



## Methods for Communication and Initial Access with RadioWeaves

<b>Project number:</b>	101013425
<b>Project acronym:</b>	<b>REINDEER</b>
<b>Project title:</b>	REsilient INteractive applications through hyper Diversity in Energy Efficient RadioWeaves technology
<b>Project Start Date:</b>	1 <sup>st</sup> January 2021
<b>Duration:</b>	42 months
<b>Programme:</b>	H2020-ICT-52-2020
<b>Deliverable Type:</b>	Report
<b>Reference Number:</b>	ICT-52-2020 / D3.2 / 1.00
<b>Workpackage:</b>	WP3
<b>Due Date:</b>	30 <sup>st</sup> September, 2022
<b>Actual Submission Date:</b>	5 <sup>th</sup> October, 2022
<b>Responsible Organisation:</b>	KU Leuven
<b>Editor:</b>	Gilles Callebaut
<b>Dissemination Level:</b>	PU
<b>Revision:</b>	1.00
<b>Abstract:</b>	This deliverable will provide a framework for communication and initial access in networks with RadioWeaves, including both communication-theoretic analysis and numerical simulations from Task 3.1 and Task 3.3, as well as the key insights and guidelines for practical design. D3.2 will in particular demonstrate how to reach the application needs set in WP1.
<b>Keywords:</b>	RadioWeaves, initial access, grant-free access, synchronisation, federations, resource allocation



The REINDEER project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101013425.

## **Editor**

Gilles Callebaut (KU Leuven)

## **Contributors**

Gilles Callebaut, Liesbet Van der Perre, François Rottenberg (KU Leuven)

Jianan Bai, Unnikrishnan Kunnath Ganesan, Sai Subramanyam Thoota, Erik G. Larsson (LiU)

Emma Fitzgerald, Liang Liu, Jesus Rodriguez Sanchez (ULund)

Pål Frenger, Andres Reial (EAB)

Benjamin J. B. Deutschmann, Thomas Wilding, Klaus Witrissal (TU Graz)

## **Internal reviewers**

Ove Edfors (ULUND) and Lieven De Strycker (KU Leuven)

## **Disclaimer**

*The information in this document is provided as is, and no guarantee or warranty is given that the information is fit for any particular purpose. The content of this document reflects only the author's view – the European Commission is not responsible for any use that may be made of the information it contains. The users use the information at their sole risk and liability.*

## Executive Summary

In this deliverable, different aspects of initial access specific to RadioWeaves networks are studied. Initial access in the broader context includes all processes and phases needed to establish communication. Following this, three phases are defined and investigated, *i.e.*, network set-up, de-registered and registered mode operation. The former embodies all synchronisation and calibration required to perform coherent processing. In de-registered mode the user equipments (UEs) are unknown to the network and need to be *registered*. After registration, all communication is in registered mode (still requiring multiple access schemes).

A frequency synchronisation protocol is proposed to achieve coherent operation between geographically distributed antenna arrays. This is required during the set-up phase of the network and, afterwards, to remain coherent. Subsequently, during de-registered mode, challenges such as powering energy neutral (EN) devices and obtaining system information are examined. An initial access scheme to power EN devices without channel state information (CSI) is given.

The high degrees-of-freedom in RadioWeaves, due to the number of spatially separated resources, demand and allow to be exploited to optimise routing and resource allocation. To do so, we explore routing resource allocation in RadioWeaves systems, where we are inherently constrained by the connections between contact service points (CSPs) and edge computing service points (ECSPs). Building on top of that, we devise a framework to orchestrate federations to dynamically allocate resources depending on the served UEs, channel conditions, application requirements and available resources.

As a consequence of the high variety of application characteristics *e.g.*, low power or low latency constraints, different access techniques are devised. Grant-based access schemes are studied for ultra-reliable low-latency communications (URLLC). Different activity detection schemes are proposed to accommodate the support of simultaneous access for a high number of devices. For this, grant-free access schemes are presented exploiting features unique to RadioWeaves systems. We explore the impact of antenna deployment topologies on the activity detection schemes.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Deliverable Outline . . . . .	1
1.2	Improving network efficiency by separation of concerns . . . . .	3
<b>I</b>	<b>Network Set-up and De-Registered Mode Operation</b>	<b>6</b>
<b>2</b>	<b>Carrier Frequency Synchronization</b>	<b>7</b>
2.1	System Model . . . . .	8
2.2	BeamSync Protocol . . . . .	9
2.3	Simulations . . . . .	13
2.4	Conclusion . . . . .	15
<b>3</b>	<b>Initial Access Phase</b>	<b>17</b>
3.1	Broadcast Information . . . . .	17
3.2	Wireless Power Transfer . . . . .	20
3.3	Conclusion . . . . .	23
<b>II</b>	<b>Registered Mode Operation</b>	<b>24</b>
<b>4</b>	<b>Establishing Routing in RadioWeaves</b>	<b>25</b>
4.1	Introduction . . . . .	25
4.2	Segmented fronthaul and related routing tasks . . . . .	26
4.3	System model . . . . .	27
4.4	Performance evaluation . . . . .	30
4.5	Conclusion . . . . .	33
<b>5</b>	<b>Federations: CSP Resource Allocation for Diverse Applications</b>	<b>35</b>
5.1	Introduction . . . . .	35
5.2	Framework Design . . . . .	41
5.3	Federation Orchestration . . . . .	48
5.4	Conclusion . . . . .	53
<b>6</b>	<b>Grant-Based and Grant-Free Access Techniques</b>	<b>54</b>
6.1	Introduction . . . . .	54
6.2	Grant-Based Access Techniques For URLLC Applications . . . . .	54
6.3	System Model and Evaluation Metrics of Grant-Free Access Techniques . . . . .	59
6.4	New Distributed AMP Algorithm for Activity Detection in Grant-Free Access . . . . .	61

6.5	Exploiting Partial Channel State Information in Grant-Free Access . . . . .	66
6.6	Impact of Antenna Deployment Topology on Performance of Grant-Free Access Detection Methods . . . . .	76
6.7	Conclusion . . . . .	83
<b>7</b>	<b>Conclusion</b>	<b>84</b>
	<b>Bibliography</b>	<b>90</b>
<b>A</b>	<b>Appendix</b>	<b>91</b>
A.1	Derivation of Distributed approximate message passing (AMP) Algorithm . . . . .	92
A.2	Device Activity Detection with Partial CSI in RadioWeaves . . . . .	95

# Glossary

**AMP** approximate message passing.

**AP** access point.

**AR** augmented reality.

**BS** base station.

**CDF** cumulative distribution function.

**CFO** carrier frequency offset.

**CP** cyclic prefix.

**CPU** central-processing unit.

**CQI** channel quality indicator.

**CRLB** Cramér-Rao lower bound.

**CRS** cell reference signal.

**CS** compressed sensing.

**CSI** channel state information.

**CSP** contact service point.

**DCC** dynamic cooperation clustering.

**DL** downlink.

**D-MIMO** distributed MIMO.

**eCDF** empirical cumulative distribution function.

**ECSP** edge computing service point.

**eMBB** enhanced Mobile Broadband.

**EN** energy neutral.

**FH** fronthaul.

**FIM** Fisher information matrix.

**GF** grant-free.

**gNB** Next Generation Node B.

**GPU** graphics processing unit.

**GSM** Global System for Mobile Communications.

**HARQ** hybrid automatic repeat request.

**HyMPRo** Hybrid Multi-Path Routing algorithm.

**i.i.d.** independently and identically distributed.

**IoT** Internet of Things.

**KPI** key performance indicator.

**LDPC** low-density parity-check.

**LMMSE** least minimum mean square error.

**LoS** line-of-sight.

**LP** linear programming.

**LPWAN** low-power wide-area network.

**LRT** likelihood-ratio test.

**LSFC** large-scale fading component.

**MCS** modulation and coding scheme.

**MF** matched filter.

**MMSE** minimum mean square error.

**mMTC** massive machine-typed communication.

**MPC** multipath component.

**MRT** maximum ratio transmission.

**MSE** mean square error.

**NOMA** non-orthogonal multiple access.

**NR** New Radio.

**OFDM** orthogonal frequency-division multiplexing.

**OFDM-IM** OFDM with index modulation.

**PAPR** peak-to-average power ratio.

**PDCCH** physical downlink control channel.

**PDSCH** physical downlink shared channel.

**PG** path gain.

**QoS** quality-of-service.

**RMSE** root-mean-square error.

**RREQ** route request packet.

**RW** RadioWeaves.

**SFN** single frequency network.

**SINR** signal-to-interference-plus-noise ratio.

**SMC** specular multipath component.

**SNR** signal-to-noise ratio.

**SSB** synchronisation signal block.

**TX** transmitter.

**UE** user equipment.

**UL** uplink.

**ULA** uniform linear array.

**URLLC** ultra-reliable low-latency communications.

**UV** unmanned vehicle.

**WPT** wireless power transfer.

**ZF** zero forcing.

**ZMCSCG** zero mean circularly symmetric complex Gaussian.

# Chapter 1

## Introduction

Prior to any form of communication, the network itself needs to be initialised. After this setup, devices can perform initial access to derive information regarding the network (or cell) and do up-link synchronisation. Subsequently, the devices can request to use the available radio resources to communicate wirelessly. Here, we focus on the different steps and define the terminology and concepts used throughout the document. It also outlines the different chapters of this deliverable. The document is divided into two parts. Part I focuses on all phases done in de-registered mode. The user equipments (UEs) are in this mode when the network is unaware of their presence or intent. Part II addresses the registered mode operations, where the UEs have performed the initial access procedure and are thus known to the network. To improve the network efficiency of the RadioWeaves system, we considered the method of separation of concerns, which is elaborated first, beforehand in Section 1.2. The section highlights the methodology and pitfalls that need to be taken into account when designing methods for communication and initial access in RadioWeaves.

The presented work has adopted the terminology introduced in Deliverable D2.1 [37, Chapter 2]. To understand this deliverable, we will briefly recapitalise the major deviations from conventional terminology for this deliverable. As a consequence of the distributed and densely populated resources of the RadioWeaves infrastructure, in combination with a variety of application requirements, new terminology was introduced. A RadioWeaves network consists of one or more edge computing service points (ECSPs) connecting different contact service points (CSPs). The ECSPs are connected to the back-haul and serve as dedicated processing units, while CSPs are equivalent to the conventional access points (APs). We adopted a new terminology as the current APs only provide communication; this in contrast to CSPs, which provides power, communication and sensing. For a more elaborate definition and more details, we refer the readers to consult the defined terminology in [25] and Deliverable D2.1 [37, Chapter 2].

### 1.1 Deliverable Outline

**Network Set-up** Next to this, due to the high number of resources in RadioWeaves, the network can optimise its resource allocation and routing through the network, which is discussed in Chapters 4 and 5. In contrast to other systems, the RadioWeaves setup can be physically tailored to the devices and applications in the network. Also, new services are supported, *e.g.*, wireless power transfer (WPT) requiring different resources and network structure than, *e.g.*, communications.

**Uplink Synchronization and Initial Access** Although being widely used in different contexts, we will make here a clear distinction between initial access and all other forms of retrieving access. The term *initial access* will denote all procedures related to getting the first access to the network. This entails that no prior information is available at the UE or network. The initial access consists of, e.g., uplink synchronisation and random access procedures to retrieve information of and access to the network. Besides synchronisation, energy neutral (EN) devices need to be discovered and powered during initial access. This is elaborated in Chapter 3. Before the initial access, the UE is in de-registered mode, meaning that is not known to or admitted by the network. After a successful initial access procedure, the UE is known to the network and enters the registered mode.

Note, that the initial access procedures are not considered in the latency performance analysis. In practical scenarios, low-latency applications require only a low latency after the first connection to the network.

**Uplink and Downlink Communication** There are two approaches taken to send/receive uplink/downlink data, i.e., i) grant-free and ii) grant-based access. In the latter, the intended resources are scheduled by the base station, being Next Generation Node B (gNB) in 5G New Radio (NR) terminology. For the uplink, each scheduled device will get a scheduling grant containing the reserved time/frequency/spatial resources. While the grant-based approach has worked for previous cellular standards, it inherently increases the latency [31] (w.r.t. intent of uplink transmission) and also mandates that the UE continuously monitors the downlink control channel, drastically reducing the battery lifetime, which is detrimental for energy-neutral devices. On top of that, the devices need to first request uplink access by issuing a scheduling request, further impacting the latency and energy consumption. Evidently, for a number of use cases, as discussed in [40], a grant-free method is preferred. Several protocols are presented in Chapter 3 using a grant-free approach to detect active devices.

## 1.2 Improving network efficiency by separation of concerns in Distributed-Massive MIMO

A good system design is the foundation that enables an energy-efficient design of an entire network. The current 3GPP 5G NR specification is an ultra-lean standard that is already a powerful enabler of low network-energy usage. The most important asset from an energy-performance perspective going forward to 6G is to maintain the ultra-lean properties that NR is based on, enabling up to 160 m sec of transmission-free periods [3]. Here we discuss additional means to further enhance the energy efficiency of 6G (beyond that of 5G/NR), and motivate why these potential enhancements are especially important for energy efficient operation of distributed MIMO (D-MIMO) deployments, such as RadioWeaves.

The NR specifications today provide signalling support for idle-mode UEs to see the full set of beams, bands, and nodes that are configured and that can be made available in active mode. The need for this observability is questionable, given the cost of the associated transmissions resulting in reduced sleep possibilities. In a D-MIMO system, with a very large number of service points each with several beams, the total number of always active beams would quickly become very large. Such a design can easily result in an unnecessarily high network energy consumption, for the relatively simple tasks of supporting basic idle mode network functions such as system information broadcast, paging, and random access. Hence, the introduction of D-MIMO in future standards calls for better separation of signalling to support idle mode functions and signalling to support user plane data transfer.

In addition, one should strive to design functionality to be self-contained, refraining from the reuse of signals specified for one functionality to support other functionality. This may sound counter-intuitive, but experience shows that the associated dependencies between different functionalities often prevent desirable sleep-mode possibilities. An example is, that active-mode synchronisation in NR relies on the same set of synchronisation signal block (SSB) signals as UEs use for cell search in idle mode. There is work ongoing in 3GPP to relax this requirement for some use-cases, e.g. related to active mode mobility. But the fundamental structure of the NR standard is still that every active-mode signal has a quasi-co-location relation with an SSB. There are “non-cell defining SSBs” defined in the NR standard (SSBs that cannot be detected by UEs in idle mode) and this concept could be extended further. But for now, this still implies that SSBs cannot be dynamically beam-formed or deactivated (in the same way that cell-specific reference signals (CRSs) in the 4G LTE standard cannot). Any signal that is used to define a cell (like an SSB in NR or a CRS in LTE) needs to use static beam-forming and always be on to avoid time-varying coverage holes. This limits the energy-efficiency gains from dynamic deactivation of nodes, for example in a distributed MIMO system.

We also lack a multi-cell-covering physical-layer broadcast channel in NR. The system information is instead transmitted on a cell, or even on a beam, basis which result in a lot of repetition. The same is true for paging messages that could also benefit from a multi-cell broadcast channel. With a stricter separation of active and idle mode in 6G, a system function relying on a multi-cell physical-layer broadcast channel (for system information, paging, and random-access response) could be optimised independently from all active mode channels. In addition, it would be possible to add more capacity for active mode when needed, without impacting the idle mode broadcast transmissions. Note that such a network separation between active and idle mode would require an extremely fast procedure to transition UEs from the “idle mode part” of the system to the “active mode part” of the system.

There are several fundamental differences between a traditional “cellular” MIMO scenario and D-MIMO scenario, see Figure 1.1. These fundamental differences motivate a redesign of how idle mode network functions (i.e., system synchronisation, system information broadcast, paging, and random access response) are designed to enable better support for D-MIMO scenarios. Some typical differences are:

- Traditional “cellular” MIMO scenario
  - The base station is surrounded by UEs (e.g. orders of magnitude more UEs than base stations).
  - The base station has more antenna elements than the UE (e.g., in a typical scenario there may be 64 base station antenna elements and 1-4 UE antenna elements).
  - All base stations are participating in transmitting wide-beam signals even when there are no ongoing data transmissions (e.g., SSB transmissions for synchronisation, system information broadcast transmissions, paging transmissions, etc.).
- RadioWeaves scenario
  - The UE is surrounded by contact service points (the number of UEs and CSPs may be of the same order of magnitude).
  - The number of antenna elements on the UE and on the CSPs are similar (e.g. in a typical scenario the number of UE and CSP antenna elements may be in the range between 1 and 8. CSPs in a D-MIMO system need to be small and low cost and that typically implies that the D-MIMO CSPs cannot have as many antenna elements as a traditional massive MIMO base station. Most UEs will have more than one antenna element).
  - Only a subset of the CSPs are needed for transmitting wide-beam signals (e.g. synchronisation signals, system information broadcast transmissions, paging transmissions, etc.).
  - Most CSPs need only be active during data transmission (in order to ensure multi-user communications with high spectral efficiency).

In the discussions for next-G technologies, there is a tendency for demanding requirements from one technical area to propagate into other technical areas. This has been an issue already in earlier standards, one example of this is the cell reference signals (CRSs) in 4G LTE that was originally intended to be used only for demodulation of physical downlink control channel (PDCCH) and physical downlink shared channel (PDSCH) but ended up being used also for supporting active and idle mode mobility. Thereby the CRSs could never be deactivated, even when no downlink control or data channels were transmitted since that would break the mobility functions of the system. To support active and idle mode mobility much simpler (*i.e.*, less costly to always maintain) signals could have been used instead, but the mixing of requirements prohibited this. Similarly, in 5G NR the SSB is used both as a reference to acquire system information broadcast (for UEs in idle mode) as well as for active mode beam management. This results in conflicting requirements where the most stringent requirement wins, in this case, the need to observe all possible beams that may be used by the UE in idle mode. As a consequence, the system information broadcast in 5G NR needs to be repeated in all beams and in all cells. A simpler and more efficient solution to achieve area covering broadcast of system information (such as using a single frequency network (SFN) transmission format) cannot be used since that

would negatively impact the active mode beam management.

