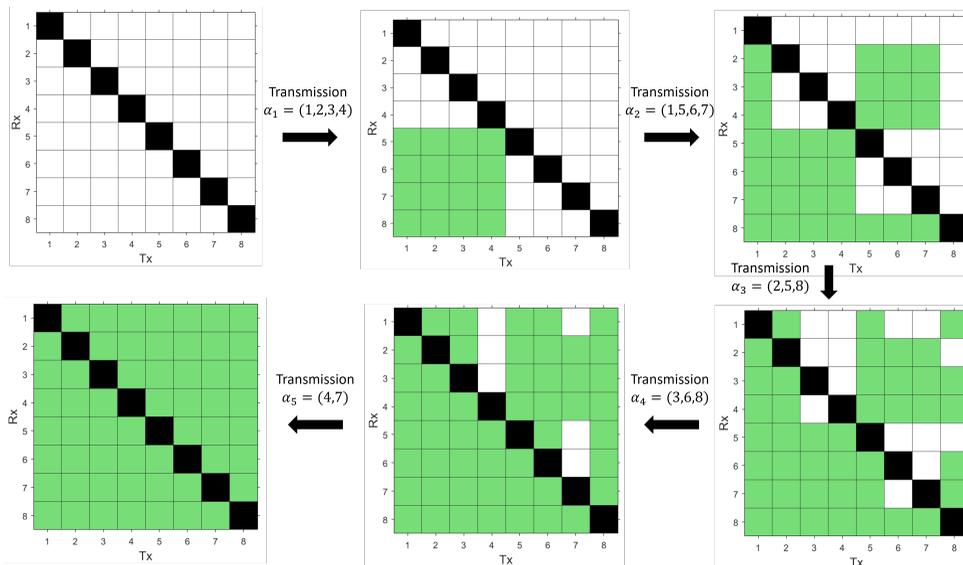


Figure 3.20: 2-transmission scheduling graphic representation with squares for $N = 8$ nodes, this method is performed in 5 steps.

maximum number of simultaneous transmissions in the cycle and $L_{S_{2TX}}$ the length of the scheduling, the upper bound of the total number of transmissions from all the nodes in the cycle is $mL_{S_{2TX}}$, which is equal to $m(m+1)$ using p_1 . As a consequence, the inequality $m(m+1) \geq 2N$ holds.

□

Next, we provide a result computing the theoretical optimal value of m , i.e. the lowest value m such that 2-transmission scheduling is successful. Once computed, the length of 2-transmission scheduling can directly be deduced from m . The proof is given in Appendix B.2.3.

Result 3.8. *Considering a network composed of $N > 4$ nodes, the optimal value of m , i.e. the smallest value of $m \geq 2$ such that 2-transmission scheduling is successful, is given as follows:*

$$m = \left\lceil \frac{\sqrt{8N+1}-1}{2} \right\rceil. \quad (3.49)$$

Using this value of m , the length of the 2-transmission scheduling can be expressed as:

$$L_{S_{2TX}} = \left\lceil \frac{\sqrt{8N+1}+1}{2} \right\rceil. \quad (3.50)$$

Result 3.8 is proved by taking the smallest value m such that the inequality provided in p_2 from Result 3.7 is respected.

We describe the procedure of the 2-transmission scheduling in Algorithm 3.4. After computing the theoretical values of m and L_{2TX} , the transmissions are updated at each

Algorithm 3.4: 2-transmission scheduling

Compute m from (3.49) and $L_{\mathcal{S}_{2TX}} = m + 1$.
Set $n_{last} = 0$.
for $t = 1, \dots, L_{2TX}$ **do**
 Update $\beta_t = (\beta_t, n_{last} \oplus (1, \dots, m - t + 1))$.
 Update $\beta_{t+i} = (\beta_{t+i}, n_{last} + i)$, for $i = 1, \dots, m - t + 1$.
 Update $n_{last} = n_{last} + m - t + 1$.
end
Return $\mathcal{S}_{2TX} = \{\beta_1, \dots, \beta_{L_{2TX}}\}$.

step t . The index n_{last} is used to deal with the transmissions at each iteration of the algorithm. In Fig. 3.19, we can see that during the first iteration, we compute the transmissions (1, 2, 3, 4), then the transmissions (5, 6, 7) during the second iteration, and so on. At each iteration, the transmissions β_t and β_{t+i} for $i = 1, \dots, m - t + 1$ are updated.

This scheduling method implementation presents two advantages:

- its complexity of implementation is much lower than the exhaustive search approach.
- its length is in $O(\sqrt{N})$ according to Result 3.8 which is lower than TDMA.

For high value of N , the divide-in-2 scheduling in $O(\log N)$ has better performance than the 2-transmission scheduling in $O(\sqrt{N})$ in terms of scheduling length. However, we show that it is the opposite when N is below a specific value. In the next section, we determine this value and propose another scheduling solution combining the divide-in-2 and 2-transmission scheduling methods.

3.3.2.3.3 Mixed scheduling

In this section, we describe another scheduling technique that we call *mixed scheduling* which consists of combining the two previous scheduling methods. Since the 2-transmission scheduling is better for lower values of N and than the divide-in-2 for higher values of N , the mixed scheduling consists of first using the divide-in-2 approach if the number of nodes N is high enough, before applying the 2-transmission scheduling technique. To be more precise, since "dividing the square in two" results in creating several subnetworks, the 2-transmission method can then be applied to each of these subnetworks independently and in parallel. An example of the application of the mixed scheduling is illustrated with the grid of squares in Fig. 3.21. During the two first steps, the divide-in-2 method is performed which results in two independent squares. Then, the 2-transmission scheduling technique is used in both squares in parallel.

Now, the remaining issue is to determine the moment when to switch from the divide-in-2 to the 2-transmission strategy. To do so, we propose to apply the divide-in-2 strategy

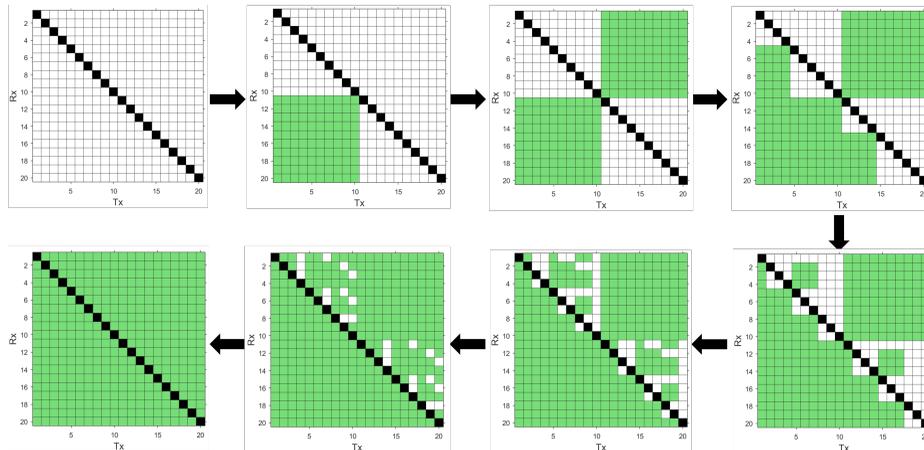


Figure 3.21: Mixed scheduling graphic representation with squares for $N = 20$ nodes performed in 7 steps. In the first two steps, the divide-in-2 strategy is performed. Then, in each square is applied in parallel the 2-transmission strategy.

until the resulting subnetworks size becomes below a threshold value N_{limit} and then, the 2-transmission strategy is performed. In the following, we provide a result to determine the theoretical length of the mixed scheduling and the optimal value of N_{limit} . The proofs are given in Appendix B.2.4.

Result 3.9. *Considering a network composed of $N > 4$ nodes, the mixed scheduling number of steps can be expressed as:*

$$L_{mix} = 2q_{mix} + \left\lceil \frac{\sqrt{8r_{mix} + 1} + 1}{2} \right\rceil, \quad (3.51)$$

with

$$q_{mix} = \left\lceil \log_2 \left(\frac{N}{N_{limit}} \right) \right\rceil, \quad (3.52)$$

$$r_{mix} = \left\lceil \frac{N}{2^{q_{mix}}} \right\rceil. \quad (3.53)$$

Moreover, there are multiple optimal values of N_{limit} and can be expressed as:

$$N_{limit} \in \{21, \dots, 28\}. \quad (3.54)$$

The expression of the number of steps for the mixed scheduling is provided in (3.51) with q_{mix} corresponding to the number of times the divide-in-2 strategy is applied, N_{limit} is the limit size of the subnetworks obtained after the divide-in-2 scheduling under which the method is changed to the 2-transmission one, and $r_{mix} \leq N_{limit}$ is the real size of the subnetworks. It is worth mentioning that for $N \leq N_{limit}$, we have $q_{mix} = 0$ and $r_{mix} = N$ which means that only the 2-transmission scheduling is performed. Furthermore, the optimal value of N_{limit} has been found numerically with an exhaustive search for $N_{limit} = 5, \dots, 1000$. In Fig. 3.22, the number of steps of the three proposed scheduling

for various numbers of nodes N is compared. The mixed scheduling implemented for $N_{limit} = 21$ has the same performance as the 2-transmission scheduling for $N \leq 28$, then outperforms the two other methods. It is worth noticing that although one could intuitively think that the divide-in-2 method becomes better than the 2-transmission method from $N \geq N_{limit}$, in reality, it is the case from $N \geq 78$.

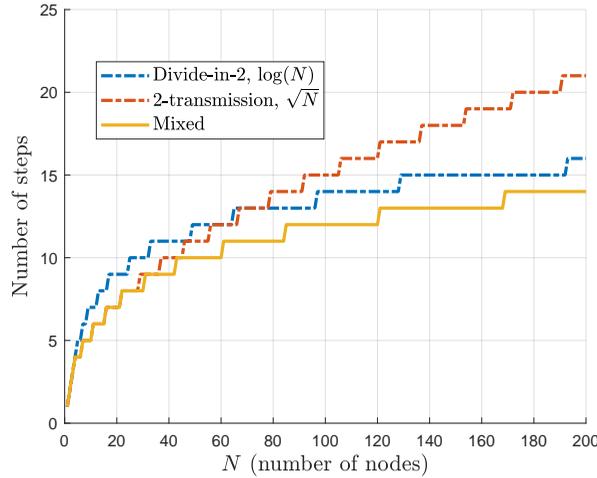


Figure 3.22: Comparison of the divide-in-2, 2-transmission, and mixed schedulings in terms of scheduling length, versus the number of nodes N .

After proposing and analyzing multiple scheduling methods to optimize the time to estimate all the channel statistics, we describe in the next section how to choose adequate orthogonal sequences.

3.3.2.4 Study of each scheduling performance

Now that we have described optimized scheduling solutions that outperforms TDMA in terms of scheduling length, in this section, we study and compare the performance of the proposed scheduling in terms of sensing time and energy consumption.

3.3.2.4.1 Sensing duration

Even though the length of the proposed scheduling is lower than TDMA's, the sensing duration is not necessarily lower because of the use of orthogonal sequences which extends the duration. Therefore, in this section, we compare the sensing duration of both solutions.

Let m be the maximum number of simultaneous transmissions during a cycle. In the mixed scheduling, we set the length of the codes at step t as $N_{ZC}(t) \leq m$. To have a fair comparison in terms of channel estimation quality between the mixed scheduling and TDMA, we set the length of the orthogonal sequences as $N_{ZC}(t) = m, \forall t$ and we assume that when using TDMA, each node transmits m samples at each step. Therefore, on the

one hand, the sensing duration for **TDMA** can be expressed as:

$$T_{TDMA} = mN. \quad (3.55)$$

On the other hand, the sensing duration of the mixed scheduling can be written as:

$$T_{mixed} = \sum_{t=1}^L N_{ZC}(t), \quad (3.56)$$

with L the length of the mixed scheduling. After plugging the equality $N_{ZC}(t) = m$ into (3.56), the following equality holds:

$$T_{mixed} = mL. \quad (3.57)$$

Since it has been proved that L is lower than N for $N \geq 5$, we can conclude that the sensing duration of the mixed scheduling is lower than **TDMA**.

3.3.2.4.2 Energy consumption

In this section, we provide a study of the energy consumption of **TDMA**, the divide-in-2 and the 2-transmission scheduling methods. To this end, we define the energy consumption of a scheduling method as the total number of transmissions from all the nodes of the network during a cycle.

For the divide-in-2 scheduling, energy consumption can be approximated by assuming that at each step, $N/2$ simultaneous transmissions occur while in reality, this number is a little lower. Moreover, its scheduling length is in $O(\log N)$ according to Result 3.6, the energy consumption can be expressed as:

$$\mathcal{E}_{div} = O(N \log N). \quad (3.58)$$

For the 2-transmission scheduling, the energy consumption can be written as:

$$\mathcal{E}_{2TX} = 2N, \quad (3.59)$$

which is optimal among the scheduling solutions with at least two simultaneous transmissions at each step.

The energy consumption for **TDMA** can be expressed as:

$$\mathcal{E}_{TDMA} = N, \quad (3.60)$$

which is optimal among all the scheduling methods.

Although the proposed scheduling methods using the divide-in-2 or the 2-transmission approaches present lower sensing duration, their energy consumption is higher than **TDMA**'s. Therefore, a trade-off between the sensing duration and the energy consumption is required depending on the needs of the systems.

3.3.3 Conclusion of Section 3.3

In this section, we have proposed a novel scheduling solution that outperforms [TDMA](#) in terms of scheduling length. This method allowing simultaneous transmissions requires the use of orthogonal sequences to remove the interference. Moreover, it combines two scheduling techniques: the divide-in-2 and the 2-transmission approaches. The scheduling lengths have been derived for both approaches and the mixed scheduling solution analytically. Finally, to complete the study, the proposed scheduling solutions are also compared with [TDMA](#) in terms of sensing duration and energy consumption.

3.4 Conclusion

In this chapter, we propose two approaches, to decrease the sensing duration as compared to [TDMA](#), for estimating the channel statistics of all pairs of nodes of the network under [TDD](#) mode assumption. In the case where orthogonal sequences cannot be used and receivers give only access to [SINR](#) samples, the first scheduling approach is performed for channel statistics estimation. Otherwise, we apply the second approach which uses orthogonal sequences since it presents better performance than the first one in terms of scheduling length.

In the first approach, we have described a scheduling solution allowing two simultaneous transmissions that outperforms [TDMA](#) in terms of sensing duration. Then, we have formulated the channel statistics estimation problem and have shown that it is equivalent to solving the estimation problem at each receiver under a single interference. Each receiver needs to estimate the direct and the interfering channels' magnitude mean power and the estimation is performed in a frequency-flat Rayleigh channel using a set of [SINR](#) samples. To do so, we introduce two new estimators based on the [MLE](#) and the [MoM](#) and implement them using Newton's method for which we have proposed initialization procedures guaranteeing the convergence of both algorithms. The performance of the estimators has been studied through simulations and has shown that the variance of the [MLE](#) is close to the [CRLB](#), while the variance of the [MoM](#) is slightly higher. However, the estimator has the advantage of being less complex than that of the [MLE](#). The [MLE](#) can be initialized with [MoM](#)'s estimates, which is less complex than if it had been randomly initialized.

In the second approach, we have proposed another scheduling method using simultaneous transmissions to estimate the channel statistics of all pairs of nodes of the network. To remove the occurring interference, a procedure using orthogonal sequences is performed. We show that the proposed scheduling solutions, i.e. the divide-in-2, 2-transmission, and mixed scheduling, outperform [TDMA](#) in terms of scheduling length. Moreover, their respective lengths of scheduling are derived. Finally, we show that although the proposed scheduling schemes have a lower sensing duration than [TDMA](#), their energy consumption is higher.

Results from Section 3.2 have been published in [52] and results from Section 3.3 have been patented in [54].

Chapter 4

Channel statistics dimensionality reduction

4.1 Introduction

4.1.1 Context and state-of-the-art

Until now, we have worked under the assumption that a central entity has access to the channel statistics related to all the links of the associated clusters at stake to compute centralized schemes such as eTIM-hybrid or JFA-eTIM-hy described in Chapter 2. However, in practice, since only local channel statistics are known by the CH of each cluster, communications overhead between the CHs are required to apply the centralized interference management solutions. Because of the high quantity of channel statistics to transmit, a large amount of signaling exchanges between the CH is needed which results in reducing the user data rate and increasing latency and complexity. In this chapter, we address the issue of compressing the channel statistics to transmit between clusters, while ensuring their effectiveness for the proposed interference management schemes.

To tackle this challenge, we propose to use data compression techniques on the channel statistics before transmitting them. In our context, *data compression* is defined as the operation of reducing the size of signaling data while preserving the essential information. Compression can be either *lossless* or *lossy*. While lossless compression allows a perfect reconstruction of the original data, lossy compression approximates the reconstruction and has usually a greater compression rate. On the one hand, several works have used lossless compression methods such as Huffman, arithmetic, and Lempel-Ziv-Welch (LZW) coding for CSI feedback compression in MIMO systems in [122], [87] and [91], respectively. Lossy compression techniques, such as discrete wavelet transform (DWT) [45] or discrete cosine transform (DCT) [2], have been mainly used for image compression, including the widely used JPEG format. However, since these compression schemes only exploit the spatial redundancies in the data, such as repetitive patterns or similar color values in

neighboring pixels, to reduce the size of the data, they do not provide the most efficient solutions.

Among the lossy compression schemes, **DR** consists of reducing the size of the input variables using a training dataset. By capturing the relationships between several data, **DR** methods can result in higher performance than conventional compression techniques. Several other **DR** techniques are described in [117]. Among these schemes, t-distributed stochastic neighbor embedding (**t-SNE**) for channel propagation measures [95] and hybrid principal components analysis (**PCA**) with Huffman coding for image compression [116], have been proposed. Nevertheless, while **t-SNE** is not easily reversible from the encoded data, techniques for image compression cannot be used in our context since it exploits the spatial correlation contained in the image, which we cannot do. Besides, many works have applied **DR** methods to **CSI** feedback compression in **MIMO** systems such as **PCA** [59] or neural network structures like autoencoders [121], recurrent neural networks [77] or convolutional neural networks [125]. However, all these methods also make use of the spatial correlation contained in the **CSI** feedback in **MIMO** systems and can thus not be exploited for the channel statistics we wish to transmit. The most relevant work in this area appears to be [11], which uses autoencoders to encode **CSI** feedback in **D2D** communication networks in **SISO** systems. However, their model is limited as it is trained with channel gain samples that assume only small-scale fading and do not consider large-scale fading. As a result, this approach is not suitable for handling varying network topologies, as the model is not trained to adapt to such variations.

To the best of our knowledge, there is no prior work on channel statistics compression in a clustered ad hoc network in **SISO**. In this chapter, we aim at finding a solution based on **DR** which reduces the size of the channel statistics for any links topology.

4.1.2 Contributions

The contributions of this chapter can be summarized as follows:

- We provide a procedure to train methods reducing the channel statistics dimension which can be applied to any links topology.
- We implement multiple **DR** methods and compare them with compression techniques, i.e. without using any training data, through numerical simulations. We also compute the **eTIM**-hybrid method using the channel statistics values estimated with the **DR** schemes to assess their performance in terms of average sum rate and minimum user rate per link.

4.2 System model

The system model is identical to the one described in Chapter 2.2.1, i.e. we consider the association of two clusters C_1 and C_2 composed of n_{C_1} and n_{C_2} links, respectively,

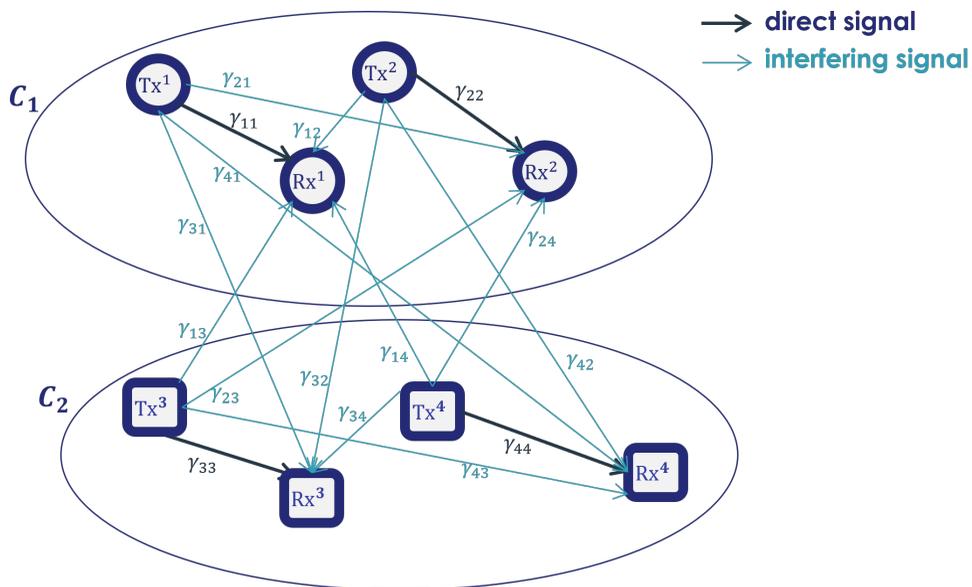


Figure 4.1: Representation of two associated clusters C_1 and C_2 with $n_{C_1} = n_{C_2} = 2$ links per cluster communicating on the same frequency band. γ_{ij} represents the channel statistics from Tx^j to Rx^i .

with $n_{C_1} + n_{C_2} = N$. It is important to note that in this chapter, we are using a system model that considers a specific number of links within each cluster, as was the case in Chapter 2, rather than considering the number of nodes within each cluster as we did in Chapter 3. Moreover, the notation γ_{ij} refers in this chapter to the channel statistics from Tx^j to Rx^i . An illustration of two adjacent clusters with $n_{C_1} = n_{C_2} = 2$ links per cluster and the corresponding channel statistics γ_{ij} are represented in Fig. 4.1.

We assume that in each cluster C_k , $k = 1, 2$, each node i have estimated the channel statistics γ_{ij} from all the possible transmitters $j = 1, \dots, i-1, i+1, \dots, N$ and have sent it to its CH. Thus, the CH of C_1 and C_2 have access to the matrices Γ^{C_1} and Γ^{C_2} , respectively, which can be expressed as:

$$\Gamma^{C_1} = \begin{pmatrix} \gamma_{1,1} & \cdots & \gamma_{1,N} \\ \vdots & & \vdots \\ \gamma_{n_{C_1},1} & \cdots & \gamma_{n_{C_1},N} \end{pmatrix}, \quad (4.1)$$

and

$$\Gamma^{C_2} = \begin{pmatrix} \gamma_{n_{C_1}+1,1} & \cdots & \gamma_{n_{C_1}+1,N} \\ \vdots & & \vdots \\ \gamma_{N,1} & \cdots & \gamma_{N,N} \end{pmatrix}. \quad (4.2)$$

To apply the interference management methods described in Chapter 3, each CH needs to know the following matrix:

$$\Gamma = \begin{pmatrix} \Gamma^{C_1} \\ \Gamma^{C_2} \end{pmatrix}. \quad (4.3)$$

As a consequence each CH has to send its locally known channel statistics matrix to the CH of the other cluster. Such a transmission may induce a significant signaling overhead (especially for large values of N) and thus we propose to use compression methods to reduce this overhead.

The general idea of the proposed signaling reduction procedure is the following one: each CH, for instance, the CH of C_1 , first uses a technique to reduce the size of Γ^{C_1} into a smaller representation $\mathbf{y} \in \mathbb{R}^{1 \times d}$, with $d < n_{C_1}N$, before transmitting it to the CH of C_2 . Then, C_2 reconstructs Γ^{C_1} from the received \mathbf{y} and forms the estimated channel statistics matrix as:

$$\tilde{\Gamma} = \begin{pmatrix} \tilde{\Gamma}^{C_1} \\ \Gamma^{C_2} \end{pmatrix}, \quad (4.4)$$

where $\tilde{\Gamma}^{C_1}$ is the estimated channel statistics matrix from C_1 . Afterward, the CH of C_2 uses an interference management technique such as eTIM-hybrid or JFA-eTIM-hy using $\tilde{\Gamma}$. Fig. 4.2 illustrates the transmission of Γ^{C_1} using an encoder at C_1 and a decoder at C_2 .

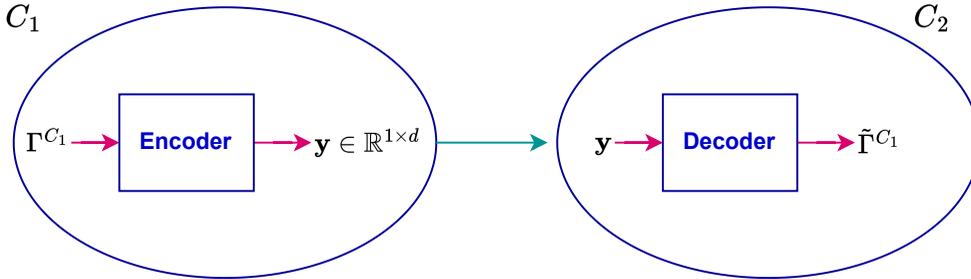


Figure 4.2: Representation of cluster C_1 encoding the channel statistics Γ^{C_1} into $\mathbf{y} \in \mathbb{R}^{1 \times d}$, with $d < n_{C_1}N$, before transmitting it to C_2 . After receiving \mathbf{y} in C_2 , reconstruction with the decoder is done to get $\tilde{\Gamma}^{C_1}$.

In this context, the channel statistics matrix Γ^{C_i} , $i = 1, 2$ is supposed constant during the encoding, transmission, and decoding process. Our goal is to find efficient methods to reduce its size while minimizing the information loss. Although multiple lossless and lossy compression schemes exist, in the following, we propose to use a DR approach to decrease the signaling exchanges between the CHs. This procedure has the benefit of using several samples, referred to as *training data*, and exploiting the relationships between them to obtain a better encoding version of the channel statistics matrix to transmit. DR methods usually provide higher performance than conventional compression techniques, at the cost of higher memory storage since it requires each CH to store the parameters of the trained DR model.

4.3 Dimensionality reduction

In what follows, we first formulate the DR problem and how to preprocess the data, before presenting several DR methods that we have implemented.

4.3.1 Problem formulation

We remind that the objective is to transmit the channel statistics matrix Γ^{C_i} from the CH of the cluster C_i , $i = 1, 2$ to the CH of the other cluster C_j , $j \neq i$. To simplify the notations, we consider the transmission of Γ^{C_1} from C_1 to C_2 since dealing with the reciprocal transmission can be proceeded similarly. To formulate the problem of DR in this context, let $\mathbf{x} \in \mathbb{R}^{1 \times p}$ be the vector representation of the flattened matrix Γ^{C_1} such that:

$$\mathbf{x} := (\Gamma^{C_1}(1, 1), \dots, \Gamma^{C_1}(1, N), \Gamma^{C_1}(2, 1), \dots, \Gamma^{C_1}(2, N), \dots, \Gamma^{C_1}(n_{C_1}, N)), \quad (4.5)$$

where $p := n_{C_1}N$ and $\Gamma^{C_1}(i, j)$ is the (i, j) th element of Γ^{C_1} . Our goal is to find the encoder φ which maps the input data \mathbf{x} to a compressed representation $\mathbf{y} := \varphi(\mathbf{x}) \in \mathbb{R}^{1 \times d}$ with $d < p$ and the decoder ψ which reconstructs the corresponding input $\tilde{\mathbf{x}} := \psi(\mathbf{y}) \in \mathbb{R}^{1 \times p}$.