For 6G discussions on “joint communication and sensing” run the risk of producing similar requirement conflicts, resulting in inefficiencies. The support for extreme capabilities such as extremely high data rates with corresponding extreme transmission bandwidths, extremely low and predictable latency and extreme reliability are important to enable the wide range of use cases envisioned for 6G. At the same time, such capabilities come with a cost in terms of network energy consumption. It is crucial that this is then limited to the situations when the specific capabilities are required and do not spill into the network operation in general. Hence, it is key to prevent requirements for active mode from also applying to idle mode network operation. This can be assured by a stricter separation between the various functions in future networks.

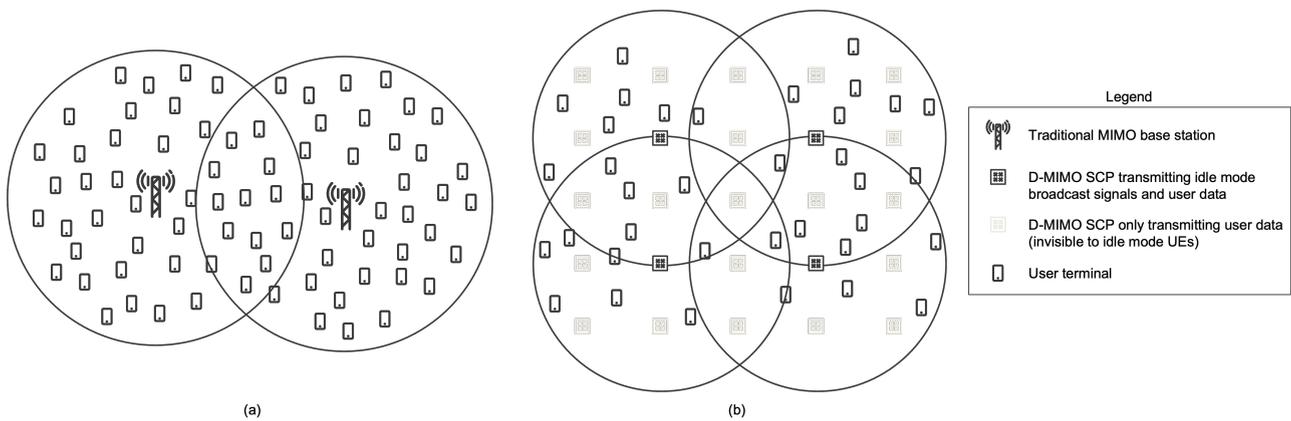


Figure 1.1: A traditional cellular MIMO scenario (a) and a distributed MIMO scenario (b) are different in several fundamental aspects. Note the black and grey CSPs in the right figure: The black CSPs illustrate CSPs that are constantly transmitting idle mode broadcast signals (performing e.g., SSB beam sweeping, system information broadcast, etc.) in idle mode, and the grey CSPs indicate CSPs that are only active during user plane data transmission and/or reception. In the left figure, all base stations are black, indicating that they are constantly transmitting idle mode broadcast signals.

# **Part I**

## **Network Set-up and De-Registered Mode Operation**

## Chapter 2

# Carrier Frequency Synchronization

Coherent reception of the signal at the user, which requires synchronization among the distributed transmitters, is critical to achieving the benefits of distributed architectures such as RadioWeaves. However, in practice, achieving synchronization is a challenging problem. Each transceiver in a communication system is equipped with a local oscillator circuit that generates carrier frequency based on a reference crystal oscillator. Due to mismatches in the reference oscillator circuits, different transceivers generate different carrier frequencies. Furthermore, the generated frequencies drift over time, for instance, due to fluctuations in temperature and voltage. Hence, the carrier frequency at different transceivers will be different. This results in a carrier frequency offset between any two transceiver nodes, which degrades the performance of the communication system. In order to avoid this problem and to achieve carrier frequency synchronization in the Global System for Mobile Communications (GSM) systems, frequency correction burst signals (FBs) are sent periodically through the frequency correction channel (FCCH) [1]. After listening to FBs, receivers tune their local oscillators to match their carrier frequency with the transmitter. Different carrier frequency synchronization techniques were studied for a point-to-point orthogonal frequency-division multiplexing (OFDM) system in [50, 72, 79].

The synchronization techniques developed for a point-to-point communication system do not extend directly to a distributed communication system. This is because the receiver observes a combined signal from different transmit nodes. One possible way to achieve carrier frequency synchronization is to provide a common carrier frequency to these distributed transmit nodes through a wired fronthaul network. However, this is not a scalable solution as the number of distributed transmitters increases. To address this issue, over-the-air carrier synchronization methods were studied in [4, 19, 77]. AirShare technique proposed in [4], uses a dedicated emitter to transmit two low-frequency tones over the air. The distributed transceivers use a dedicated circuit to receive these tones and generate their reference signal with the frequency equal to the difference of the two tones. This technique is robust to variations in temperature and supply voltage at the emitter. However, it uses out of band frequency resources. In AirSync technique studied in [19], a primary AP transmits pilots continuously in the out of the data transmission band. The secondary APs receive these pilots to estimate the frequency offset. This technique requires continuous transmission of the pilots from the primary AP and one dedicated receive antenna at each secondary AP. A pilot signaling between anchor APs, which form a connected cover of the network, in a special synchronization slot to estimate the frequency offset is proposed in [77]. These estimates are exchanged through a wired fronthaul connecting the distributed transmit nodes. The scheme requires geographically dispersed anchor nodes and requires high anchor

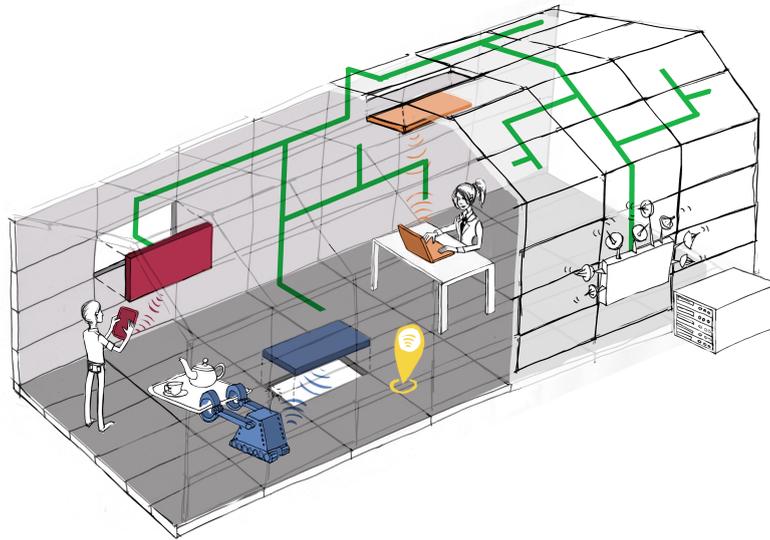


Figure 2.1: Distributed RadioWeaves array deployment system model.

AP density.

In this section, we discuss an over-the-air, carrier frequency synchronization protocol based on digital beamforming, which we shall refer to as BeamSync, for distributed multi-antenna CSPs in RadioWeaves infrastructure. In BeamSync, we consider one of the CSPs as the primary CSP and others as secondary CSPs, which need to synchronize with the primary CSP. BeamSync removes the requirement of dedicated circuits for synchronization at transceiver nodes unlike [4, 19]. Moreover, the scheme does not exchange calibration data through wired fronthaul connections and enables a faster carrier frequency synchronization. BeamSync exploits the diversity benefits of the multiple antennas at each CSP to beamform the synchronization signal. The primary CSP beamforms the frequency synchronization signal towards the secondary CSPs in the dominant direction of the channel between the primary and secondary CSPs. The secondary CSPs estimate their frequency offset with respect to primary using signal processing techniques. We show that the optimal beamforming direction, which minimizes the offset estimation error, is the dominant direction of the channel<sup>1</sup> between the CSPs in which the signal is received.

## 2.1 System Model

We consider a distributed RadioWeaves array deployment, which consists of multiple geographically separated CSPs communicating simultaneously to the users as shown in Figure 2.1. Each CSP is equipped with multiple antennas, each with its own radio frequency (RF) chain, and one local oscillator circuit to generate the carrier frequency. There is no mismatch between the carrier frequency among RF chains in the same CSP, as all of them are driven by the same oscillator circuit of the CSP. However, the carrier frequencies generated at different CSPs will differ. Hence, to synchronize the carrier frequency among the CSPs to a common reference, we nominate one of the CSPs as the primary CSP and consider its carrier frequency as the reference. The remaining CSPs, which we refer to as secondary CSPs, synchronize with the primary.

Let  $N_s$  denote the number of secondary CSPs. Let  $M_p$  denote the number of antennas at

<sup>1</sup>The dominant direction is here defined as the singular vector corresponding to the largest singular value of the channel.

the primary CSP and  $M_{s,i}$  denote the number of antennas at the  $i^{\text{th}}$  secondary CSP, for  $i \in \{1, 2, \dots, N_s\}$ . Let  $\mathbf{G}_i \in \mathbb{C}^{M_p \times M_{s,i}}$  denote the complex channel gain matrix between the primary CSP and the  $i^{\text{th}}$  secondary CSP. We assume that the channel is reciprocal. Let  $f_p$  and  $f_{s,i}$  denote the carrier frequencies at the primary CSP and the  $i^{\text{th}}$  secondary CSP, respectively. Then, the carrier frequency offset of the  $i^{\text{th}}$  secondary CSP with respect to the primary CSP is given by  $\Delta_i = f_p - f_{s,i}$ . Each of the secondary CSPs estimates its  $\Delta_i$  and compensates during data transmission to the users.

## 2.2 BeamSync Protocol

We describe the BeamSync protocol for one secondary CSP. For simplicity, we drop the index  $i$  of the secondary CSP in the notations. Therefore,  $M_s$ ,  $\mathbf{G} \in \mathbb{C}^{M_p \times M_s}$ , and  $\Delta$ , denote the number of antennas at the secondary CSP, the channel matrix between the primary and secondary CSP, and the frequency offset, respectively. The protocol consists of two stages described as follows:

### Stage-I

The secondary CSP transmits an orthonormal pilot sequence of length  $\tau_p \geq M_s$  from each of its antennas. Let the columns of the matrix  $\Phi \in \mathbb{C}^{\tau_p \times M_s}$ , where  $\Phi^H \Phi = \mathbf{I}_{\tau_p}$ , denote the set of orthonormal pilot sequences. Let  $\phi(n)$ , denote the  $n^{\text{th}}$  row of  $\Phi$ . Thus, at the  $n^{\text{th}}$  time instant, the signal received at the primary CSP  $\mathbf{y}_p \in \mathbb{C}^{M_p \times 1}$  can be expressed as

$$\mathbf{y}_p(n) = \sqrt{\rho} \mathbf{G} \phi(n) e^{j2\pi n \Delta} + \mathbf{w}_p(n), \quad (2.1)$$

where  $\rho$  is the normalised signal-to-noise ratio (SNR) and  $\mathbf{w}_p(n) \in \mathbb{C}^{M_p \times 1}$  is the additive noise with each of the elements independent and identically distributed (i.i.d.)  $\mathcal{CN}(0, 1)$ . Let

$$\mathbf{D}_{\Delta, \tau} = \text{diag}\{e^{j2\pi\Delta}, e^{j2\pi 2\Delta}, \dots, e^{j2\pi\tau\Delta}\} \in \mathbb{C}^{\tau \times \tau}. \quad (2.2)$$

The collective signal received in  $\tau_p$  time instants at the primary CSP,  $\mathbf{Y}_p = [\mathbf{y}_p(1) \mathbf{y}_p(2) \dots \mathbf{y}_p(\tau_p)]$ , can be written as

$$\mathbf{Y}_p = \sqrt{\rho} \mathbf{G} \Phi^H \mathbf{D}_{\Delta, \tau_p} + \mathbf{W}_p, \quad (2.3)$$

where  $\mathbf{W}_p = [\mathbf{w}_p(1) \mathbf{w}_p(2) \dots \mathbf{w}_p(\tau_p)]$ .

### Stage-II

The primary CSP processes the signal  $\mathbf{Y}_p$  received in stage-I and determines a beamforming vector  $\mathbf{a} \in \mathbb{C}^{M_p \times 1}$  such that  $\|\mathbf{a}\|=1$ . It then beamforms a length  $N$  frequency synchronization signal  $\mathbf{x}$ . The received signal at the secondary CSP,  $\mathbf{y}_s(n) \in \mathbb{C}^{M_s \times 1}$  at the  $n^{\text{th}}$  time instant is given by

$$\mathbf{y}_s(n) = \sqrt{\rho} \mathbf{G}^T \mathbf{a} x(n) e^{-j2\pi n \Delta} + \mathbf{w}_s(n), \quad (2.4)$$

where  $x(n)$  is the  $n^{\text{th}}$  component of signal  $\mathbf{x}$  and  $\mathbf{w}_s(n) \in \mathbb{C}^{M_s \times 1}$  is the additive noise with independently and identically distributed (i.i.d.)  $\mathcal{CN}(0, 1)$  entries. The collective signal received over  $N$  times instants at the secondary CSP,  $\mathbf{Y}_s = [\mathbf{y}_s(1) \mathbf{y}_s(2) \dots \mathbf{y}_s(N)]$ , can be written as

$$\mathbf{Y}_s = \sqrt{\rho} \mathbf{G}^T \mathbf{a} \mathbf{x}^T \mathbf{D}_{\Delta, N}^* + \mathbf{W}_s, \quad (2.5)$$

where  $\mathbf{W}_s = [\mathbf{w}_s(1) \ \mathbf{w}_s(2) \ \cdots \ \mathbf{w}_s(N)]$ . Secondary CSP needs to estimate its frequency offset  $\Delta$ , with respect to the primary CSP via (2.5). The channel  $\mathbf{G}$  and the beamforming vector  $\mathbf{a}$  are unknown at the secondary CSP. Let  $\mathbf{b} = \mathbf{G}^T \mathbf{a}$  denote the effective channel. Then (2.5) can be rewritten as

$$\mathbf{Y}_s = \sqrt{\rho} \mathbf{b} \mathbf{x}^T \mathbf{D}_{\Delta, N}^* + \mathbf{W}_s. \quad (2.6)$$

The joint maximum likelihood estimates of  $\mathbf{b}$  and  $\Delta$  are given by

$$(\hat{\mathbf{b}}, \hat{\Delta}) = \underset{\mathbf{b}, \Delta}{\operatorname{argmin}} \|\mathbf{Y}_s - \sqrt{\rho} \mathbf{b} \mathbf{x}^T \mathbf{D}_{\Delta, N}^*\|^2. \quad (2.7)$$

Solving (2.7) using non-linear least squares estimation in Gaussian noise [59, Sec. 8.9] with  $\mathbf{b}$  as nuisance parameter, estimates of  $\mathbf{b}$  and  $\Delta$  are given by

$$\hat{\mathbf{b}} = \frac{\mathbf{Y}_s \mathbf{D}_{\Delta, N} \mathbf{x}^*}{\sqrt{\rho} \|\mathbf{x}\|^2}, \quad (2.8)$$

$$\hat{\Delta} = \underset{\Delta}{\operatorname{argmax}} \|\mathbf{Y}_s \mathbf{D}_{\Delta, N} \mathbf{x}^*\|^2. \quad (2.9)$$

The secondary CSP uses  $\hat{\Delta}$  to de-rotate its transmitted signals to synchronise with the primary CSP.

## 2.2.1 Optimal Beamforming Direction

In this subsection, we derive the optimal beamforming direction that minimises the offset estimation error. We look at the conditions for which the Cramér-Rao lower bound (CRLB) on the estimate of  $\Delta$  is minimised.

Let  $(\cdot)_R$  and  $(\cdot)_I$  denote the real and imaginary parts of a complex number, respectively. Then  $\mathbf{b} = \mathbf{b}_R + j\mathbf{b}_I$  and  $\mathbf{y}_s(n) = \mathbf{y}_{sR}(n) + j\mathbf{y}_{sI}(n)$ . Let

$$\boldsymbol{\theta} = [\mathbf{b}_R^T \ \mathbf{b}_I^T \ \Delta]^T, \quad (2.10)$$

be the unknown parameter at the secondary CSP. From (2.6), the signal received at the  $n^{\text{th}}$  time instant,  $\mathbf{y}_s(n)$  is distributed as  $\mathcal{CN}(\sqrt{\rho} \mathbf{b} x(n) e^{-j2\pi n \Delta}, \mathbf{I})$ . We assume that the frequency synchronisation signal  $\mathbf{x}$  is real-valued. Thus,  $\bar{\mathbf{y}}_s(n) = [\mathbf{y}_{sR}^T(n) \ \mathbf{y}_{sI}^T(n)]^T \in \mathbb{R}^{2M_s \times 1}$  is distributed as  $\mathcal{N}(\boldsymbol{\mu}_n(\boldsymbol{\theta}), \mathbf{C}(\boldsymbol{\theta}))$ , where  $\boldsymbol{\mu}_n(\boldsymbol{\theta})$  and  $\mathbf{C}(\boldsymbol{\theta})$  denote the mean and covariance of  $\bar{\mathbf{y}}_s$ , respectively parameterised by  $\boldsymbol{\theta}$ , and are given by

$$\boldsymbol{\mu}(\boldsymbol{\theta}) = \sqrt{\rho} x(n) \begin{bmatrix} \mathbf{b}_R \cos(2\pi n \Delta) + \mathbf{b}_I \sin(2\pi n \Delta) \\ -\mathbf{b}_R \sin(2\pi n \Delta) + \mathbf{b}_I \cos(2\pi n \Delta) \end{bmatrix}, \quad (2.11)$$

$$\mathbf{C}(\boldsymbol{\theta}) = \frac{1}{2} \mathbf{I}_{2M_s}. \quad (2.12)$$

Using the Slepian-Bang theorem [59, Sec. 3.9], each element of the Fisher information matrix (FIM) of  $\boldsymbol{\theta}$  at the  $n^{\text{th}}$  time instant,  $\mathbf{J}_n(\boldsymbol{\theta}) \in \mathbb{R}^{(2M_s+1) \times (2M_s+1)}$ , can be computed as

$$[\mathbf{J}_n(\boldsymbol{\theta})]_{k,l} = \left[ \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_k} \right]^T \mathbf{C}^{-1}(\boldsymbol{\theta}) \left[ \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_l} \right] + \frac{1}{2} \operatorname{tr} \left[ \mathbf{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \mathbf{C}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_k} \mathbf{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \mathbf{C}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_l} \right], \quad (2.13)$$

where  $[\cdot]_{k,l}$  denotes the  $(k, l)^{\text{th}}$  element. By computing the partial derivatives of (2.11) and (2.12), we obtain

$$\mathbf{J}_n(\boldsymbol{\theta}) = 2\rho x^2(n) \begin{bmatrix} \mathbf{I}_{M_s} & \mathbf{0} & 2\pi n \mathbf{b}_I \\ \mathbf{0} & \mathbf{I}_{M_s} & -2\pi n \mathbf{b}_R \\ 2\pi n \mathbf{b}_I^T & -2\pi n \mathbf{b}_R^T & 4\pi^2 n^2 \|\mathbf{b}\|^2 \end{bmatrix}. \quad (2.14)$$