We aim to create a DR technique that can efficiently encode Γ^{C_1} for various link topologies. The goal is to avoid the need for multiple training phases, which can be computationally expensive. To this end, the training data is generated by collecting several i.i.d samples where each sample corresponds to a random topology of the links. The different steps to train the DR solution are the following:

1. We generate N_{train} random topology and for each topology n we compute the resulting channel statistics matrix $\Gamma_n^{C_1}$. From all these topologies, we form $\mathbf{X} := (\mathbf{x}_1, \dots, \mathbf{x}_{N_{train}})^T$, where \mathbf{x}_n is the flattened representation of $\Gamma_n^{C_1}$.
2. The training data \mathbf{X} are preprocessed to have values between 0 and 1, which is expected to improve the performance of the DR methods [3]. Since the input data \mathbf{x}_i have values that span several orders of magnitude, it is first converted in dB and then, normalized using min-max scaling:

$$\forall n \in \{1, \dots, N_{train}\}, \quad \forall m \in \{1, \dots, p\}, \quad x_{scale,n}(m) = \frac{x_n(m) - x_{min}}{x_{max} - x_{min}}, \quad (4.6)$$

where $x_n(m)$ is the m th element of \mathbf{x}_n and x_{min} and x_{max} are set according to the simulation parameters (the details are provided in Section 4.4.1).

3. At each CH, the DR solution is trained with the same preprocessed dataset $\mathbf{X}_{scale} := (\mathbf{x}_{scale,1}, \dots, \mathbf{x}_{scale,N_{train}})^T$, with $\mathbf{x}_{scale,n} := (x_{scale,n}(1), \dots, x_{scale,n}(p))$, $\forall n \in \{1, \dots, N_{train}\}$.

In practice, after performing the offline training, the parameters of the trained model are stored at the CH of each cluster.

Then, considering testing data Γ^{C_1} to transmit from C_1 to C_2 , which is independent of the training dataset, the different steps to compress and send it are:

1. The input data is flattened at the **CH** of C_1 into a vector $\mathbf{x} \in \mathbb{R}^{1 \times p}$ and is then preprocessed like in the second step of the training phase (4.6), i.e. converted in dB and scaled like in (4.6) to get \mathbf{x}_{std} .
2. Using the trained **DR** solution, \mathbf{x}_{std} is encoded into a vector $\mathbf{y} \in \mathbb{R}^{1 \times d}$ with $d < p$ and sent to the **CH** of C_2 .
3. At the **CH** of the other cluster C_2 , the received data \mathbf{y} is decoded into the estimated $\tilde{\mathbf{x}} \in \mathbb{R}^{1 \times p}$, from which we get the estimated channel statistics matrix $\tilde{\mathbf{I}}^{C_1}$.

Afterward, considering the flattened channel statistics matrix to send $\mathbf{x} \in \mathbb{R}^{1 \times p}$ from the **CH** of C_1 , \mathbf{x} is encoded into $\mathbf{c} \in \mathbb{R}^{1 \times d}$ instead of \mathbf{x} , with $d < p$, and sent to the **CH** of C_2 where it will be decoded using the common **DR** model to $\tilde{\mathbf{x}} \in \mathbb{R}^{1 \times p}$. The value of d is set by the system designer to achieve a desired trade-off between the compression of the signaling and the information loss, indeed, the higher the value of d , the lower the information loss, but also the lower the compression of the signaling. Simulation results enabling the study of this trade-off are provided in Section 4.4.

We describe different **DR** methods in the next sections.

4.3.2 Linear and non-linear **DR** techniques: a comparison of **PCA** and kernel **PCA**

Among the **DR** methods, **PCA** and kernel **PCA** (**KPCA**) are often used in many fields of applications usually including computer vision [25] and **MIMO** channel [78].

4.3.2.1 Principal components analysis

Linear **DR** methods are convex techniques where the optimization problem is linear. **PCA** [92] is one of the most popular techniques among the linear methods for reducing the dimensionality of data. The goal of **PCA** is to find the best projection of data into a linear subspace of lower dimensionality by maximizing the variance of the projected data. Let us notice that this "variance" we wish to maximize measures how well the data is spread in the directions of the projected subspace and differs from the "variance of an estimator" like in Chapter 2, which measures the uncertainty of an estimate.

Formally, an implementation procedure of the **PCA** is provided in [104, Section 23.1.2] and we provide a summary of this method. Let us consider a dataset $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_{N_{\text{train}}})^T$, with $\mathbf{x}_i \in \mathbb{R}^{1 \times p}$, and center it, which consists in performing the following operation:

$$\tilde{\mathbf{x}}_i := \mathbf{x}_i - \frac{1}{N_{\text{train}}} \sum_{i=1}^{N_{\text{train}}} \mathbf{x}_i, \quad \forall i = 1, \dots, N_{\text{train}}. \quad (4.7)$$

where $\mathbf{0}_{1 \times p}$ is the $1 \times p$ vector only composed of 0 and we define $\tilde{\mathbf{X}} = (\tilde{\mathbf{x}}_1, \dots, \tilde{\mathbf{x}}_{N_{\text{train}}})^T$.

Then, the orthogonal projection of $\mathbf{x} \in \mathbb{R}^{1 \times p}$ onto a direction $\mathbf{w} \in \mathbb{R}^{p \times 1}$, denoted by the function $h_{\mathbf{w}} : \mathbb{R}^{1 \times p} \rightarrow \mathbb{R}$, can be computed as:

$$h_{\mathbf{w}}(\mathbf{x}) = \mathbf{x} \frac{\mathbf{w}}{\|\mathbf{w}\|_2}. \quad (4.8)$$

The goal of the procedure is to maximize the empirical variance captured by $h_{\mathbf{w}}$, which is denoted by $Var(h_{\mathbf{w}})$ and is defined as:

$$Var(h_{\mathbf{w}}) := \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} h_{\mathbf{w}}(\bar{\mathbf{x}}_i)^2, \quad (4.9)$$

which can be rewritten as:

$$Var(h_{\mathbf{w}}) = \frac{1}{N_{train}} \frac{\mathbf{w}^T \bar{\mathbf{X}}^T \bar{\mathbf{X}} \mathbf{w}}{\mathbf{w}^T \mathbf{w}}. \quad (4.10)$$

The optimization problem that we wish to solve consists of finding the i th principal directions $\mathbf{w}_i, i = 1, \dots, d$ by maximizing the variance, which can be written as follows:

$$\mathbf{w}_i = \arg \max_{\mathbf{w} \perp \{\mathbf{w}_1, \dots, \mathbf{w}_{i-1}\}} \mathbf{w}^T \bar{\mathbf{X}}^T \bar{\mathbf{X}} \mathbf{w}, \quad \text{s.t. } \|\mathbf{w}\|_2 = 1, \quad (4.11)$$

whose solutions can be proved to be the successive eigenvectors of $\bar{\mathbf{X}}^T \bar{\mathbf{X}}$, ranked by decreasing eigenvalues [104]. After computing the d eigenvectors corresponding to the d highest eigenvalues denoted as $\mathbf{W}_d = (\mathbf{w}_1, \dots, \mathbf{w}_d)$, we can compute the d principal components which can be written as $\mathbf{T}_d := \mathbf{X} \mathbf{W}_d$.

The PCA training computation in our context can be summarized as follows:

1. The training data \mathbf{X}_{std} is computed by collecting and preprocessing [i.i.d](#) samples as described in [Section 4.3.1](#), before centering it as the following:

$$\mathbf{y}_i := \mathbf{x}_{scale,i} - \boldsymbol{\mu}, \quad (4.12)$$

with $\boldsymbol{\mu}$ defined as:

$$\boldsymbol{\mu} := \frac{1}{N_{train}} \sum_{i=1}^{N_{train}} \mathbf{x}_{scale,i}. \quad (4.13)$$

2. Compute the d eigenvectors $\mathbf{W}_d = (\mathbf{w}_1, \dots, \mathbf{w}_d)$ corresponding to the d highest eigenvalues of the covariance matrix $\mathbf{Y}^T \mathbf{Y}$, with $\mathbf{Y} := (\mathbf{y}_1, \dots, \mathbf{y}_{N_{train}})$.

For the testing phase of PCA, we consider the channel statistics matrix Γ^{C_1} that we wish to transmit from C_1 to C_2 and which is a different one from the training dataset. Then, we proceed with the DR scheme with PCA according to the following phases:

1. At the CH of C_1 , considering the channel statistics matrix Γ^{C_1} to send, compute \mathbf{x}_{scale} by flattening and normalizing Γ^{C_1} as in (4.6). Then, center it as the following:

$$\mathbf{y} = \mathbf{x}_{scale} - \boldsymbol{\mu}, \quad (4.14)$$

with $\boldsymbol{\mu}$ defined in [4.13](#).

2. Compress \mathbf{y} using the following encoding procedure:

$$\mathbf{y}_{\text{enc}} = \mathbf{y}\mathbf{W}_d, \quad (4.15)$$

before sending it from the CH of C_1 to the CH of C_2 .

3. At the CH of C_2 , reconstruct the original data as the following:

$$\tilde{\mathbf{x}}_{\text{std}} = \mathbf{y}_{\text{enc}}\mathbf{W}_d^T + \boldsymbol{\mu}, \quad (4.16)$$

which assumes that the received signal at C_2 is obtained without measurement noise.

4. Reverse the normalization to get the reconstructed flattened vector $\tilde{\mathbf{x}} := (\tilde{x}(1), \dots, \tilde{x}(p))$ which can be written as:

$$\tilde{x}(m) = \tilde{x}_{\text{std}}(m)(x_{\text{max}} - x_{\text{min}}) + x_{\text{min}}, \quad \forall m \in \{1, \dots, p\}. \quad (4.17)$$

Remark 4.1. It can be shown that among the linear DR methods, PCA minimizes the total squared reconstruction error [104, Section 23.1]. In other words, let us consider the following optimization problem:

Problem 4.1.

$$\arg \min_{\mathbf{W}_e, \mathbf{W}_d \in \mathbb{R}^{p \times d}} \sum_{i=1}^{N_{\text{train}}} \|\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_i \mathbf{W}_{\text{enc}} \mathbf{W}_{\text{dec}}^T\|_2^2, \quad (4.18)$$

with $\tilde{\mathbf{x}}_i$ defined in (4.7) and with the encoding and decoding matrices $\mathbf{W}_{\text{enc}}, \mathbf{W}_{\text{dec}} \in \mathbb{R}^{p \times d}$.

It can be proved that the optimal solution of Problem 4.1 can be provided with PCA, i.e. the solution which consists of setting \mathbf{W}_{enc} as the matrix whose columns are the d eigenvectors of the matrix $\tilde{\mathbf{X}}^T \tilde{\mathbf{X}}$ corresponding to its d highest eigenvalues and setting $\mathbf{W}_{\text{dec}} = \mathbf{W}_{\text{enc}}$.

Remark 4.2. SVD is a known and easy-implemented linear technique for DR and it can be proved that it is equivalent to applying the PCA method [104, Section 23.6]. Indeed, let us consider the SVD decomposition of a centered matrix $\mathbf{X} \in \mathbb{R}^{N_{\text{train}} \times p}$ which can be written as:

$$\mathbf{X} = \mathbf{U}_{\text{svd}} \mathbf{S} \mathbf{V}_{\text{svd}}^T, \quad (4.19)$$

where \mathbf{U}_{svd} and \mathbf{V}_{svd} are real orthogonal matrices. The covariance matrix $\mathbf{X}^T \mathbf{X}$ can be expressed as:

$$\mathbf{X}^T \mathbf{X} = \mathbf{V}_{\text{svd}} \mathbf{S}^2 \mathbf{V}_{\text{svd}}^T, \quad (4.20)$$

with $\mathbf{S}^2 := \mathbf{S}\mathbf{S}$ whose elements are the square of \mathbf{S} 's elements since \mathbf{S} is a diagonal matrix.

(4.20) means that $\mathbf{V}_{\text{svd}} \in \mathbb{R}^{p \times p}$ is the matrix composed of the eigenvectors of $\mathbf{X}^T \mathbf{X}$, and thus the p principal directions, denoted as \mathbf{w}_i in the PCA.

Remark 4.3. Although in the machine learning community PCA/SVD are usually performed with training data, they can also be used for image compression by considering the image as a matrix [83]. As a consequence, the matrix Γ^{C_1} can be compressed without training data by directly applying either PCA or SVD to it. We implement this solution in the numerical simulations to compare it with the other DR methods.

4.3.2.2 Kernel PCA

KPCA [103] is a convex DR method which is a non-linear extension of the conventional PCA method. This technique uses the kernel trick, which consists of projecting the data onto a higher dimensional space, *feature space*, where the data points are more separable. This allows for a linear separation of the data points in the higher-dimensional space, even if they were not linearly separable in the original space.

Let us consider a non-linear transformation of a data point $\mathbf{x} \in \mathbb{R}^{1 \times p}$ given by $\phi(\mathbf{x})$. Although performing standard PCA in the feature space is possible, this is typically avoided due to the high dimensionality of the feature space. To this end, we use kernel methods which are an alternative approach for comparing data points by defining a function called a kernel function. The kernel function measures the similarity between two data points and outputs a similarity score. By using kernel functions, we can avoid computing an arbitrary function ϕ explicitly in very high-dimensional feature spaces, which can be computationally expensive. Instead, we can compare data points in the feature space using their similarity scores. This simplifies computations and makes it easier to compare data points.

To this end, we consider the kernel matrix \mathbf{K} whose (i, j) th components can be written as:

$$\mathbf{K}_{i,j} = \kappa(\mathbf{x}_i, \mathbf{x}_j), \quad (4.21)$$

with $\kappa(\mathbf{x}_i, \mathbf{x}_j) := \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$. It is worth noticing that using the linear kernel function $\kappa(\mathbf{x}, \mathbf{y}) = \mathbf{x}^T \mathbf{y}$ means applying the standard PCA.

The Gaussian kernel is commonly used in practice. It can be written as:

$$\kappa(\mathbf{x}, \mathbf{y}) = \exp\left(\frac{-\|\mathbf{x} - \mathbf{y}\|_2}{2\sigma^2}\right), \quad (4.22)$$

with σ a parameter to set.

The optimization problem to solve for the i th principal directions, $i = 1, \dots, d$, is provided in [120] and can be formulated as:

$$\alpha_i = \arg \max_{\alpha \in \mathbb{R}^{N_{train}}} \alpha^T \tilde{\mathbf{K}} \alpha, \quad \text{s.t.} \begin{cases} \alpha_i^T \mathbf{K} \alpha_j = 0, \\ \alpha_i^T \mathbf{K} \alpha_i = 1, \end{cases} \quad (4.23)$$

for a centered kernel matrix \mathbf{K} , which is detailed in the following procedure.

The implementation of KPCA is provided in [120] and can be described as follows:

1. Compute the kernel matrix $\mathbf{K} \in \mathbb{R}^{N_{train} \times N_{train}}$ using (4.21).
2. If the projected dataset $(\phi(\mathbf{x}_1), \dots, \phi(\mathbf{x}_{N_{train}}))$ is not centered, center it using the following substitution [12]:

$$\tilde{\mathbf{K}} = \mathbf{K} - \frac{1}{N} \mathbf{1}_N \mathbf{K} - \frac{1}{N} \mathbf{K} \mathbf{1}_N + \frac{1}{N^2} \mathbf{1}_N \mathbf{K} \mathbf{1}_N, \quad (4.24)$$

where $\mathbf{1}_N$ is the $N \times N$ matrix with all elements equal to 1.

3. The optimization problem that we wish to solve consists of finding the i th principal directions $\mathbf{w}_i, i = 1, \dots, d$ by maximizing the variance, which can be written as follows:

Solve the optimization problem provided in (4.23) by computing and sorting the d eigenvalues and eigenvectors $(\lambda_i, \mathbf{a}_i), i = 1, \dots, d$ of $\tilde{\mathbf{K}}$ and normalize the eigenvectors as follows:

$$\boldsymbol{\alpha}_i = \frac{1}{\sqrt{\lambda_i}} \mathbf{a}_i, \quad \forall i = 1, \dots, d. \quad (4.25)$$

The projections of the points onto the i th eigenvector are given by $\tilde{\mathbf{K}}\boldsymbol{\alpha}_i$.

While **KPCA** has the advantage of handling non-linear relationships between the data points, contrarily to standard **PCA**, it has the drawback of using for reconstruction an approximation which usually has both high reconstruction error and computational complexity [9].

After implementing both **PCA** and **KPCA** methods, we have observed through numerical simulations that the reconstructed channel statistics $\tilde{\mathbf{I}}^{C_1}$ can contain negative terms. As a consequence, before using this estimated parameter in interference management methods, the positive part function $x^+ := \max(x, 0)$ is applied.

In the next section, we consider a non-convex and non-linear technique for **DR** based on autoencoders.

4.3.3 Non-linear **DR** with autoencoders

In this section, we consider using autoencoder [99] which is a conventional method for non-linear **DR**. We first recall its principle before analyzing the choice of its structures and hyperparameters.

4.3.3.1 Description of an autoencoder

An autoencoder is a type of neural network where the output layer has the same number of neurons as the input layer. Its goal consists of minimizing the difference between the input and the output data.

Its structure is composed of two components:

- an **encoder** which is a neural network compressing the input into a compressed representation in a reduced dimension called *latent space*.
- a **decoder** which is also a neural network that takes the latent representation as input and tries to reconstruct the original data as output.

In the case where the encoder and decoder consist of a single hidden layer without any non-linear activation functions, i.e., both apply a linear transformation, the autoencoder produces a latent representation that is equivalent to the optimal linear dimensionality reduction scheme in terms of reconstruction error, which is **PCA**[13].

In the next section, we detail the parameters and architecture of the autoencoder that we implement.

4.3.3.2 Autoencoder architecture

First of all, the input and output layer of the autoencoder is composed of p neurons since it corresponds to the dimension of each data. Moreover, we only consider two structures of symmetric autoencoder, i.e. with the same number of layers and neurons in the encoder and decoder, and illustrate them in Figs. 4.3 and 4.4. The first structure represented in Fig. 4.3 contains hidden layers composed of the same number of neurons as the input and output layers which is p and is referred to as constant number of neurons autoencoder (CstNeu-AE). The second structure, illustrated in Fig. 4.4, consists of hidden layers with varying numbers of neurons. Let us set the parameter $\text{Neu}_{\max} > p$ as the maximum number of neurons in the layers of this autoencoder. In the encoder, while the first hidden layer contains p neurons, the number of neurons of the next hidden layers linearly increases until it reaches Neu_{\max} , then it linearly decreases until the last layer of the encoder which has p neurons. Similarly for the decoder, the number of neurons in the hidden layers starts from d , then increases until Neu_{\max} , and decreases until the output layer which contains p neurons. For example, for $p = 50$, $\text{Neu}_{\max} = 80$, and 5 layers, the number of neurons per layer in the encoder and the decoder would be 50, 65, 80, 65, 50.

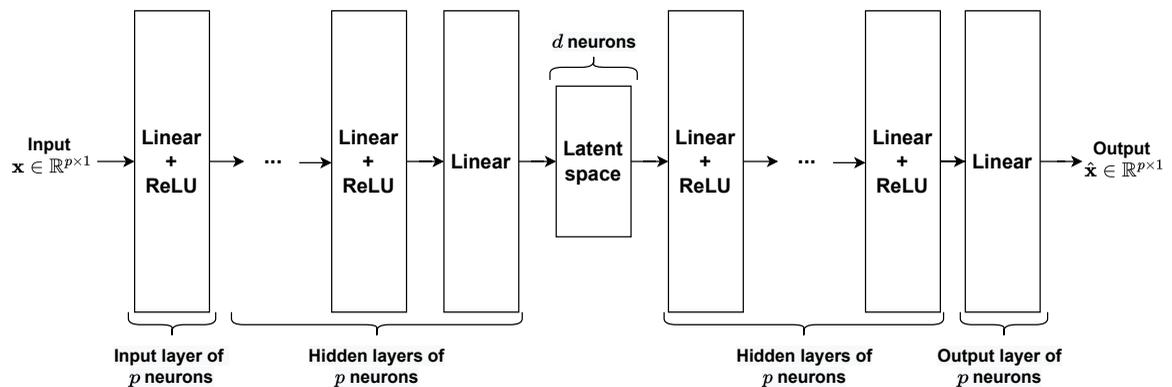


Figure 4.3: Constant number of neurons autoencoder (CstNeu-AE).

Most studies use autoencoders with a "funnel structure" towards the latent space, i.e. composed of an encoder with a decreasing number of neurons and a decoder with an increasing number of neurons. In our simulations, we did not use the funnel structure in our simulations because we varied the latent space dimension d , and using a funnel structure would require modifying the number of neurons and layers for each value of d , making the study unstable

The networks used in our study consist of a linear layer followed by a ReLU activation function, except for the last layers of the encoder and decoder. Through simulations, we found that adding a ReLU at these points slightly degraded performance, so we only

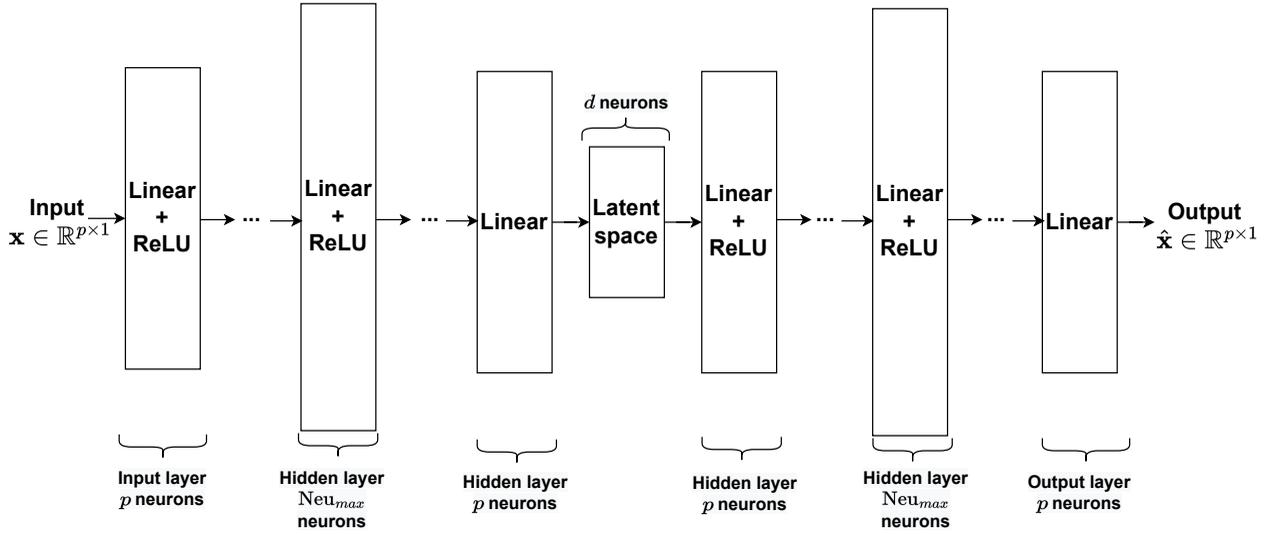


Figure 4.4: Increasing and decreasing number of neurons autoencoder (IncDecNeu-AE).

applied the ReLU function at the final layer after training. This involves applying the positive part function $x^+ = \max(x, 0)$ before using interference management methods, as with PCA.

Finally, the neural network's encoder and decoder parameters are trained through the MSE loss function between the output of the decoder and the target value, which is the original input data. After computing the MSE, the Adam optimizer [66] is used for optimizing the weights of the autoencoder since it provides a good trade-off between robustness, computation, and performance in a wide range of non-convex optimization problems in the field of deep learning. Furthermore, we define an *epoch* which represents a complete cycle composed of one forward pass and one backward pass of all the training samples through the neural network. The training exit criteria is the following one: either the maximum number of epochs is reached, or the loss has converged in the last $m_{AE} = 100$ epochs, which can be written as:

$$\frac{\left| \sum_{i=0}^{m_{AE}/2-1} \text{MSE}(t-i) - \sum_{i=m_{AE}/2}^{m_{AE}-1} \text{MSE}(t-i) \right|}{\sum_{i=0}^{m_{AE}/2-1} \text{MSE}(t-i)} < \epsilon_{AE}, \quad (4.26)$$

where $t > m_{AE}$ is the current epoch number, $\text{MSE}(t)$ is the loss function at epoch t and ϵ_{AE} is a parameter to set. The exit criteria are met if the loss function does not improve for m_{AE} consecutive epochs. Moreover, the minimum number of epochs of the training process is m_{AE} .

In the next section, we provide numerical results showing the influence of hyperparameters such as the number of layers and the training learning rate, and also compare the performance of the different DR techniques in terms of MSE and when using interference management methods.

4.4 Numerical results

In this section, we first present numerical simulations to compare the performance of the implemented DR techniques in terms of average NMSE, and then in terms of sum rate and minimum user rate per link when applying the eTIM-hybrid method described in Chapter 2 with compressed and reconstructed channel statistics. The average NMSE is defined as:

$$\text{NMSE}(\tilde{\Gamma}) := \frac{1}{n_C N} \sum_{i=1}^{n_C} \sum_{j=1}^N \left(\frac{\tilde{\gamma}_{ij} - \gamma_{ij}}{\gamma_{ij}} \right)^2, \quad (4.27)$$

with $\tilde{\Gamma}$ defined as the estimated Γ after the reconstruction of the DR method, $\tilde{\gamma}_{ij}$ and γ_{ij} are respectively $\tilde{\Gamma}$ and Γ 's (i, j) th element.

4.4.1 Setup

To assess the performance of the proposed solutions, the same setup as in Section 2.3.4 is employed, i.e. two adjacent clusters composed of n_C links each. The simulations are performed for $n_C = 5$ and 10 links per cluster and use the parameters listed in Table 2.1. Moreover, the channel is modeled such that γ_{ij} is expressed in (2.36).

To train and evaluate the DR methods, 10000 i.i.d samples of channel statistics are collected where each sample corresponds to a random topology of the links in the clusters. Among these samples, we set $N_{train} = 8000$, $N_{val} = 1000$, and $N_{test} = 1000$ which corresponds to the number of samples of the training, validation, and testing datasets, respectively. While the training and validation datasets are used for training the DR solutions, the testing dataset is utilized for evaluating them. Then, when the data are normalized using (4.6), x_{min} and x_{max} are computed using (2.36) with the highest and smallest distance values, which are respectively $d_{ij} = d_{size}$ and $d_{ij} = d_{min}$. We set $d_{min} = 100$ m to avoid the presence of links with huge rates because of the transmitter-receiver proximity.

Besides, in the following figures, each value of NMSE is averaged over the N_{test} random topologies.

4.4.2 Compression and DR methods performance

In this section, we first assess the performance of the autoencoders for various hyperparameters values in terms of NMSE versus the number of parameters we need to transmit, which corresponds to the dimensionality reduction parameter d and is related the compression rate with the following:

$$\rho_C := 1 - \frac{d}{p}. \quad (4.28)$$

Then, after choosing the optimal values of the hyperparameters, we compare the NMSE of several DR methods, versus the number of transmitted parameters.

4.4.2.1 Study of the autoencoder hyperparameters influence

For the autoencoder parameters, we set a high value of the maximum number of epochs $n_{epochs} = 10000$ to ensure that the neural network's loss converges, $m_{AE} = 100$ epochs and $\epsilon_{AE} = 10^{-4}$. During the training phase, we compute at each epoch the training loss and the validation loss, i.e. the **MSE** computed with the training data and the validation data, respectively. At the end of the training, the saved model corresponds to the one with the smallest validation loss. In the optimizer, the batch size is set to 32. Furthermore, when implementing the IncDecNeu-AE structure, the maximum number of neurons is set as $Neu_{max} = 80$ for $n_C = 5$ links and 300 for $n_C = 10$ links.

In the following, we study the influence of the learning rate of the autoencoder training phase, the number of layers in the encoder and decoder each, and the autoencoder architecture. We remind that the number of transmitted parameters without **DR** techniques is $p = 50$ for $n_C = 5$ links and $p = 200$ for $n_C = 10$ links.

4.4.2.1.1 Influence of the learning rate α

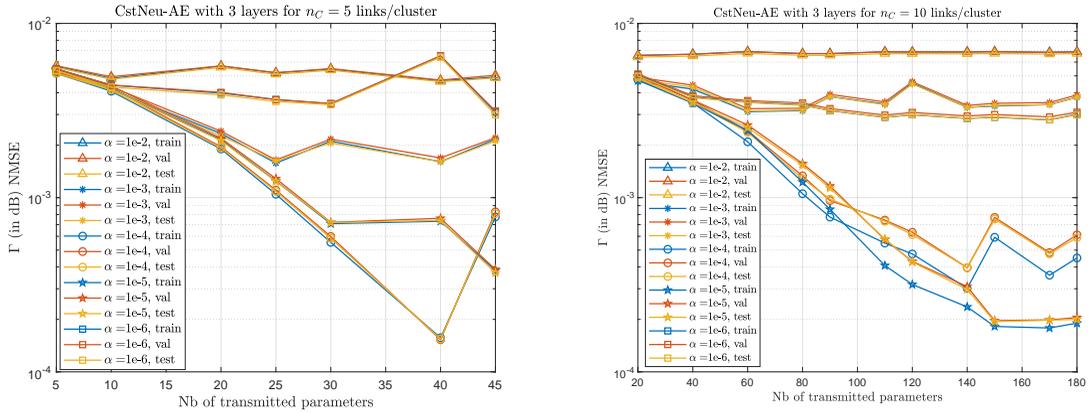


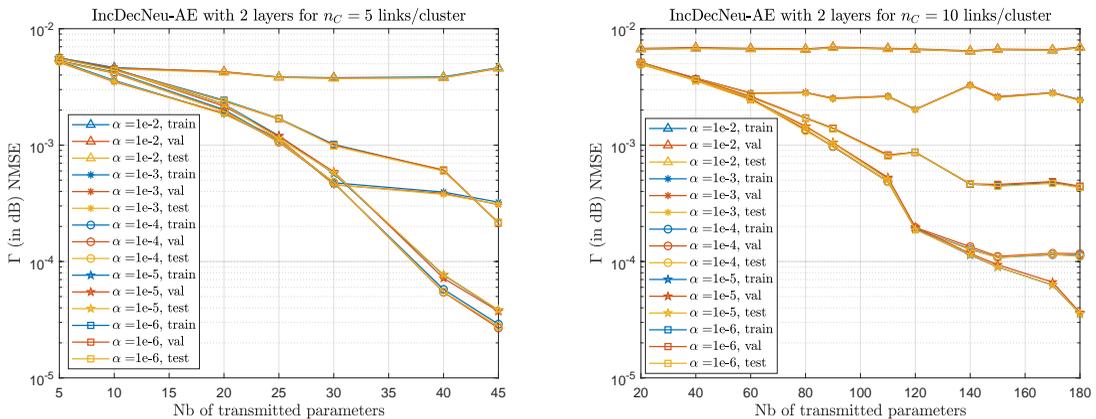
Figure 4.5: Study of the learning rate influence with CstNeu-AE structure with 3 layers. Without **DR** methods, the number of transmitted parameters is 50 for $n_C = 5$ and 200 for $n_C = 10$.

In Figs. 4.5a, 4.5b, 4.6a and 4.6b, we study the influence of the learning rate by plotting the autoencoder reconstruction **NMSE** applied to the training, validation and testing datasets, versus d , for $n_C = 5, 10$ links per cluster. Figs. 4.5a and 4.5b illustrate the performance of CstNeu-AE structure with 3 layers for $n_C = 5$ and 10 links, respectively, whereas Figs 4.6a and 4.6b display the case with IncDecNeu-AE structure with 2 layers for $n_C = 5$ and 10 links, respectively.

In Fig. 4.5a, we can see that the **NMSE** decreases between 5 and 25 for $\alpha = 10^{-3}$,

between 5 and 40 for $\alpha = 10^{-4}$, between 5 and 45 for $\alpha = 10^{-5}$ and between 5 and 30 for $\alpha = 10^{-6}$, while it is steady for $\alpha = 10^{-2}$. We can observe that the lowest **NMSE** is obtained for $\alpha = 10^{-4}, 10^{-5}$, while the highest are for $\alpha = 10^{-2}, 10^{-6}$. This can be explained by the fact that choosing a large learning rate that is too high results in convergence issues since it oscillates around the convergence point while selecting a low learning rate can lead to a slow convergence with a premature ending of the training process before reaching the desired convergence if it reaches the maximum number of epochs. Moreover, we can see that the **NMSE** is the lowest for $\alpha = 10^{-4}$ for d between 5 and 40, but is higher than the case $\alpha = 10^{-5}$ for $d = 45$. This outlier shows some convergence issues with the CstNeu-AE structure that we can also observe for the case 10^{-6} for $d = 40$. The reason why a very low learning rate such as $\alpha = 10^{-6}$ provides higher **NMSE** can be due to the loss function being trapped in a local minimum during the optimization. Besides, the **NMSE** corresponding to the training, validation and testing datasets are identical, which shows the absence of overfitting and underfitting of the autoencoder.

In Fig. 4.5b, we can see that the **NMSE** decreases between 20 and 60 for $\alpha = 10^{-3}$, between 20 and 140 for $\alpha = 10^{-4}$, between 20 and 150 for $\alpha = 10^{-5}$, and between 20 and 110 for $\alpha = 10^{-6}$, while it is steady for $\alpha = 10^{-2}$. The **NMSE** for $\alpha = 10^{-4}$ is the lowest for d between 20 and 90, whereas for $\alpha = 10^{-5}$, the **NMSE** is the lowest for $d \geq 110$. Furthermore, we can observe the presence of overfitting for $\alpha = 10^{-4}, 10^{-5}$ since the **NMSE** computed with the training dataset is lower than with the validation and testing datasets. This overfitting can be caused by the fact that the number of layers in the autoencoder is 3, which can be too high. However, we show in the next section that reducing the number of layers of the autoencoder with the CstNeu-AE structure provides similar or even lower performance in terms of **NMSE**.



(a) Case $n_C = 5$ links. The best is $\alpha = 10^{-4}$.

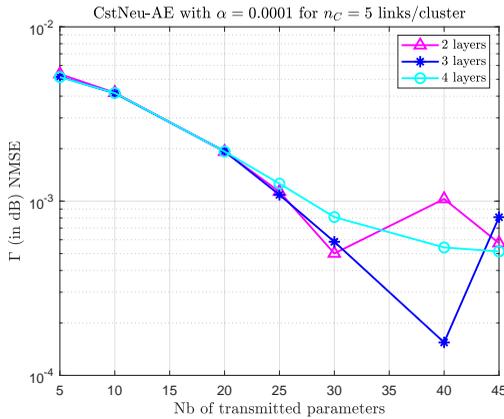
(b) Case $n_C = 10$ links. The best is $\alpha = 10^{-5}$.

Figure 4.6: Study of the learning rate influence with IncDecNeu-AE structure with 2 layers. Without **DR** methods, the number of transmitted parameters is 50 for $n_C = 5$ and 200 for $n_C = 10$.

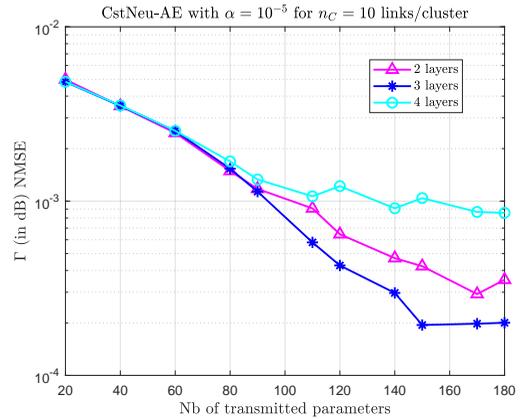
In Fig. 4.6a, we can see that the **NMSE** decreases between 5 and 45 for $\alpha = 10^{-3}, 10^{-4}, 10^{-5}$, or 10^{-6} , while it is steady for $\alpha = 10^{-2}$. The **NMSE** for $\alpha = 10^{-4}$ is the lowest when $d > 20$ and is similar to the others for $d \leq 20$, which makes it the best setting in this context. In Fig. 4.6b, we can see that the **NMSE** decreases between 20 and 60 for $\alpha = 10^{-3}$, between 20 and 150 for $\alpha = 10^{-4}$, between 20 and 180 for $\alpha = 10^{-5}$, and between 20 and 150 for $\alpha = 10^{-6}$, while it is steady for $\alpha = 10^{-2}$. The **NMSE** for $\alpha = 10^{-5}$ is the lowest when $d > 110$ and is similar or lower to the others for $d \leq 110$, which makes it the best setting in this context. Moreover, in both figures, the **NMSE** computed with the training, validation and testing datasets are similar. This can be explained by the fact that the autoencoder contains 2 layers, which is lower than in Figs. 4.5a and 4.5b and thus, decreases the overfitting. Moreover, unlike with CstNeu-AE structure, these figures do not present any outlier, which underlines that the IncDecNeu-AE structure is more stable than CstNeu-AE's.

4.4.2.1.2 Influence of the number of layers

In this section, we study the influence of the number of layers in the encoder and the decoder each for $n_C = 5, 10$ links and for both structures CstNeu-AE and IncDecNeu-AE. To this end, we plot in Figs 4.7a, 4.7b, 4.8a, and 4.8b the autoencoder reconstruction **NMSE** applied to the testing dataset, versus the number of transmitted parameters, which is d . The learning rate α is set with the best ones that we have previously determined.



(a) $n_C = 5$. The best are 2, 3 or 4 layers.



(b) $n_C = 10$. The best is 3 layers.

Figure 4.7: Study of the number of layers influence for CstNeu-AE structure with $\alpha = 10^{-4}$ for $n_C = 5$ and $\alpha = 10^{-5}$ for $n_C = 10$.

In Fig 4.7a, we can see that the **NMSE** decreases between 5 and 30 for 2 layers, between 5 and 40 for 3 layers, and between 5 and 45 for 4 layers. In this situation, choosing the right number of layers is hard since there is not any clear choice of a number of layers where the **NMSE** is always lower. Indeed, the lowest **NMSE** for $d = 30$ is reached with 2 layers, for $d = 40$, it is reached with 3 layers and for $d = 45$, with 4 layers. This can be

explained by the fact that increasing the number of layers in the autoencoder can make its training more stable, at the expense of higher overfitting as seen in the previous section. In Fig 4.7b, we can see that the NMSE decreases between 20 and 170 for 2 layers, between 20 and 150 for 3 layers, and between 20 and 110 for 4 layers. However, the choice of the number of layers is easier since the lowest NMSE is always reached for 3 layers.

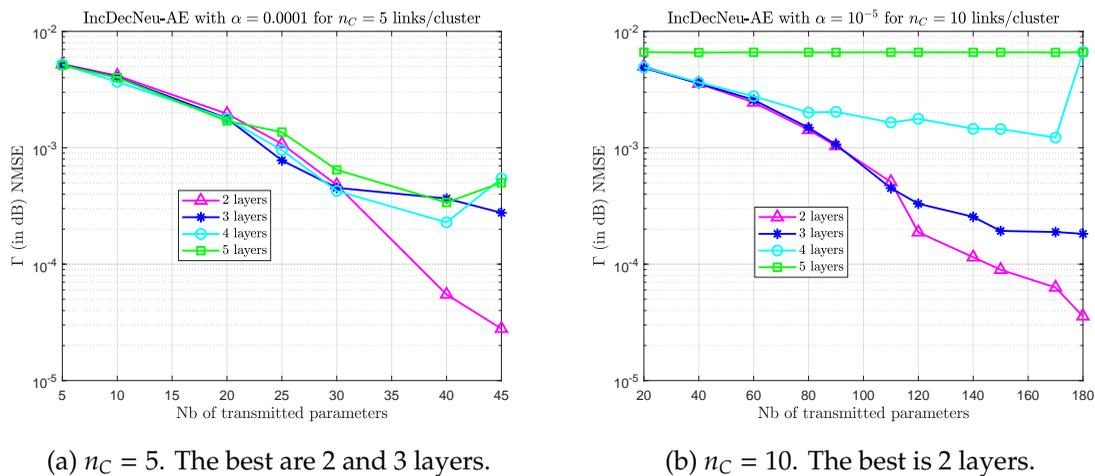


Figure 4.8: Study of the influence of the number of layers for IncDecNeu-AE structure with $\alpha = 10^{-4}$ for $n_C = 5$ and $\alpha = 10^{-5}$ for $n_C = 10$.

Regarding IncDecNeu-AE structure, Figs 4.8a and 4.8b show that we should choose an autoencoder with 2 layers since the NMSE for 2 layers is similar or lower than with the other number of layers, except for one point which is $d = 25$ for $n_C = 5$ and $d = 110$ for $n_C = 10$. This easier choice of a number of layers emphasizes the greater stability of the IncDecNeu-AE structure over CstNeu-AE's.

4.4.2.1.3 Influence of the structure of the autoencoder

In Figs 4.9a and 4.9b, we study the influence of the autoencoder structure by plotting the autoencoder reconstruction NMSE, versus the number of transmitted parameters, of three architectures:

- CstNeu-AE with 3 layers and $\alpha = 10^{-4}$.
- IncDecNeu-AE with 2 layers and $\alpha = 10^{-4}$.
- IncDecNeu-AE without ReLU with 2 layers and $\alpha = 10^{-5}$: this corresponds to the same structure as IncDecNeu-AE by removing the non-linear activation function ReLU.

It is worth mentioning that to compute IncDecNeu-AE without ReLU, we have chosen to set the best learning rate and number of layers. Besides, removing the non-linear

activation function in IncDecNeu-AE without ReLU structure is equivalent to applying a linear DR method.

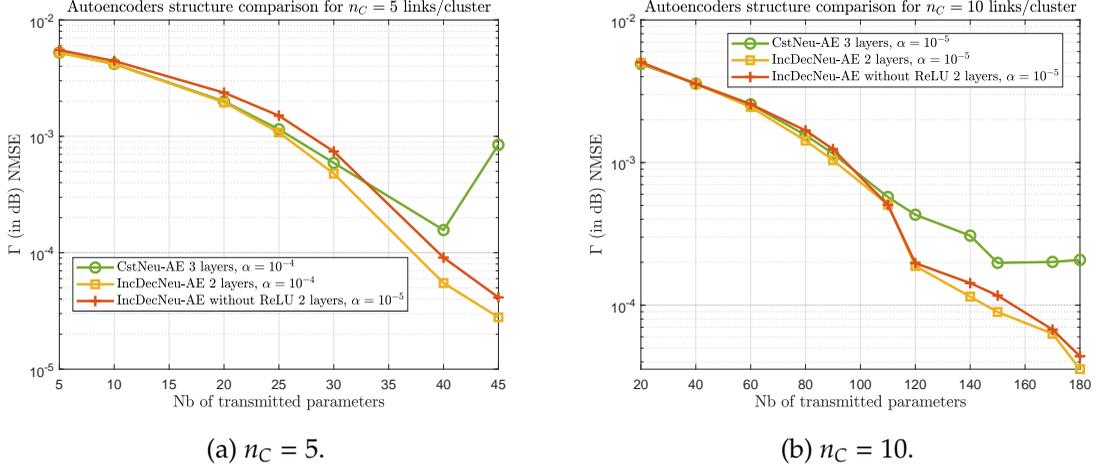


Figure 4.9: Study of the autoencoder structure for $n_C = 5$ and for $n_C = 10$. IncDecNeu-AE structure has the lowest NMSE.

Moreover, we can observe in the figures that IncDecNeu-AE shows the lowest NMSE for $n_C = 5$ and 10. Indeed, for $n_C = 5$, its NMSE is lower than IncDecNeu-AE without ReLU's by around $3 \cdot 10^{-4}$ for d between 20 and 30, and than CstNeu-AE by more than 10^{-4} for $d \geq 40$. For $n_C = 10$, while the gap between IncDecNeu-AE and CstNeu-AE is around $2 \cdot 10^{-4}$ for $d \geq 120$, the one between IncDecNeu-AE with and without ReLU is smaller and is around 10^{-5} for d between 80 and 90, and between 140 and 150. As a consequence, in addition to having shown previously its greater stability, IncDecNeu-AE shows better performance than the other structures in terms of NMSE.

4.4.2.2 Compression and DR methods performance comparison

4.4.2.2.1 Performance analysis in terms of NMSE

In this section, we assess the performance of compression and DR techniques in terms of NMSE, versus the number of transmitted parameters which is d . The implemented methods are the following:

- Compression with SVD: the input matrix Γ^{C_i} is decomposed using SVD into a product of two matrices which are both transmitted to the receiving cluster. It does not require any other data.
- DR with PCA described in Section 4.3.2.1: although it is the same as the DR method with SVD, it differs from the compression with SVD since the dimension reduction is applied using several training points.

- DR with KPCA described in Section 4.3.2.2 using the Gaussian kernel expressed in (4.22) with $\sigma = 0.01$.
- IncDecNeu-AE with 2 layers and $\alpha = 10^{-4}$.
- IncDecNeu-AE without ReLU with 2 layers and $\alpha = 10^{-5}$

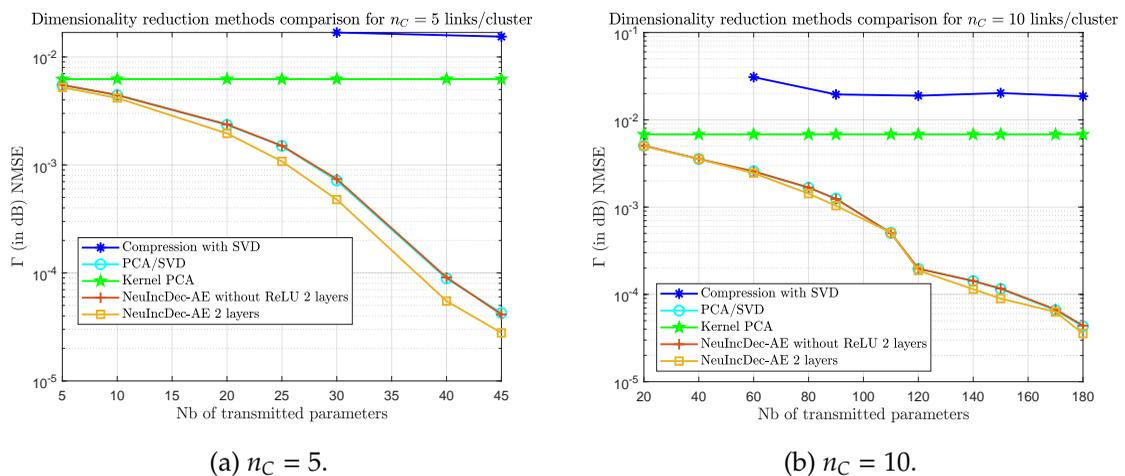


Figure 4.10: Comparison of compression and DR methods in terms of NMSE, versus the number of transmitted parameters. IncDecNeu-AE structure shows the lowest NMSE.

In Figs. 4.10a and 4.10b, we plot the reconstruction NMSE of various DR techniques for various number of transmitted parameters. We can observe that for decreasing compression, i.e. the number of transmitted parameters increases, every solution's NMSE decreases, except for KPCA whose NMSE is steady. This can be explained by the fact that the NMSE is dominated by KPCA's reconstruction error as explained in Section 4.3.2.2. Moreover, the compression with SVD presents higher NMSE than the other techniques. This result is expected since compression methods do not use any training data and have thus lower performance compared to DR schemes. Another expected outcome is the DR techniques with PCA and IncDecNeu-AE without ReLU having similar performance since they are both linear DR methods. Finally, IncDecNeu-AE presents the lowest NMSE compared to the others even though they can be close to the linear DR schemes' which can be much less numerically complex to implement with PCA or SVD.

4.4.2.2.2 Performance analysis by applying eTIM-hybrid in terms of average sum rate and minimum user rate per link

In this section, we assess the performance of the proposed compression and DR methods by using the estimated channel statistics matrix to compute the eTIM-hybrid scheme described previously in Section 2.4. Like in the numerical simulations from Section 2.4.4,

the average sum rates and minimum user rates are estimated with 10^5 Monte-Carlo simulations of channel realizations, and in the following figures, each of these values are averaged over 100 random topologies of links in the clusters.

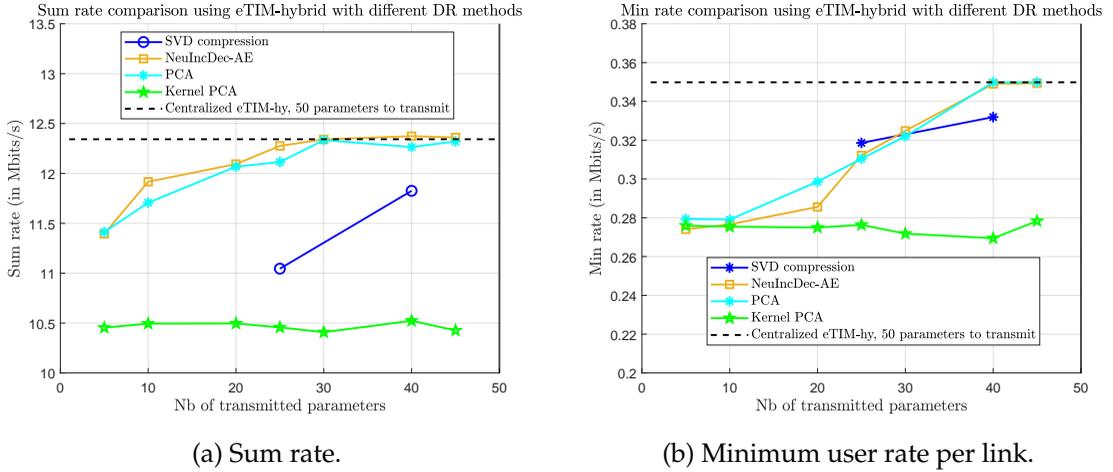
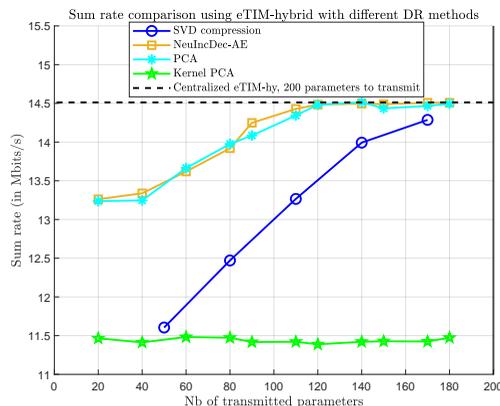


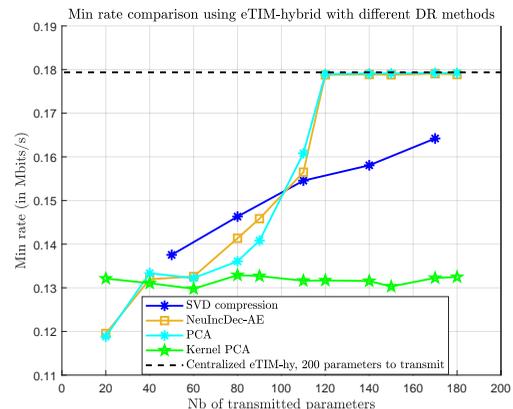
Figure 4.11: Performance comparison of eTIM-hybrid with other implemented methods versus the number of links per cluster for $n_C = 5$ links per cluster, $d_{\text{size}} = 10$ km and $\delta = 0$ m.

In Figs 4.11a and 4.11b, we respectively plot the average sum rate and minimum user rate per link for $n_C = 5$ links when using eTIM-hybrid with SVD compression, IncDecNeu-AE, PCA and KPCA, as long as with the channel statistics true values that we refer to as Centralized eTIM-hy. We can observe in Fig 4.11a that the performance of IncDecNeu-AE is much higher than SVD compression and KPCA in terms of average sum rate, but only slightly higher than PCA by around 0.1-0.2 Mbits/s regardless of the compression rate. Moreover, the average sum rate difference between IncDecNeu-AE and the centralized eTIM-hybrid is less than 0.1 Mbits/s for $d > 25$ parameters to transmit, which corresponds to a signaling reduction of 50%, whereas for PCA it is the case for around $d > 30$ which corresponds to 40% of compression rate. KPCA presents a lower average sum rate than even SVD compression, which is likely to be caused by the high error reconstruction from KPCA. In terms of average minimum user rate per link, IncDecNeu-AE and PCA minimum user rate per link are only higher than SVD compression's for $d \geq 30$ and is almost identical to the centralized method's one for $d \geq 40$, which corresponds to a signaling reduction of 20%. While IncDecNeu-AE and PCA minimum user rate per link are always similar except for $d = 20$, KPCA's is lower than the others.

In Figs 4.12a and 4.12b, we perform the same plots as previously but for $n_C = 10$. Like for $n_C = 5$ links, Fig 4.12a shows a lower average sum rate for SVD compression and KPCA than for IncDecNeu-AE and PCA, which are very close. The gap between IncDecNeu-AE and PCA average sum rate and centralized eTIM-hybrid is lower than



(a) Sum rate.



(b) Minimum user rate per link.

Figure 4.12: Performance comparison of eTIM-hybrid with other implemented methods versus the number of links per cluster for $n_C = 10$ links per cluster, $d_{\text{size}} = 10$ km and $\delta = 0$ m.

0.1 Mbits/s for $d \geq 110$, which corresponds to a compression rate of 45%. In terms of average minimum user rate, we can observe in Fig. 4.12b that IncDecNeu-AE and PCA's are higher than when using SVD compression for $d > 110$ and almost the same as centralized eTIM-hybrid's one for $d \geq 120$, which corresponds to a signaling reduction of 40%. KPCA has again a lower minimum user rate per link than the others.

Remark 4.4. We can observe that PCA has a close performance to the autoencoder's, which can be explained by the fact that in the simulations, the channel statistics, i.e. γ_{ij} , may be a "simple" function of the distance d_{ij} , since it is a deterministic fraction of polynomials. With more complex data, e.g. with random fluctuations or with shadowing, the performance gap between autoencoder and PCA can be higher.

4.5 Conclusion

In this chapter, we have first proposed a procedure reducing the channel statistics exchanges between clusters by compressing them using DR techniques. Specifically, we introduce a DR-based encoding and decoding procedure for channel statistics that can be applied to any link topology. Since many studies in the literature evaluate their DR solutions only using MSE, we performed numerical simulations to assess several DR methods using not only MSE but also other metrics such as the average sum rate and the minimum user rate per link. We computed these metrics using the previously proposed interference management scheme eTIM-hybrid. Both autoencoder and PCA achieve a high compression rate of up to 45-50%, with only a 1% loss in the average sum rate and minimum user rate per link compared to the centralized solution. While our results show

that Autoencoder performs slightly better than [PCA](#), it should be noted that [PCA](#) has the advantage of being much less complex.

Conclusion and perspectives

Summary of contributions

In this thesis, we addressed the problem of interference management in clustered ad hoc networks, assuming [SCSI](#) knowledge, a frequency-flat Rayleigh channel, and [SISO](#) systems. To achieve this objective, we have formulated three intermediate sub-problems:

1. Proposing an interference management solution with good performance in terms of sum rate and minimum rate per link that relies solely on [SCSI](#).
2. Developing a procedure for estimating the statistical [CSI](#).
3. Devising a framework that compresses the channel statistics between the [CH](#) to reduce the signaling exchanges between clusters.