The received signal  $\mathbf{y}_s(n)$  is independent for different time instants. Thus, using the additive property of FIM, the overall FIM of  $\boldsymbol{\theta}$ ,  $\mathbf{J}(\boldsymbol{\theta})$ , is given by

$$\mathbf{J}(\boldsymbol{\theta}) = \sum_{n=1}^N \mathbf{J}_n(\boldsymbol{\theta}). \quad (2.15)$$

The CRB of  $\hat{\Delta}$  can be computed from  $\mathbf{J}(\boldsymbol{\theta})$  as

$$\text{CRB}(\hat{\Delta}) = [\mathbf{J}^{-1}(\boldsymbol{\theta})]_{2M_s+1, 2M_s+1}, \quad (2.16)$$

which is the lower right corner element of  $\mathbf{J}^{-1}(\boldsymbol{\theta})$ . Using the inverse of a block partitioned matrix [49, Sec. 0.7.3], the CRB of  $\hat{\Delta}$  is given by

$$\text{CRB}(\hat{\Delta}) = \frac{1}{8\pi^2\rho\|\mathbf{b}\|^2 \left( \sum_{n=1}^N n^2 x^2(n) - \frac{(\sum_{n=1}^N n x^2(n))^2}{\sum_{n=1}^N x^2(n)} \right)}. \quad (2.17)$$

From (2.17), the CRB of  $\hat{\Delta}$  will be minimized when  $\|\mathbf{b}\|^2 = \|\mathbf{G}^T \mathbf{a}\|^2$  is maximized. Let the singular value decomposition (SVD) of the channel  $\mathbf{G}$  be

$$\mathbf{G} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^H, \quad (2.18)$$

where  $\mathbf{U} \in \mathbb{C}^{M_p \times M_p}$  and  $\mathbf{V} \in \mathbb{C}^{\tau_p \times \tau_p}$  are unitary matrices and  $\boldsymbol{\Sigma} \in \mathbb{R}^{M_p \times \tau_p}$  is a diagonal matrix with singular values of  $\mathbf{G}$  in decreasing order. Then,

$$\begin{aligned} \|\mathbf{G}^T \mathbf{a}\|^2 &= \mathbf{a}^H \mathbf{G}^* \mathbf{G}^T \mathbf{a} \\ &= \mathbf{a}^H \mathbf{U}^* \boldsymbol{\Sigma} \boldsymbol{\Sigma}^T \mathbf{U}^T \mathbf{a}. \end{aligned} \quad (2.19)$$

From (2.19),  $\|\mathbf{G}^T \mathbf{a}\|^2$  is the Rayleigh quotient of matrix  $\mathbf{G}^* \mathbf{G}^T$  with vector  $\mathbf{a}$  and can be maximised by choosing  $\mathbf{a} = \mathbf{u}_1^*$ . Vector  $\mathbf{u}_1$ , is the first column of matrix  $\mathbf{U}$ . Hence, the optimal beamforming direction  $\mathbf{a}$ , corresponds to the dominant direction of the channel in which the signal is received at the secondary CSP.

From (2.17), for estimating  $\Delta$ , the synchronisation signal length  $N$  should be at least 2. Moreover, from (2.17), the frequency offset estimate  $\hat{\Delta}$  can be improved by increasing the SNR  $\rho$ , as well as increasing the synchronisation sequence length  $N$ .

## 2.2.2 Estimating Beamforming Direction in BeamSync

From Sec. 2.2.1, it is evident that the optimal direction to beamform the synchronization signal is the dominant direction of the channel in which the secondary CSP receives the signal from the primary CSP. In practice, the channel  $\mathbf{G}$  is not perfectly known at the primary CSP. However, as the channel is reciprocal, the primary CSP can estimate the dominant direction of the signal received from the secondary CSP, without the need to estimate the channel. The primary CSP listens to the pilot signal  $\Phi$  which is transmitted in all directions by the secondary CSP in Stage-I of synchronization protocol and computes the dominant direction in which the signal was received. It can be mathematically expressed as SVD of  $\mathbf{Y}_p$  given by

$$\mathbf{Y}_p = \mathbf{U}_p \boldsymbol{\Sigma}_p \mathbf{V}_p^H, \quad (2.20)$$

where  $\mathbf{U}_p \in \mathbb{C}^{M_p \times M_p}$  and  $\mathbf{V}_p \in \mathbb{C}^{\tau_p \times \tau_p}$  are unitary matrices and  $\Sigma_p \in \mathbb{R}^{M_p \times \tau_p}$  is a diagonal matrix with singular values of  $\mathbf{Y}_p$  in decreasing order. The columns of  $\mathbf{U}_p$  correspond to the direction of the received signal ordered according to the dominance of power received in each direction. Hence the optimal beamforming direction is given by  $\mathbf{a} = \mathbf{u}_{p1}^*$ , where  $\mathbf{u}_{p1}$  is the first column of  $\mathbf{U}_p$ . As SNR increases, the primary CSP will be able to perfectly determine the dominant direction of the channel asymptotically, i.e.,  $\mathbf{u}_{p1} \rightarrow \mathbf{u}_1$ .

### 2.2.3 Over-The-Air Carrier Synchronization Protocol

We generalize the proposed BeamSync protocol for multiple secondary CSPs and the communication flow is shown in Figure 2.2. During the cold start or initialization of the entire communication system, all the distributed transceivers will be out of sync. After the initial power up, all the secondary CSPs will synchronize with the primary CSP in a sequential fashion using the BeamSync protocol. Afterwards, the distributed CSPs can start joint coherent transmission to the users in the data transmission phase. Moreover, as the carrier frequency synchronization is done over-the-air without the need of wired fronthaul connections, it enables a faster carrier frequency synchronization.

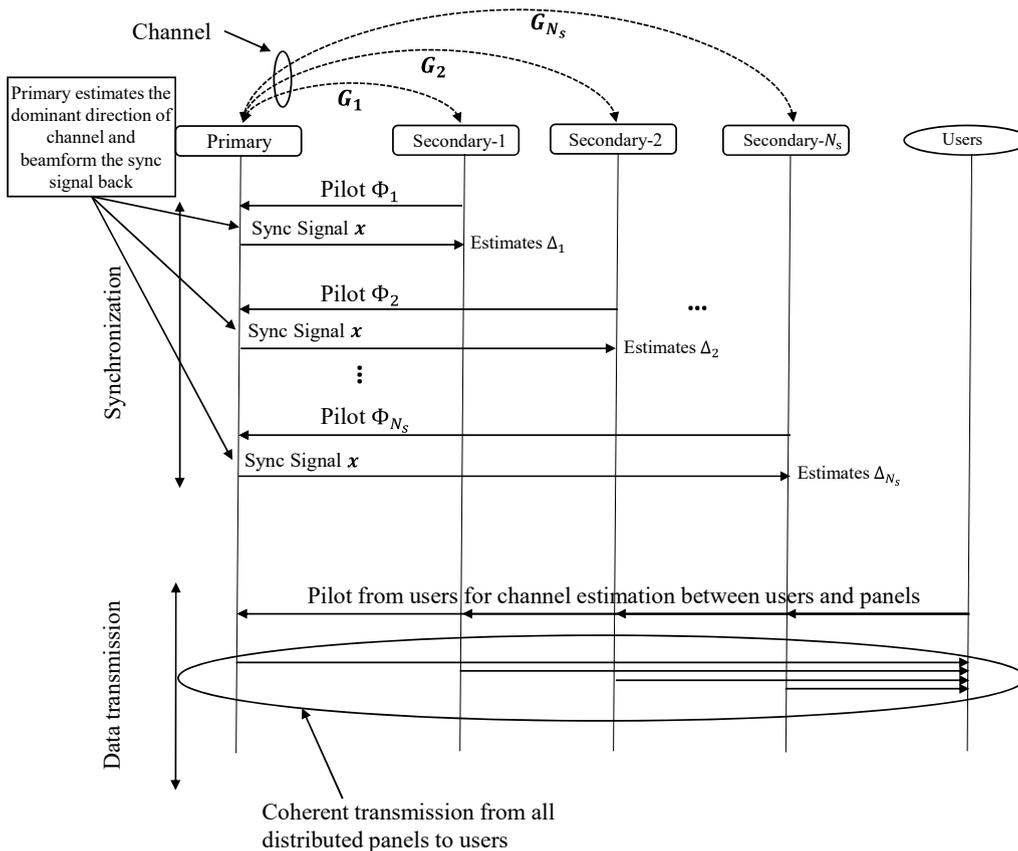


Figure 2.2: BeamSync protocol.

The frequency generated through the local oscillator circuit can drift over time, for instance, due to fluctuations in temperature and voltage. This frequency drift is negligible in a coherence interval.

Hence, after the cold start, the synchronisation procedure needs to be done when the secondary CSP goes out of sync, based on a need basis. Thus, the synchronisation procedure dispersed over time is represented in Figure 2.3.

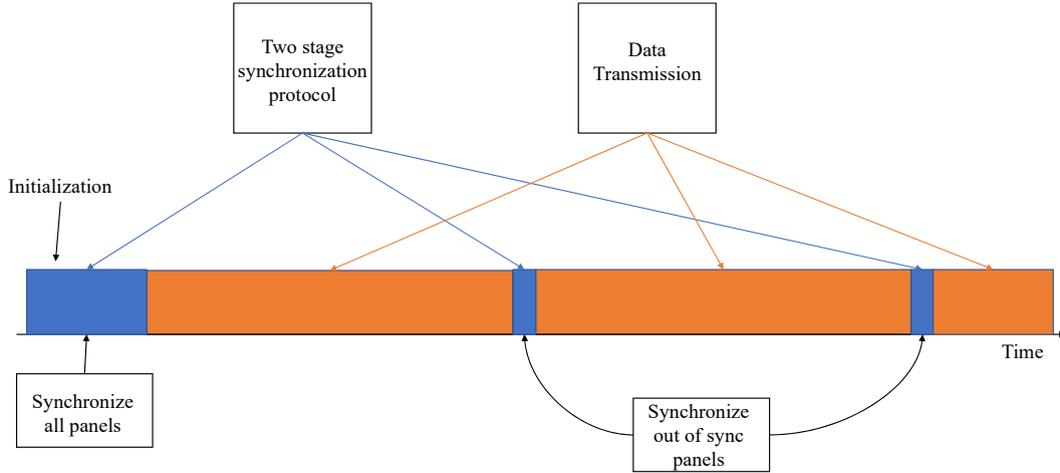


Figure 2.3: Synchronization procedure over time.

## 2.3 Simulations

*Simulation Parameters:* For simulations, we consider the number of antennas at the primary CSP and secondary CSP as  $M_p = M_s = 16$ . The pilot signal length  $\tau_p = M_s$ . The length of the synchronization signal transmitted from the primary CSP,  $N = 100$ . The Monte Carlo trials considered is  $10^5$ . We consider the frequency synchronization signal  $\mathbf{x}$  as

$$\mathbf{x} = [1 \sin(2\pi f) \sin(4\pi f) \cdots \sin(2\pi f(N - 1))]^T, \quad (2.21)$$

where  $f$  is the frequency and is chosen to have four full cycles of sinusoid signal in  $N$  time instants.

### 2.3.1 Performance Benchmarking Schemes

With BeamSync, the beamforming vector can be pointed in any direction in 3-dimensional environment and can be done by digital signal processing techniques. Hence, BeamSync is a fully digital beamforming scheme. For comparison in the figures we refer to our proposed schemes as follows:

1. *BeamSync*: Proposed scheme, where the received signal at primary is used to find the beamforming direction  $\mathbf{a} = \mathbf{u}_{p1}^*$ .
2. *BeamSync-Genie*: Proposed scheme, where through an aid of a genie, we consider that the primary CSP perfectly knows the channel  $\mathbf{G}$ . The beamforming direction is  $\mathbf{a} = \mathbf{u}_1^*$ .

We compare our proposed schemes with the following beamforming techniques:

3. *Analog beamforming*: In this scheme, the primary CSP performs transmit beamforming and the secondary CSP performs receive beamforming. The beamforming vectors at both CSPs are chosen from a fixed set of beams. In our numerical example, we consider columns of a DFT matrix as the possible set of orthogonal beams. Let  $\{\mathbf{f}_{p,k} \in \mathbb{C}^{M_p \times 1}, k = 1, 2, \dots, M_p\}$  and  $\{\mathbf{f}_{s,l} \in \mathbb{C}^{M_s \times 1}, l = 1, 2, \dots, M_s\}$  be the fixed set of beams available at the primary and secondary CSPs, respectively. The transmit and receive beamforming vectors are chosen such that the received signal power is maximized. Let

$$k = \operatorname{argmax}_{k'} \|\mathbf{f}_{p,k'}^H \mathbf{Y}_p\|^2, \quad l = \operatorname{argmax}_{l'} \|\mathbf{f}_{s,l'}^H \mathbf{Y}_s\|^2. \quad (2.22)$$

Then, the transmit beamforming vector is  $\mathbf{a}_p = \mathbf{f}_{p,k}^*$ , and the receive beamforming vector is  $\mathbf{a}_s = \mathbf{f}_{s,l}$ .

4. *Analog beamforming-Genie*: Same as 3), but we choose the beamforming vectors based on the perfectly known channel  $\mathbf{G}$  through a genie at both primary and secondary CSPs. Let

$$(k, l) = \operatorname{argmax}_{k', l'} |\mathbf{f}_{p,k'}^H \mathbf{G} \mathbf{f}_{s,l'}|^2. \quad (2.23)$$

Then, the transmit beamforming vector is  $\mathbf{a}_p = \mathbf{f}_{p,k}^*$ , and the receive beamforming vector is  $\mathbf{a}_s = \mathbf{f}_{s,l}$ .

## 2.3.2 Results

First, we consider a Rayleigh fading channel between the primary and the secondary CSPs. Thus, each element in  $\mathbf{G}$  is i.i.d.  $\mathcal{CN}(0, 1)$ . We consider the antennas to be omni-directional such that the signal can be transmitted and received in all directions. We use the root mean square error (root-mean-square error (RMSE)) of the frequency offset estimate as to the performance metric for comparison. The performance of different schemes in the Rayleigh fading scenario is shown in Figure 2.4a. From the plot, it can be seen that as the SNR increases, the RMSE decreases. When beamforming is done in the dominant direction determined from the perfect channel matrix, RMSE is lower for all SNR values among all the techniques. The performance of the proposed BeamSync protocol, which uses the dominant direction determined from the received vector, improves as SNR increases and matches with BeamSync-genie scheme at high SNR. This is because, as the SNR increases, the dominant direction chosen by BeamSync scheme becomes close to the one chosen from the perfect knowledge of the channel. Scheme 3, which uses analog beamforming, performs worse compared to other two schemes for all the SNR values. This shows that the fully digital beamforming in the dominant direction yields significant performance gain compared to the analog beamforming with fixed beams. For example, for a fixed RMSE requirement, the SNR gain is approximately 10 dB for BeamSync.

Figure 2.4b compares the performance of the proposed synchronization schemes with analog beamforming in a LoS scenario [43]. We consider the CSPs are distributed in a  $100\text{m} \times 100\text{m} \times 10\text{m}$  room. We consider directional patch antennas on the CSPs, and the primary and secondary CSPs are on adjacent walls. The channel and antenna design parameters used are as in [43]. Due to the strong line of sight signal, frequency offset can be better estimated at low SNR values compared to Rayleigh fading scenario. Similar to the Rayleigh fading case, BeamSync scheme matches with BeamSync-genie scheme at high SNR and performs better than the analog beamforming scheme. In this example, for a fixed RMSE requirement, the SNR gain is approximately 5 dB for BeamSync compared to the analog scheme.

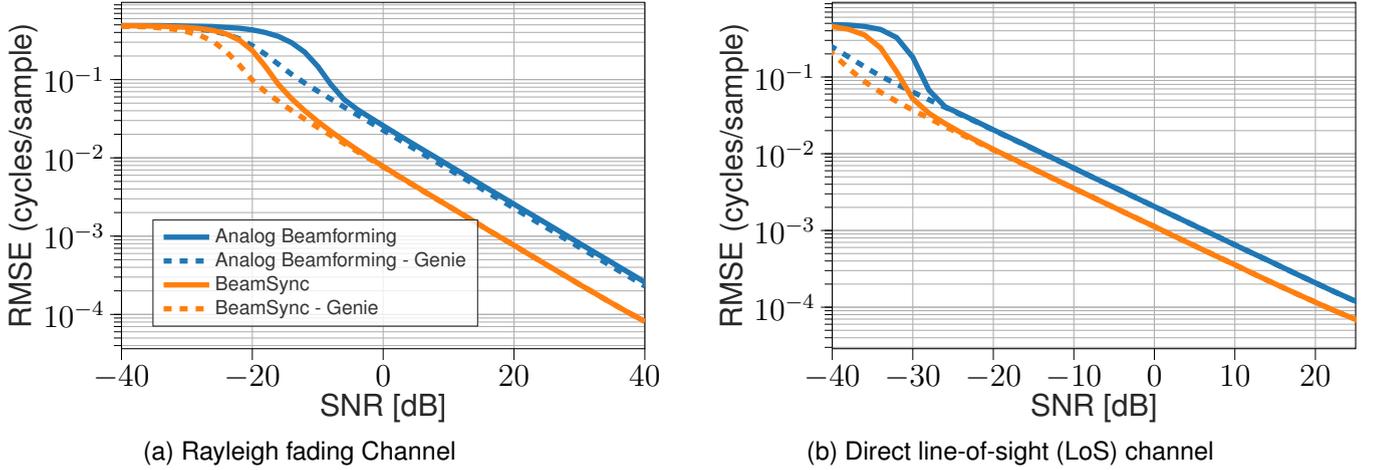


Figure 2.4: The RMSE of the carrier frequency estimate when using analog beamforming and the BeamSync protocol for a Rayleigh and LoS channel.