In Chapter 2, we addressed the first sub-problem assuming we had two adjacent clusters that share the same frequency band because of a shortage of frequencies. We considered the [TIM](#) problem in a frequency-flat Rayleigh channel with statistical [CSI](#) knowledge using a hybrid approach combining [eTIM](#) and [TDMA](#) access, called [eTIM-hybrid](#). This scheme involves two steps: a grouping procedure based on rate criteria, with either [eTIM](#) or [TDMA](#) and a scheduling optimization for slot allocation, for which we presented the optimal closed-form solution. The performance of various interference management techniques were compared through numerical simulations in different cluster configurations. Our results showed that [eTIM-hybrid](#) outperforms conventional techniques such as [TDMA](#) in terms of average sum rate while guaranteeing an average minimum user rate for every link. We also extended the proposed solution to the case when there are two available [FBs](#) for both clusters and proposed a novel method, [JFA-eTIM-hy](#), that jointly optimizes the [FB](#) and [eTIM-hybrid](#). Simulations showed that [JFA-eTIM-hy](#) outperformed the standard orthogonal channel allocation solutions, even when combined with [eTIM-hybrid](#).

In Chapter 3, we tackled the second sub-problem by proposing two approaches, to reduce the sensing duration, for estimating the channel statistics of all pairs of nodes in the network, under the [TDD](#) mode assumption. In the first approach, we introduced a scheduling solution that enables two simultaneous transmissions, outperforming [TDMA](#)

in terms of sensing duration. Next, we formulated the channel statistics estimation problem for each receiver which consists of estimating the average power of the direct and interfering channels, under a single interference and using a set of SINR samples. We then introduced two new estimators based on the MLE and the MoM, implemented using Newton's method, for which we proposed initialization procedures to ensure the convergence of both algorithms. Simulation results indicate that while the MLE variance is close to the CRLB, the MoM estimator has a slightly higher variance but is less complex than the MLE. Additionally, we demonstrated that the MLE can be initialized with the MoM's estimates, which is less complex than random initialization. In the second approach, we presented an alternative scheduling method that enables simultaneous transmissions to estimate the channel statistics of all pairs of nodes in the network. While this scheduling solution offers improved performance over the first approach in terms of sensing duration, it is only feasible when orthogonal sequences can be utilized to eliminate interference. Our analysis demonstrates that the divide-in-2, 2-transmission and mixed scheduling solutions outperform TDMA in terms of the scheduling length, and we provided closed-form expressions for their respective lengths. However, we also note that the energy consumption of these proposed schemes is higher than that of TDMA, despite their reduced sensing duration.

In Chapter 4, we dealt with the third sub-problem by reducing the dimensionality of the channel statistics, thus decreasing the signaling exchanges between the CHs. To this end, we proposed two methods: a linear DR scheme and an autoencoder-based approach that both encode and decode channel statistics for any links topology. Numerical simulations evaluated these solutions in terms of NMSE, average sum rate, and minimum user rate per link using the previously proposed interference management method eTIM-hybrid. Results showed that compressing channel statistics by up to 45%, using the linear DR and autoencoder methods, achieved at least a 99% average sum rate of the centralized performance, while guaranteeing a minimum user rate per link. Although the autoencoder method performed slightly better, the linear DR has the advantage of being less complex.

Perspectives

To conclude, we will share some possible future directions of research based on the work carried out in this thesis.

System design

1. As future directions, it would be interesting to extend the current work to include different types of propagation channels, such as Rician or Nakagami-m, as well as frequency-selective and time-correlated channels, to assess the performance of the proposed techniques under these scenarios.

2. Another possible extension is to explore the use of **MIMO** systems instead of **SISO**, which can potentially increase the system's capacity and robustness to interference.

Enhanced topological interference management

1. Possible future directions include extending the proposed interference management solutions to more clusters and frequency bands by combining them with coloring algorithms, e.g. **TEL**, to improve channel allocation in ad hoc networks.
2. Although our system model assumed a predefined set of communication links, in practice, nodes can communicate with any other nodes in the cluster. One possible future work is to extend the proposed **eTIM** solution that can handle varying communication links between deployed nodes, where the connections between nodes may change depending on the traffic.
3. Moreover, to speed up the computation of the proposed **eTIM** solution, it would be worth exploring other optimization techniques, instead of the **AO** which is suboptimal, such as geometric programming.
4. Additionally, the TIM model can be improved to accommodate mobile nodes, by devising an online solution that has faster computation, as shown in [28]. Alternatively, an offline training approach can also be used with neural networks to design coders **U** and **V** for any node and link topology. However, the primary challenge in this case would be to consider the variable size of output neural networks since the length of the coders r also needs to be optimized.
5. Another possibility is to extend the interference management problem by, for instance, jointly optimizing interference management and power allocation, or modifying the objective function to incorporate a fairness metric.

Channel statistics estimation

1. In the estimation problem formulated from the first scheduling approach, one possible extension of the first scheduling approach is to handle an arbitrary number of interferers, as it is currently designed for a single interference scenario.
2. In the first approach, another perspective is to explore the use of neural network techniques to avoid prior knowledge on the channel.

Decreasing signaling exchanges using dimensionality reduction

1. Other compression schemes than the linear **DR** and the autoencoder can be considered, such as uniform manifold approximation and projection (**UMAP**), which has shown promising results in reducing the dimensionality of high-dimensional data.
-

Another possibility is to investigate the use of quantization with neural networks, as has been done in recent research [22].

2. Additionally, it would be interesting to train a model that considers a varying number of links in the network since in practice, some links may go off while new links may appear.
-

Appendix A

Appendix related to Chapter 2

A.1 Derivative expressions in Section 2.3.2

In this Appendix, we derive the necessary partial derivative expression to compute gradient descent. Most of the calculations are inspired from [113].

A.1.1 Partial derivation expression of $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ for Problem 2.5

In this part, we derive the partial derivatives of $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ with respect to $\tilde{\mathbf{u}}_k$. We remind that we have $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}} = \sum_{i=1}^N \bar{R}_{\mathbf{V},i}^{\text{hard}}$ with $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ defined in (2.27). After some manipulations, $\bar{R}_{\mathbf{V},i}^{\text{hard}}$ can be expressed as:

$$\bar{R}_{\mathbf{V},i}^{\text{eTIM}} = \frac{1}{\ln 2} (\ln \Phi_{\mathbf{V},i} - \ln \Pi_{\mathbf{V},i}), \quad (\text{A.1})$$

where $\Phi_{\mathbf{V},i}$ and $\Pi_{\mathbf{V},i}$ are defined as the following:

$$\begin{aligned} \Phi_{\mathbf{V},i} &:= \sum_{j=1}^N \beta_{\mathbf{V},ij} (\tilde{\mathbf{u}}_i \mathbf{v}_j^T)^2 + P_n \\ \Pi_{\mathbf{V},i} &:= \sum_{\substack{j=1 \\ j \neq i}}^N \beta_{\mathbf{V},ij} (\tilde{\mathbf{u}}_i \mathbf{v}_j^T)^2 + P_n. \end{aligned} \quad (\text{A.2})$$

Besides, we use the linearity of the derivation to express $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ derivative as follows:

$$\frac{\partial \bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}(\tilde{\mathbf{U}})}{\partial \tilde{\mathbf{u}}_k} = \sum_{i=1}^N \frac{\partial \bar{R}_{\mathbf{V},i}^{\text{hard}}(\tilde{\mathbf{U}})}{\partial \tilde{\mathbf{u}}_k} \quad (\text{A.3})$$

We define $\delta_{i,k}$ as the Kronecker symbol which can be expressed as:

$$\delta_{i,k} = \begin{cases} 1, & \text{if } i = k, \\ 0, & \text{else.} \end{cases} \quad (\text{A.4})$$

Then, we compute the partial derivative of $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ expression as:

$$\begin{aligned}
\frac{\partial \bar{R}_{\mathbf{V},i}^{\text{hard}}(\tilde{\mathbf{U}})}{\partial \tilde{\mathbf{u}}_k} &= \frac{1}{\ln 2} \left(\frac{\partial \ln \Phi_{\mathbf{V},i}}{\partial \tilde{\mathbf{u}}_k} - \frac{\partial \ln \Pi_{\mathbf{V},i}}{\partial \tilde{\mathbf{u}}_k} \right) \\
&= \frac{\delta_{i,k}}{\ln 2} \left[\Phi_{\mathbf{V},k}^{-1} \sum_{j=1}^N \left(2\beta_{\mathbf{V},kj}(\tilde{\mathbf{u}}_k \mathbf{v}_j^T) \mathbf{v}_j - \frac{2\beta_{\mathbf{V},kj}(\tilde{\mathbf{u}}_k \mathbf{v}_j^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \tilde{\mathbf{u}}_k \right) \right. \\
&\quad \left. - \Pi_{\mathbf{V},k}^{-1} \sum_{\substack{j=1 \\ j \neq k}}^N \left(2\beta_{\mathbf{V},kj}(\tilde{\mathbf{u}}_k \mathbf{v}_j^T) \mathbf{v}_j - \frac{2\beta_{\mathbf{V},kj}(\tilde{\mathbf{u}}_k \mathbf{v}_j^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \tilde{\mathbf{u}}_k \right) \right] \\
&= \frac{2\delta_{i,k}}{\ln 2} \left[(\Phi_{\mathbf{V},k}^{-1} - \Pi_{\mathbf{V},k}^{-1}) \sum_{j=1}^N \beta_{\mathbf{V},kj} \left(\mathbf{v}_j^T \mathbf{v}_j - \frac{(\tilde{\mathbf{u}}_k \mathbf{v}_j^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \mathbf{I}_r \right) \right. \\
&\quad \left. + \Pi_{\mathbf{V},k}^{-1} \beta_{\mathbf{V},kk} \left(\mathbf{v}_k^T \mathbf{v}_k - \frac{(\tilde{\mathbf{u}}_k \mathbf{v}_k^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \mathbf{I}_r \right) \right] \tilde{\mathbf{u}}_k,
\end{aligned} \tag{A.5}$$

with $\Phi_{\mathbf{V},k}$ and $\Pi_{\mathbf{V},k}$ defined in (A.2).

Finally, by injecting (A.5) into (A.3), we obtain the partial derivative of $\bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}$ which can be expressed as:

$$\begin{aligned}
\frac{\partial \bar{R}_{\text{sum},\mathbf{V}}^{\text{hard}}(\tilde{\mathbf{U}})}{\partial \tilde{\mathbf{u}}_k} &= \frac{2}{\ln 2} \left[(\Phi_{\mathbf{V},k}^{-1} - \Pi_{\mathbf{V},k}^{-1}) \sum_{j=1}^N \beta_{\mathbf{V},kj} \left(\mathbf{v}_j^T \mathbf{v}_j - \frac{(\tilde{\mathbf{u}}_k \mathbf{v}_j^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \mathbf{I}_r \right) \right. \\
&\quad \left. + \Pi_{\mathbf{V},k}^{-1} \beta_{\mathbf{V},kk} \left(\mathbf{v}_k^T \mathbf{v}_k - \frac{(\tilde{\mathbf{u}}_k \mathbf{v}_k^T)^2}{\|\tilde{\mathbf{u}}_k\|_2^2} \mathbf{I}_r \right) \right] \tilde{\mathbf{u}}_k,
\end{aligned} \tag{A.6}$$

A.1.2 Partial derivation expression of $\bar{R}_{\text{sum},\mathbf{U}}^{\text{hard}}$ for Problem 2.6

In this section, we derive the partial derivatives of $\bar{R}_{\text{sum},\mathbf{U}}^{\text{hard}}$ where $\bar{R}_{\text{sum},\mathbf{U}}^{\text{hard}} = \sum_{i=1}^N \bar{R}_{\mathbf{U},i}^{\text{hard}}$ and $\bar{R}_{\mathbf{U},i}^{\text{hard}}$ is defined in (2.29) with respect to $\tilde{\mathbf{v}}_k$. Using similar derivations as in the previous section, $\bar{R}_{\mathbf{U},i}^{\text{hard}}$ can be expressed as:

$$\bar{R}_{\mathbf{U},i}^{\text{hard}} = \frac{1}{\ln 2} (\ln \Phi_{\mathbf{U},i} - \ln \Pi_{\mathbf{U},i}), \tag{A.7}$$

with $\Phi_{\mathbf{U},i}$ and $\Pi_{\mathbf{U},i}$ defined as the following:

$$\begin{aligned}
\Phi_{\mathbf{U},k} &:= \sum_{j=1}^N \beta_{\mathbf{U},kj} (\mathbf{u}_k \tilde{\mathbf{v}}_j^T)^2 + P_n \|\mathbf{u}_k\|_2^2 \\
\Pi_{\mathbf{U},k} &:= \sum_{\substack{j=1 \\ j \neq k}}^N \beta_{\mathbf{U},kj} (\mathbf{u}_k \tilde{\mathbf{v}}_j^T)^2 + P_n \|\mathbf{u}_k\|_2^2
\end{aligned} \tag{A.8}$$

Then, the partial derivative of $\bar{R}_{U,i}^{\text{eTIM}}$ is given by

$$\begin{aligned} \frac{\partial \bar{R}_{U,i}^{\text{hard}}(\tilde{\mathbf{V}})}{\partial \tilde{\mathbf{v}}_k} &= \frac{1}{\ln 2} \left(\frac{\partial \ln \Phi_{U,i}}{\partial \tilde{\mathbf{v}}_k} - \frac{\partial \ln \Pi_{U,i}}{\partial \tilde{\mathbf{v}}_k} \right) \\ &= \frac{1}{\ln 2} \left[\Phi_{U,i}^{-1} \left(2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T) \mathbf{u}_i - \frac{2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \tilde{\mathbf{v}}_k \right) \right. \\ &\quad \left. - (1 - \delta_{i,k}) \Pi_{U,i}^{-1} \left(2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T) \mathbf{u}_i - \frac{2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \tilde{\mathbf{v}}_k \right) \right], \end{aligned} \quad (\text{A.9})$$

with $\Phi_{U,k}$ and $\Pi_{U,k}$ defined in (A.8). Thus, using the linearity of the derivation, the partial derivative of $\bar{R}_{\text{sum,U}}^{\text{eTIM}}$ with respect to $\tilde{\mathbf{v}}_k$ can be expressed as:

$$\begin{aligned} \frac{\partial \bar{R}_{\text{sum,U}}^{\text{hard}}(\tilde{\mathbf{V}})}{\partial \tilde{\mathbf{v}}_k} &= \frac{1}{\ln 2} \left[\sum_{i=1}^N \Phi_{U,i}^{-1} \left(2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T) \mathbf{u}_i - \frac{2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \tilde{\mathbf{v}}_k \right) \right. \\ &\quad \left. - \sum_{\substack{i=1 \\ i \neq k}}^N \Pi_{U,i}^{-1} \left(2\beta_{U,ik}(\tilde{\mathbf{u}}_k \mathbf{v}_i^T) \mathbf{u}_i - \frac{2\beta_{U,ik}(\mathbf{u}_i \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \tilde{\mathbf{v}}_k \right) \right] \end{aligned} \quad (\text{A.10})$$

In the end, its final expression can be written as:

$$\begin{aligned} \frac{\partial \bar{R}_{\text{sum,U}}^{\text{hard}}(\tilde{\mathbf{V}})}{\partial \tilde{\mathbf{v}}_k} &= \frac{2}{\ln 2} \left[\sum_{i=1}^N \beta_{U,ik} (\Phi_{U,i}^{-1} - \Pi_{U,i}^{-1}) \left(\mathbf{u}_i^T \mathbf{u}_i - \frac{(\mathbf{u}_i \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \mathbf{I}_r \right) \right. \\ &\quad \left. + \Pi_{U,k}^{-1} \beta_{U,kk} \left(\mathbf{u}_k^T \mathbf{u}_k - \frac{(\mathbf{u}_k \tilde{\mathbf{v}}_k^T)^2}{\|\tilde{\mathbf{v}}_k\|_2^2} \mathbf{I}_r \right) \right] \tilde{\mathbf{v}}_k. \end{aligned} \quad (\text{A.11})$$

A.2 Implementation of eTIM max sum rate with minimum rate constraint

A.2.1 AO procedure

Similarly to Problem 2.2, we solve Problem 2.8 by using an AO framework combined with gradient descent methods. Moreover, we apply the same change of variables as for Problem 2.3 to remove the norm constraint on $\mathbf{u}_i, \mathbf{v}_i$. Hence, to optimize Problem 2.8, we execute at iteration t the following procedure:

First, we formulate the optimization problem alternately between the decoders, which can be written for all $k \in \{1, \dots, N\}$ as:

Problem A.1.

$$\tilde{\mathbf{u}}_k^{(t)} = \arg \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum,V}}^{\text{hard}}(\tilde{\mathbf{u}}_1^{(t)}, \dots, \tilde{\mathbf{u}}_{k-1}^{(t)}, \tilde{\mathbf{u}}_k, \tilde{\mathbf{u}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{u}}_N^{(t-1)}) \quad (\text{A.12})$$

$$\text{s.t.} \quad \frac{\bar{R}_{\mathbf{V},k}^{\text{hard}}}{r} \geq R_0^u. \quad (\text{A.13})$$

with fixed $\tilde{\mathbf{u}}_1^{(t)}, \dots, \tilde{\mathbf{u}}_{k-1}^{(t)}, \tilde{\mathbf{u}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{u}}_N^{(t-1)}, \tilde{\mathbf{v}}_1^{(t-1)}, \dots, \tilde{\mathbf{v}}_N^{(t-1)}$.

Then, we formulate the optimization problem alternately between the precoders, which can be expressed for all $k \in \{1, \dots, N\}$ as:

Problem A.2.

$$\tilde{\mathbf{v}}_k^{(t)} = \arg \max_{\tilde{\mathbf{v}}_k \in \mathbb{R}^{1 \times r}} \bar{R}_{\text{sum}, \mathbf{U}}^{\text{hard}}(\tilde{\mathbf{v}}_1^{(t)}, \dots, \tilde{\mathbf{v}}_{k-1}^{(t)}, \tilde{\mathbf{v}}_k, \tilde{\mathbf{v}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{v}}_N^{(t-1)}) \quad (\text{A.14})$$

$$\text{s.t.} \quad \frac{\bar{R}_{\mathbf{U}, i}^{\text{hard}}}{r} \geq R_0^u, \quad \forall i \in \{1, \dots, N\}, \quad (\text{A.15})$$

with fixed $\tilde{\mathbf{u}}_1^{(t)}, \dots, \tilde{\mathbf{u}}_N^{(t)}, \tilde{\mathbf{v}}_1^{(t)}, \dots, \tilde{\mathbf{v}}_{k-1}^{(t)}, \tilde{\mathbf{v}}_{k+1}^{(t-1)}, \dots, \tilde{\mathbf{v}}_N^{(t-1)}$, and with $\bar{R}_{\mathbf{V}, k}^{\text{hard}}$ and $\bar{R}_{\mathbf{U}, i}^{\text{hard}}$ defined in (2.27) and (2.29), respectively.

A.2.2 Gradient descent and subgradient method

Since Problems A.1 and A.2 are constrained optimization problems, we optimize them by considering their associated Lagrangian function. The Lagrangian multipliers are optimized using the subgradient method [110].

A.2.2.1 Problem A.1 resolution

We first consider Problem A.1, we define its associated Lagrangian function and the Lagrangian derivative function which can be written as:

$$\mathcal{L}_{\mathbf{V}, k}(\tilde{\mathbf{u}}_k, \lambda_k) := \bar{R}_{\text{sum}, \mathbf{V}}^{\text{hard}}(\tilde{\mathbf{u}}_k) - \lambda_k (rR_0^u - \bar{R}_{\mathbf{V}, k}^{\text{hard}}(\tilde{\mathbf{u}}_k)) \quad (\text{A.16a})$$

$$\frac{\partial \mathcal{L}_{\mathbf{V}, k}(\tilde{\mathbf{u}}_k, \lambda_k)}{\partial \tilde{\mathbf{u}}_k} = (1 + \lambda_k) \frac{\partial \bar{R}_{\mathbf{V}, k}^{\text{hard}}(\tilde{\mathbf{u}}_k)}{\partial \tilde{\mathbf{u}}_k}, \quad (\text{A.16b})$$

with the Lagrange multiplier $\lambda_k \geq 0$ and $\partial \bar{R}_{\mathbf{V}, k}^{\text{hard}} / \partial \tilde{\mathbf{u}}_k$ defined in (A.5).

Then, Problem A.1's equivalent optimization problem can be formulated as maximizing its Lagrangian function:

Problem A.3.

$$\tilde{\mathbf{u}}_k^{(t)(m)} = \arg \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}_{\mathbf{V}, k}(\tilde{\mathbf{u}}_k, \lambda_k^{(m)}), \quad (\text{A.17})$$

which we solve with a gradient descent algorithm using (A.16b).

Then, the Lagrangian multiplier λ_k is optimized using the subgradient method. To this end, we consider the following function that we want to maximize:

$$g_{\mathbf{V}}(\lambda_k) := \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}_{\mathbf{V}, k}(\tilde{\mathbf{u}}_k, \lambda_k). \quad (\text{A.18})$$

Since $g_{\mathbf{V}}$ is non-differentiable, we need to compute its subgradient function instead of its gradient function to maximize it. Moreover, when optimizing the Lagrangian multipliers with the subgradient method, it can be proved that the constraint has the same direction as

the subgradient function [110]. Therefore, instead of computing $g_{\mathbf{v}}$ subgradient function, we can maximize $g_{\mathbf{v}}$ by updating λ at iteration m as the following:

$$\lambda_k^{(m+1)} = \left[\lambda_k^{(m)} + \alpha_k^{(m)} (rR_0^u - \bar{R}_{\mathbf{v},k}^{\text{hard}}(\tilde{\mathbf{u}}_k^{(t)(m)})) \right]^+, \quad (\text{A.19})$$

with the step size at iteration m set as $\alpha_k^{(m)} = \frac{1}{m+1}$ and $(\cdot)^+$ defined as $(x)^+ = \max(0, x)$. This operator is used to guarantee the positivity of the Lagrangian multipliers.

In summary, the constraint optimization is described in Algorithm A.1 and is provided by repeating the following two-stage procedure until the constraint $\bar{R}_{\mathbf{v},k}^{\text{hard}} \geq rR_0^u$ is respected:

1. we use a gradient descent algorithm on Problem A.3 using (A.16b) until convergence.
2. we update the Lagrangian multiplier λ_k using (A.19).

Since there are cases when the constraints cannot be verified, this procedure is not guaranteed to converge toward the optimal solution.

Algorithm A.1: Subgradient method for solving Problem A.1

Solve $\tilde{\mathbf{u}}_k^{(t)(0)} = \arg \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}(\tilde{\mathbf{u}}_k, \lambda_k^{(0)})$ using gradient descent in Algorithm 2.2, with $\mathcal{L}(\tilde{\mathbf{u}}_k, \lambda_k^{(0)})$ defined in (A.16).

while $(rR_0^u - \bar{R}_{\mathbf{v},k}^{\text{eTIM}}) > 0$ **do**

Solve $\tilde{\mathbf{u}}_k^{(t)(m+1)} = \arg \max_{\tilde{\mathbf{u}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}(\tilde{\mathbf{u}}_k, \lambda_k^{(m)})$ using gradient descent in Algorithm 2.2.

$\lambda_k^{(m+1)} = \lambda_k^{(m)} + \alpha^{(m)} (rR_0^u - \bar{R}_{\mathbf{v},k}^{\text{eTIM}})$.

Increment $m = m + 1$

end

A.2.2.2 Problem A.2 resolution

Similarly to Problem A.1, Problem A.2 is optimized by maximizing its associated Lagrangian function and the Lagrangian multipliers are optimized using the subgradient method. The Lagrangian function and its derivative expressions can be written as:

$$\mathcal{L}_{\mathbf{U},k}(\tilde{\mathbf{v}}_k, \boldsymbol{\mu}) := \bar{R}_{\text{sum},\mathbf{U}}^{\text{hard}}(\tilde{\mathbf{v}}_k) - \sum_{i=1}^N \mu_i (R_0^u - \bar{R}_{\mathbf{U},i}^{\text{hard}}(\tilde{\mathbf{v}}_k)), \quad (\text{A.20a})$$

$$\frac{\partial \mathcal{L}_{\mathbf{U},k}(\tilde{\mathbf{v}}_k, \boldsymbol{\mu})}{\partial \tilde{\mathbf{v}}_k} = \sum_{i=1}^N (1 + \mu_i) \frac{\partial \bar{R}_{\mathbf{U},i}^{\text{eTIM}}(\tilde{\mathbf{v}}_k)}{\partial \tilde{\mathbf{v}}_k}, \quad (\text{A.20b})$$

with the Lagrange multipliers $\forall i \in \{1, \dots, N\}, \mu_i \geq 0$ and $\partial \bar{R}_{\mathbf{U},k}^{\text{eTIM}} / \partial \tilde{\mathbf{v}}_k$ defined in (A.9).

Using the Lagrangian function, Problem A.1's equivalent optimization problem can be formulated as:

Problem A.4.

$$\tilde{\mathbf{v}}_k^{(t)(m)} = \arg \max_{\tilde{\mathbf{v}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}_{U,k}(\tilde{\mathbf{v}}_k, \boldsymbol{\mu}_k^{(m)}), \quad (\text{A.21})$$

Then, the Lagrangian multipliers $\boldsymbol{\mu}_k$ are optimized using the subgradient method. To this end, we consider the following function that we want to maximize:

$$g_U(\boldsymbol{\mu}_k) := \max_{\tilde{\mathbf{v}}_k \in \mathbb{R}^{1 \times r}} \mathcal{L}_{U,k}(\tilde{\mathbf{v}}_k, \boldsymbol{\mu}_k). \quad (\text{A.22})$$

Since g_U is non-differentiable, its subgradient is required to maximize it. As said in Section 2.3.3.2, it can be proved that the constraint $(R_0^u - \bar{R}_{U,i}^{\text{eTIM}}(\tilde{\mathbf{v}}_k^{(t)(m)}))$ is in the same direction as g_U 's subgradient function, we can maximize g_U by updating λ at iteration m as:

$$\mu_k^{(m+1)}(i) = [\mu_k^{(m)}(i) + \alpha_k^{(m)}(R_0^u - \bar{R}_{U,i}^{\text{eTIM}}(\tilde{\mathbf{v}}_k^{(t)(m)}))]^+, \quad \forall i \in \{1, \dots, N\}, \quad (\text{A.23})$$

with the step size at iteration m set as $\alpha_k^{(m)} = \frac{1}{m+1}$.

In summary, the constraint optimization is described in the following two-stage procedure whose implementation can be obtained by adapting Algorithm A.1:

1. we use a gradient descent algorithm on Problem A.4 using (A.20b) until convergence.
2. we update the Lagrangian multipliers $\boldsymbol{\mu}_k$ using (A.23).

Stages 1 and 2 are repeated until the constraint $\bar{R}_{U,k}^{\text{eTIM}} \geq rR_0^u$ is respected.

A.3 Scheduling optimization proofs in Section 2.4.2

A.3.1 Proof of Result 2.2

Proof. We want to show that Problem 2.11 is feasible if and only if we have the following inequality:

$$\sum_{k \in \mathcal{T}_d} n_k^{\min} + n_{\text{eTIM}}^{\min} r \leq L, \quad (\text{A.24})$$

with $n_k^{\min}, n_{\text{eTIM}}^{\min}$ defined in (2.58).