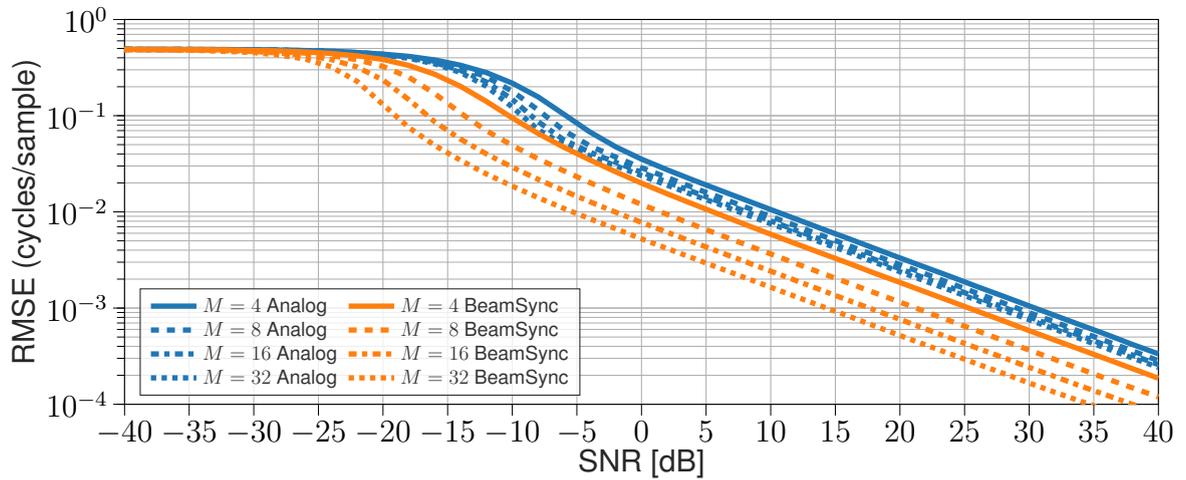


Figure 2.5: Performance of BeamSync with different number of antennas  $M_p = M_s = M$ , pilot signal length  $\tau_p = M_s$ .

Figure 2.5 compares the performance of the BeamSync protocol and analog scheme, when a different number of antennas are deployed at the CSP. From the figure, it can be seen that for a fixed RMSE requirement, the SNR requirement reduces by 3 dB when the number of antennas is doubled at the CSPs for the BeamSync protocol. This is because the signal can be steered better in the desired direction as the number of antennas increases in BeamSync [68]. However, when the analog beamforming is used, the gain in performance is negligible as the number of antennas increases.

## 2.4 Conclusion

In this chapter, we studied the carrier frequency synchronization in the distributed RadioWeaves array deployment. We proposed a novel, over-the-air carrier synchronization protocol, BeamSync, based on digital beamforming to synchronize different multi-antenna CSPs in Radioweaves. We showed that sending the frequency synchronization signal burst in the dominant direction of the channel between the panels is optimal. We also proposed a scheme to estimate the beamforming direction without estimating the channel. We compared our scheme with an analog beam-

forming scheme and showed that our proposed protocol can achieve better carrier frequency offset estimation. This is due to the improved SNR by beamforming and spatial processing gain. Moreover, the proposed protocol allows fast synchronization among the distributed CSPs. Also, we showed that, the better the synchronization signal burst is steered towards the secondary panel, the better is the offset estimation performance.

## Chapter 3

### Initial Access Phase

During the initial access phase the UEs retrieve the necessary information in order to connect to a system, *i.e.*, broadcast information. This is discussed in Section 3.1. On top of that, EN devices also need to be powered at first in order to perform the initial access procedure. The proposed wireless power protocol is elaborated in Section 3.2.

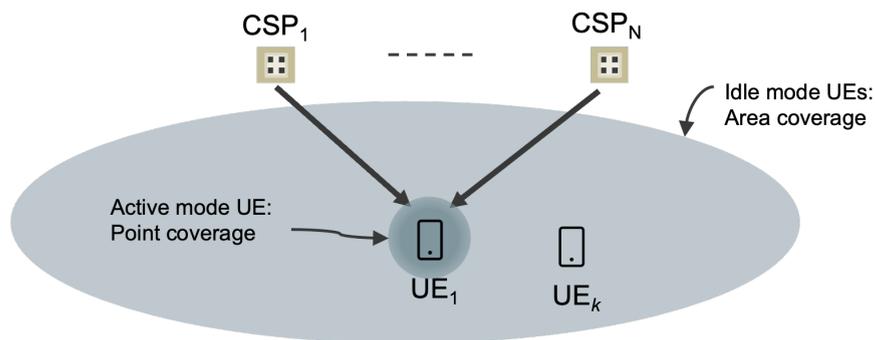


Figure 3.1: Active mode transmissions to a UE in D-MIMO aims at achieving coherent phase additions of signals from multiple CSPs at the point in space where the targeted UE is located. This type of “point coverage” transmission requires channel state information that is not available when transmitting idle mode signals such as system information broadcast. Instead, idle mode signals need to use area covering transmissions.

### 3.1 Broadcast Information

Active mode beams (point coverage) and idle mode beams (area coverage) are very different in D-MIMO systems, see Figure 3.1. To achieve phase coherent joint transmission from multiple CSPs, accurate channel state information (CSI) is required at the transmitting end. Active mode data signals can then be transmitted such that they add up coherently at a point in space where the targeted UE is located. During *e.g.*, system information broadcast, no such CSI is available. The idle mode signals (system information broadcast, paging transmissions, basic system synchronisation) needs to be transmitted over the entire coverage area of the system. This requires a fundamentally different beam-forming approach. As discussed in Section 1.2, this calls for a decoupling of “beams used in active mode” from “beams in idle mode”. For an overall energy-efficient system operation, the beams used in idle mode (for area covering transmissions of system information and paging signals) need to be optimised independently, without

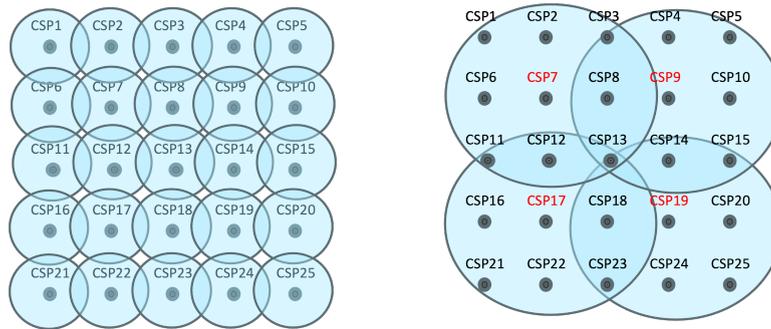


Figure 3.2: Left: Each CSP has responsibilities for active and idle mode functions. This can easily result in high idle mode energy consumption. Right: A sub-set of CSPs are responsible for idle mode functions. This can significantly reduce the overall network energy consumption.

constraints induced by active mode operation. What needs to be avoided, is that all CSPs must sweep through all active mode beams for the purpose of covering the area with system information broadcast signalling, as this could quickly result in very high network energy usage.

Interesting techniques to achieve both energy-efficient and area covering beams in D-MIMO are *e.g.*,

- Use a **sub-set** of the CSPs for transmitting signals related to initial system access, see Figure 3.1. In a D-MIMO system, the CSPs are typically not all deployed for area coverage reasons. Rather, a dense deployment of CSPs is used to enhance the channel richness (*i.e.*, the channel rank) and the overall system capacity. It is important that the overall system design is such that additional CSPs can be added to *e.g.*, to increase the system capacity, without requiring that additional CSPs are involved in the transmission of energy-consuming idle mode broadcast signalling.
- **Dual-polarised array-size invariant beam-forming** is a technique to maintain a wide beam shape from a large antenna array. For single polarised antenna arrays, the beams become narrower as the array size increase. But using dual polarised arrays, the overall beam-width can be made invariant of the array size. See [75] for further details. Using dual-polarised array size invariant beam-forming, service points with multiple antennas don't need to perform excessive beams-weeping for transmitting area covering signals.
- **Space-time-coding with port hopping**, see Figure 3.4. Space-time codes, such as the Alamouti code, are successfully used in *e.g.*, 3GPP LTE for providing additional diversity and robustness for system information broadcast signals. In a centralised MIMO system the antenna ports have the same average path gain, which is not the case in a D-MIMO deployment, see Figure 3.3. By applying port hopping, the diversity gain of space-time codes when used in D-MIMO can be significantly enhanced. This is achieved by ensuring that the spatial antenna ports seen by the UE have approximately equal average path gain. To get full diversity gain from this type of port hopping, the hopping sequence length needs to be at least as long as the number of spatial ports of the space-time code.

A good solution for idle mode signal transmission in RadioWeaves needs to support all of the above-mentioned techniques.

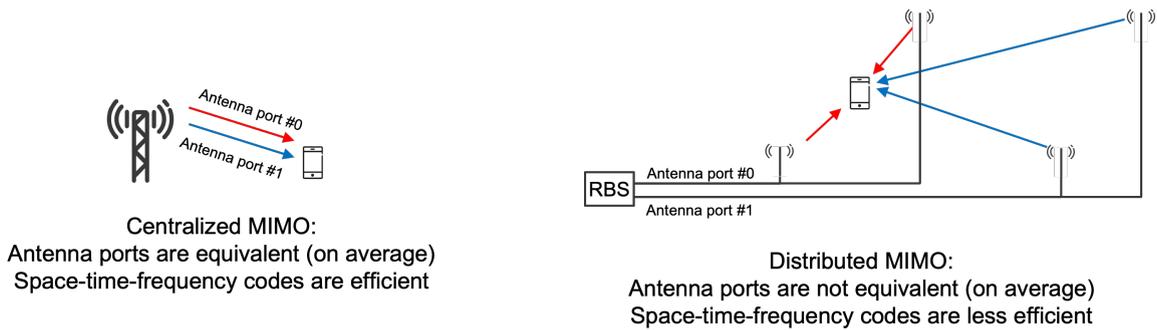


Figure 3.3: Space time coding (such as Alamouti codes) is efficient for exploiting channel diversity in centralised MIMO systems (left). In D-MIMO systems, the antenna ports are on average not equal. To archive good performance a port hopping solution is proposed to be used in D-MIMO systems.

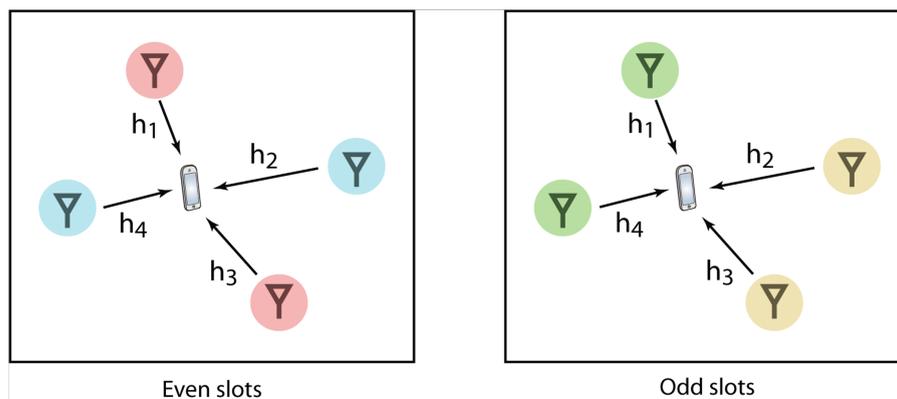


Figure 3.4: Efficient exploitation of macro-diversity gains provided by space-time coding (such as Alamouti codes) in D-MIMO can be achieved by port hopping. By periodically re-mapping the antenna ports used by the space-time diversity code, the antenna ports will have a more equal average path gain resulting in enhanced diversity gain.

## 3.2 Wireless Power Transfer

The initial access to EN devices targets the first wake-up and sufficient wireless power supply of devices at unknown positions and with unknown CSI. Under usual circumstances, in absence of CSI, downlink precoding schemes like maximum ratio transmission (MRT), cannot be applied and no array gain can be leveraged. Channel state estimation can be performed on the first signal sent by an EN device, possibly through backscatter communication [58], which can be performed efficiently when EN devices start up in class 0 mode [40]. With no array gain available, supplying an EN device with sufficient power to exceed the device sensitivity, *i.e.*, the minimum power required for wake-up and backscatter communication [37], strongly reduces the initial access distance.

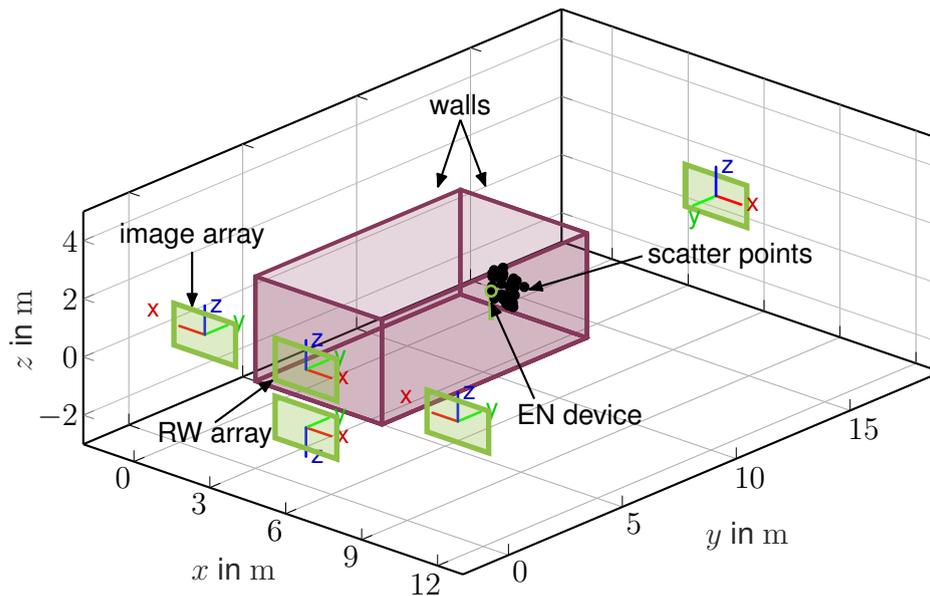


Figure 3.5: A scenario with four walls, as well as a floor, simulated in [34]. A (5 m × 9 m × 3.5 m) large room is simulated. Virtual image arrays with their local coordinate systems are indicated in the figure along with the EN device. Scatter points are distributed along the front wall in the direction of LoS to the EN device.

To overcome this impairment, one possible approach for initial access is beam sweeping, where the transmit array sweeps beams sequentially according to a predefined codebook [95] to power up the EN device for the first time and thereby exploit some array gain. This requires the array to be calibrated in case a single array is used for power transmission. If several arrays are transmitting power to a single receiving device simultaneously, they further need to be synchronized. In indoor scenarios, environment awareness can aid the choice of predefined codebooks, assuming that the possible locations of EN devices, as well as the propagation environment, are at least partially known. However, beam sweeping in indoor scenarios suffers from fading due to severe multipath propagation, possibly originating from unknown objects in the environment.

The REINDEER consortium has investigated the potential of overcoming severe fading and increasing the initial access distance by exploiting beam diversity [34]: In a simultaneous multi-beam transmission from one RadioWeaves panel, the phases of the individual beams can be varied to reduce the necessary fading margin for the initial access to EN devices.

We demonstrated that environment awareness and beam diversity yield an array gain usable without CSI and reduce the necessary fading margin. For a specific environment setting, we were able to enlarge the initial access distance by a factor of 4 using 16 variations of beamforming

vectors generated through geometric environment information. This section includes the beam-diversity scheme and achievable gains, while the approximate power budget and expected initial access distance are described in the REINDEER deliverable D4.1 [23].

### 3.2.1 Environment-aware Initial Access to EN Devices

Beam sweeping can be an effective method for the initial access to EN devices. In near-field WPT, as is expected in RadioWeaves, not only the angular domain needs to be scanned, but also the range domain. This would make exhaustive near-field beam scanning time-consuming, even if wide beams are used in sub-10 GHz bands. However, we exploit environment-awareness to reduce the three-dimensional search space to the most likely positions of EN devices (*e.g.*, along walls or the floor). If the complete environment and the EN device position were perfectly known, power could be focused at the position of the device. Power is transmitted via multiple beams simultaneously, reflected at the walls and the floor and sums up coherently at the EN device position. The scatterers may dominantly cause diffuse reflections, while the specular reflections along walls introduce strong fading patterns attributable to standing waves. Particularly, the wall opposite to an array in Figure 3.5 causes deep fades parallel to the  $xz$ -plane, spaced at  $\lambda/2$ . If beam sweeping using a predefined codebook is employed for the initial wake-up of EN devices, deep fades may pose a problem with initial access. EN devices located at unfavourable positions would require a large fading margin necessary for successful initial access, severely reducing the achievable initial access distance. A necessary fading margin would decrease the initial access distance by a factor  $1/\sqrt{M}$  under the assumption of a free-space path loss. That is, every 6 dB fading margin would halve the initial access distance.

Figure 3.6 (a) shows a close-up of the path gain (PG) distribution around the focal point when conventional MRT is employed with perfect CSI. The PG is evaluated on a circular disc in the  $xy$ -plane centred around the focal point at the position  $\mathbf{p}_{\text{EN}}$ . We initially chose its diameter as,  $2 y_{\text{EN}} \lambda / l_x$  which is a measure of the beamwidth at a distance  $y_{\text{EN}}$  from the array, resulting from geometric considerations. However, we reduced the diameter by a factor of 0.62 to omit a performance decrease due to low powers at the disc edges. When awareness of the channel model is used for beam sweeping, the deterministic part of the channel<sup>1</sup>  $\mathbf{h}_k$  can be predicted based on a target position  $\mathbf{p}_{\text{EN}}$ . Unfortunately, this will lead to strong local fading, as illustrated in Figure 3.6 (a), and a large fading margin is needed to overcome this local fading.