Before proving this equivalence, we start by simplifying the constraints of Problem 2.11 into a simpler expression to manipulate, we remind that these constraints can be written as:

$$\bar{R}_{\text{TDMA},k}^u \geq R_0^u, \quad \forall k \in \mathcal{T}_d, \quad (\text{A.25})$$

$$\bar{R}_{\text{eTIM},\min}^u \geq R_0^u, \quad (\text{A.26})$$

$$\sum_{k \in \mathcal{T}_d} n_k + n_{\text{eTIM}} r = L. \quad (\text{A.27})$$

following:

$$\begin{cases} n_k = n_k^{\min} & \forall k \in \mathcal{T}_d \setminus \{k_0\}, & (\text{A.32a}) \\ n_{k_0} = n_{k_0}^{\min} - L_{\text{res}}, & & (\text{A.32b}) \\ n_{\text{eTIM}} = n_{\text{eTIM}}^{\min} & & (\text{A.32c}) \end{cases}$$

which satisfies constraint (A.31c). □

A.3.2 Proof of Result 2.3

Proof. In this proof, we provide the optimal solution of Problem 2.11. This solution is composed of two steps: i) a step where the slots are allocated so that each link satisfies its constraint, and ii) a greedy step where the remaining slots are allocated to the best links.

1. Asserting Problem 2.11's feasibility: we first allocate the required slots, i.e. n_{eTIM}^{\min} and n_k^{\min} for all $k \in \mathcal{T}_d$ using (2.58a) and (2.58b), to ensure that the constraints are satisfied.
2. Greedy allocation: once the required slots to guarantee the constraints are allocated, we proceed to a greedy allocation, i.e. we allocate the slots to the links with the highest rates. To maximize the average sum rate, we allocate the L_{res} remaining slots, with L_{res} defined in (2.60), to either k_{max} in \mathcal{T}_d or the links in \mathcal{T}_e which apply MSR-eTIM. To this end, we compare the sum of average physical rates on r slots restrained to \mathcal{T}_e , namely $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi$, and the highest possible sum of physical rates on r slots in \mathcal{T}_d , which is $r\bar{R}_{k_{\text{max}}}^\varphi$.

- For case $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\text{max}}}^\varphi$, allocating slots to the links in \mathcal{T}_e would provide a lower gain than allocating slots to k_{max} . Therefore, we allocate the L_{res} remaining slots to link k_{max} in \mathcal{T}_d , which can be expressed as:

$$n_{k_{\text{max}}} = n_{k_{\text{max}}}^{\min} + L_{\text{res}}. \quad (\text{A.33})$$

- For case $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{\text{max}}}^\varphi$, we allocate the remaining slots to \mathcal{T}_e . Since each repetition of MSR-eTIM lasts r slots, the new allocation for the links in \mathcal{T}_e is:

$$n_{\text{eTIM}} = n_{\text{eTIM}}^{\min} + \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor. \quad (\text{A.34})$$

Since some slots can remain (strictly less than r), we allocate the rest of them to link k_{max} and its allocation can be written as:

$$n_{k_{\text{max}}} = n_{k_{\text{max}}}^{\min} + L_{\text{res}} - \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor r. \quad (\text{A.35})$$

Therefore, the proof for the optimal allocation is concluded. □

A.3.3 Derivations leading to (2.65)

Since Result 2.3 provides the optimal solutions $n_k^*, n_{e\text{TIM}}^*$ as a function of L , we can express $\bar{R}_{\text{sum}}^{\text{eTIM-hy}}$ using (2.49) with the optimal transmissions allocation to have a function depending on L only. We call it $R_{\text{sum,opt}}^{\text{eTIM-hy}}$ and express it as:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) := \frac{n_{e\text{TIM}}^*}{L} \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi + \sum_{k \in \mathcal{T}_d} \frac{n_k^*}{L} \bar{R}_k^\varphi. \quad (\text{A.36})$$

Since $n_k^*, n_{e\text{TIM}}^*$ depends on the cases whether $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi$ is lower or higher than $r\bar{R}_{k_{\max}}^\varphi$, we calculate $R_{\text{sum,opt}}^{\text{eTIM-hy}}$ for both cases.

Case 1: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$. Plugging (2.61) into (A.36), we obtain:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = \frac{1}{L} \left(n_{e\text{TIM}}^{\min} \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi + \sum_{k \in \mathcal{T}_d \setminus \{k_{\max}\}} n_k^{\min} \bar{R}_k^\varphi + (n_{k_{\max}}^{\min} + L_{\text{res}}) \bar{R}_{k_{\max}}^\varphi \right) \quad (\text{A.37})$$

Plugging (2.60) into (A.37), we can obtain:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = \frac{1}{L} \left(n_{e\text{TIM}}^{\min} \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi + \sum_{k \in \mathcal{T}_d \setminus \{k_{\max}\}} n_k^{\min} \bar{R}_k^\varphi + \left(n_{k_{\max}}^{\min} + L - \sum_{k \in \mathcal{T}_d} n_k^{\min} - n_{e\text{TIM}}^{\min} r \right) \bar{R}_{k_{\max}}^\varphi \right), \quad (\text{A.38})$$

which can be computed after simplifications as:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = \frac{1}{L} \left(n_{e\text{TIM}}^{\min} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d \setminus \{k_{\max}\}} n_k^{\min} (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi. \quad (\text{A.39})$$

Case 2: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi > r\bar{R}_{k_{\max}}^\varphi$. Plugging (2.62) into (A.36), we obtain:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = \frac{1}{L} \left(\left(n_{e\text{TIM}}^{\min} + \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \right) \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi + \sum_{k \in \mathcal{T}_d \setminus \{k_{\max}\}} n_k^{\min} \bar{R}_k^\varphi + (n_{k_{\max}}^{\min} + L_{\text{res}} - \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor) \bar{R}_{k_{\max}}^\varphi \right). \quad (\text{A.40})$$

which can be computed after plugging (2.60) into (A.40) simplifications as:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = \frac{1}{L} \left(\left(n_{e\text{TIM}}^{\min} + \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \right) (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d \setminus \{k_{\max}\}} n_k^{\min} (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi. \quad (\text{A.41})$$

A.3.4 Proof of Lemma 2.1

Proof. We suppose that: $\forall L \in \mathbb{N}^*, \sum_{k \in \mathcal{T}_d} x_k + r x_{e\text{TIM}} = 1$. We want to show that Problem 2.14 is feasible.

We remind that the feasible set of Problem 2.14 is the set of values L verifying the inequality (2.66). Moreover, using the known inequality $x \leq \lceil x \rceil$, we can obtain:

$$\underbrace{\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}}}_{=1} \leq \underbrace{\frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right)}_{\leq 1}, \quad (\text{A.42})$$

from which we can deduce the following equations:

$$\begin{aligned} & \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) = 1, \\ \Leftrightarrow & \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) = \sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}}, \quad (\text{A.43}) \\ \Leftrightarrow & \sum_{k \in \mathcal{T}_d} (\lceil Lx_k \rceil - Lx_k) + r(\lceil Lx_{e\text{TIM}} \rceil - Lx_{e\text{TIM}}) = 0. \end{aligned}$$

Since we have a sum of positive terms which is equal to 0, each of these terms equals 0. Thus, (A.43) enables us to deduce that:

$$\begin{cases} \lceil Lx_k \rceil = Lx_k, & \forall k \in \mathcal{T}_d, \\ \lceil Lx_{e\text{TIM}} \rceil = Lx_{e\text{TIM}}. \end{cases} \quad (\text{A.44})$$

Since we have $L \in \mathbb{N}^*$, (A.44) is equivalent to the following system:

$$\begin{cases} Lx_k \in \mathbb{N}, & \forall k \in \mathcal{T}_d, \\ Lx_{e\text{TIM}} \in \mathbb{N}. \end{cases} \quad (\text{A.45})$$

To find values of L verifying (A.45), we use the assumption that $(x_k)_{k \in \mathcal{T}_d}$ and $x_{e\text{TIM}}$ are rational, which enables us to state the following:

$$\begin{aligned} \forall k \in \mathcal{T}_d, \quad \exists p_k, q_k \in \mathbb{N}^* : x_k &= \frac{p_k}{q_k} \\ \exists p_{e\text{TIM}}, q_{e\text{TIM}} \in \mathbb{N}^* : x_{e\text{TIM}} &= \frac{p_{e\text{TIM}}}{q_{e\text{TIM}}}, \end{aligned} \quad (\text{A.46})$$

with $\text{gcd}(p_{e\text{TIM}}, q_{e\text{TIM}}) = 1$ and $\forall k \in \mathcal{T}_d, \text{gcd}(p_k, q_k) = 1$, where $\text{gcd}(a, b)$ stands for greatest common divisor between a and b . q_k and $q_{e\text{TIM}}$ are respectively referred to as the smallest denominators of x_k and $x_{e\text{TIM}}$. By injecting (A.46) into (A.45), we can deduce that the values of L verifying (A.45) are the ones such that L is a multiplier of $q_{e\text{TIM}}$ and q_k for all $k \in \mathcal{T}_d$, with $q_{e\text{TIM}}$ and q_k . One solution is thus to choose:

$$L = \text{lcm}(\mathcal{Q}_{\text{feas}}), \quad (\text{A.47})$$

with $\text{lcm}(a, b, c, \dots)$ standing for least common multiple between a, b, c, \dots and $\mathcal{Q}_{\text{feas}}$ defined as follows:

$$\mathcal{Q}_{\text{feas}} := \{q_k \mid k \in \mathcal{T}_d\} \cup \{q_{e\text{TIM}}\}. \quad (\text{A.48})$$

□

In the following, we deliver a corollary of Lemma 2.1 which characterizes the feasible set of Problem 2.14. Its proof can be obtained using (A.46) and (A.45).

Corollary A.1. *Let $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} = 1$ and let assume that x_{eTIM} and x_k are rational positive values.*

In this case, the feasible set of Problem (2.14) is the set of values $L \in \mathbb{N}^$ such that L is a multiplier of q_{eTIM} and q_k for all $k \in \mathcal{T}_d$, which are defined in (A.46). In other words, the feasible set of Problem (2.14) is equal to \mathcal{F}_0 , where \mathcal{F}_0 can be expressed as:*

$$\mathcal{F}_0 := \{L \in \mathbb{N}^* \text{ such that } : \exists m_0 \in \mathbb{N}^* : L = m_0 q_{eTIM} \text{ and } \forall k \in \mathcal{T}_d, \exists m_k \in \mathbb{N}^* : L = m_k q_k\}. \quad (\text{A.49})$$

A.3.5 Proof of Lemma 2.2

Proof. We suppose that: $\forall L \in \mathbb{N}^*, \sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} < 1$. We want to show that there exists a value $L_0 \in \mathbb{N}^*$ such that for all $L \geq L_0$, L is a feasible solution of Problem 2.14. Formally, we want to prove the following statement:

$$\exists L_0 \in \mathbb{N}^* : \forall L \geq L_0, \quad g(L) \leq 1, \quad (\text{A.50})$$

with $g(L) := \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{eTIM} \rceil \right)$.

Let us define η such that $\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} + \eta = 1$. After applying the following ceiling function bounding on $\lceil Lx_k \rceil$ and $\lceil Lx_{eTIM} \rceil$:

$$\forall z \in \mathbb{R}, z \leq \lceil z \rceil < z + 1, \quad (\text{A.51})$$

we obtain the following inequalities:

$$\underbrace{\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM}}_{=1-\eta} \leq g(L) < \underbrace{\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM}}_{=1-\eta} + \frac{Q+r}{L}. \quad (\text{A.52})$$

Since the upper bound limit can be expressed as:

$$\lim_{L \rightarrow +\infty} \left(\sum_{k \in \mathcal{T}_d} x_k + rx_{eTIM} + \frac{Q+r}{L} \right) = 1 - \eta, \quad (\text{A.53})$$

we can deduce the following limit expression:

$$\lim_{L \rightarrow +\infty} g(L) = 1 - \eta, \quad (\text{A.54})$$

which is strictly inferior to 1 according to the initial assumption.

The main idea to prove the infinite number of feasible solutions of Problem 2.14 is the following. On the one hand, L is a feasible solution if $g(L) \leq 1$. On the other hand, $g(L)$ tends to a value strictly lower than 1 when L tends to $+\infty$. Therefore, for increasing

values of L , $g(L)$ will get closer to its limit which is strictly lower than 1. Then, when L is high enough, $g(L)$ will always be lower than 1 which means that all these high values of L are feasible solutions of Problem 2.14.

The formal proof based on this idea is obtained by expressing the formal definition of the limit for (A.54) which can be written as:

$$\forall \epsilon > 0, \exists L_0 \in \mathbb{N}^* : \forall L \geq L_0, \quad |g(L) - 1 + \eta| \leq \epsilon, \quad (\text{A.55})$$

whose absolute value can be removed according to (A.52), which provides the following result:

$$\forall \epsilon > 0, \exists L_0 \in \mathbb{N}^* : \forall L \geq L_0, \quad g(L) \leq 1 + \epsilon - \eta. \quad (\text{A.56})$$

Since (A.56) is true for all $\epsilon > 0$, we choose it such that $\epsilon \leq \eta$, which shows the following statement:

$$\exists L_0 \in \mathbb{N}^* : \forall L \geq L_0, \quad g(L) \leq 1, \quad (\text{A.57})$$

which concludes the proof. \square

A.3.6 Proof of Result 2.4

Since we need to prove an equivalence statement, we first prove the direct statement, then we prove its converse.

Suppose Problem 2.14 is feasible. We want to prove that we have either 1. $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} < 1$ or 2. $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} = 1$ with $(x_k)_{k \in \mathcal{T}_d}, x_{e\text{TIM}}$ rational positive numbers.

Since Problem 2.14 has at least one feasible solution $L \in \mathbb{N}^*$, we have the following feasibility inequality from (2.66):

$$\frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) \leq 1. \quad (\text{A.58})$$

By using the ceiling function bounding provided in (A.51) on $\lceil Lx_k \rceil$ and $\lceil Lx_{e\text{TIM}} \rceil$, we can deduce the following inequality:

$$\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} \leq \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right), \quad (\text{A.59})$$

and thus, using (A.58), we equivalently have the following:

$$\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} \leq 1. \quad (\text{A.60})$$

Now that we have obtained the desired inequality, we only need to prove that the case where there is an equality in (A.60) necessarily leads to rational values of $(x_k)_{k \in \mathcal{T}_d}$ and $x_{e\text{TIM}}$.

Since we have the following inequalities:

$$\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} \leq \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) \leq 1, \quad (\text{A.61})$$

the following implications hold:

$$\begin{aligned} \sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} = 1 &\implies \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) = 1, \\ &\implies \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \lceil Lx_k \rceil + r \lceil Lx_{e\text{TIM}} \rceil \right) = \sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}}, \\ &\implies \sum_{k \in \mathcal{T}_d} (\lceil Lx_k \rceil - Lx_k) + r(\lceil Lx_{e\text{TIM}} \rceil - Lx_{e\text{TIM}}) = 0, \quad (\text{A.62}) \\ &\implies \begin{cases} \forall k \in \mathcal{T}_d, \lceil Lx_k \rceil = Lx_k, \\ \lceil Lx_{e\text{TIM}} \rceil = Lx_{e\text{TIM}}, \end{cases} \\ &\implies \begin{cases} \forall k \in \mathcal{T}_d, Lx_k \in \mathbb{N}^*, \\ Lx_{e\text{TIM}} \in \mathbb{N}^*. \end{cases} \end{aligned}$$

The third implication is justified by the fact that when a sum of positive terms is equal to 0, each of these terms equals 0.

Finally, as stated in Lemma 2.1, the last system requires the rationality of the terms $x_k, x_{e\text{TIM}}$ so that Lx_k and $Lx_{e\text{TIM}}$ can be integer values, which concludes the first implication of the proof.

Suppose either $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} < 1$ or $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} = 1$ with $(x_k)_{k \in \mathcal{T}_d}, x_{e\text{TIM}}$ rational positive numbers. We want to prove that Problem 2.14 is feasible.

- We suppose that $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} < 1$ is true. Lemma 2.2 shows that Problem 2.14 has an infinite number of feasible solutions, so it is feasible.
- We suppose that $\sum_{k \in \mathcal{T}_d} x_k + rx_{e\text{TIM}} = 1$ and $(x_k)_{k \in \mathcal{T}_d}, x_{e\text{TIM}}$ are rational positive numbers. Using Lemma 2.1, we show that Problem 2.14's feasible set is \mathcal{F}_0 defined in (A.49), which is not empty and thus, proves Problem 2.14's feasibility.

A.3.7 Proof of Lemma 2.3

We first use the ceiling function inequalities defined in (A.51) on (2.58a) and (2.58b) and we obtain the following inequalities:

$$Lx_k \leq n_k^{\min} < Lx_k + 1, \quad \forall k \in \mathcal{T}_d, \quad (\text{A.63})$$

$$Lx_{e\text{TIM}} \leq n_{e\text{TIM}}^{\min} < Lx_{e\text{TIM}} + 1. \quad (\text{A.64})$$

Now, we inject these inequalities into $R_{\text{sum,opt}}^{\text{eTIM-hy}}(L)$ expression.

Case 1: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$. Plugging (A.63) and (A.64) into (2.65) yields:

$$\begin{aligned} & \frac{1}{L} \left((Lx_{\text{eTIM}} + 1) (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} (Lx_k + 1) (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi \\ & < R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) \\ & \leq \frac{1}{L} \left(Lx_{\text{eTIM}} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} Lx_k (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi, \end{aligned} \quad (\text{A.65})$$

whose inequalities have been reverse since the terms $(\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi)$ and $(\sum_{j \in \mathcal{T}_e} \bar{R}_{\text{eTIM},j}^\varphi - r\bar{R}_{k_{\max}}^\varphi)$ are negative.

After simplifications, (A.65) can be computed as:

$$R_{\text{inf}}(L) < R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) \leq R_{\text{sup}}, \quad (\text{A.66})$$

where R_{sup} and $R_{\text{inf}}(L)$ are defined for $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$ as:

$$R_{\text{sup}} := x_{\text{eTIM}} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + N_{\mathcal{T}_d} R_0^\mu + \bar{R}_{k_{\max}}^\varphi \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right), \quad (\text{A.67})$$

$$R_{\text{inf}}(L) := R_{\text{sup}} + \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \bar{R}_k^\varphi + \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - (N_{\mathcal{T}_d} + r) \bar{R}_{k_{\max}}^\varphi \right). \quad (\text{A.68})$$

It is worth mentioning that the upper bound R_{sup} does not depend on L . This concludes Lemma 2.3's proof for the case $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$.

Case 2: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi > r\bar{R}_{k_{\max}}^\varphi$. Using the floor function inequality: $x - 1 < \lfloor x \rfloor \leq x$, we obtain the following inequalities:

$$\frac{L_{\text{res}}}{r} - 1 < \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \leq \frac{L_{\text{res}}}{r}, \quad (\text{A.69})$$

which can be computed by adding the term n_{eTIM}^{\min} and injecting (2.60) into (A.69) as:

$$\frac{L(1 - \sum_{k \in \mathcal{T}_d} x_k)}{r} - 1 < n_{\text{eTIM}}^{\min} + \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \leq \frac{L(1 - \sum_{k \in \mathcal{T}_d} x_k)}{r}. \quad (\text{A.70})$$

After injecting (A.63) and (A.70) into (2.65), we obtain the following inequalities:

$$\begin{aligned} & \frac{1}{L} \left(\left(\frac{L(1 - \sum_{k \in \mathcal{T}_d} x_k)}{r} - 1 \right) (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} (Lx_k + 1) (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi \\ & < R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) \\ & \leq \frac{1}{L} \left(\frac{L(1 - \sum_{k \in \mathcal{T}_d} x_k)}{r} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} Lx_k (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) \right) + \bar{R}_{k_{\max}}^\varphi, \end{aligned} \quad (\text{A.71})$$

which can be simplified as:

$$R_{\text{inf}}(L) < R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) \leq R_{\text{sup}}, \quad (\text{A.72})$$

where R_{sup} and $R_{\text{inf}}(L)$ are defined for $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi > r\bar{R}_{k_{\text{max}}}^\varphi$ as:

$$R_{\text{sup}} := \frac{(1 - \sum_{k \in \mathcal{T}_d} x_k)}{r} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\text{max}}}^\varphi) + QR_0^u + \bar{R}_{k_{\text{max}}}^\varphi \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right), \quad (\text{A.73})$$

$$R_{\text{inf}}(L) := R_{\text{sup}} + \frac{1}{L} \left(\sum_{k \in \mathcal{T}_d} \bar{R}_k^\varphi - \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - (Q - r)\bar{R}_{k_{\text{max}}}^\varphi \right). \quad (\text{A.74})$$

Similarly, in this case, the upper bound R_{sup} does not depend on L . This concludes Lemma 2.3's proof for the case $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi > r\bar{R}_{k_{\text{max}}}^\varphi$.

Corollary A.2. R_{sup} is $\bar{R}_{\text{sum,opt}}^{\text{eTIM-hy}}(L)$'s limit when $L \rightarrow +\infty$, which can formally be written as:

$$\lim_{L \rightarrow +\infty} R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = R_{\text{sup}}. \quad (\text{A.75})$$

Proof. Case 1: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\text{max}}}^\varphi$.

Using R_{sup} and R_{inf} 's expression defined in (A.67) and (A.68), respectively, we can derive the following:

$$\lim_{L \rightarrow +\infty} R_{\text{inf}}(L) = R_{\text{sup}}, \quad (\text{A.76})$$

from which we can thus deduce that:

$$\lim_{L \rightarrow +\infty} R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = R_{\text{sup}}. \quad (\text{A.77})$$

Case 2: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi > r\bar{R}_{k_{\text{max}}}^\varphi$.

Using R_{sup} and R_{inf} 's expression defined in (A.73) and (A.74), respectively, we can derive the following:

$$\lim_{L \rightarrow +\infty} R_{\text{inf}}(L) = R_{\text{sup}}, \quad (\text{A.78})$$

from which we can thus deduce that:

$$\lim_{L \rightarrow +\infty} R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = R_{\text{sup}}. \quad (\text{A.79})$$

□

A.3.8 Proof of Result 2.5

In this proof, we want to find the solutions of the following equation:

$$R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = R_{\text{sup}}, \quad (\text{A.80})$$

with $R_{\text{sum,opt}}^{\text{eTIM-hy}}(L)$ and R_{sup} defined in (2.65) and (2.68), respectively.

Case 1: $\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$. (A.80) can be rewritten as:

$$R_{\text{sup}} - R_{\text{sum,opt}}^{\text{eTIM-hy}}(L) = 0. \quad (\text{A.81})$$

Plugging (2.65) and (2.68) into (A.81), we obtain:

$$\begin{aligned} & \frac{n_{\text{eTIM}}}{L} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) - x_{\text{eTIM}} (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + QR_0^u + \bar{R}_{k_{\max}}^\varphi \left(1 - \sum_{k \in \mathcal{T}_d} x_k \right) \\ & - \frac{1}{L} \sum_{k \in \mathcal{T}_d} n_k^{\min} (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi) - \bar{R}_{k_{\max}}^\varphi = 0. \end{aligned} \quad (\text{A.82})$$

Using the following equalities:

$$\begin{aligned} QR_0^u - \bar{R}_{k_{\max}}^\varphi \sum_{k \in \mathcal{T}_d} x_k &= \sum_{k \in \mathcal{T}_d} R_0^u - \bar{R}_{k_{\max}}^\varphi \sum_{k \in \mathcal{T}_d} x_k \\ &= \sum_{k \in \mathcal{T}_d} \bar{R}_k^\varphi x_k - \bar{R}_{k_{\max}}^\varphi \sum_{k \in \mathcal{T}_d} x_k \\ &= \sum_{k \in \mathcal{T}_d} x_k (\bar{R}_k^\varphi - \bar{R}_{k_{\max}}^\varphi), \end{aligned} \quad (\text{A.83})$$

we inject (A.83) into (A.82) and we obtain the following:

$$(n_{\text{eTIM}} - Lx_{\text{eTIM}}) (r\bar{R}_{k_{\max}}^\varphi - \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi) + \sum_{k \in \mathcal{T}_d} (n_k^{\min} - Lx_k) (\bar{R}_{k_{\max}}^\varphi - \bar{R}_k^\varphi) = 0. \quad (\text{A.84})$$

Then, by plugging (2.58a) and (2.58b) into (A.84), we equivalently obtain:

$$([\lfloor Lx_{\text{eTIM}} \rfloor - Lx_{\text{eTIM}}]) (r\bar{R}_{k_{\max}}^\varphi - \bar{R}_{\text{sum},\mathcal{T}_e}^\varphi) + \sum_{k \in \mathcal{T}_d} ([\lfloor Lx_k \rfloor - Lx_k]) (\bar{R}_{k_{\max}}^\varphi - \bar{R}_k^\varphi) = 0. \quad (\text{A.85})$$

Since this is a sum of positive terms which is equal to 0, each of these terms equals 0.

Thus, we can deduce the conditions on the solutions of (A.80) which can be written as:

$$\begin{cases} [\lfloor Lx_k \rfloor = Lx_k, & \forall k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}, \\ ([\lfloor Lx_{\text{eTIM}} \rfloor = Lx_{\text{eTIM}}]) \vee (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi = r\bar{R}_{k_{\max}}^\varphi), \end{cases} \quad (\text{A.86})$$

with $\mathcal{K}_{\max} = \{k | \bar{R}_k^\varphi = \bar{R}_{k_{\max}}^\varphi\}$. We can then write the following conditions:

$$\begin{cases} Lx_k \in \mathbb{N}^*, & \forall k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}, \\ (Lx_{\text{eTIM}} \in \mathbb{N}^*) \vee (\bar{R}_{\text{sum},\mathcal{T}_e}^\varphi = r\bar{R}_{k_{\max}}^\varphi), \end{cases} \quad (\text{A.87})$$

Similarly, since we started our reasoning from (A.80), the conditions we have found in (A.97) are necessary ones. However, as we have only used equivalence to find them, these conditions are also sufficient.

Then, using the rates rationality assumption, we compute $(p_{\text{eTIM}}, q_{\text{eTIM}})$ and (p_k, q_k) , $\forall k \in \mathcal{T}_d$ using (2.70) and we deduce that the solutions of (A.80) are the set $L \in \mathbb{N}^*$ such that the following conditions are satisfied:

1. $\forall k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}$, L is a multiplier of q_k .

2. Either L is a multiplier of $q_{e\text{TIM}}$ or we have $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi = r\bar{R}_{k_{\max}}^\varphi$,

which concludes the proof for $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi \leq r\bar{R}_{k_{\max}}^\varphi$.

Case 2: $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{\max}}^\varphi$.

After injecting (2.65) and (2.68) into (A.81) and after similar derivations as in the first case, (A.81) can equivalently be expressed as:

$$\left(\frac{L(1 - \sum_{k \in \mathcal{T}_d} Lx_k)}{r} - n_{e\text{TIM}} - \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \right) (\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} (\lceil Lx_k \rceil - Lx_k) (\bar{R}_{k_{\max}}^\varphi - \bar{R}_k^\varphi) = 0, \quad (\text{A.88})$$

from which we can derive the following equation:

$$\left(\frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} - \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \right) (\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi - r\bar{R}_{k_{\max}}^\varphi) + \sum_{k \in \mathcal{T}_d} (\lceil Lx_k \rceil - Lx_k) (\bar{R}_{k_{\max}}^\varphi - \bar{R}_k^\varphi) = 0. \quad (\text{A.89})$$

On the one hand, using the ceiling function inequality $x \leq \lceil x \rceil$, on n_k^{\min} and $n_{e\text{TIM}}$, we can deduce the following inequality:

$$L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r \geq L_{\text{res}}. \quad (\text{A.90})$$

On the other hand, using the flooring function inequality $\lfloor x \rfloor \leq x$, on $\left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor$, we can deduce the following inequality:

$$\left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \leq \frac{L_{\text{res}}}{r}, \quad (\text{A.91})$$

Thus, using (A.90) and (A.91), we can deduce the positivity of the following term:

$$\left(\frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} - \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor \right) \geq 0. \quad (\text{A.92})$$

Therefore, we can deduce that (A.89) corresponds to a sum of positive terms which is equal to 0. Thus, we can deduce the conditions on the solutions of (A.80) which can be written as:

$$\begin{cases} \lceil Lx_k \rceil = Lx_k & \forall k \in \mathcal{T}_d \setminus \mathcal{K}_{\max}, \end{cases} \quad (\text{A.93a})$$

$$\begin{cases} \frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} = \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor. \end{cases} \quad (\text{A.93b})$$

We solve the last equation of the system (A.93b) and we express the following equivalences to find the conditions on the solutions of (A.80):

$$\frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} = \left\lfloor \frac{L_{\text{res}}}{r} \right\rfloor, \quad (\text{A.94})$$

which is equivalent to the following:

$$\left\{ \begin{array}{l} \frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} \in \mathbb{N}^*, \\ \frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} = \frac{L_{\text{res}}}{r}, \end{array} \right. \quad (\text{A.95a})$$

$$\left\{ \begin{array}{l} \frac{L - \sum_{k \in \mathcal{T}_d} Lx_k - n_{e\text{TIM}}r}{r} = \frac{L_{\text{res}}}{r}, \end{array} \right. \quad (\text{A.95b})$$

from which we equivalently deduce the following conditions:

$$\left\{ \begin{array}{l} \lceil Lx_k \rceil = Lx_k, \quad \forall k \in \mathcal{T}_d, \\ L \left(\frac{1 - \sum_{k \in \mathcal{T}_d} x_k}{r} \right) \in \mathbb{N}^*. \end{array} \right. \quad (\text{A.96a})$$

$$\left\{ \begin{array}{l} L \left(\frac{1 - \sum_{k \in \mathcal{T}_d} x_k}{r} \right) \in \mathbb{N}^*. \end{array} \right. \quad (\text{A.96b})$$

Since the conditions in (A.93a) are included in the conditions in (A.96a), the conditions to solve (A.80) can be written as:

$$\left\{ \begin{array}{l} Lx_k \in \mathbb{N}^*, \quad \forall k \in \mathcal{T}_d, \\ L \left(\frac{1 - \sum_{k \in \mathcal{T}_d} x_k}{r} \right) \in \mathbb{N}^*. \end{array} \right. \quad (\text{A.97a})$$

$$\left\{ \begin{array}{l} L \left(\frac{1 - \sum_{k \in \mathcal{T}_d} x_k}{r} \right) \in \mathbb{N}^*. \end{array} \right. \quad (\text{A.97b})$$

Similarly, since we started our reasoning from (A.80), the conditions we have found in (A.97) are necessary. However, as we have only used equivalence to find them, these conditions are also sufficient.

Then, using the rates rationality assumption, we compute $(p_k, q_k), \forall k \in \mathcal{T}_d$ using (2.70) and (p_0, q_0) (2.72). Thus, we deduce that the solutions of (A.80) are the set $L \in \mathbb{N}^*$ such that the following conditions are satisfied:

1. $\forall k \in \mathcal{T}_d, L$ is a multiplier of q_k .
2. L is a multiplier of q_0 .

which concludes the proof for $\bar{R}_{\text{sum}, \mathcal{T}_e}^\varphi > r\bar{R}_{k_{\text{max}}}^\varphi$.

Appendix B

Appendix related to Chapter 3

B.1 Scheduling with two simultaneous transmissions

B.1.1 Maximum likelihood estimation: Hessian matrix expression

Given samples $\mathbf{S} = (\mathcal{S}_1, \dots, \mathcal{S}_N)$, the log-likelihood partial derivatives of $\ell(\mathbf{S}; \gamma)$ defined in (3.5), we calculate $\ell(\mathbf{S}; \gamma)$'s Hessian matrix \mathbf{H}_ℓ , whose (i, j) -th entry can be written as

$$[\mathbf{H}_\ell]_{ij} := \frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_i \partial \gamma_j}, i, j = 1, 2, \quad (\text{B.1})$$

with $\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1 \partial \gamma_2} = \frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_2 \partial \gamma_1}$ according to Schwarz's theorem since $\ell(\mathbf{S}; \gamma)$ is of class C^2 .

In the following, we provide the following second-order partial derivatives $\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1^2}$ and $\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_2^2}$ which can be written as:

$$\begin{aligned} \frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1^2} &= \sum_{i=1}^{N_s} \left(-\frac{2\mathcal{S}_i}{a\gamma_1^3} - \left(\frac{a(1+b\gamma_2)}{a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2} \right)^2 + 2 \left(\frac{a}{a\gamma_1 + b\gamma_2\mathcal{S}_i} \right)^2 \right) \\ \frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_2^2} &= b^2 \left(2 \sum_{i=1}^{N_s} \left(\frac{\mathcal{S}_i}{a\gamma_1 + b\gamma_2\mathcal{S}_i} \right)^2 - \sum_{i=1}^{N_s} \left(\frac{\mathcal{S}_i + a\gamma_1}{a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2} \right)^2 \right) \end{aligned} \quad (\text{B.2})$$

In the following, we derive the expression of $\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1 \partial \gamma_2}$. To do so, we first differentiate $\frac{\partial \ell(\mathbf{S}; \gamma)}{\partial \gamma_2}$ with respect to γ_1 and we obtain the second-order partial cross-derivative of the log-likelihood function which can be expressed as:

$$\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1 \partial \gamma_2} = \sum_{i=1}^{N_s} \left(q'(\gamma_1) + \frac{2ab\mathcal{S}_i}{(a\gamma_1 + b\gamma_2\mathcal{S}_i)^2} \right), \quad (\text{B.3})$$

with

$$q(\gamma_1) := \frac{b(\mathcal{S}_i + a\gamma_1)}{a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2}. \quad (\text{B.4})$$

Differentiating (B.4) leads to $q'(\gamma_1)$ expression which can be written as:

$$\begin{aligned} q'(\gamma_1) &= \frac{ab(a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2) - b(\mathcal{S}_i + a\gamma_1)a(1 + b\gamma_2)}{(a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2)^2} \\ &= -\frac{ab\mathcal{S}_i}{(a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2)^2} \end{aligned} \quad (\text{B.5})$$

Therefore, the second-order partial cross-derivative can be written as the following:

$$\begin{aligned} \frac{\partial^2 \ell(\mathbf{S}; \boldsymbol{\gamma})}{\partial \gamma_1 \partial \gamma_2} &= \frac{\partial^2 \ell(\mathbf{S}; \boldsymbol{\gamma})}{\partial \gamma_2 \partial \gamma_1} \\ &= ab \sum_{i=1}^{N_S} \left(\mathcal{S}_i \left(-\frac{1}{(a\gamma_1 + b\gamma_2\mathcal{S}_i + ab\gamma_1\gamma_2)^2} + \frac{2}{(a\gamma_1 + b\gamma_2\mathcal{S}_i)^2} \right) \right) \end{aligned} \quad (\text{B.6})$$

Therefore, the Hessian matrix term expressions are provided in (B.2) and (B.6).

B.1.2 Method of moments

B.1.2.1 Theoretical moments derivation

The first and second-order theoretical moments $\phi_1(\boldsymbol{\gamma}) := \mathbb{E}[\mathcal{S}]$, $\phi_2(\boldsymbol{\gamma}) := \mathbb{E}[\mathcal{S}^2]$ have been computed in [109]. In this part, we detail how to derive them step by step using our notations.

B.1.2.1.1 First-order moment of \mathcal{S}

Using \mathcal{S} definition in (3.2) and the expected value property, we obtain:

$$\begin{aligned} \phi_1(\boldsymbol{\gamma}) &= \iint_{\mathbb{R}_+^2} \frac{a\gamma_1 x_1}{1 + b\gamma_2 x_2} e^{-x_1} e^{-x_2} dx_1 dx_2 \\ &= a\gamma_1 \underbrace{\int_0^{+\infty} x_1 e^{-x_1} dx_1}_{=1} \int_0^{+\infty} \frac{e^{-x_2}}{1 + b\gamma_2 x_2} dx_2 \end{aligned} \quad (\text{B.7})$$

Making the substitution $t = x_2 + 1/b\gamma_2$ yields:

$$\phi_1(\boldsymbol{\gamma}) = a\gamma_1 \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2), \quad (\text{B.8})$$

with E_1 the exponential integral function defined as $E_1(z) = \int_z^{+\infty} \frac{e^{-t}}{t} dt$.

Using the change of notations $y = 1/(b\gamma_2)$, we get the expression provided in (3.11).

B.1.2.1.2 Second-order moment of \mathcal{S}

Using \mathcal{S} definition in (3.2) and the law of the unconscious statistician (LOTUS), $\phi_2(\gamma)$'s expression can be written as:

$$\begin{aligned}\phi_2(\gamma) &= \iint_{\mathbb{R}_+^2} \left(\frac{a\gamma_1 x_1}{1 + b\gamma_2 x_2} \right)^2 e^{-x_1} e^{-x_2} dx_1 dx_2 \\ &= (a\gamma_1)^2 \underbrace{\int_0^{+\infty} x_1^2 e^{-x_1} dx_1}_{=2} \int_0^{+\infty} \frac{e^{-x_2}}{(1 + b\gamma_2 x_2)^2} dx_2.\end{aligned}\quad (\text{B.9})$$

Making the substitution $t = x_2 + 1/b\gamma_2$, $\phi_2(\gamma)$ can be expressed as:

$$\phi_2(\gamma) = 2(a\gamma_1)^2 \frac{e^{1/b\gamma_2}}{(b\gamma_2)^2} \int_{1/b\gamma_2}^{+\infty} \frac{e^{-t}}{t^2} dt, \quad (\text{B.10})$$

which can be written after performing an integration by parts as the following:

$$\phi_2(\gamma) = \frac{2(a\gamma_1)^2}{b\gamma_2} \left(1 - \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) \right), \quad (\text{B.11})$$

which corresponds to the expression in (3.11) when using the change of notations $y = 1/(b\gamma_2)$.

B.1.2.2 Proof of Result 3.2

Hereafter, we provide the derivations to get the MoM estimates $\gamma_{\text{MoM}} = (\gamma_{1,\text{MoM}}, \gamma_{2,\text{MoM}})$ by solving (3.10). For the purpose of clarity, we simplify the notations in this part by replacing $\gamma_{\text{MoM}} = (\gamma_{1,\text{MoM}}, \gamma_{2,\text{MoM}})$ by $\gamma = (\gamma_1, \gamma_2)$.

Plugging (3.11) into (3.10) yields the following system to solve:

$$\begin{cases} a\gamma_1 \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) = M_1 \end{cases} \quad (\text{B.12a})$$

$$\begin{cases} \frac{2(a\gamma_1)^2}{b\gamma_2} \left(1 - \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) \right) = M_2. \end{cases} \quad (\text{B.12b})$$

Next, we inject (B.12a) into (B.12b) and we obtain the following system:

$$\begin{cases} a\gamma_1 \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) = M_1 \\ \frac{2(a\gamma_1)^2}{b\gamma_2} - \frac{2a\gamma_1}{b\gamma_2} M_1 = M_2 \end{cases} \quad (\text{B.13})$$

$$\begin{cases} a\gamma_1 \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) = M_1 \\ \gamma_1^2 - \frac{M_1}{a} \gamma_1 - \frac{M_2 b\gamma_2}{2a^2} = 0. \end{cases} \quad (\text{B.14a})$$

$$\gamma_1^2 - \frac{M_1}{a} \gamma_1 - \frac{M_2 b\gamma_2}{2a^2} = 0. \quad (\text{B.14b})$$

We solve (B.14b) which is a second-order polynomial and we get γ_1 's two possible roots:

$$\gamma_1 = \frac{M_1}{2a} \left(1 \pm \sqrt{1 + 2b\gamma_2 \frac{M_2}{(M_1)^2}} \right). \quad (\text{B.15})$$

Since γ_1 is a positive value, its expression is provided by the positive solution from (B.15), which can be written as:

$$\gamma_1 = \frac{M_1}{2a} \left(1 + \sqrt{1 + 2b\gamma_2 \frac{M_2}{(M_1)^2}} \right). \quad (\text{B.16})$$

Then, plugging (B.16) into (B.14a) yields to the following system:

$$\begin{cases} \gamma_1 = \frac{M_1}{2a} \left(1 + \sqrt{1 + 2b\gamma_2 \frac{M_2}{(M_1)^2}} \right) \\ \frac{1 + \sqrt{1 + 2b\gamma_2 M_2 / (M_1)^2}}{2} \frac{\exp(1/b\gamma_2)}{b\gamma_2} E_1(1/b\gamma_2) - 1 = 0, \end{cases} \quad (\text{B.17})$$

which corresponds to the system provided in (3.13) by setting $\rho := M_2 / (M_1)^2$ and f_{MoM} as defined in (3.14).

B.1.2.3 Function f_{MoM} derivative

In this part, we provide the expression of f'_{MoM} which corresponds to the derivative of f_{MoM} . We remind that f_{MoM} is defined in (3.14) and can be expressed as:

$$f_{\text{MoM}}(z) = g_1(z)g_2(z)g_3(z)g_4(z) - 1 \quad (\text{B.18})$$

with $g_1(z) = 1/(2bz)$, $g_2(z) = 1 + \sqrt{1 + 2b\rho z}$, $g_3(z) = \exp(1/bz)$ and $g_4(z) = E_1(1/bz)$.

Therefore, f'_{MoM} expression can be written as:

$$f'_{\text{MoM}}(z) = \sum_{i=1}^4 g'_i(z) \prod_{j=1, j \neq i}^4 g_j(z) \quad (\text{B.19})$$

with $g'_1(z) = -1/(2bz^2)$, $g'_2(z) = b\rho(1 + 2b\rho z)^{-1/2}$, $g'_3(z) = -\exp(1/bz)/(bz^2)$ and $g'_4(z) = \exp(-1/bz)/z$.

Let us define $g_{34}(z)$ as the following:

$$g_{34}(z) = g_3(z)g_4(z) = e^{1/bz} E_1\left(\frac{1}{bz}\right) \quad (\text{B.20})$$

After tedious calculations, we get f'_{MoM} expression which can be written as:

$$f'_{\text{MoM}}(z) = g_1(z) \left(g'_2(z)g_{34}(z) + \frac{g_2(z)}{z} \left(1 - \frac{g_{34}(z)}{bz} (1 + bz) \right) \right). \quad (\text{B.21})$$

Remark B.1. The notation $g_{34}(z)$ is defined to make the integral exponential computation easier. Indeed, by taking the continued fraction formula approximation from [1, Chapter 5] to implement the exponential integral, we notice that the product $e^y E_1(y)$ is easier to implement than separately. Moreover, for high values of y , computing the term e^y can result in an infinite term which leads to $e^y E_1(y)$ being indefinite. In reality, the term $e^y E_1(y)$ is well defined and equals 0 when y tends to $+\infty$.

B.1.2.4 Proof of Result 3.3**B.1.2.4.1 Proofs of p_1**

The goal of this part is to find the limits of f_{MoM} in 0^+ whose expression is reminded in the following:

$$f_{\text{MoM}}(z) = \frac{1 + \sqrt{1 + 2b\rho z}}{2bz} \exp(1/bz) E_1(1/bz) - 1, \quad (\text{B.22})$$

which can be written as:

$$f_{\text{MoM}}(z) = \frac{1 + \sqrt{1 + 2b\rho z}}{2} J\left(\frac{1}{bz}\right) - 1, \quad (\text{B.23})$$

with J defined as the following:

$$\begin{aligned} J: \mathbb{R}_+ &\rightarrow \mathbb{R}_+ \\ y &\mapsto ye^y E_1(y). \end{aligned} \quad (\text{B.24})$$

Using [49, Inequation (75)], we have the following inequality on J :

$$\forall y > 0, \quad \frac{y}{y+1} < J(y) < 1, \quad (\text{B.25})$$

from which we deduce that $\lim_{y \rightarrow +\infty} J(y) = 1$. Therefore, using the limit of $J(y)$ and $f_{\text{MoM}}(z)$'s expression from (B.23), we deduce p_1 stating that:

$$\lim_{z \rightarrow 0^+} f_{\text{MoM}}(z) = 0. \quad (\text{B.26})$$

B.1.2.4.2 Proofs of p_2

We consider function J defined in (B.24) and we perform the following change of variables: $t = s + y$, yielding:

$$\forall y \in \mathbb{R}_+, J(y) = \int_y^{+\infty} \frac{ye^{y-t}}{t} dt = \int_0^{+\infty} \frac{ye^{-s}}{s+y} ds. \quad (\text{B.27})$$

Since the function $(s, y) \mapsto \frac{ye^{-s}}{s+y}$ converges simply to $s \mapsto 0$ when $y \rightarrow 0^+$ and is dominated by the integrable function $s \mapsto e^{-s}$ on \mathbb{R}_+ , we can use the particular case of Lebesgue's dominated convergence theorem to get J 's limit in 0^+ which states that the limit and integral operator can be swapped:

$$\lim_{y \rightarrow 0^+} J(y) = \int_0^{+\infty} 0 ds = 0. \quad (\text{B.28})$$

Therefore, we deduce p_2 stating that

$$\lim_{z \rightarrow +\infty} f_{\text{MoM}}(z) = -1. \quad (\text{B.29})$$

B.1.2.4.3 Proofs of p_3

The aim of this part is to express $f'(0)$ which corresponds to $\lim_{z \rightarrow 0^+} f'_{\text{MoM}}(z)$ with f'_{MoM} defined in (3.15). To do so, let us define G_1 and G_2 such that:

$$\forall z > 0, f'(z) = G_1(z) + G_2(z), \quad (\text{B.30})$$

with $G_1(z)$ and $G_2(z)$ defined as:

$$G_1(z) := \frac{g'_2(z)g_{34}(z)}{2bz}, \quad (\text{B.31})$$

$$G_2(z) := \frac{g_2(z)}{2bz^2} \left(1 - \frac{g_{34}(z)}{bz}(1+bz) \right). \quad (\text{B.32})$$

In the following, we will calculate G_1 and G_2 's limit in 0^+ to find f'_{MoM} limit in 0^+ .

Limit of G_1 Plugging the expressions of $g'_2(z)$ and $g_{34}(z)$ into (B.31) yields:

$$G_1(z) = \frac{b\rho}{2\sqrt{1+2b\rho z}} J\left(\frac{1}{bz}\right) \quad (\text{B.33})$$

Since we have $\lim_{y \rightarrow +\infty} J(y) = 1$, we can deduce that:

$$\lim_{z \rightarrow 0^+} G_1(z) = \frac{b\rho}{2}. \quad (\text{B.34})$$

Limit of G_2 Starting from E_1 , we integrate by parts three times to have the following expression for all $z > 0$:

$$\begin{aligned} E_1(y) &= \frac{e^{-y}}{y} - \frac{e^{-y}}{y^2} + \frac{2e^{-y}}{y^3} - 3 \int_y^{+\infty} \frac{e^{-t}}{t^4} dt \\ &= \frac{e^{-y}}{y} \left(1 - \frac{1}{y} + \frac{2}{y^2} + q_3(y) \right), \end{aligned} \quad (\text{B.35})$$

with $q_3(z) = -3ye^y \int_y^{+\infty} \frac{e^{-t}}{t^4} dt$.

In addition, we prove that $q_3(y)$ is dominated by $\frac{1}{y^2}$ with the following:

$$\begin{aligned} y^2 |q_3(y)| &\leq 3y^3 e^{-y} \int_y^{+\infty} \frac{e^{-t}}{y^4} dt \\ &\leq \frac{3}{y}, \end{aligned} \quad (\text{B.36})$$

which tends to 0 when $y \rightarrow +\infty$. Thus, $q_3(y)$ is dominated by $\frac{1}{y^2}$ which can be written as follows using the Bachmann–Landau notation:

$$q_3(y) = o_{y \rightarrow +\infty} \left(\frac{1}{y^2} \right). \quad (\text{B.37})$$

Then, by plugging (B.35) and (B.37) into (B.24) yields:

$$J\left(\frac{1}{bz}\right) = 1 - bz + 2(bz)^2 + \underset{z \rightarrow 0^+}{o}(z^2), \quad (\text{B.38})$$

Finally, we plug (B.38) into (B.32) and we obtain after simplifications:

$$\forall z > 0, G_2(z) = \frac{g_2(z)}{2} \left(-b + \underset{z \rightarrow 0^+}{o}(1) \right). \quad (\text{B.39})$$

Since we have $\lim_{z \rightarrow 0^+} g_2(z) = 2$, (B.39) can thus be written as:

$$\lim_{z \rightarrow 0^+} G_2(z) = -b. \quad (\text{B.40})$$

Finally, we deduce p_3 which is:

$$\lim_{z \rightarrow 0^+} f'_{\text{MoM}}(z) = b \left(\frac{\rho}{2} - 1 \right). \quad (\text{B.41})$$

B.1.3 Cramer-Rao lower bound

In this appendix, we derive the CRLB expression by providing the Fischer information matrix components defined in equation (3.23). To do so, we determine the expected value of the partial second derivatives of $\ell(\mathbf{S}; \gamma)$ detailed in (B.6). In the following we calculate Fischer information matrix components $[\mathbf{I}(\gamma)]_{11}$, $[\mathbf{I}(\gamma)]_{12}$, $[\mathbf{I}(\gamma)]_{21}$, $[\mathbf{I}(\gamma)]_{22}$.

B.1.3.1 Deriving Cramer-Rao lower bound expression

B.1.3.1.1 Computing $[\mathbf{I}(\gamma)]_{11}$

By plugging (B.6) into (3.23), we can compute the first term $[\mathbf{I}(\gamma)]_{11}$ which can be written as:

$$\begin{aligned} [\mathbf{I}(\gamma)]_{11} &:= -\mathbb{E} \left[\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1^2} \right] \\ &= -N_S \mathbb{E} \left[-\frac{2\mathcal{S}_1}{a\gamma_1^3} - \left(\frac{a(1+b\gamma_2)}{a\gamma_1 + b\gamma_2\mathcal{S}_1 + ab\gamma_1\gamma_2} \right)^2 + 2 \left(\frac{a}{a\gamma_1 + b\gamma_2\mathcal{S}_1} \right)^2 \right] \\ &= -N_S(A_1 + \mathcal{I}_1 + A_2), \end{aligned} \quad (\text{B.42})$$

where $A_1 := -\frac{2}{a\gamma_1^3} \mathbb{E}[\mathcal{S}_1]$, $\mathcal{I}_1 := \mathbb{E} \left[-\left(\frac{a(1+b\gamma_2)}{a\gamma_1 + b\gamma_2\mathcal{S}_1 + ab\gamma_1\gamma_2} \right)^2 \right]$ and $A_2 := \mathbb{E} \left[2 \left(\frac{a}{a\gamma_1 + b\gamma_2\mathcal{S}_1} \right)^2 \right]$.

Let us first express A_1 . By using the first order moment expression in (B.8), A_1 can be written as:

$$A_1 = -\frac{2}{b\gamma_1^2\gamma_2} \exp(1/b\gamma_2) E_1(1/b\gamma_2). \quad (\text{B.43})$$

Second, let us express express \mathcal{I}_1 . After applying the LOTUS, \mathcal{I}_1 can be expressed as:

$$\begin{aligned} \mathcal{I}_1 &= \int_0^{+\infty} -\frac{a^2(1+b\gamma_2)^2}{(a\gamma_1+b\gamma_2y)^2(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\ &= \int_0^{+\infty} -\frac{(1+b\gamma_2)^2 e^{-t}}{\gamma_1^2(1+b\gamma_2t)^2(1+b\gamma_2(t+1))} dt, \quad (\text{change of variables: } t = y/a\gamma_1), \end{aligned} \quad (\text{B.44})$$

which is a non-closed form expression and requires a numerical integration.

Finally, A_3 is computed by using the LOTUS and can be written as:

$$\begin{aligned} A_2 &= \int_0^{+\infty} \frac{2a^2(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)}{(a\gamma_1+b\gamma_2x)^4} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\ &= \frac{2}{\gamma_1^2} \int_0^{+\infty} \frac{1+b\gamma_2(t+1)}{(1+b\gamma_2t)^4} e^{-t} dt, \quad (\text{change of variables: } t = y/a\gamma_1), \\ &= \frac{2}{\gamma_1^2} \left(\int_0^{+\infty} \frac{e^{-t}}{(1+b\gamma_2t)^3} dt + \int_0^{+\infty} \frac{b\gamma_2 e^{-t}}{(1+b\gamma_2t)^4} dt \right) \\ &= \frac{2}{\gamma_1^2} \left(\int_1^{+\infty} \frac{\exp\left(-\frac{u-1}{b\gamma_2}\right)}{b\gamma_2 u^3} du + \int_1^{+\infty} \frac{\exp\left(-\frac{u-1}{b\gamma_2}\right)}{u^4} du \right) \\ &= \frac{2 \exp(1/b\gamma_2)}{\gamma_1^2} \left(\int_1^{+\infty} \frac{\exp(u/b\gamma_2)}{b\gamma_2 u^3} du + \int_1^{+\infty} \frac{\exp(u/b\gamma_2)}{u^4} du \right) \\ &= \frac{2 \exp(1/b\gamma_2)}{\gamma_1^2} \left(\frac{E_3(1/b\gamma_2)}{b\gamma_2} + E_4(1/b\gamma_2) \right), \end{aligned} \quad (\text{B.45})$$

where $E_n(x) := \int_1^{+\infty} \frac{e^{xt}}{t^n} dt$ is defined as the generalized exponential integral and follows the following recurrence [1]:

$$\forall n \geq 1, \quad E_{n+1}(x) = \frac{1}{n} (e^{-x} - xE_n(x)). \quad (\text{B.46})$$

Let us define $y := 1/b\gamma_2$. When using the recurrence (B.46), A_2 's expression can be

simplified as the following:

$$\begin{aligned}
A_2 &= \frac{2e^y}{\gamma_1^2} (yE_3(y) + E_4(y)) \\
&= \frac{2e^y}{\gamma_1^2} \left(yE_3(y) + \frac{1}{3} (e^{-y} - yE_3(y)) \right) && \text{using recurrence for } E_4 \\
&= \frac{2}{3\gamma_1^2} (2ye^y E_3(y) + 1) \\
&= \frac{2}{3\gamma_1^2} \left(2ye^y \frac{1}{2} (e^{-y} - yE_2(y)) + 1 \right) && \text{using recurrence for } E_3 \\
&= \frac{2}{3\gamma_1^2} (-y^2 e^y E_2(y) + y + 1) \\
&= \frac{2}{3\gamma_1^2} (-y^2 e^y (e^{-y} - yE_1(y)) + y + 1) && \text{using recurrence for } E_2 \\
&= \frac{2}{3\gamma_1^2} (y^3 e^y E_1(y) - y^2 + y + 1).
\end{aligned} \tag{B.47}$$

Thus, A_2 expression can be written as:

$$A_2 = \frac{2}{3\gamma_1^2} \left(\frac{\exp(1/b\gamma_2)}{(b\gamma_2)^3} - \frac{1}{(b\gamma_2)^2} + \frac{1}{b\gamma_2} + 1 \right). \tag{B.48}$$

Thus, $[\mathbf{I}(\gamma)]_{11} = -N_S(A_1 + \mathcal{I}_1 + A_2)$.

B.1.3.1.2 Computing $[\mathbf{I}(\gamma)]_{22}$

Plugging (B.6) into (3.23) yields the expression of $[\mathbf{I}(\gamma)]_{22}$ which can be written as:

$$\begin{aligned}
[\mathbf{I}(\gamma)]_{22} &:= -\mathbb{E} \left[\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_2^2} \right] \\
&= -N_S(B_1 + B_2),
\end{aligned} \tag{B.49}$$

with $B_1 := \mathbb{E} \left[-\left(\frac{b(\mathbf{S}_1 + a\gamma_1)}{a\gamma_1 + b\gamma_2 \mathbf{S}_1 + ab\gamma_1 \gamma_2} \right)^2 \right]$ and $B_2 := \mathbb{E} \left[2 \left(\frac{b\mathbf{S}_1}{a\gamma_1 + b\gamma_2 \mathbf{S}_1} \right)^2 \right]$.