### 3.2.2 Beam Diversity

The distribution of power in the region surrounding the focal point can be improved through a suitable precoding scheme. Beam diversity can be effectively exploited to even out deep fades and generate a smoother power distribution in proximity of the focal point. As a simple scheme in a multibeam transmission, the phases  $\varphi_k$  of specular multipath component (SMC) beams can be varied to reduce the necessary fading margin at the cost of lower peak powers. In this regard, we propose to assign equal power to each of the beams to maximally affect the local fading. That is, we choose the transmit signal  $\mathbf{s} = \sqrt{P_{\text{TX}}} \mathbf{w}_{\text{BD}}$  with the weights defined as

$$\mathbf{w}_{\text{BD}} = \frac{\sum_{k=1}^K \mathbf{w}_{\text{BD},k}}{\left\| \sum_{k=1}^K \mathbf{w}_{\text{BD},k} \right\|} \quad \text{with} \quad \mathbf{w}_{\text{BD},k} = \frac{\mathbf{h}_k^*}{\|\mathbf{h}_k\|} e^{j\varphi_k}. \quad (3.1)$$

<sup>1</sup> For more details on our WPT channel model, please refer to the references [34] or [23].

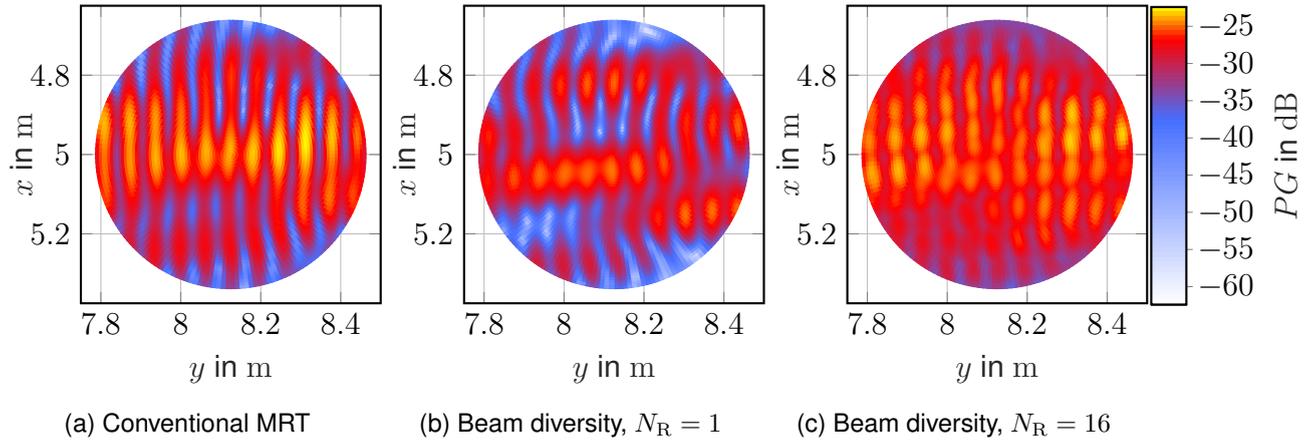


Figure 3.6: Exploitation of beam diversity for initial access to EN devices: Path gains  $PG$  are evaluated on a circular disc in the  $xy$ -plane located around the focal point. The PG of conventional MRT in (a) is computed assuming full CSI. The PG in (b) and (c) is computed using only environment information and by varying the phases of the individual SMC beams randomly using  $N_R \in \{1, 16\}$  realizations and taking the highest values.

The individual weight vectors  $w_{BD,k}$  are computed for some target position  $p_{EN}$  using the WPT channel vector  $h_k$  defined in deliverable D4.1 [23]. Figure 3.6 (b) shows the PG distribution for  $N_R = 1$  realization of  $K$  beam phases  $\varphi_k$  drawn from uniform distributions, i.e.,  $\varphi_k \sim \mathcal{U}(0, 2\pi)$ . This approach assumes known environment information in terms of the SMC model, while the scatterer components  $h_{sc,k}$  are unknown.

Deep fades still occur, but their depth has been reduced, i.e., the use of several realizations of random phases  $\varphi_k$  increases the probability of waking up an EN device located at an unfavourable position. This effect can be further exploited through multiple iterations of the beam diversity scheme: Figure 3.6 (c) shows the maximum PG for every position on the circular disc from  $N_R = 16$  realizations of  $K$  random beam phases. The smooth power distribution depicted in the figure will not exist at a single time instance, but after  $N_R$  attempts, every position has at least once experienced the peak PGs depicted. If the power received by an EN device exceeds the device sensitivity for one realization, this is sufficient to transmit a signal on which the RadioWeaves infrastructure can perform channel state estimation. An exhaustive beam sweep exploiting this scheme may initially take several tens of minutes. However, the procedure has to be completed only once, for the initial wake-up of EN devices.

To analyse the possible gain in initial access distance, we evaluate the cumulative distribution function (CDF) of the PG distribution across the circular disc. We compute the distribution from PGs across an equally-spaced Cartesian grid on the disc. Figure 3.7 depicts the CDFs of the maximum PG of  $N_R \in \{1, 2, 4, 8, 16\}$  realizations. For comparison, the CDF for MRT with perfect CSI (including point scatterers) is depicted, which shows higher peak powers, but also a higher probability of deep fades. For the first drawn realization ( $N_R = 1$ ), the complete range of power values lies below what is achievable with MRT. However, drawing more realizations, it is observable that low power values, i.e., deep fades, become less likely. Targeting an outage probability of less than 1%, i.e., the horizontal line at  $\Pi(PG) = 10^{-2}$ , the beam diversity scheme using  $N_R = 16$  realizations would have reduced the necessary fading margin by 12 dB when compared with conventional MRT. That corresponds to an improvement in initial access distance by a factor of 4, assuming a free-space path loss. Four repetitions  $N_R = 4$  reduce the fading margin by 12 dB compared to a single shot ( $N_R = 1$ ), which even leaves a gain of 6 dB when factoring in the loss

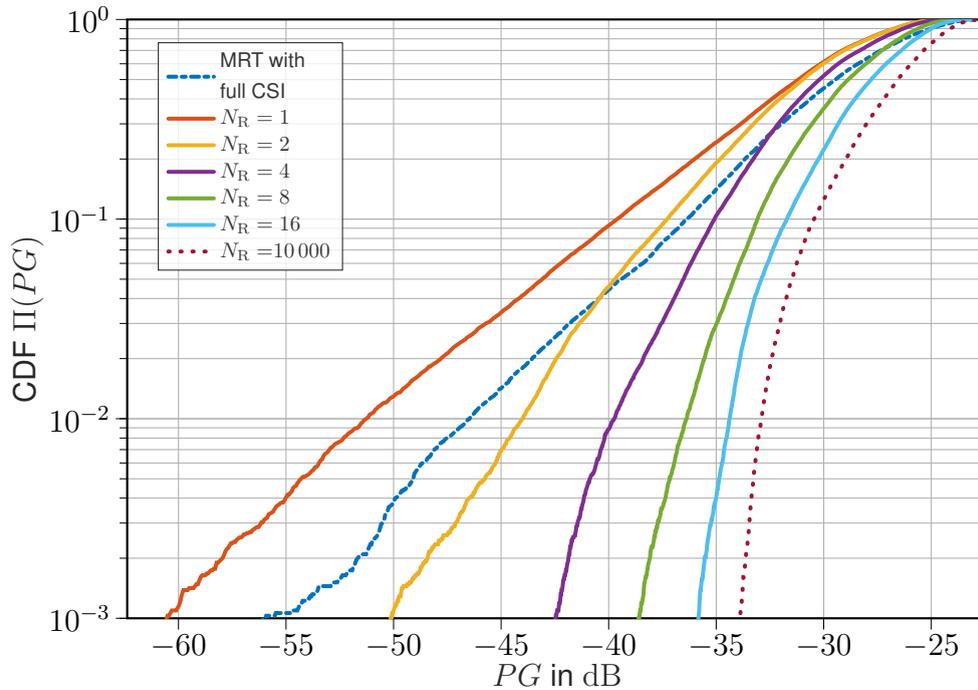


Figure 3.7: Beam diversity exploited for the initial access to EN devices [34]: Path gain  $PG$  evaluated within the focal point. Varying the phases of the individual multipath component (MPC) beams randomly using  $N_R \in \{1, 2, 4, 8, 16\}$  realizations and taking the highest values can help to reduce the necessary fading margin for initial access.

of energy due to the four re-transmissions. We would like to argue, however, that an absolute power gain is not the key aim of the proposed technique. The aim is merely to wake up the EN device to enable CSI estimation in the initial-access phase. After the channel estimation, MRT will be used for WPT. Finally, we performed a Monte Carlo simulation ( $N_R = 10\,000$ ) to hint at the maximum achievable gain obtainable with beam diversity in the given simulation scenario.

### 3.3 Conclusion

This chapter elaborates on the de-registered mode operation, *i.e.*, broadcast information and initial power transfer. Here, we advocate decoupling idle and active mode. The covered area by the network is constrained by the idle mode operation, while the active mode is responsible for point coverage. Hence, in this chapter, we focus on how to optimise the idle mode operation for communication and power transfer, without being constrained by the active mode operation. We present an approach to delivering wireless power transfer to EN devices without CSI by using information about the environment.

# **Part II**

## **Registered Mode Operation**

# Chapter 4

## Establishing Routing in RadioWeaves

In this chapter, joint grouping of UEs and fronthaul data routing through the RadioWeaves network is discussed. It provides an initial exploration of routing resource allocation in a distributed system given the constraints inherently imposed by the connections –or lack of connections– between the CSPs and ECSPs. This work is extended in the next Chapter, where next to communication, other services such as positioning and wireless power are considered.

### 4.1 Introduction

To make deployment of a large number of distributed radio elements simple and cost-efficient, a shared fronthaul (FH) together with a high degree of integration and miniaturization can be used. The FH structure used by RadioWeaves may be referred to as segmented fronthaul –each CSP is connected to one or more neighboring CSP via ECSPs, interfacing segments that can be used for transferring power, downlink (DL) data packets and precoding weights, uplink (UL) combining weights and symbol estimates, etc. An important property of such FH structure is that a given CSP is generally not directly connected to a central-processing unit (CPU) but signals to and from it need to pass multiple segments to reach their destinations. To transfer signals to and from multiple users (UEs) and multiple CSPs, careful routing solutions are required to utilize the data transfer capabilities of the segments as fully as possible.

In segmented FH solutions, CSP grouping for each co-scheduled UE may first be performed to define the serving CSP subset that transmits data to the UE, using joint coherent precoding. The grouping may be based on radio considerations, e.g., UE-CSP pair link qualities, maximum number of UEs that a CSP can simultaneously serve, etc. and per-group combining weights are determined. In simple solutions, a routing solution is then determined to distribute DL data from the CPU to the CSPs, and/or forward UL data from CSPs to the CPU. If a routing path for a CSP cannot be established due to FH segment capacity limitations, the resulting DL precoding may be highly suboptimal since a missing transmitter (TX) component will distort the spatial energy focusing and interference null steering, or resulting UL combining.

Individual UEs could be accommodated sequentially, selecting the preferred CSP group for each UE and determining whether it can be routed, and subsequently only keeping UEs whose CSPs are successfully routed. However, this would be highly inefficient in terms of FH resource utilization and would result in blocking many UEs.

To improve resource utilization, joint grouping and routing approaches that explicitly test different grouping options and check their resulting routing solutions to determine the best grouping options may be envisioned, but full joint grouping and routing solutions may be feasible only for a very low number of UEs and CSPs; their complexity for scenarios of practical interest is prohibitive.

We thus see a need for a low-complexity approach to jointly consider CSP grouping and routing algorithms that would avoid blocking UEs unnecessarily and can provide robust precoding in the presence of routing failures for some CSPs. In this chapter, performance of DL D-MIMO with routing constraints in segmented FH is analysed. The aim of routing is to select well-performing CSPs data routing paths for each of the users to their CSP sub-groups, such that the user signal-to-interference-plus-noise ratio (SINR) is kept high in the presence of the routing constraints. The results indicate that limited segment capacity combined with long FH path lengths may cause problematic failures when delivering data to serving CSPs for the scheduled UEs.

## 4.2 Segmented fronthaul and related routing tasks

Figure 4.1 depicts a system configuration where each UE is assigned serving CSPs and an ECSP that receives the data packets for the UE from the CPU and distributes them further to additional serving CSPs; this way the amount of unique packets to be routed is reduced compared to the number of individual CPU-CSP pairs per UE. In Figure 4.1, each UE is assigned three best CSPs (e.g., based on long-term path-gain).

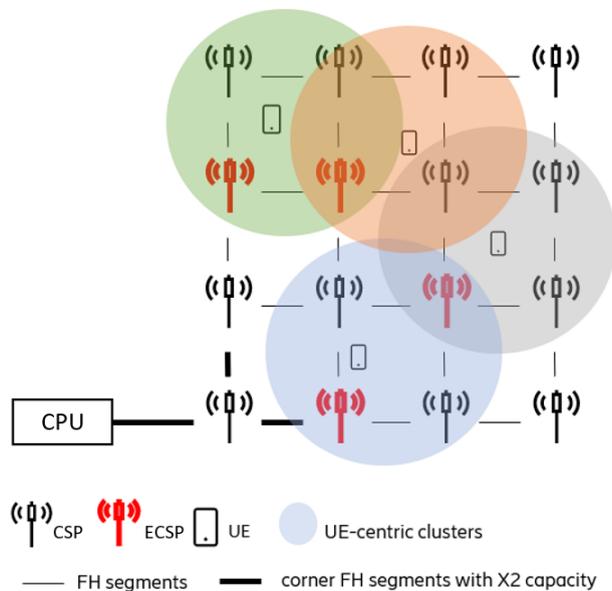


Figure 4.1: An example layer where each UE is assigned serving CSPs and ECSP that receives the data packets for the UE from the CPU and distributes them further to additional serving CSPs.

In early RadioWeaves studies, the FH segment capacity has been assumed unlimited. However, in reality, there are FH practical limitations, causing missed CSPs connections, where such failures have not been taken into account until now. So, a proper multi-path routing strategy is desired that can help find an appropriate route from the source, i.e., CPU, to the destination(s), i.e., serving CSPs considering the FH limitations. In realistic scenarios, when there is high UE

traffic and the segment capacities are limited, some of the segments will be congested after a certain number of paths have been established, and therefore some serving CSPs cannot provide service to some UEs. To minimize this impact, we introduce a routing algorithm with two levels:

1. Connecting the closest CSPs to the CPU: these CSPs will function as ECSPs, performing an aggregation function, and are responsible for the connection between the CSPs and the CPU. In particular, missing an ECSP connection would result in dropping the UE. The proposed routing algorithm can help find alternative serving CSPs to select as an ECSP where the closest CSP's connecting segment is occupied.
2. Connecting the serving CSPs to the ECSP, where the CSPs are based on the received power of the UE's reference signals. There are two different scenarios to select the number of serving CSPs:
  - (a) Fixed (predefined) CSP subgroup size.
  - (b) Flexible CSP subgrouping: e.g. depending on the available segment capacity and the number of UEs. This can be handled as a further optimization problem.

In this study, we assume the first scenario, i.e., fixed CSP initial subgroups before updating due to routing constraints, for the sake of simplicity of analysing the performance of the implemented routing algorithm, and to show some important results such as segment utilisation, serving CSPs' successful connection ratio, and path length used from source ECSP to the destination serving CSPs. These results help understand in which situations the second scenario can/cannot be implemented, considering the time complexity of the flexible CSP subgrouping algorithm to find the optimal number of the subgroups.

## 4.3 System model

In this section, a UE-centric CSP subset selection, data routing, and updating the CSP subset in RadioWeaves with segmented fronthaul is presented. The structure of the proposed model is depicted in Figure 4.2.

The considered network consists of one CPU,  $N$  CSPs and  $K$  UEs (Figure 4.1), connected via segments that serve as FH links (e.g., fiber-optic cables) between two CSPs. Each CSP is connected to one or more of its neighbours via segments. A UE will connect to the  $M$  CSPs where  $M$  is the number of serving CSPs in a sub-group. It is assumed that all CSPs and UEs are stationary. It is assumed that CSPs are equipped with multiple antennas and UEs have a single antenna, nonetheless, it is straightforward to generalise the equations suitably for multi-antenna UEs. In our configuration, we separate four steps in the process: i) CSP subset selection, ii) routing discovery, iii) subsets update, and iv) performing routing and precoding. These steps are presented below in more detail.

### 4.3.1 UE-centric CSP subset selection

UE-centric CSP selection/clustering method is used in which each UE is served by a subset of CSPs, typically the ones that provide the best, or a sufficiently high, average SNR. The subsets may be partially overlapping between neighbouring UEs and thus the CSPs cannot always be divided into disjoint sets, as is the case in conventional cellular networks. We select the CSPs with the best channel quality that contribute at least  $\alpha\%$ , e.g., 95%, towards the UE. The CSP

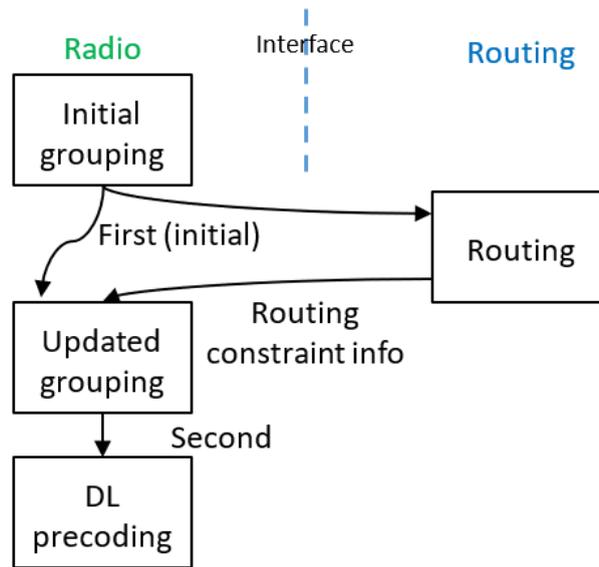


Figure 4.2: System with proposed processing sequence.

grouping may be performed centrally at the CPU that possesses the information on the channel large-scale fading coefficients for all UE, or alternatively, each CSP can decide to be part of the serving cluster of a given UE based on local observations. CSPs can have a limited maximum number of UEs to serve, where CSPs will prioritise the UEs with the better channel gains. For the sake of simplicity, in our solution, each initial subset consists of  $M$  CSPs which have the best channel gains for a given the UE.

### 4.3.2 Routing discovery

A route discovery step determines the paths used to forward data packets from the CPU to the serving CSPs for the co-scheduled UEs. The routing algorithm attempts to connect  $M$  CSPs to serve the  $k$ -th UE, as determined by the CSP subset selection algorithm. A two-level routing strategy is proposed, which is done for each device  $k$ :

1. L1 generates the route from the CPU to ECSP $_k$ .
2. L2 discovers the route(s) from the ECSP $_k$  to the other serving CSPs, i.e., CSP $_{k,m}$ ,  $m = 2..M$

We use the strategy of the previously proposed multi-path routing algorithm Hybrid Multi-Path Routing algorithm (HyMPRo) defined in [27]. For the sake of simplicity, we assume a single data flow at each FH segment in the route discovery process. A route can be established if the involved segment(s) have available capacity. Before starting route discovery, routing constraints are determined for the multi-path route discovery utility function defined in (4.3). The route discovery constraints, i.e., FH segment capacity and path length, in this work are given as follows:

**FH segment capacity.** The state of the FH segment capacity can be defined as:

$$\beta(S_z) = \frac{\text{UsedCap}(S_z)}{\text{TotalCap}(S_z)}, \quad (4.1)$$

where  $\text{UsedCap}(S_z)$  is the number of existing packets in the segment and  $\text{TotalCap}(S_z)$  is the

maximum number of packets that can be conveyed in the segment  $z$ , where  $z = 1..Z$  with  $Z$  the total number of segments in the network topology. The piggybacked segment capacity information along route  $r$  from the CPU to ECSP in L1 or from the ECSP to CSP at L2 will be collected and evaluated at the CPU through the following:

$$\rho(r) = \prod_{w=1}^W 1 - \beta(S_z), \quad z \in Z, \quad (4.2)$$

where  $\rho(r)$  is the congestion awareness metric with  $0 \leq \beta(S_z) < 1$ .  $W$  is the total number of involved segments at each route  $R$  during the route discovery process. As the segments between the intermediate CSPs become congested,  $\rho(r)$  will approach 0. The amount of free capacity on the segments of a chosen route becomes significant if the overall data traffic increases. In this case, it is more rational to forward the data in a newly generated route through the segments with available capacity. Otherwise, the data packages of the next route are confronted with the risk of being dropped in a region of high traffic load.

**Path length ( $L$ ).** Apart from the segment capacity, the number of segments that can be involved in the route, has a major role in successful connection versus dropping. The path length is the number of segments in a route  $R$ , where the maximum path length limits the number of segments. If  $\beta(S_z) \cong 1$ , meaning that the segment is almost congested, there are a lot of active co-scheduled UEs, or that segment capacity is limited. That is why a higher maximum path length value relaxes the problem to find alternative transmission routes, where it can be adjusted based on the current requirements. Although it brings computational complexity and communication delay, the UE can still be served. Considering both constraints, the utility function can be calculated as follows:

$$f(r) = \frac{\rho(r)}{L(r)}. \quad (4.3)$$

A route request packet (RREQ) is created by the source nodes, i.e., CPU or ECSPs, and flooded through the network to determine the alternative routers towards the destination, i.e., the serving CSPs. The utility function values are evaluated for each candidate route after receiving the RREQ, which includes the path length and congestion metrics. The best route, i.e.,  $\operatorname{argmax}_{r \in \mathcal{R}} f(r)$  is selected by the CPU, with  $\mathcal{R}$  the set of all routes.

### 4.3.3 CSP subset updates

The radio algorithm uses the routing report info to modify the original UE-centric CSP subsets, if needed. The first CSP sets will be modified in the “radio unit” based on the routing outcome info to determine the actual set of CSPs that may serve as serving CSPs to the co-scheduled UEs, subject to practical routing limitations in the fronthaul network. In the current solution, when a CSP can not be connected to the subset it is removed from that set prior to computing the precoding weights.

If all CSPs in the first set of  $N$  CSPs for a UE could not be connected due to any limitation (e.g., segment capacity, UE prioritisation), that UE may be reallocated in the following co-scheduling update interval, possibly with elevated prioritisation to satisfy fairness; this improvement is out of scope of the current study.

### 4.3.4 Performing routing and precoding

In this step, a controlling node in the network or the individual ECSPs finalize the routing algorithm to determine paths for DL packet forwarding. It then performs DL precoding according to known techniques. This includes e.g.:

- computing precoding weights based on channel estimates and/or interference estimates for the UE-CSP links (we have used zero forcing (ZF) precoding in this work)
- routing data packets to the CSPs, e.g., over the CPU to ECSP and ECSP to CSP routing stages, using paths found according to the determined routing parameters, like the determined pipelining and routing slot budget choices.
- performing modulation (in some cases previously applying coding)
- applying precoding weights and transmitting the modulated and precoded symbols to the UEs in the DL.

## 4.4 Performance evaluation

In this section, we discuss example cases in which the number of UEs and FH segment capacity strongly affect the successful connection ratio and the resulting radio performance.

Table 4.1: Simulation parameters.

Parameter	Symbol	Value
Scenario		Indoor Factory
Carrier frequency		3.5 GHz
Bandwidth		100 MHz
Simulation area		100×100 m
Number of CSPs	$N$	16 - deployed in a regular 4x4 pattern
Array size at CSP		2 x 2 elements
Number of blockers		1000 – uniformly randomly distributed
Duplex		Symmetric TDD, DL/UL 50/50
UE / CSP height		1.5 m / 1.5 m
Propagation model		3GPP InH [71]
CSP power		13 dBm
UE noise figure		10 dB
FH segment capacity	TotalCap( $S_z$ )	5, 10, 100 (x2 for segments 1, 2 as in Figure 4.1)
Max path length	$L$	5, 10, 100
Number of UEs	$K$	8, 15 – uniformly randomly distributed
Desired serving CSP set size	$M$	5
Number of realizations		50

### 4.4.1 Simulation parameters and assumptions

In this section, performance of the large-scale D-MIMO system depicted in Figure 4.1 is analysed in a downlink indoor scenario over an area of 100m × 100m with randomly distributed 1000 blockers. The performance has been evaluated for  $N = 16$  CSPs, and  $K = 8$  and 15 uniformly distributed UEs. It is assumed that UEs and CSPs are stationary. The configuration parameters used in the simulations are given in Table 4.1. Perfect channel estimation is assumed. RF

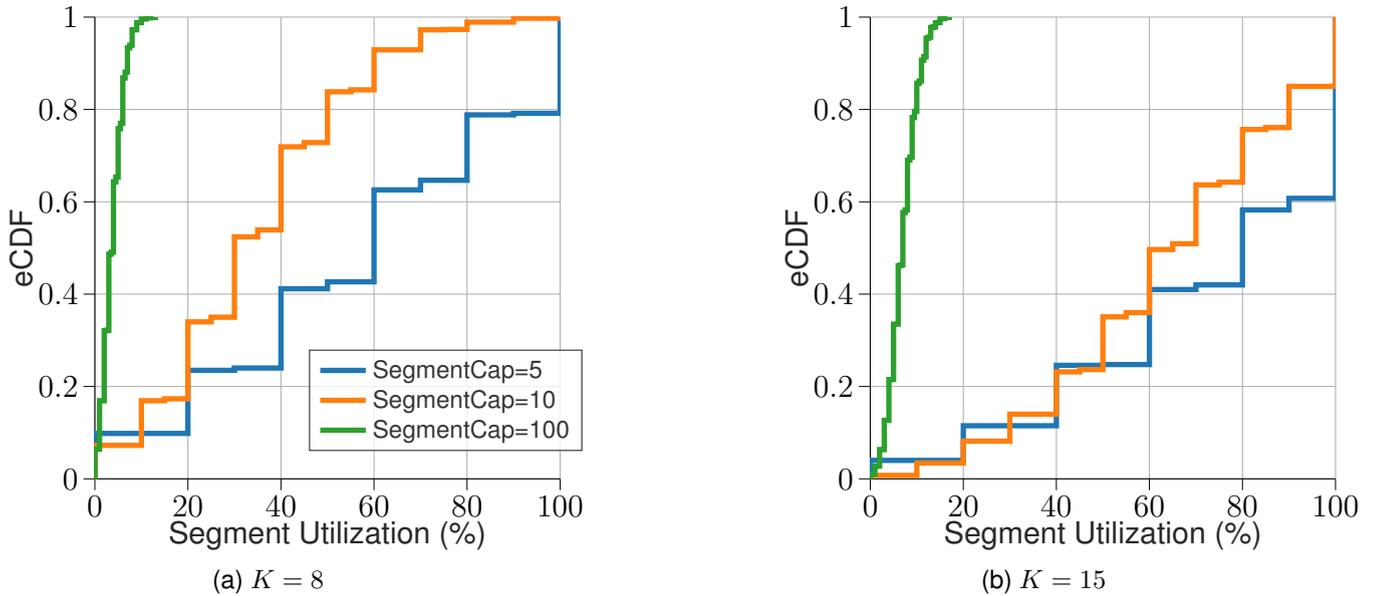


Figure 4.3: Empirical cumulative distribution function (eCDF) of the segment utilisation for different segment capacities with serving CSP subset of 5.

imperfections, hardware impairments, phase noise, power amplifier non-linearities are omitted from the scope of this work.

## 4.4.2 Performance evaluations

### Segment Utilization

The CDF of the segment utilisation for given segment capacities are shown in Figure 4.3. The results illustrate that, when segment capacity is low, there is a higher level of segment utilisation. As segment capacity increases, the utilisation reduces. However, as the number of UEs increases, the average segment utilisation also increases. It should be noted that, when segment capacity is limited, such as 5 packets per time unit, increasing the number of UEs in a case of low utilisation samples would not further enhance the segment utilisation due to the fact that some of the UEs are dropped even with only 8 UEs.

### SINR

Figure 4.4 shows the distribution of received SINR with the post-routing updated CSP set for different segment capacities. It shows that when the number of UEs are high and more routing failures occur, Figure 4.4b, the resulting SINR is decreased especially for lower segment capacities. It should be noted that when segment capacity is set to 100, the segment capacity is essentially unlimited and yields SINR performance equal to unconstrained routing. Figure 4.4b also shows that when segment capacity is 10, there are limited route discovery failures and consequently some UE drops which slightly affect the SINR.

### Successful Connection Ratio

Figure 4.5 shows one of the important results that the UE drop ratio increases as segment capacity constraint gets tighter. When the segment capacity is limited, some of the segments that

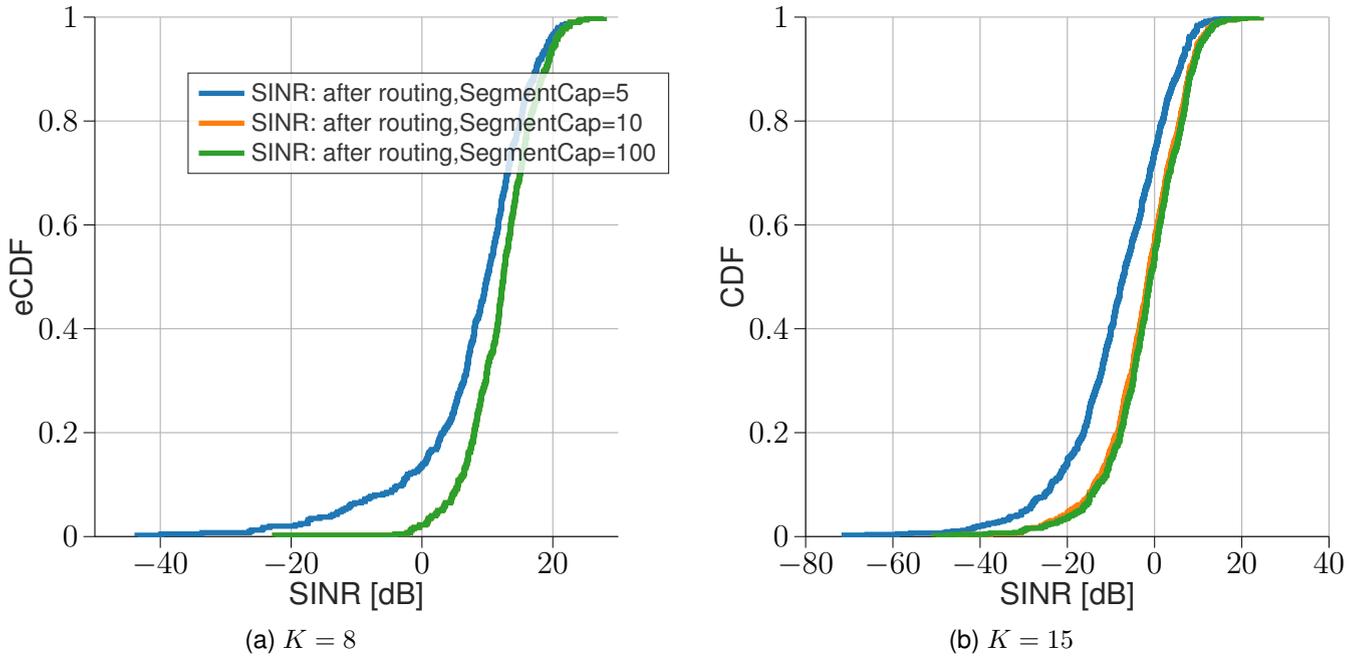


Figure 4.4: ECDF of the radio link SINR for different segment capacities and users served.

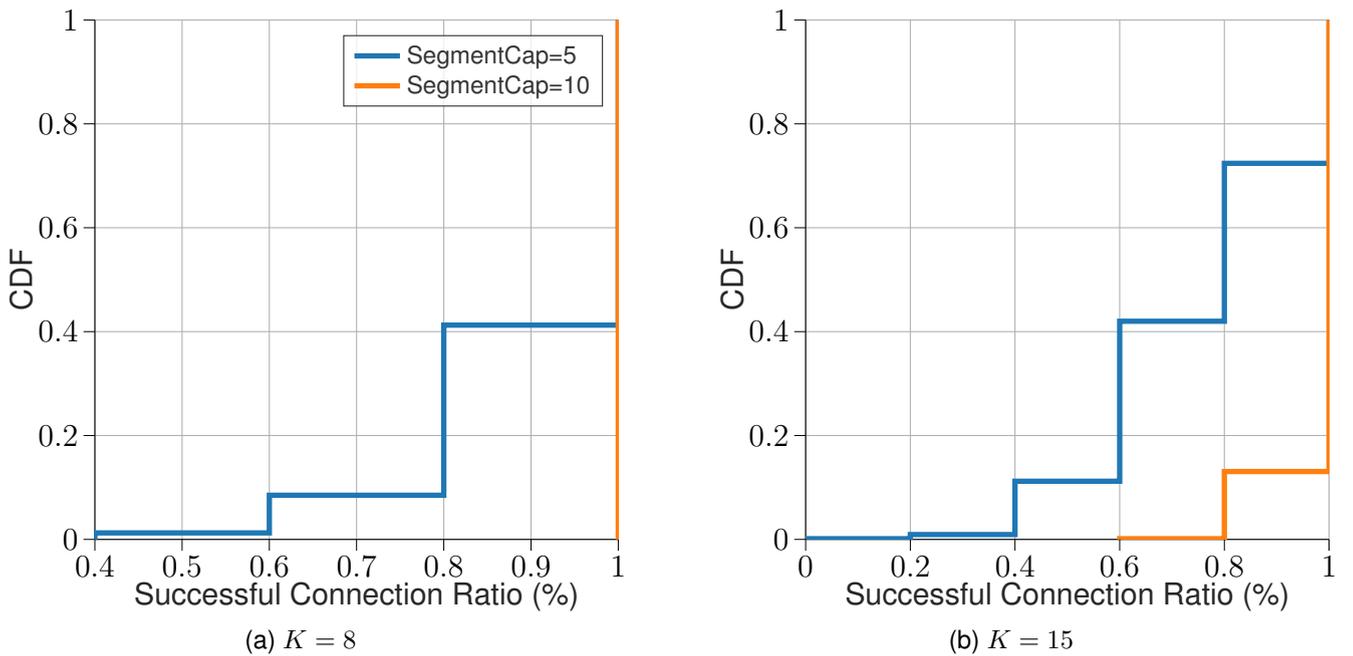


Figure 4.5: Successful Connection Ratio.

are connected to the serving CSPs will be occupied even for a lower number of UEs. Figure 4.5a shows that there is a full successful connection ratio for all CSPs in each subset of  $M = 5$  for a segment capacity 10 or higher. However, when the number of UEs increases from 8 to 15, Figure 4.5b, connections to some of serving CSPs will fail, hence the results become worse for practical segment capacities.

## Path Length from ECSPs to the destination/serving CSPs

Here, the total length of the established path from the source ECSP to the destination CSP in L2 is evaluated. We observe that the constraints such as the segment capacity and the number of UEs affect the length of the resulting routing path. In Figure 4.6a, the maximum path length for both segment capacities 5 and 10 is 5, whereas, for unlimited segment capacity, such as 100, it may increase up to 7. This is because the routing algorithm tries to find alternative paths, albeit with a high path length, to avoid UE drop.

We may observe that, for unlimited capacity, longer paths appear in the distribution while it could intuitively be expected that the routing algorithm could select the shortest paths for all connections since even the busiest segments never run out of resources. This is an artefact of the algorithm since we are not explicitly trying to minimize the path lengths but just ensure a connecting path (any viable path), and the algorithm is configured to look for unused segments first to minimize the search time/complexity in normal constrained situations. If the routing algorithm would be configured to prioritize the shortest paths, then the paths found for the unlimited capacity case would also be the shortest.

Figure 4.6b shows that, when increasing the number of UEs, segment utilisation is increased where the fraction of routes with path length up to 2 is increased to 80 percent. The length of the path is increased to 8 when unlimited segment capacity is offered. It is directly related to (4.3) from [27] which consider  $L$  values for applying load balancing in individual segment utilisation. Thus, whenever the segment capacity is high, the route discovery algorithm calculates more alternative routes with higher path lengths. So, in delay-sensitive scenarios, e.g., ultra-reliable low-latency communications (URLLC), it may be preferable to set a lower maximum path length to avoid considering a large number of alternative routes from the source to the destination. On the other hand, this may cause UE drops at lower segment capacities. Thus, to strike a balance in the proposed algorithm, there is a correlation between maximum path length and segment capacity. As a result, Figure 4.6b shows that whenever there are no limitations in both parameters such as maximum path length and segment capacities, a suitable max path length can be 8 when there are 15 UEs.