We simplify B_1 and B_2

$$\begin{aligned}
B_1 &= \int_0^{+\infty} -\left(\frac{b(y+a\gamma_1)}{a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2}\right)^2 \frac{(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)}{(a\gamma_1+b\gamma_2y)^2} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= \int_0^{+\infty} -\frac{b^2(y+a\gamma_1)^2}{(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)(a\gamma_1+b\gamma_2y)^2} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= b^2 \int_0^{+\infty} -\frac{(t+1)^2 e^{-t}}{(1+b\gamma_2(t+1))(1+b\gamma_2t)^2} dt \\
B_2 &= \int_0^{+\infty} 2\left(\frac{by}{a\gamma_1+b\gamma_2y}\right)^2 \frac{(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)}{(a\gamma_1+b\gamma_2y)^2} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= \int_0^{+\infty} \frac{(by)^2(a\gamma_1+b\gamma_2y+ab\gamma_1\gamma_2)}{(a\gamma_1+b\gamma_2y)^4} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= 2b^2 \int_0^{+\infty} \frac{t^2(1+b\gamma_2(t+1))e^{-t}}{(1+b\gamma_2t)^4} dt, \quad (\text{change of variables: } t = y/a\gamma_1).
\end{aligned} \tag{B.50}$$

Hence, we have

$$[\mathbf{I}(\gamma)]_{22} = -N_S \mathcal{I}_2, \tag{B.51}$$

where

$$\begin{aligned}
\mathcal{I}_2 &:= (B_1 + B_2) \\
&= \int_0^{+\infty} b^2 \left(-\frac{(t+1)^2}{(1+b\gamma_2(t+1))(1+b\gamma_2t)^2} + \frac{2t^2(1+b\gamma_2(t+1))}{(1+b\gamma_2t)^4} \right) e^{-t} dt,
\end{aligned} \tag{B.52}$$

which is not a closed-form expression and, like (B.44), requires a numerical integration.

B.1.3.1.3 Computing $[\mathbf{I}(\gamma)]_{12}$ and $[\mathbf{I}(\gamma)]_{21}$

Plugging (B.6) into (3.23), we can compute $[\mathbf{I}(\gamma)]_{12}$ which is equal to $[\mathbf{I}(\gamma)]_{21}$. $[\mathbf{I}(\gamma)]_{21}$ can be written as:

$$\begin{aligned}
[\mathbf{I}(\gamma)]_{21} &= [\mathbf{I}(\gamma)]_{12} \\
&= -\mathbb{E} \left[\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1 \partial \gamma_2} \right] \\
&= N_S (C_1 + C_2),
\end{aligned} \tag{B.53}$$

where $C_1 := \mathbb{E} \left[-\frac{ab\mathcal{S}_1}{(a\gamma_1+b\gamma_2\mathcal{S}_1+ab\gamma_1\gamma_2)^2} \right]$ and $C_2 := \mathbb{E} \left[\frac{2ab\mathcal{S}_1}{(a\gamma_1+b\gamma_2\mathcal{S}_1)^2} \right]$.

We simplify C_1 and C_2 by performing the change of variables $t = y/a\gamma_1$ and obtain the

following expressions:

$$\begin{aligned}
C_1 &= \int_0^{+\infty} -\frac{aby}{(a\gamma_1 + b\gamma_2 y + ab\gamma_1\gamma_2)(a\gamma_1 + b\gamma_2 y)^2} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= \int_0^{+\infty} -\frac{bte^{-t}}{\gamma_1(1 + b\gamma_2(t+1))(1 + b\gamma_2 y)^2} dt, \quad (\text{change of variables: } t = y/a\gamma_1), \\
C_2 &= \int_0^{+\infty} \frac{2aby(a\gamma_1 + b\gamma_2 y + ab\gamma_1\gamma_2)}{(a\gamma_1 + b\gamma_2 t)^4} \exp\left(-\frac{y}{a\gamma_1}\right) dy \\
&= \int_0^{+\infty} \frac{2bt(1 + b\gamma_2(t+1))e^{-t}}{\gamma_1(1 + b\gamma_2 t)^4} dt, \quad (\text{change of variables: } t = y/a\gamma_1).
\end{aligned} \tag{B.54}$$

Thus, $[\mathbf{I}(\gamma)]_{12}$ and $[\mathbf{I}(\gamma)]_{21}$ expression can be written as:

$$\begin{aligned}
[\mathbf{I}(\gamma)]_{12} &= [\mathbf{I}(\gamma)]_{21} \\
&= -\mathbb{E}\left[\frac{\partial^2 \ell(\mathbf{S}; \gamma)}{\partial \gamma_1 \partial \gamma_2}\right] \\
&= -N_S \mathcal{I}_3,
\end{aligned} \tag{B.55}$$

where

$$\begin{aligned}
\mathcal{I}_3 &:= C_1 + C_2 \\
&= \int_0^{+\infty} \frac{bt}{\gamma_1} \left(-\frac{1}{(1 + b\gamma_2(t+1))(1 + b\gamma_2 y)^2} + \frac{2(1 + b\gamma_2(t+1))}{(1 + b\gamma_2 t)^4} \right) e^{-t} dt,
\end{aligned} \tag{B.56}$$

which also requires a numerical integration method.

B.1.3.2 Integral approximation with Gaussian quadrature

In this section, we describe integral approximation methods based on Gaussian quadrature to implement the CRLB expressed in (3.24). First, the Gauss-Laguerre quadrature and its limitations are provided, and then the Gauss-Kronrod quadrature, which is implemented in the simulations, is described.

B.1.3.2.1 Gauss-Laguerre quadrature

In this section, we describe a technique that aims at approximating the integrals involved in \mathcal{I}_1 , \mathcal{I}_2 and \mathcal{I}_3 using Gauss-Laguerre quadrature (GLQ), which performs the following integral approximation:

$$\int_0^{+\infty} e^{-x} f_1(x) dx \approx \sum_{i=1}^{N_{\text{GLQ}}} w_i f_1(\chi_i), \tag{B.57}$$

where N_{GLQ} is the GLQ order and, for $i = 1, \dots, N_{\text{GLQ}}$, χ_i is the i th root of Laguerre polynomials $L_{N_{\text{GLQ}}}(x)$ and the weights are given by $w_i = \frac{\chi_i}{(N_{\text{GLQ}} + 1)^2 L_{N_{\text{GLQ}}+1}(\chi_i)^2}$ [1]. It is worth mentioning that the nodes $\chi_1, \dots, \chi_{N_{\text{GLQ}}}$ are chosen such that the approximation in (B.57) is exact if $f_1(x)$ is a polynomial of degree $2N_{\text{GLQ}} - 1$ or less on $[0, +\infty]$.

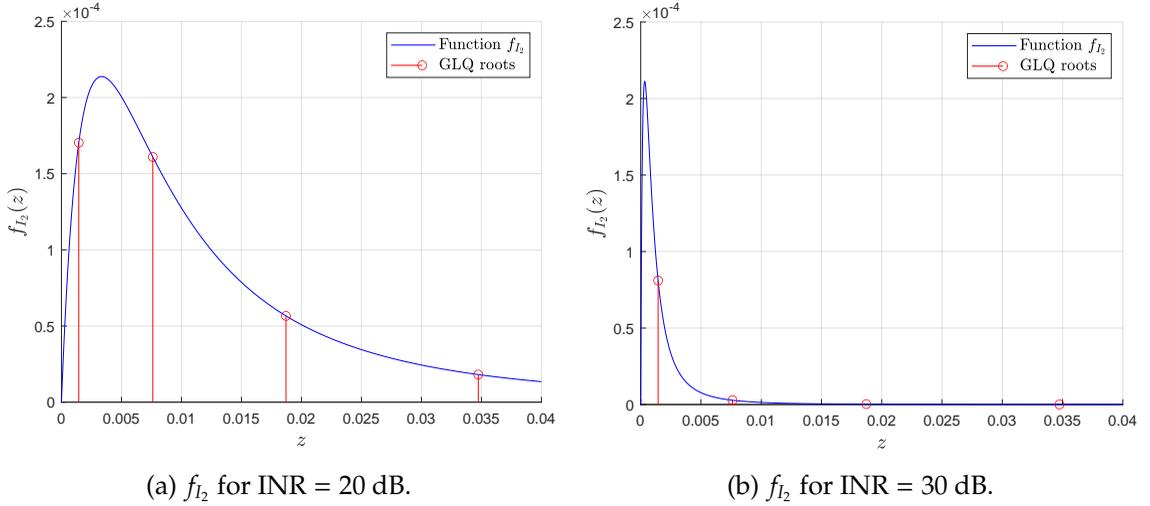


Figure B.1: Integral approximation using a $N_{\text{GLQ}} = 1000$ order GLQ.

Among the numerous types of Gaussian quadrature, GLQ seems to be the appropriate choice since the integrals involved in \mathcal{I}_1 , \mathcal{I}_2 and \mathcal{I}_3 contain the product of e^{-t} and a given function. However, static quadrature rules like GLQ fail to approximate integrals when the integrand function contains integrable singularities. In Fig. B.1a and B.1b, we respectively represent the following integrand function:

$$f_{\mathcal{I}_2} := \frac{2t^2(\omega + t + 1)^2 - (t + 1)^2(\omega + t)^2}{(\omega + t)^4(\omega + t + 1)}, \quad (\text{B.58})$$

which is used in (3.26) for $\text{INR} = 20$ and $\text{INR} = 30$ dB, and use the GLQ approximation with order $N_{\text{GLQ}} = 1000$. The first roots of the GLQ of the Laguerre polynomials with order $N_{\text{GLQ}} = 1000$ are given as follows: $\chi_1 = 0.001445$, $\chi_2 = 0.007614$ and $\chi_3 = 0.01871$. As represented in Figs B.1a and B.1b, we can observe that for increasing INR , the peak of $f_{\mathcal{I}_2}(z)$ moves to lower values of z . When the peak is located at the left of the first GLQ root $\chi_1 = 0.001445$ as illustrated in Fig. B.1b, its integral approximation using GLQ thus provides poor performance. To deal with this issue, one naive solution is to increase the already high order of the GLQ. Even though the complexity of the quadrature implementation increases, it can be computed by using the efficient Golub and Welsch algorithm to calculate Gauss-Laguerre roots [39]. Nevertheless, this scheme presents a scalability issue since the maximum value that the INR can reach is unknown, which makes it difficult to set the order N_{GLQ} . Another solution is to implement adaptive quadrature which only increases the number of interpolation points in specific subintervals which requires an increased precision. In the next section, we describe an example of adaptive quadrature.

B.1.3.2.2 Adaptive quadrature and Gauss-Kronrod quadrature

Adaptive quadrature is a numerical integration technique that consists in computing adaptive refined intervals and applying a quadrature rule to them. Each interval is recursively subdivided in subintervals until the integral error estimation is below a chosen threshold. In the end, the intervals containing singularities are allocated more interpolation points (higher Gaussian quadrature order), whereas the other intervals have fewer interpolation points (lower Gaussian quadrature order). Although its implementation is provided in [84], we described it in Algorithm B.1 for the sake of completeness.

Algorithm B.1: Adaptive quadrature algorithm [84].

procedure `integrate`(f_I , a , b , ϵ).

 Compute $Q \approx \int_a^b e^{-x} f_I(x) dx$.

 Compute $\epsilon_Q = \left| Q - \int_a^b e^{-x} f_I(x) dx \right|$.

if $\epsilon_Q > \epsilon$ **then**

 | Compute $m = \frac{a+b}{2}$.

 | Update $Q = \text{integrate}(f_I, a, m, \epsilon/2) + \text{integrate}(f_I, m, b, \epsilon/2)$.

 Return Q .

Adaptive quadrature methods have the drawback of requiring to compute the error estimation ϵ_Q defined as the following:

$$\epsilon_Q = \left| Q - \int_a^b e^{-x} f_I(x) dx \right|, \quad (\text{B.59})$$

with Q the proposed quadrature and f_I the integrand function. Stoer and Burlisch [111] have shown that the quadrature error to integrate f_I requires computing the $2N_{GLQ}$ th order partial derivative of f_I . Since this term is very complex to calculate in our case, we propose to use the Gauss-Kronrod quadrature (GKQ) which is an extension of the regular Gaussian quadrature.

GKQ is a numerical integration that combines the Gaussian quadrature and the adaptive quadrature methods to achieve higher accuracy while being robust to integrable singularities. To implement it, we first compute the Gaussian quadrature Q^{GQ} with order N_{GQ} which is obtained after computing its roots $\chi_1, \dots, \chi_{N_{GQ}}$ like in (B.57). Then, the GKQ is initialized by using the nodes from the Gaussian quadrature and adding $N_{GQ} + 1$ additional nodes $\zeta_1, \dots, \zeta_{N_{GQ}+1}$ to the same set, which leads to the quadrature Q^{GKQ} . These additional nodes are chosen in such a way that the resulting quadrature can exactly integrate polynomials of degree as high as possible. In other words, the resulting quadrature Q^{GKQ} is of higher order than Q^{GQ} which means its approximation is of higher performance. To that purpose, it can be proved that the additional nodes $\zeta_1, \dots, \zeta_{N_{GQ}+1}$ to compute GKQ are the roots of the Stieltjes polynomials [8]. Afterwards, to compute the adaptive quadrature procedure from Algorithm B.1, the quadrature error is approximated

as the following:

$$e^{GKQ} = |Q^{GKQ} - Q^{GQ}| \quad (\text{B.60})$$

Many works have shown the efficiency of using such an error approximation using properties on Stieltjes polynomials [41]. With this approximation error, an adaptive quadrature method can be implemented as proposed in [105] which provides a Matlab implementation of the GKQ by using the function integral. Its performance is illustrated by comparing the integration using GLQ with $N_{GLQ} = 1000$ nodes in Fig B.2a with the one using GKQ in Fig. B.2b with $N_{GKQ} = 30$ nodes which is much lower than with GLQ.

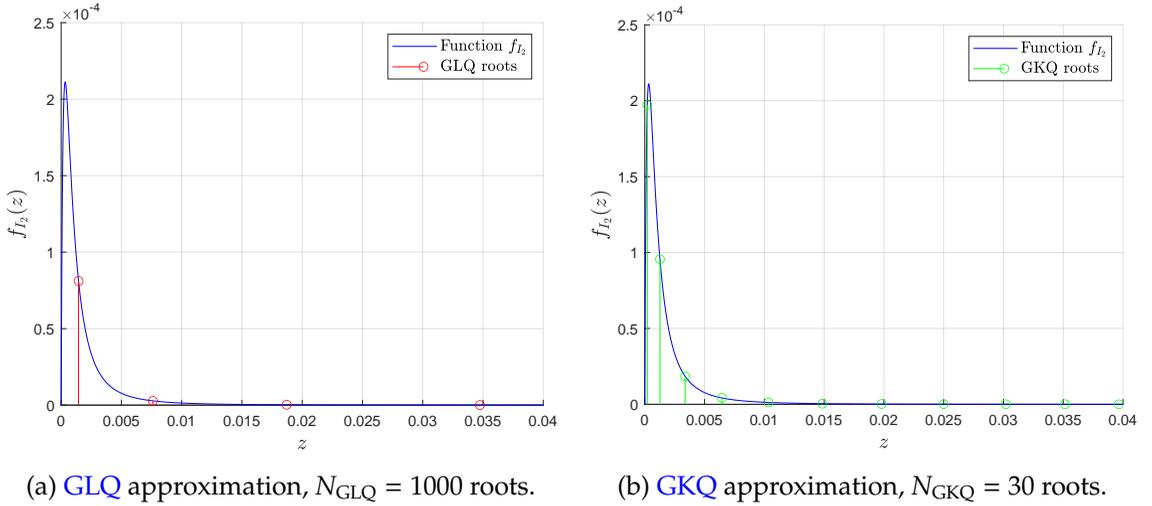


Figure B.2: Comparison of integral approximation between GLQ and GKQ approximation for $a = 1, b = 1, \text{SNR} = 30$ dB and $\text{INR} = 30$ dB.

B.2 Scheduling optimization with simultaneous transmissions

B.2.1 Proof of Lemma 3.4

Let $p \in \mathbb{N}^*$ and let us consider the networks \mathcal{G}_{2p-1} and \mathcal{G}'_{2p} composed of $2p-1$ and $2p$ nodes, respectively. After applying the "divide-in-2" strategy on \mathcal{G}_{2p-1} , i.e. the transmissions $\alpha_1 = (1, \dots, p)$ and $\alpha_2 = (p+1, \dots, 2p-1)$, we obtain the following equalities:

$$L_{div}(2p-1) = L_{div}(p), \quad (\text{B.61})$$

which is equal to $L(\mathcal{S}_{div}(2p))$.

B.2.2 Proof of Result 3.6

In this section, we want to prove the following statement that we refer to as $P(N)$:

$$\forall N \geq N_{min}, \quad L_{div}(N) = 2q + r, \quad (\text{B.62})$$

where $q := \left\lceil \log_2 \left(\frac{N}{N_{min}} \right) \right\rceil$ and $r := \left\lceil \frac{N}{2^q} \right\rceil$.

To this end, we distinguish the cases when N is even or odd.

B.2.2.1 Case N is even.

In the following, we prove $P(N)$ by strong induction, i.e. we first prove that the result is true for $N = N_{min}$, i.e. $P(N_{min})$, and then, after assuming that the propositions $P(N_{min}), \dots, P(N-1)$, we prove it for $P(N)$.

First, for proving $P(N_{min})$, we obtain for $N = N_{min}$ the following: $q = 0$ and $r = N$, which corresponds to the linear method that we use when $N \leq N_{min}$.

Second, to prove the induction step, we assume that $P(m)$ holds for all $m \in \{N_{min}, \dots, N-1\}$ and we prove it for $m = N$. Using the assumption that N is even, the number of steps of the "divide-in-2" scheduling applied to a network of N nodes can be expressed as:

$$L_{div}(N) = L_{div}(N/2) + 2. \quad (\text{B.63})$$

Using the induction hypothesis, (B.63) can then be written as:

$$L_{div}(N) = 2(q + 1) + r, \quad (\text{B.64})$$

with $q = \left\lceil \log_2 \left(\frac{N/2}{N_{min}} \right) \right\rceil$ and $r = \left\lceil \frac{N}{2^{q+1}} \right\rceil$. Moreover, using the following equality:

$$q + 1 = \left\lceil \log_2 \left(\frac{N/2}{N_{min}} \right) \right\rceil + 1, \quad (\text{B.65})$$

which can be written after simplification as:

$$q + 1 = \left\lceil \log_2 \left(\frac{N}{N_{min}} \right) \right\rceil, \quad (\text{B.66})$$

which proves the induction for N .

B.2.2.2 Case N is odd.

Since N is odd, there exists a value $p \in \mathbb{N}^*$ such that $N = 2p - 1$ with $p \geq 3$ since $N_{min} \geq 5$. According to Lemma 3.3, the following statement holds:

$$L_{div}(2p - 1) = L_{div}(2p). \quad (\text{B.67})$$

Plugging (3.48) into (B.67) yields the number of steps expression in the odd case which can be written as:

$$L_{div}(2p - 1) = 2q + r, \quad (\text{B.68})$$

with $q = \left\lceil \log_2 \left(\frac{2p}{N_{min}} \right) \right\rceil$ and $r = \left\lceil \frac{2p}{2^q} \right\rceil$.

Let $q' = \left\lceil \log_2 \left(\frac{2p-1}{N_{min}} \right) \right\rceil$ and $r' = \left\lceil \frac{2p-1}{2^q} \right\rceil$. On the one hand, we have $q' \leq q$. On the other hand, we have:

$$q - 1 = \left\lceil \log_2 \left(\frac{p}{N_{min}} \right) \right\rceil, \quad (\text{B.69})$$

which is strictly lower than q' for $p \geq 3$. Since q is an integer, then we can deduce that $q' = q$.

Similarly for r and r' , we have on the one hand $r' \leq r$, and on the other hand, we have:

$$r - 1 = \left\lceil \frac{2p}{2^q} \right\rceil - 1, \quad (\text{B.70})$$

which can be reexpressed as:

$$r - 1 = \left\lceil \frac{2p - 2^q}{2^q} \right\rceil, \quad (\text{B.71})$$

which is strictly lower than r' for $q \geq 1$, which is the case for $N \geq N_{min}$.

As a consequence, since we have proved that $q = q'$ and $r = r'$, the statement in (B.67) is proved.

B.2.3 Proof of Result 3.8

To have a successful scheduling method, all the nodes need to transmit exactly twice during the cycle of $m + 1$ steps, while at maximum m nodes are allowed to transmit simultaneously. This condition can be formulated as respecting the following inequality:

$$m(m + 1) \geq 2N. \quad (\text{B.72})$$

Solving (B.72) leads to the following inequality:

$$m \geq \frac{\sqrt{8N + 1} - 1}{2}, \quad (\text{B.73})$$

from which we deduce the smallest value of m respecting (B.72), which can be expressed as:

$$m \geq \left\lceil \frac{\sqrt{8N + 1} - 1}{2} \right\rceil. \quad (\text{B.74})$$

Since we have: $L(\mathcal{S}_{2TX}) = m + 1$, we deduce the number of steps of the "2-emission" scheduling, which can be written as:

$$L_{2TX} = \left\lceil \frac{\sqrt{8N + 1} + 1}{2} \right\rceil. \quad (\text{B.75})$$

B.2.4 Proof of Result 3.9

Since for $N \geq N_{limit}$, the "divide-in-2" method is applied, we can deduce from Result 3.6 that the number of steps for the mixed scheduling can be expressed as:

$$L_{mix} = 2q_{mix} + L_{2TX}(r_{mix}), \quad (\text{B.76})$$

with

$$q_{mix} = \left\lceil \log_2 \left(\frac{N}{N_{limit}} \right) \right\rceil, \quad (\text{B.77})$$

$$r_{mix} = \left\lceil \frac{N}{2^{q_{mix}}} \right\rceil. \quad (\text{B.78})$$

where N_{limit} is the value such that if the resulting subnetworks size is lower than N_{limit} , we stop applying the "divide-in-2" method to the subnetworks. Thus, the size of the largest subnetwork is r_{mix} and is then treated with the "2-transmission" method. Hence, the number of steps $L_{2TX}(r_{mix})$ can be computed using (3.50), which proves the result provided in (3.51).

Moreover, the optimal value of N_{limit} provided in (3.52) has been found numerically with an exhaustive search for $N_{limit} = 5, \dots, 1000$. For each value of N_{limit} , we have computed the function $N \mapsto L_{mix}(N, N_{limit})$ for $N = 5, \dots, 1000$. In the end, we have numerically found that for $N_{limit} \in \{21, \dots, 28\}$, $N \mapsto L_{mix}(N, N_{limit})$ for $N = 5, \dots, 1000$ is always below the other functions $L_{mix}(N, N'_{limit})$ for $N'_{limit} \notin \{21, \dots, 28\}$.

Publications

- C1.** D. Jia, X. Leturc, M. Assaad and C. J. Le Martret: “Rayleigh Channel Statistics Estimation Using SINR Samples Under Single Interference”, *IEEE 95th Vehicular Technology Conference (VTC2022-Spring)*, Helsinki (Finland), June 2022.
 - C2.** D. Jia, X. Leturc, M. Assaad and C. J. Le Martret: “Estimation des statistiques d’un canal de Rayleigh à partir de mesures du rapport signal à interférences plus bruit”, *XXVIIIème Colloque GRETSI*, Nancy (France), September 2022.
 - P1.** D. Jia, C. J. Le Martret, X. Leturc, and M. Assaad: “Procédé et dispositif d’ordonnement des transmissions dans un réseau de radiocommunication ad hoc procédé d’évaluation des interférences et réseau de radiocommunication associés”, déposé le 21/12/2021, demande de brevet no. 2114057.
 - P2.** D. Jia, X. Leturc, C. J. Le Martret and M. Assaad: “Procédé de communications hybrides NOMA/TDMA pour réseaux ad hoc clusterisés”, déposé le 19/06/2023, demande de brevet no. 2306291.
 - J1.** D. Jia, X. Leturc, C. J. Le Martret and M. Assaad: “Enhanced Hybrid TIM-TDMA to maximize the total throughput with Fairness Constraints”, submitted to *IEEE Open Journal of the Communications Society*.
 - J2.** D. Jia, X. Leturc, C. J. Le Martret and M. Assaad: “Distributed Enhanced TIM scheme in clustered ad hoc networks with reduced statistical CSI exchange”, submitted to *IEEE Communications Letters*.
-

Bibliography

- [1] M. Abramowitz, **Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables**. USA: Dover Publications, Inc., 1974. Cited pages [156](#), [160](#), and [163](#)
 - [2] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," **IEEE Transactions on Computers**, vol. C-23, no. 1, pp. 90–93, 1974. Cited page [109](#)
 - [3] L. Al Shalabi, Z. Shaaban, and B. Kasasbeh, "Data mining: A preprocessing engine," **Journal of Computer Science**, vol. 2, 09 2006. Cited page [113](#)
 - [4] P. Aquilina, "Advanced interference management techniques for future generation cellular networks," 2017, Ph. D. dissertation, University of Edinburgh. [Online]. Available: <https://era.ed.ac.uk/handle/1842/28714> Cited page [9](#)
 - [5] P. Aquilina and T. Ratnarajah, "Performance analysis of IA techniques in the MIMO IBC with imperfect CSI," **IEEE Transactions on Communications**, vol. 63, no. 4, pp. 1259–1270, 2015. Cited page [8](#)
 - [6] —, "Linear interference alignment in full-duplex MIMO networks with imperfect CSI," **IEEE Transactions on Communications**, vol. 65, no. 12, pp. 5226–5243, 2017. Cited page [8](#)
 - [7] —, "On the degrees of freedom of interference broadcast channels with topological interference management," **IEEE Transactions on Communications**, vol. 64, no. 4, pp. 1477–1489, 2016. Cited page [10](#)
 - [8] B. Baillaud and H. Bourget, **Correspondance d’Hermite et de Stieltjes I, II**. Gauthier-Villars, Paris, 1905. Cited page [165](#)
 - [9] G. Bakir, J. Weston, and B. Schölkopf, "Learning to find pre-images," in **Advances in Neural Information Processing Systems**, S. Thrun, L. Saul, and B. Schölkopf, Eds., vol. 16. MIT Press, 2003. [Online]. Available: <https://proceedings.neurips.cc/paper/2003/file/ac1ad983e08ad3304a97e147f522747e-Paper.pdf> Cited page [118](#)
 - [10] M. Bakshi, M. Kaddour, B. Jaumard, and L. Narayanan, "An efficient method to minimize TDMA frame length in wireless sensor networks," in **2015 IEEE Wireless Communications and Networking Conference (WCNC)**, 2015, pp. 825–830. Cited page [43](#)
 - [11] T.-W. Ban, "Compressed feedback using autoencoder based on deep learning for D2D communication networks," **IEEE Wireless Communications Letters**, pp. 1–1, 2023. Cited page [110](#)
 - [12] C. M. Bishop, **Pattern Recognition and Machine Learning (Information Science and Statistics)**. Berlin, Heidelberg: Springer-Verlag, 2006. Cited page [117](#)
 - [13] H. Boursard and Y. Kamp, **Auto-association by multilayer perceptrons and singular value decomposition**. Biological cybernetics, 1988, vol. 59. Cited page [118](#)
-