## 4.5 Conclusion

In RadioWeaves environments with many CSPs, determining an efficient data routing solution for which segments should be used for data transmission is a vital issue. We have studied a 2-level routing-based downlink transmission where in the first level CPU will connect to the ECSPs, and in the second level, the selected ECSPs make a connection with the selected serving CSPs to send/receive data related to the corresponding UEs. Sufficient segment capacity is a prerequisite to avoid serving CSP routing failures and resulting per-UE SINR reduction.

It is motivated to further investigate opportunities for joint optimization of routing and CSP selection where alternative ECSP selection (not closest to CPU) could be considered to show the performance of the routing with limited/unlimited segment capacity and path length.

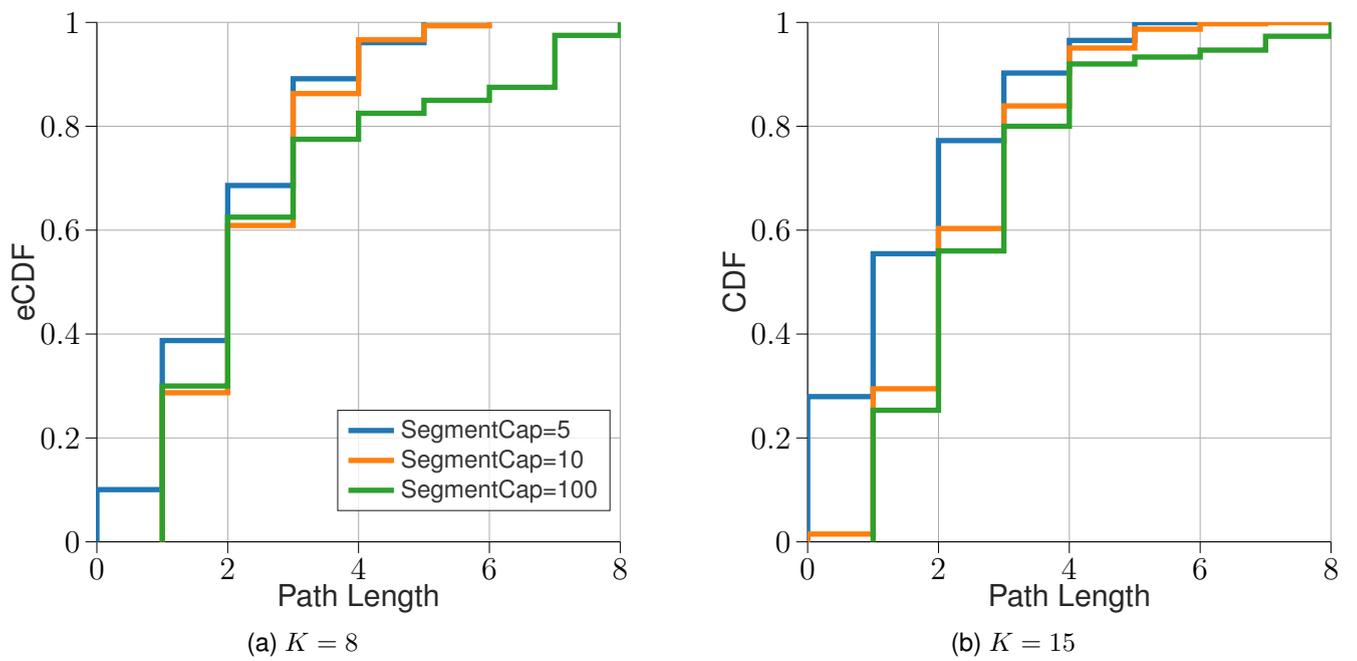


Figure 4.6: ECDF of path lengths from ECSP to serving CSPs

# Chapter 5

## Federations: CSP Resource Allocation for Diverse Applications

### 5.1 Introduction

In this chapter we will discuss resource allocation for registered mode operation, primarily in terms of spatial resource allocation, that is, associating CSPs and other resources with the groups of UEs they will serve. A group of serving resources and served UEs we call a *federation*, a concept introduced in Deliverable 2.1 [37] and further elaborated in [25]. We will begin with an overview of the problem using an illustrative example, along with defining the scope for the use of federations. We will then discuss existing work on fair allocation and its relevance to REINDEER and RadioWeaves. From there we will describe our framework for federation creation and operation, and then detail methods for orchestration of federations, in particular how allocation of CSPs and UEs can be performed and can be dynamically updated as the situation changes.

#### 5.1.1 Problem Overview

A *federation* is a set of resources that serve a particular application running on a set of UEs, together with the federation anchor that coordinates the federation. The resources can include communication resources, i.e., CSPs, resources for wireless power transfer, resources for positioning, edge computing resources, data storage resources, or any other resources provided by the RadioWeaves infrastructure and needed by the application. Note that the UEs belonging to a given federation do not need to constitute all UEs running the application, nor does the application have to be the only one running on the UEs belonging to the federation. It is thus possible to split the UEs running an application into multiple groups, where each group belongs to one federation, in order to facilitate easier or more efficient resource allocation. Equally, it is possible for a physical UE running multiple applications to belong to multiple federations, represented by multiple virtual UEs, each with one application.

The federation anchor is responsible for the operation of the federation. This may include such functions as scheduling of radio resources (e.g., time or frequency resources) for the users of the federation, communicating with other federation anchors to negotiate resource allocation, signalling between the infrastructure and the served applications, and admission control of UEs to the federation. Of note is that there is no central coordinator for all federations defined in the RadioWeaves architecture, in keeping with the distributed, cell-free paradigm. This does not

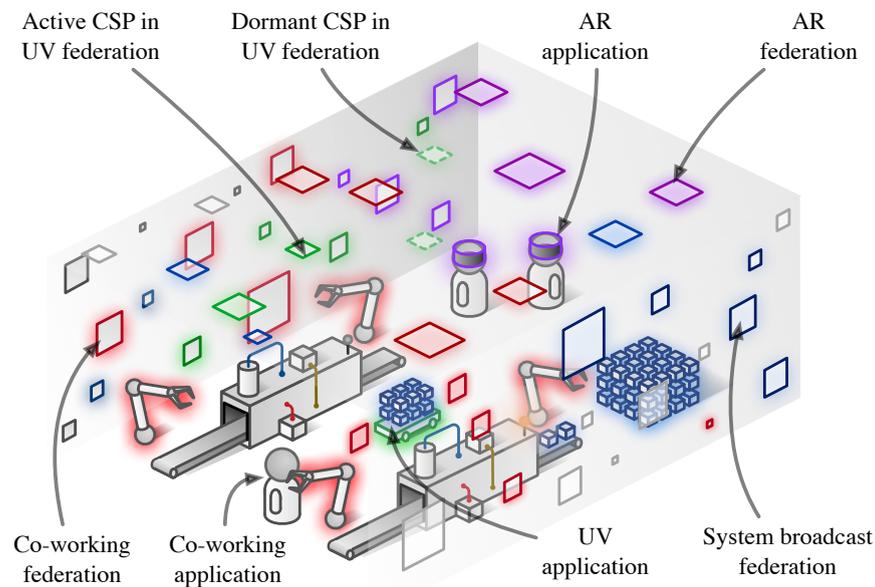


Figure 5.1: An example RadioWeaves (RW) deployment in a smart factory, with federations and their served devices colour coded. The four applications are augmented reality (AR) for professional applications (purple), tracking of robots and unmanned vehicles (UVs) (green), tracking of goods and real-time inventory (blue), and human-robot co-working (red).

however, exclude the use of centralised resource allocation methods, for all or a subset of the resources and applications. In such cases, the federations may either elect one of their numbers to manage the resource allocation process or utilise a separate process running on one of the ECSPs present in the infrastructure.

The overall problem of federation orchestration then includes the following:

- Creation and removal of federations as necessary
- Association of UEs and applications to federations
- Allocation of resources to federations
- Dynamic update of resource allocations and UEs' associations to federations

We will in this chapter focus on CSPs as the primary resources to be allocated to federations.

### 5.1.1.1 Example

Figure 5.1 shows an example deployment of RadioWeaves in a smart factory, with four federations, shown in different colours. RadioWeaves CSPs are deployed throughout the production hall on the walls and ceiling, and are dynamically assigned to federations to serve the devices and their running applications that are present at any given time. The constellation of CSPs assigned to each federation is tailored to the particular application's requirements.

In Figure 5.1, there are currently four different applications running, taken from the use cases presented in [40]. These are AR for professional applications (shown in purple), tracking of robots and UVs (green), tracking of goods and real-time inventory (blue), and human-robot co-working (red). Each application is served by a federation, with its CSPs shown in the same colour as the application.

For the AR for professional applications use case, human workers wear energy-neutral AR goggles, which display digital information overlaid on the physical area in which they are working. As described in [40], using energy-neutral devices allows the goggles to be extremely light and thus comfortable to wear, but it increases the requirements on the infrastructure, which must transmit uncompressed video to the goggles as they lack the processing capability to decode a compressed video stream. This means that this application requires an extremely high data rate (potentially up to 3 Gbps [40]), as well as very low latency to prevent motion sickness. The goggles also need to be powered via wireless power transfer. To meet these requirements, the federation thus consists of a cluster of CSPs located on the wall and ceiling close to the user devices. The close proximity of the CSPs gives a good link budget both for communication and wireless power transfer, and the short distances between CSPs keeps the latency low as CSI does not need to be transmitted a long through the X-haul links between the CSPs.

Meanwhile, for the UV tracking use case, one of the biggest challenges is mobility. Tracking of the UV also requires low latency, high reliability, and a relatively high data rate. To serve this application, we again show a cluster of nearby CSPs, though somewhat more spread out to also provide spatial diversity for positioning. In order to account for the UV's mobility, the federation is adapted as the UV moves. Some CSPs are not currently serving the UV but are standing by to join the federation as it is predicted the UV will move in their direction. This dynamic adaptation of the federation as it "follows" the robot around the factory floor ensures a consistently good channel even as the robot passes objects that may cause shadowing, while also allowing the federation's panels to be physically close to each other to provide low latency communications.

In the tracking of goods use case, the requirements on latency, data rate, and reliability are significantly relaxed, but instead positioning accuracy becomes the most important requirement as the items to be tracked move around the production hall. A key ingredient in providing high-accuracy positioning is aperture size, and so for this application the federation is assigned panels spread out over the deployment area. The spatial diversity thus provided also ensures goods can be located and the tracking devices communicated with anywhere in the production hall. For this use case, a dedicated federation could be used, as shown here, or instead the UEs could be served by the same set of CSPs as for system information broadcast, since only low data rate, infrequent communication is needed. When to create a federation for a given application and set of UEs is a design issue that depends on the overhead of federation creation and orchestration, as well as the capabilities of the broadcast mode of the infrastructure.

Finally, for the human-robot co-working use case, we again see a dense cluster of CSPs located close to the user devices. As in the AR use case, this provides good data rates, high reliability, and low latency. This use case also does not require wireless power transfer, as it does not employ energy-neutral devices. This, combined with a somewhat lower data rate compared with the AR use case, means that fewer CSPs are needed per user for the human-robot co-working federation.

### 5.1.2 Scope

Federations should be used where the cost of instantiating and managing them is justified in terms of the application requirements. This implies that for applications with low data rates or sporadic data, it may not be appropriate to create federations to serve them. More specifically, UEs should be placed in federations only when they are in registered mode, and grant-based access should usually apply for the UEs belonging to federations. Some applications, even some with demanding and strict requirements, are therefore out of the scope of federation orchestra-

tion. These applications are instead served by the same infrastructure resources as are used for broadcasting of system information, with either grant-based or grant-free access. Typical applications served by federations are then those that have high data rates, especially over an extended period of time (long or frequent packet sessions), and these applications may then have additional requirements that affect the formation of the federation, such as low latency, high reliability, positioning, or wireless power transfer.

### 5.1.3 Existing Work on Fair Allocation

There is a large body of existing work on fair allocation of indivisible goods, so we will here only discuss the most relevant work. Surveys on current research on fair allocation can be found in [9, 14]. The general setting is that there is a set of *goods*, also sometimes called items or resources, to be divided amongst a set of *agents*, also sometimes called players. The goods are indivisible, that is, a good must be entirely allocated to one agent and cannot be split up. This mirrors many real-world goods such as pieces of jewellery, books, or furniture, whose value derives from the item being intact. Each agent has a set of *utilities*, also called valuations or preferences, for either individual items, or more generally sets of items, known as *bundles*. For RadioWeaves, the agents are federations, and the goods are CSPs and other resources. Given the combinatorial nature of fair allocation with indivisible goods, many of the problems in this area have been proven to be NP-complete or NP-hard, including those most closely related to the federation allocation problem. It is therefore likely that approximation methods will be the most promising for solving this problem in practice in real RadioWeaves deployments.

Much of the existing work focuses on proving the possibility or impossibility of fair allocations in different settings, for some notion of fairness. Some commonly used notions of fairness include:

- **Envy-free:** An allocation is *envy-free* if no agent wants to swap their allocation with that of another agent. In practice, envy-freeness is usually not obtainable for indivisible goods (for example if two agents compete for a single item), so other notions are applied for fair allocation.
- **Envy-free-but-1 (EF1):** An allocation is *EF1* if, for every agent, the agent does not prefer the allocation of any other agent to their own, if one good is removed from the other agent's allocation.
- **Envy-free-but- $c$  (EF $c$ ):** This is similar to EF1, except that up to a certain number  $c$  of goods can be removed from another agent's allocation to make it no longer preferred for another agent.
- **Maxmin share (MMS):** Each agent divides all the available goods into as many shares as there are agents, in any way they like. An agent receives an MMS-fair allocation if the allocation they actually receive is at least as valuable to them as the least valuable share in their own division of the goods. The entire allocation is MMS-fair if every agent receives an MMS-fair allocation. This is similar to the notion of cut-and-choose with two agents: the first agent divides the goods into two bundles, and the second agent can then choose from these two bundles.
- **Proportionally fair:** If there are  $k$  agents, an allocation is *proportionally fair* if every agent values their own allocation at at least  $\frac{1}{k}$  of their valuation of the entire collection of goods.

The fairness notions above are listed in order of difficulty, for example it is generally more difficult to find an allocation that satisfies EF1 than EF $c$  or MMS fairness. There are also many more

fairness notions defined in the literature not listed here, but these are the most common ones and many of the others are relaxations of or variations on them.

For RadioWeaves, while fairness is one of the goals of federation resource allocation, efficiency is also important, that is, that resources in RadioWeaves are used in an efficient way, with no more resources allocated than are needed to satisfy the requirements of our targeted use cases. We may thus prefer an allocation that is unfair but where all application requirements are still met, to one that is fair, in cases where this improves efficiency. We may even wish to depart from fairness altogether and prioritise critical applications over less critical ones. Nonetheless, the literature on fair allocation provides a rich source of existing algorithms that we can draw upon, as well as complexity results that illustrate the difficulty of such allocation problems, and the results that can be achieved.

In [22], different notions of fairness such as those above are also used to characterise the degree of conflict between agents' preferences. The fewer agents' preferences are in conflict, the more difficult notions of fairness can be satisfied for a given problem. Preferences that are identical between agents, or that place the items in the same preferential order, are more conflicting than other types of preferences and make it more difficult to find fair allocations. Understanding the level of conflict between federations' utility functions can therefore be useful for admission control and for determining whether we are likely to be able to satisfy all federations' requirements simultaneously.

The allocation problem from the literature that most closely fits our federation resource allocation is what is known as the Santa Claus problem [20]. In this problem, there is a set of children (agents) and a set of presents (goods), and the presents need to be distributed amongst the children as fairly as possible, taking into account the children's' preferences for the different presents. In the RadioWeaves setting, the children are analogous to federations, while the presents represent the resources to be allocated. The Santa Claus problem has been shown to be NP-hard [53].

In [20], the authors introduce two different linear programming (LP) formulations for (relaxations of) the Santa Claus problem. The first, known as the assignment LP, is the natural way to formulate the problem as described, however, it suffers from a large integrality gap to the corresponding integer programming formulation. For this reason, they then introduce a second formulation known as the configuration LP, which instead has a variable for every possible bundle, or configuration, of goods that can be allocated. This yields a much stronger linear relaxation of the problem. By taking the dual problem of the configuration LP and applying the ellipsoid method, combined with a binary search on the objective value, the configuration LP can be solved to any desired accuracy in polynomial time. However, for arbitrary item utility values, the configuration LP can still have a large integrality gap; in particular a hard case is when some items have large values such that some agents can be satisfied by single items. Other approximation algorithms for the Santa Claus problem that deal with this issue can be found in [11, 12]. Unfortunately, there are a number of properties of the Santa Claus problem and other similar problems that make algorithms designed to solve them not directly applicable to federation resource allocation in RadioWeaves.

The federation resource allocation problem distinguishes itself in three main ways from the problems previously studied in existing work. The first is that the focus of the majority of the existing research has considered particular classes of agents' utility functions. The most common is additive utility functions, where an agent's utility  $U(\mathcal{G})$  for a bundle of goods  $\mathcal{G}$  is given by the sum

of its utilities for each good in the bundle, i.e.,

$$U(\mathcal{G}) = \sum_{g \in \mathcal{G}} U(g) \quad (5.1)$$

where  $U(g)$  is the utility for an individual item  $g \in \mathcal{G}$ . Some results are obtained only for binary utilities, a more restricted form of additive utilities in which agents assign a utility of either 0 or 1 to each good, i.e. each agent simply either wants each item or not. Other work considers monotonic utility functions, where the only restriction placed on agents' utilities is that each agent's utility should not decrease if it is allocated any additional good. This is thus a much more general class of utility functions. Unfortunately, as we will see in Section 5.2.7, the utility functions of UEs and federations in RadioWeaves are not additive and may not even be monotonic. This immediately renders many of the existing results and algorithms not applicable to our case.

One class of utility functions that could be applicable to RadioWeaves is that of  $k$ -additive utilities [22, 29]. With  $k$ -additive utilities, agents' utilities are additive for subsets of the available goods up to some size  $k$ . When  $k = 1$ , this is then equivalent to additive utilities.  $k$ -additive utilities are able to capture synergies between goods in a bundle. If  $k$  is allowed to be any integer up to the number of goods, any utility function can be represented as a  $k$ -additive function, although this does not necessarily result in a more succinct representation of agents' utilities, nor lead to more tractable allocation algorithms [29]. Nonetheless, some useful results have been proved for  $k$ -additive utilities but not general ones, and for smaller  $k$ , algorithms targeting  $k$ -additive utilities could apply to RadioWeaves, since synergies between CSPs will likely be restricted in the number of synergising CSPs.