-
- [14] S. Boyd and L. Vandenberghe, **Convex Optimization**. Cambridge University Press, 2004. Cited pages 26 and 72
- [15] G. Bresler and D. Tse, "3 user interference channel: Degrees of freedom as a function of channel diversity," in **2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)**, 2009, pp. 265–271. Cited page 8
- [16] P. Bélis, R. Bonnefoi, H. Farès, and Y. Louët, "Optimal power and resource allocation for transmit power minimization in OFDMA-based NOMA networks," in **2019 IEEE Wireless Communications and Networking Conference (WCNC)**, 2019, pp. 1–7. Cited page 5
- [17] V. R. Cadambe and S. A. Jafar, "Interference alignment and degrees of freedom of the k -user interference channel," **IEEE Transactions on Information Theory**, vol. 54, no. 8, pp. 3425–3441, 2008. Cited page 6
- [18] E. J. Candes and B. Recht, "Exact low-rank matrix completion via convex optimization," in **2008 46th Annual Allerton Conference on Communication, Control, and Computing**, 2008, pp. 806–812. Cited page 10
- [19] E. J. Candes and T. Tao, "The power of convex relaxation: Near-optimal matrix completion," **IEEE Transactions on Information Theory**, vol. 56, no. 5, pp. 2053–2080, 2010. Cited page 10
- [20] J. Cao, X. Liu, W. Dong, T. Peng, R. Duan, Y. Yuan, and W. Wang, "A neural network based conflict-graph construction approach for ultra-dense networks," in **2018 IEEE Globecom Workshops (GC Wkshps)**, 2018, pp. 1–6. Cited pages 12 and 69
- [21] J. Cao, T. Peng, X. Liu, W. Dong, R. Duan, Y. Yuan, W. Wang, and S. Cui, "Resource allocation for ultradense networks with machine-learning-based interference graph construction," **IEEE Internet of Things Journal**, vol. 7, no. 3, pp. 2137–2151, 2020. Cited pages 12, 66, and 69
- [22] Z. Cao, J. Guo, C.-K. Wen, and S. Jin, "Deep-unfolding-based bit-level CSI feedback in massive MIMO systems," **IEEE Wireless Communications Letters**, vol. 12, no. 2, pp. 371–375, 2023. Cited page 134
- [23] M. A. Charafeddine, A. Sezgin, Z. Han, and A. Paulraj, "Achievable and crystallized rate regions of the interference channel with interference as noise," **IEEE Transactions on Wireless Communications**, vol. 11, no. 3, pp. 1100–1111, 2012. Cited page 5
- [24] S. Chen and R. S. Cheng, "Clustering for interference alignment in multiuser interference network," **IEEE Transactions on Vehicular Technology**, vol. 63, no. 6, pp. 2613–2624, 2014. Cited pages 8 and 30
- [25] F. De la Torre and M. Black, "Robust principal component analysis for computer vision," in **Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001**, vol. 1, 2001, pp. 362–369 vol.1. Cited page 114
- [26] S. Doumiati, "Advanced interference management techniques for future generation cellular networks," 2019, Ph. D. dissertation, American University of Beirut. [Online]. Available: <https://scholarworks.aub.edu.lb/handle/10938/10815> Cited page 10
- [27] S. Doumiati, M. Assaad, and H. A. Artail, "Topological interference management framework for device-to-device communication," **IEEE Wireless Communications Letters**, vol. 7, no. 4, pp. 602–605, 2018. Cited page 11
- [28] S. Doumiati, H. Artail, and M. Assaad, "Application of the topological interference management method in practical scenarios," in **2019 International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)**, 2019, pp. 1–4. Cited pages 13 and 133
-

-
- [29] S. Doumiati, M. Assaad, and H. A. Artail, "A framework of topological interference management and clustering for D2D networks," **IEEE Transactions on Communications**, vol. 67, no. 11, pp. 7856–7871, 2019. Cited pages 7 and 11
- [30] R. Frank, S. Zadoff, and R. Heimiller, "Phase shift pulse codes with good periodic correlation properties (corresp.)," **IEEE Transactions on Information Theory**, vol. 8, no. 6, pp. 381–382, 1962. Cited page 91
- [31] A. Galantai, "The theory of Newton's method," **Journal of Computational and Applied Mathematics**, vol. 124, no. 1, pp. 25–44, December 2000. Cited page 72
- [32] H. Gao, P. Smith, and M. Clark, "Theoretical reliability of MMSE linear diversity combining in Rayleigh-fading additive interference channels," **IEEE Transactions on Communications**, vol. 46, no. 5, pp. 666–672, 1998. Cited page 70
- [33] J. Gaveau, "Allocation des ressources pour la gestion dynamique du spectre dans les réseaux ad hoc clustérisés," July 2018, Ph.D. dissertation, CentraleSupélec, ED STIC. [Online]. Available: <http://www.theses.fr/2018SACL060> Cited page 12
- [34] J. Gaveau, X. Leturc, C. J. Le Martret, and M. Assaad, "Trial and error learning for dynamic distributed channel allocation in random medium," **IEEE Transactions on Wireless Communications**, vol. 20, no. 12, pp. 8177–8190, 2021. Cited page 1
- [35] C. Geng, H. Sun, and S. A. Jafar, "Multilevel topological interference management," in **2013 IEEE Information Theory Workshop (ITW)**, 2013, pp. 1–5. Cited page 5
- [36] —, "Multilevel topological interference management: A TIM-TIN perspective," **IEEE Transactions on Communications**, vol. 69, no. 11, pp. 7350–7362, 2021. Cited pages 14 and 30
- [37] S. Georgiou, C. Koukouvinos, and J. Seberry, **Hadamard Matrices, Orthogonal Designs and Construction Algorithms**, W. D. Wallis, Ed. Springer US, 2003. [Online]. Available: https://doi.org/10.1007/978-1-4613-0245-2_7 Cited page 91
- [38] S. Gherekhloo, A. Chaaban, and A. Sezgin, "Resolving entanglements in topological interference management with alternating connectivity," in **2014 IEEE International Symposium on Information Theory**, 2014, pp. 1772–1776. Cited page 10
- [39] G. H. Golub and J. H. Welsch, "Calculation of Gauss quadrature rules," **Mathematics of Computation**, vol. 23, no. 106, pp. 221–230, 1969. Cited page 164
- [40] K. Gomadam, V. R. Cadambe, and S. A. Jafar, "A distributed numerical approach to interference alignment and applications to wireless interference networks," **IEEE Transactions on Information Theory**, vol. 57, no. 6, pp. 3309–3322, 2011. Cited pages 8 and 30
- [41] P. Gonnet, "A review of error estimation in adaptive quadrature," **CoRR**, vol. abs/1003.4629, 2010. [Online]. Available: <http://arxiv.org/abs/1003.4629> Cited page 166
- [42] G. Goyal and S. Goel, "A comparative review on state-of-the-art clustering techniques for vehicular ad-hoc networks," in **2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC)**, 2022, pp. 276–279. Cited page 1
- [43] Y. Guo, C. Hu, T. Peng, H. Wang, and X. Guo, "Regression-based uplink interference identification and SINR prediction for 5G ultra-dense network," in **IEEE International Conference on Communications (ICC)**, 2020, pp. 1–6. Cited page 69
-

- [44] M. A. Gutierrez-Estevez, R. L. G. Cavalcante, S. Stańczak, J. Zhang, and H. Zhuang, "A distributed solution for proportional fairness optimization in load coupled OFDMA networks," in **2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)**, 2016, pp. 550–554. Cited page [27](#)
- [45] A. Haar, "Zur Theorie der orthogonalen Funktionensysteme," **Mathematische Annalen**, vol. 69, no. 3, pp. 331–371, 1910. [Online]. Available: <https://hal.science/hal-01333722> Cited page [109](#)
- [46] M. Hardt, "Understanding alternating minimization for matrix completion," in **2014 IEEE 55th Annual Symposium on Foundations of Computer Science**, 2014, pp. 651–660. Cited page [11](#)
- [47] B. Hassibi, "Topological interference alignment in wireless networks," in **Smart Antennas Workshop**, 2014. Cited pages [2](#), [7](#), [8](#), [10](#), [11](#), [13](#), [17](#), and [30](#)
- [48] A. G. Helmy, A. R. Hedayat, and N. Al-Dhahir, "Robust weighted sum-rate maximization for the multi-stream MIMO interference channel with sparse equalization," **IEEE Transactions on Communications**, vol. 63, no. 10, pp. 3645–3659, 2015. Cited page [8](#)
- [49] E. Hopf, "Mathematical problems of radiative equilibrium," vol. 31, no. 236, p. 123, 1934. Cited page [157](#)
- [50] S. A. Jafar, "Topological interference management through index coding," **IEEE Transactions on Information Theory**, vol. 60, no. 1, pp. 529–568, 2014. Cited pages [2](#), [8](#), [11](#), and [13](#)
- [51] M. Jain, N. Sharma, A. Gupta, D. Rawal, and P. Garg, "Performance analysis of NOMA assisted mobile ad hoc networks for sustainable future radio access," **IEEE Transactions on Sustainable Computing**, vol. 6, no. 2, pp. 347–357, 2021. Cited page [5](#)
- [52] D. Jia, X. Leturc, M. Assaad, and C. Le Martret, "Rayleigh channel statistics estimation using SINR samples under single interference," in **2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)**, 2022, pp. 1–5. Cited page [107](#)
- [53] D. Jia, C. Le Martret, X. Leturc, and M. Assaad, "Procédé de communications hybrides NOMA/TDMA pour réseaux ad hoc clusterisés," déposé le 19/06/2023, demande de brevet n°FR2306291. Cited page [62](#)
- [54] D. Jia, X. Leturc, C. Le Martret, and M. Assaad, "Procédé et dispositif d'ordonnement des transmissions dans un réseau de radiocommunication ad hoc procédé d'évaluation des interférences et réseau de radiocommunication associés," déposé le 21/12/2021, demande de brevet n°FR2114057. Cited page [107](#)
- [55] X. Jiang, B. Zheng, W.-P. Zhu, L. Wang, and X. Hou, "Topological interference management with inaccurate topology of network," **IEEE Communications Letters**, vol. 25, no. 11, pp. 3724–3728, 2021. Cited pages [9](#) and [11](#)
- [56] X. Jiang, B. Zheng, L. Wang, and X. Hou, "Clustering for topological interference management," **Chinese Journal of Electronics**, vol. 31, no. 5, pp. 844–850, 2022. [Online]. Available: <https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/cje.2021.00.277> Cited page [11](#)
- [57] X. Jiang, B. Zheng, W.-P. Zhu, L. Wang, and X. Hou, "Topological interference management via low-rank tensor completion for time-varying topology networks," **Signal Processing**, vol. 208, p. 108972, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0165168423000464> Cited page [11](#)

-
- [58] J. Jose, A. Ashikhmin, P. Whiting, and S. Vishwanath, "Scheduling and pre-conditioning in multi-user MIMO TDD systems," in **2008 IEEE International Conference on Communications**, 2008, pp. 4100–4105. Cited page [64](#)
- [59] J. Joung and E. Kurniawan, "CSI partitioning method with PCA-based compression for low-complexity feedback of large-dimensional channels," **IEEE Communications Letters**, vol. 21, no. 3, pp. 544–547, 2017. Cited page [110](#)
- [60] M. Kaddour, "A column generation approach to maximize capacity of multi-rate power controlled TDMA wireless sensor networks," in **ADHOC-NOW**, 2013. Cited page [43](#)
- [61] J. Kaleva, A. Tölli, and M. Juntti, "Decentralized sum rate maximization with QoS constraints for interfering broadcast channel via successive convex approximation," **IEEE Transactions on Signal Processing**, vol. 64, no. 11, pp. 2788–2802, 2016. Cited page [27](#)
- [62] H. Kallam, L. S. Cardoso, and J. Marie Gorce, "On the impact of normalized interference threshold for topological interference management," in **2020 European Conference on Networks and Communications (EuCNC)**, 2020, pp. 105–110. Cited pages [9](#) and [12](#)
- [63] V. Kalokidou, O. Johnson, and R. Piechocki, "A hybrid TIM-NOMA scheme for the SISO broadcast channel," in **2015 IEEE International Conference on Communication Workshop (ICCW)**, 2015, pp. 387–392. Cited page [14](#)
- [64] S. M. Kay, **Fundamentals of Statistical Signal Processing: Estimation Theory**. Prentice Hall Inc., 1997. Cited pages [71](#), [74](#), and [82](#)
- [65] P. d. Kerret and D. Gesbert, "Interference alignment with incomplete CSIT sharing," **IEEE Transactions on Wireless Communications**, vol. 13, no. 5, pp. 2563–2573, 2014. Cited page [8](#)
- [66] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: <https://arxiv.org/abs/1412.6980> Cited page [120](#)
- [67] L. Kleinrock, "Analysis of a time-shared processor," **Naval Research Logistics Quarterly**, vol. 11, pp. 59–73, 1964. Cited page [21](#)
- [68] Y.-C. Ko and M.-S. Alouini, "Estimation of Nakagami- m fading channel parameters with application to optimized transmitter diversity systems," **IEEE Transactions on Wireless Communications**, vol. 2, no. 2, pp. 250–259, 2003. Cited page [63](#)
- [69] S. Kompella, J. E. Wieselthier, A. Ephremides, H. D. Sherali, and G. D. Nguyen, "On optimal SINR-based scheduling in multihop wireless networks," **IEEE/ACM Transactions on Networking**, vol. 18, no. 6, pp. 1713–1724, 2010. Cited page [64](#)
- [70] N. Lee and R. W. Heath Jr, "Advanced interference management technique: potentials and limitations," **IEEE Wireless Communications**, vol. 23, no. 3, pp. 30–38, 2016. Cited page [5](#)
- [71] X. Leturc, P. Ciblat, and C. J. Le Martret, "Estimation of the Ricean K -factor from noisy complex channel coefficients," in **2016 50th Asilomar Conference on Signals, Systems and Computers**, 2016, pp. 1092–1096. Cited page [63](#)
- [72] —, "Estimation of the Ricean K factor in the presence of shadowing," **IEEE Communications Letters**, vol. 24, no. 1, pp. 108–112, 2020. Cited page [63](#)
-

- [73] L. Liang, S. Xie, G. Y. Li, Z. Ding, and X. Yu, "Graph-based resource sharing in vehicular communication," *IEEE Transactions on Wireless Communications*, vol. 17, no. 7, pp. 4579–4592, 2018. Cited page 12
- [74] Y.-C. Liang, C.-S. Liao, and X. Yi, "Topological interference management with adversarial topology perturbation: An algorithmic perspective," *IEEE Transactions on Communications*, vol. 70, no. 12, pp. 8153–8166, 2022. Cited page 10
- [75] J. J.-N. Liu and I. Chlamtac, **Mobile Ad Hoc Networking with a View of 4G Wireless: Imperatives and Challenges**. John Wiley & Sons, Ltd, 2004, ch. 1, pp. 1–45. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/0471656895.ch1> Cited page 1
- [76] W. Liu, K. Liu, L. Tian, C. Zhang, and Y. Yang, "Joint interference alignment and subchannel allocation in ultra-dense networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 7, pp. 7287–7296, 2022. Cited page 8
- [77] C. Lu, W. Xu, H. Shen, J. Zhu, and K. Wang, "MIMO channel information feedback using deep recurrent network," *IEEE Communications Letters*, vol. 23, no. 1, pp. 188–191, 2019. Cited page 110
- [78] X. Ma, J. Zhang, Y. Zhang, Z. Ma, and Y. Zhang, "A PCA-based modeling method for wireless MIMO channel," in **2017 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)**, 2017, pp. 874–879. Cited page 114
- [79] J. Mao, J. Gao, Y. Liu, and G. Xie, "On the ergodic capacity of fading cognitive radio MIMO Z-interference channel," *IEEE Communications Letters*, vol. 15, no. 11, pp. 1166–1168, 2011. Cited page 21
- [80] T. L. Marzetta, E. G. Larsson, H. Yang, and H. Q. Ngo, **Fundamentals of Massive MIMO**. Cambridge University Press, 2016. Cited pages 22 and 23
- [81] R. Massin, C. J. Le Martret, and P. Ciblat, "A coalition formation game for distributed node clustering in mobile ad hoc networks," *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3940–3952, 2017. Cited page 1
- [82] F. Maturana, M. C. Estela, and P. Adasme, "Precoder optimization for interference channels with interference alignment," in **2022 IEEE International Conference on Automation/XXV Congress of the Chilean Association of Automatic Control (ICA-ACCA)**, 2022, pp. 1–6. Cited page 8
- [83] C. McGoldrick, W. Dowling, and A. Bury, "Image coding using the singular value decomposition and vector quantization," in **Fifth International Conference on Image Processing and its Applications, 1995.**, 1995, pp. 296–300. Cited page 116
- [84] W. M. McKeeman, "Algorithm 145: Adaptive numerical integration by simpson's rule," *Commun. ACM*, vol. 5, no. 12, p. 604, dec 1962. [Online]. Available: <https://doi.org/10.1145/355580.369102> Cited page 165
- [85] L. Miretti, E. Björnson, and D. Gesbert, "Team MMSE precoding with applications to cell-free massive MIMO," *IEEE Transactions on Wireless Communications*, vol. 21, no. 8, pp. 6242–6255, 2022. Cited page 23
- [86] R. K. Mishra, K. Chahine, H. Kim, S. Jafar, and S. Vishwanath, "Distributed interference alignment for K-user interference channels via deep learning," in **2021 IEEE International Symposium on Information Theory (ISIT)**, 2021, pp. 2614–2619. Cited page 8
-

-
- [87] K. Mittal, S. Pathania, P. Reddy, and D. Rawal, "Channel state information feedback overhead reduction using arithmetic coding in massive mimo systems," in **2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN)**, 2016, pp. 328–331. Cited page [109](#)
- [88] S. Mohajer, S. N. Diggavi, C. Fragouli, and D. Tse, "Transmission techniques for relay-interference networks," in **2008 46th Annual Allerton Conference on Communication, Control, and Computing**, 2008, pp. 467–474. Cited page [5](#)
- [89] K. K. Mohan and S. S. Das, "Parametric estimation of SINR distribution using quantized SINR samples for maximizing average spectral efficiency," in **National Conference on Communications (NCC)**, 2021, pp. 1–6. Cited pages [66](#) and [69](#)
- [90] N. Naderializadeh and A. S. Avestimehr, "Interference networks with no CSIT: Impact of topology," **IEEE Transactions on Information Theory**, vol. 61, no. 2, pp. 917–938, 2015. Cited page [10](#)
- [91] P. NDIAYE, M. DIALLO, M. MBAYE, I. DIOP, and M. R. SEYE, "Efficient lossless feedback compression for FDD massive MIMO," in **2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)**, 2019, pp. 579–585. Cited page [109](#)
- [92] K. Pearson, "LIII. on lines and planes of closest fit to systems of points in space," **The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science**, vol. 2, no. 11, pp. 559–572, 1901. [Online]. Available: <https://doi.org/10.1080/14786440109462720> Cited page [114](#)
- [93] S. W. Peters and R. W. Heath, "Interference alignment via alternating minimization," in **2009 IEEE International Conference on Acoustics, Speech and Signal Processing**, 2009, pp. 2445–2448. Cited pages [8](#) and [30](#)
- [94] J. G. Proakis, **Digital Communications**. McGraw-Hill, 2001. Cited page [70](#)
- [95] A. Ramírez-Arroyo, L. García, A. Alex-Amor, and J. F. Valenzuela-Valdés, "An introduction to dimensionality reduction for propagation measurements," in **2022 16th European Conference on Antennas and Propagation (EuCAP)**, 2022, pp. 1–5. Cited page [110](#)
- [96] M. Razaviyayn, M. Sanjabi, and Z.-Q. Luo, "Linear transceiver design for interference alignment: Complexity and computation," **IEEE Transactions on Information Theory**, vol. 58, no. 5, pp. 2896–2910, 2012. Cited page [8](#)
- [97] B. Recht, M. Fazel, and P. A. Parrilo, "Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization," **SIAM Review**. Cited page [11](#)
- [98] L. Rose, S. M. Perlaza, C. J. Le Martret, and M. Debbah, "Self-organization in decentralized networks: A trial and error learning approach," **IEEE Transactions on Wireless Communications**, vol. 13, no. 1, pp. 268–279, 2014. Cited page [1](#)
- [99] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, **Learning Internal Representations by Error Propagation**. Cambridge, MA, USA: MIT Press, 1986, p. 318–362. Cited page [118](#)
- [100] I. Santamaria, O. Gonzalez, R. W. Heath Jr., and S. W. Peters, "Maximum sum-rate interference alignment algorithms for MIMO channels," in **2010 IEEE Global Telecommunications Conference GLOBECOM 2010**, 2010, pp. 1–6. Cited pages [8](#), [18](#), [22](#), [23](#), [24](#), and [30](#)
- [101] L. L. Scharf, **Statistical signal processing**. Addison–Wesley, 1969. Cited pages [63](#) and [91](#)
-

-
- [102] D. A. Schmidt, C. Shi, R. A. Berry, M. L. Honig, and W. Utschick, "Minimum mean squared error interference alignment," in **2009 Conference Record of the Forty-Third Asilomar Conference on Signals, Systems and Computers**, 2009, pp. 1106–1110. Cited page 8
- [103] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in **Artificial Neural Networks – ICANN'97**, W. Gerstner, A. Germond, M. Hasler, and J.-D. Nicoud, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997, pp. 583–588. Cited page 117
- [104] S. Shalev-Shwartz and S. Ben-David, **Understanding Machine Learning: From Theory to Algorithms**. USA: Cambridge University Press, 2014. Cited pages 114, 115, and 116
- [105] L. Shampine, "Vectorized adaptive quadrature in MATLAB," **Journal of Computational and Applied Mathematics**, vol. 211, no. 2, pp. 131–140, 2008. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S037704270600700X> Cited page 166
- [106] M. J. Sherman, "Directional broadcast method," 2017, u.S. Patent 2017/011 1905 A1. Cited page 95
- [107] Y. Shi, J. Zhang, and K. B. Letaief, "Low-rank matrix completion for topological interference management by Riemannian pursuit," **IEEE Transactions on Wireless Communications**, vol. 15, no. 7, pp. 4703–4717, 2016. Cited page 11
- [108] Y. Shi, B. Mishra, and W. Chen, "Topological interference management with user admission control via Riemannian optimization," **IEEE Transactions on Wireless Communications**, vol. 16, no. 11, pp. 7362–7375, 2017. Cited page 11
- [109] A. Shojaeifard, K. A. Hamdi, E. Alsusa, D. K. C. So, and J. Tang, "Exact SINR statistics in the presence of heterogeneous interferers," **IEEE Transactions on Information Theory**, vol. 61, no. 12, pp. 6759–6773, December 2015. Cited pages 75 and 154
- [110] N. Z. Shor, **Minimization methods for non-differentiable functions**. Springer Series in Computational Mathematics, 1985. Cited pages 138 and 139
- [111] J. Stoer and R. Bulirsch, **Introduction to Numerical Analysis**, 3rd ed., ser. Texts in Applied Mathematics. Springer-Verlag, 2002. Cited page 165
- [112] X. Su, Y. Yuan, and Q. Wang, "Performance analysis of rate splitting in K-user interference channel under imperfect CSIT: Average sum rate, outage probability and SER," **IEEE Access**, vol. 8, pp. 136 930–136 946, 2020. Cited page 21
- [113] H. Sung, S.-H. Park, K.-J. Lee, and I. Lee, "Linear precoder designs for K-user interference channels," **IEEE Transactions on Wireless Communications**, vol. 9, no. 1, pp. 291–301, 2010. Cited pages 8, 22, 23, 24, 25, 30, and 135
- [114] D. Tse and P. Viswanath, **Fundamentals of Wireless Communication**. Cambridge University Press, 2005. Cited page 5
- [115] P.-H. Tseng and T.-C. Lee, "Numerical evaluation of exponential integral: Theis well function approximation," **Journal of Hydrology**, vol. 205, no. 1, pp. 38–51, February 1998. Cited page 83
- [116] A. Vaish and M. Kumar, "A new image compression technique using principal component analysis and Huffman coding," in **2014 International Conference on Parallel, Distributed and Grid Computing**, 2014, pp. 301–305. Cited page 110
- [117] L. Van Der Maaten, E. Postma, and J. Van den Herik, "Dimensionality reduction: a comparative review," **Journal of Machine Learning Research**, vol. 10, pp. 66–71, 2009. Cited page 110
-

-
- [118] C. Wang, D. Deng, L. Xu, W. Wang, and F. Gao, "Joint interference alignment and power control for dense networks via deep reinforcement learning," **IEEE Wireless Communications Letters**, vol. 10, no. 5, pp. 966–970, 2021. Cited page 8
- [119] L. Wang, H. Tang, H. Wu, and G. L. Stüber, "Resource allocation for D2D communications underlay in Rayleigh fading channels," **IEEE Transactions on Vehicular Technology**, vol. 66, no. 2, pp. 1159–1170, 2017. Cited page 12
- [120] Q. Wang, "Kernel principal component analysis and its applications in face recognition and active shape models," **CoRR**, vol. abs/1207.3538, 2012. [Online]. Available: <http://arxiv.org/abs/1207.3538> Cited page 117
- [121] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," **IEEE Wireless Communications Letters**, vol. 7, no. 5, pp. 748–751, 2018. Cited page 110
- [122] C. Xu, L. Zhang, L. Song, Y. Zhao, and B. Jiao, "Feedback compression for time-correlated MIMO block-fading channels using Huffman coding," in **2011 IEEE 13th International Conference on Communication Technology**, 2011, pp. 411–415. Cited page 109
- [123] K. Yang, Y. Shi, and Z. Ding, "Low-rank matrix completion for mobile edge caching in Fog-RAN via Riemannian optimization," in **2016 IEEE Global Communications Conference (GLOBECOM)**, 2016, pp. 1–6. Cited page 11
- [124] K. Yang, Y. Shi, J. Zhang, Z. Ding, and K. B. Letaief, "A low-rank approach for interference management in dense wireless networks," in **2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)**, 2016, pp. 708–712. Cited pages 2, 13, and 17
- [125] Q. Yang, M. B. Mashhadi, and D. Gündüz, "Deep convolutional compression for massive MIMO CSI feedback," in **2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)**, 2019, pp. 1–6. Cited page 110
- [126] X. Yi and D. Gesbert, "Topological interference management with transmitter cooperation," **IEEE Transactions on Information Theory**, vol. 61, no. 11, pp. 6107–6130, 2015. Cited page 10
- [127] X. Zhang, J. Wang, and H. V. Poor, "NOMA-based statistical QoS provisioning for wireless ad-hoc networks with finite blocklength," in **2019 IEEE Global Communications Conference (GLOBECOM)**, 2019, pp. 1–6. Cited page 5
- [128] C. Zhao, X. Xu, Z. Gao, and L. Huang, "A coloring-based cluster resource allocation for ultra dense network," in **2016 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC)**, 2016, pp. 1–5. Cited page 12
- [129] J. Zhao, X. He, H. Wang, X. Zheng, J. Lv, T. Luo, and X. Hou, "Cluster-based resource selection scheme for 5G V2X," in **2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)**, 2019, pp. 1–5. Cited page 1
- [130] M. Zhou, Z. Ding, K. Wang, S. Zhang, and X. Wang, "Achievable sum-rate of full-duplex-based small cells with clustered interference alignment," **IEEE Transactions on Vehicular Technology**, vol. 70, no. 8, pp. 8367–8372, 2021. Cited page 30
- [131] —, "Achievable sum-rate of full-duplex-based small cells with clustered interference alignment," **IEEE Transactions on Vehicular Technology**, vol. 70, no. 8, pp. 8367–8372, 2021. Cited page 8
-