The other two ways in which the RadioWeaves case differs from existing work is that we consider allocations to groups of agents, i.e., federations of UEs, instead of individual agents, and that we have online agents, that is, UEs may arrive to and depart from the system, and their preferences may change over time. For group allocation, there is some existing work that can be of assistance to us, which we will discuss below. However, for online agents, there is very little work that has been done [7, 9]. One exception is [57], however, their setting is quite different to ours: their agents can arrive but not depart, agents' demands are in fixed proportions, allocations are irrevocable, and the resources are homogeneous. Updating allocations of resources to federations over time is thus a new area that we need to develop solutions for in REINDEER.

The existing work on group allocation is somewhat more substantial. Group fair division of divisible goods was first introduced in [80], while the case with indivisible goods was introduced in [67]. [67] also introduced a new notion of fairness suitable for group settings, namely *democratic fairness*, in which a certain fraction of the agents in each group should be satisfied according to some more fundamental fairness notion, e.g., EF1-fairness. If the fraction of satisfied agents is  $h \in [0, 1]$ , then this is known as  $h$ -democratic fairness. Work considering group allocation problems includes [44, 61, 81, 86]. In particular, [44] and [61] consider settings in which the groups are not pre-determined, but rather agents can choose which group they belong to, or are assigned to groups by a central authority. This more closely reflects the case in RadioWeaves, where the choice of which UEs should be placed in which federation is also part of the problem to be solved. While many of the results in these studies concern the existence or impossibility of fair allocations, as well as the complexity of finding them, [81] also provides a number of practical protocols that could be useful as starting points for designing an algorithm to solve the federation allocation problems. We will discuss these further in Section 5.3.3.

One other area within fair allocation that is relevant for federation resource allocation is that of allocation with constraints. [87] surveys work in this area, and discusses constraints of various

forms, including connectivity, cardinality, geometric, separation, budget, and conflict constraints. Of these, the most relevant for RadioWeaves are connectivity and conflict constraints. In RadioWeaves, we inherently have a graph of CSPs connected by X-haul links, and for some applications, it is desirable to provide an allocation in which the CSPs serving the federation are connected in this graph, for example to facilitate coherent processing with minimal delays. For other use cases, we may wish to avoid allocating more than one CSP from a given group of CSPs to the same federation. This could be used to ensure spatial diversity in the case of positioning-based applications, where additional CSPs close to those already serving the federation would not markedly improve the positioning accuracy, while CSPs that are further away would do so, by increasing the aperture size of the serving antennas.

## 5.2 Framework Design

We will now describe our framework for federation orchestration. The overall problem of federation orchestration can be broken down into five subproblems.

1. Collection of input information from the RadioWeaves infrastructure, UEs, or other sources
2. Modelling and prediction of application-level performance metrics
3. Evaluation of resource allocation solutions
4. Performing resource allocation
5. Dynamic update of resource allocation

### 5.2.1 Collection of Input Information

In order to make a good allocation of resources and UEs to federations, input information is needed that describes the current situation and the applications' needs. A good resource allocation is one in which both application performance is good, or at least acceptable, and resources in the RadioWeaves infrastructure are used efficiently. (We will more formally define what is meant by a good allocation in Section 5.2.7.)

Relevant information that can be collected includes the applications' performance requirements, the locations and wireless channel characteristics to the UEs, information about the environment such as the location of objects and walls, and information about the infrastructure itself, such as the locations of and connections between CSPs. Some of this information is straightforward to collect and collate; for instance the topology of X-haul connections between CSPs is stable over time and should be already known to the RadioWeaves infrastructure. Other information may be easy to collect but require additional signalling and overhead; an example of this is the application performance requirements, which need to be provided by the UEs in the form of a traffic specification. The collection of CSI is also an inherent part of the operation of RadioWeaves, but since this information changes frequently (depending on the channel coherence time), it incurs a significant overhead if it is used for federation orchestration. Still other information may be difficult to collect or require sophisticated models and processing algorithms. This might include information about the environment, which could be learnt over time with the help of historical channel measurements and UE locations, coupled with machine learning algorithms.

## 5.2.2 Modelling and Prediction of Application-Level Performance Metrics

Given the performance requirements for each application, we also need a way to determine whether a candidate resource allocation will meet those requirements. This is a difficult problem in itself, since it is not straightforward to know what performance in terms of application-level metrics such as data rate (at the application layer), end-to-end latency, or packet error rate will result from a given constellation of CSPs and other resources serving the application. For this, models are needed that can give the application's performance based on a particular resource allocation with sufficient accuracy. It is also desirable to be able to predict the application's performance at a future time, especially for applications requiring high reliability. This allows for predictive orchestration of federations, for example adding CSPs to the federation that can mitigate an upcoming loss in channel quality, or reach a UE as it moves behind a shadowing object. With such predictions, loss of coverage can be avoided since problems are anticipated and solved in advance.

## 5.2.3 Evaluation of Resource Allocation Solutions

The third problem we need to address for successful federation orchestration is how to evaluate candidate allocations. This occurs in two parts, firstly, in the allocation process itself in order to allow us to decide the best allocation amongst multiple possible alternatives, and secondly, once an allocation has been implemented to know whether the applications are performing as expected. The first type of evaluation must necessarily be based on models of application performance, since we will not know the real performance until the allocation is actually activated. However, this is a critical step needed for various types of algorithms that can actually produce allocations: optimisation algorithms inherently have such an evaluation embodied in their objective functions, while many heuristic algorithms take an iterative approach in which the next step needs to be compared to the previous one. Evaluating running allocations is in some ways easier, as UEs can directly report the performance they are experiencing, although this still needs to be carefully designed to avoid excessive signalling overhead. This second type of evaluation can then be used to dynamically adjust the federations to maintain good application performance, or to trigger a re-allocation when performance drops too low.

## 5.2.4 Performing Resource Allocation

Perhaps the most central problem in federation orchestration is how to actually produce a good resource allocation. Here we can to some extent draw on existing work on fair allocation, but as discussed in Section 5.1.3, no existing algorithms directly apply to the RadioWeaves case and so work needs to be done to develop new ones. We will discuss resource allocation for federation orchestration in Section 5.3. Another more practical issue is, having decided on a particular resource allocation, how to activate it in the RadioWeaves infrastructure, that is, what protocol should be applied to notify UEs, CSPs, and other entities that a particular allocation should now be applied, and ensure that this is done in a synchronised way.

## 5.2.5 Dynamic Update of Resource Allocation

For the types of applications served by RadioWeaves, a one-time, static allocation of resources to federations is not enough. Application performance will change over time as channel conditions change and users move around, and new UEs and applications may connect to the system,

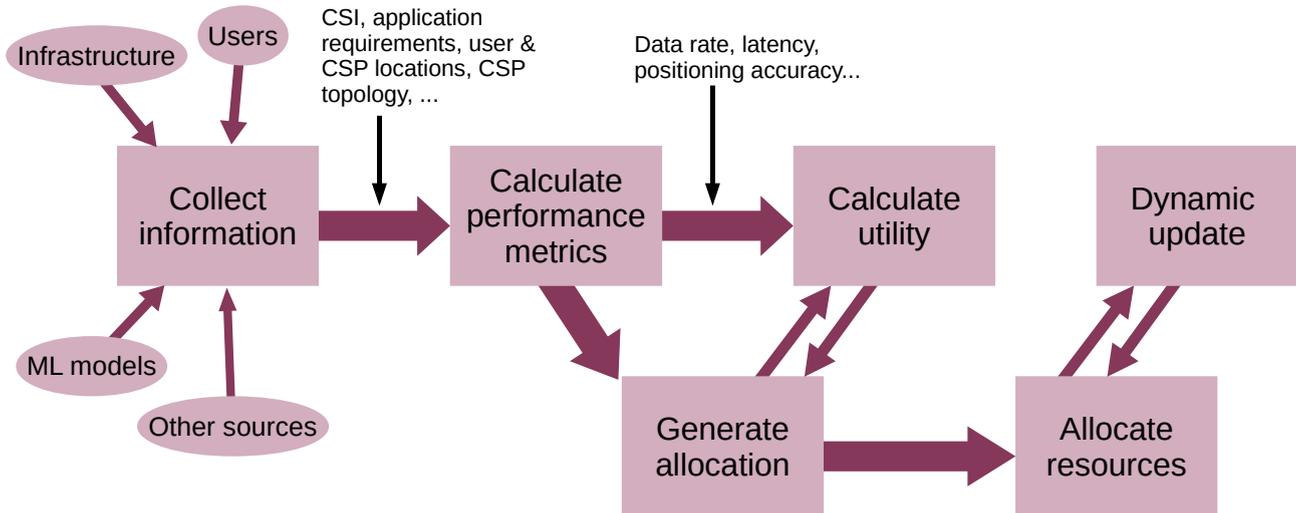


Figure 5.2: Framework for federation orchestration

or existing ones leave, or the applications' needs may also change. This leads to the problem of dynamically updating the resources assigned to each federation, and even the set of UEs served. A full allocation of resources and UEs to federations is likely to be time-consuming and computationally expensive, given the complexity results discussed in Section 5.1.3. We therefore envisage a more lightweight mechanism to adjust the set of resources allocated to the federations more frequently, for example swapping a small number of CSPs between federations as their users move. Depending on the method used for full allocation, dynamic update could use a more limited version of the same method, for example running a few steps of an iterative algorithm, or it may require its own custom solution.

## 5.2.6 Federation Orchestration Framework

Combining these parts gives an overall framework for federation orchestration, shown in Figure 5.2. First, information is collected from various sources (problem 1), and if necessary is transformed into a usable form. Next, application-level performance metrics are calculated from performance models (problem 2). The performance metrics can then be used both to determine the *utility* of a proposed allocation (problem 3), as well as to actually generate candidate allocations, either in a single pass, or in an iterative manner, depending on the algorithm used. The resources are then allocated, and performance is monitored and dynamic updates performed as needed (problem 5).

In the rest of this section, we will discuss the problem of evaluation of resource allocations (problem 3). We will introduce the concept of utility functions and define them as they will be used in RadioWeaves. We will also describe how utility functions can be combined at the UE, federation, and deployment levels to give an overall performance measure for resource allocation for federations.

## 5.2.7 Utility Functions

While federations can include other resources such as data storage and edge computing resources, in the following we will focus on CSPs as the primary resource to be allocated to federations. CSP allocation is also the part of resource allocation for federations that is the most

challenging and least covered by existing work, as discussed in Section 5.1.3, whereas allocation of storage and computing resources are well-investigated topics in computer science, including in distributed systems such as RadioWeaves.

In order to determine a good assignment of UEs and CSPs to federations, we need to define the *utility* that UEs and their applications derive from being served by particular constellations of CSPs. Each UE or set of UEs needs a defined *utility function*, specifying the value given by a set of serving CSPs. In the fair allocation parlance (see Section 5.1.3), UEs are *agents*, CSPs are *goods* (or sometimes *items* or *resources*), and constellations of CSPs are *bundles* (or *configurations*).

We define a utility function as follows.  $U(\mathcal{K}, \mathcal{S}) \in [0, 1]$  is the (normalised) utility that a set of UEs  $\mathcal{K}$  derives from being served by a set of CSPs  $\mathcal{S}$ . We use normalised utility functions in order to simplify comparison and combination of multiple utility functions later on, as will be discussed in Section 5.2.9.  $\mathcal{K}$  may be a singleton set, i.e. a single UE can also have a utility function.  $U = 0$  implies that the application run by the UE(s) is not able to function at all, typically because minimum requirements are not met, for example a minimum data rate, or the energy received from WPT is not sufficient to turn on the device(s). Meanwhile,  $U = 1$  implies that the application is performing at the maximum level possible. For some applications, it may be always possible to improve the performance, so for these applications a maximum needs to be defined where the improvement in performance is not in practice noticeable by end users. Also, for some applications, the application may be able to function to some degree even with very poor performance. In such cases,  $U = 0$  might only occur when  $\mathcal{S} = \emptyset$ , i.e., there are no CSPs in the federation at all.

As discussed in Section 5.1.3, our utility functions are likely to have some “nasty” behaviour, compared with many classical examples in fair allocation. The utility a UE or federation derives from a bundle of CSPs cannot be calculated by simply adding the utilities for each individual CSP, since combinations of CSPs may have added value (e.g. by providing spatial diversity for positioning) or lowered value (e.g. if there is a high level of interference between them). The utility may also not change with the addition or removal of individual CSPs, and can even decrease when more CSPs are added (e.g. if this increases the latency beyond the application’s maximum limit).

In general, our utilities are *subjective*, *complementary*, and *substitutable*.

**Subjective:** Different sets of UEs can have different utility values for different CSPs or constellations of CSPs. This is not unusual in fair allocation problems, but does exclude some existing algorithms, for example those developed for the restricted Santa Claus problem, which is a version of the Santa Claus problem (see Section 5.1.3) where all agents have equal utilities for every item.

**Complementary:** A group of CSPs can have synergistic value, e.g. multiple CSPs close to each other (in terms of the CSPs topology) for low latency

**Substitutable:** Having one of a group of CSPs has value, but more than one is not needed, e.g. CSPs located close to each other may give poor spatial diversity for positioning

The latter two properties imply that our utility functions are not *additive*, which excludes most of the available fair allocation algorithms. However, algorithms developed for  $k$ -additive utilities (see Section 5.1.3) could be applicable to our case.

Further, it is likely that our utility functions cannot be expressed in closed form and may in general

be difficult to compute. We therefore target resource allocation algorithms that do not rely on easily expressible utility functions. However, as a minimum, we assume that the functions can be computed by a function we will refer to as an *oracle*, similar to [87]. The oracle is a function that, when given a set of UEs  $\mathcal{K}$  and a set of CSPs  $\mathcal{S}$  can return the value of  $U(\mathcal{K}, \mathcal{S})$ , within a reasonable time. If our problem size is small enough, we could exhaustively query the oracle to create a lookup table of all possible values of  $U$ , however in practice this is unlikely to be feasible, and we will instead need algorithms that query the oracle iteratively while they are running.

### 5.2.8 Evaluation of Resource Allocation Solutions (The Oracle)

The oracle encompasses the collection of performance metrics and calculation of utility functions in the federation orchestration framework in Figure 5.2. It needs to take available information about the UEs, applications, CSPs, and environment, transform it first into performance metrics that relate to application requirements, and from there compute utility functions. Some examples of information that may be available for the oracle to use are:

- The locations of the UEs and CSPs, as well as the topology of how the CSPs are connected by X-haul links. The topology, together with the precoding and uplink combining algorithms used, has implications for the latency added by coherent processing for a group of CSPs (see [37]).
- A map of the physical environment, possibly including radio propagation characteristics, such as surface reflectivity or shadowing
- A specification of requirements for each application, and the mapping of applications to UEs
- CSI between UEs and CSPs, both instantaneous, and longer term metrics such as effective SINR. The CSI could be either measured or predicted based on channel models, and could be complete (between every pair of UE and CSP antennas) or partial. In particular, the CSI may include, or be used to derive
  - Angular channel information, e.g., set of angles of arrival to a CSP from a UE, useful for positioning.
  - Received power of a UE from a set of CSPs, useful for WPT
- Current load on or usage of CSPs and ECSPs
- Information about how CSI and data are processed, for example the precoding scheme and modulation and coding scheme (MCS) used.

The oracle needs to translate the above information into usable network performance metrics. These should be the same metrics used for specifying application requirements, and for each metric, a model is needed to derive it from the available information. Some performance metrics that could be used are:

- Data rate (peak and average)
- Latency (end-to-end delay to backhaul or ECSP)
- Expected packet error rate
- Positioning accuracy
- Received power for WPT

- Energy usage

Some of the above performance metrics are not absolute but rather depend on each other, or on other factors such as the precoding scheme and MCS used. For example, a high packet error rate can be reduced by choosing a different MCS, at the cost of a lower data rate and/or longer latency, or a higher data rate could be achieved at the cost of increased energy usage. Because of this, lower level metrics such as the SINR could also be used, leaving the mapping of SINR to higher-level performance as an extra, application-dependent step.

## 5.2.9 Computing and Combining Utility Functions

Once the oracle has derived performance metrics for a candidate resource allocation of CSPs to UEs, these need to be synthesised into a utility value between 0 and 1. There are several levels at which utility functions need to be combined. First, on a per-UE level, the performance metrics are used as input to give a utility function for that UE. Next, the utilities of all the UEs assigned to a given federation need to be combined to give an overall utility for that federation. This can be used to indicate whether the federation is performing well, and to compare resource allocations between federations, for example to determine if the allocation is fair. Finally, we can also combine the utilities of the federations into one overall utility to give a performance metric for the resource allocation and performance across the whole deployment. In fair allocation, this is often referred to as *social welfare*: how good the resource allocation is on a system (or society) level. There are different ways of combining utility functions at each level. Some proposed approaches are give below.

### 5.2.9.1 UE Level

At the UE level, we take our performance metrics and synthesise them into one utility value, between 0 and 1, for the given UE. First, we check if all necessary minimum requirements for the application have been fulfilled. If any are not, we set the UE's overall utility to 0. For example, for an application with strict real-time limits on latency, if the latency exceeds the given limit, then the overall utility is 0, regardless of the values of the other performance metrics.

Mathematically, we can express this as follows. Suppose we have a set of performance metrics  $\mathcal{P}$ , and let  $p_k^i$  be the value of the  $i$ -th performance metric for UE  $k \in \mathcal{K}$ . For metrics where a lower value is more desirable,  $p_k^i$  should be expressed as the negative of the raw value. Then, suppose we have some set  $\mathcal{P}_k$  of *critical metrics*, that is, those that must exceed a certain value, say  $P_k^i$ , for UE  $k$ 's application to function at all. Then, we have

$$U(\{k\}, \mathcal{S}_{f(k)}) = \max(\min_{i \in \mathcal{P}_k} (p_k^i - P_k^i), 0) \quad (5.2)$$

where  $U$  is the utility function,  $f(k)$  is the federation to which  $k$  has been assigned and  $\mathcal{S}_f$  is the set of CSPs allocated to federation  $f$ .

Once all minimum requirements are fulfilled, we can consider the case where an application's performance does not depend further on the performance metrics, but rather is binary, that is, the application does not work at all if the minimum requirements are not fulfilled, but if they are, it works at a consistent level regardless of their values. For such on-off applications, the utility function should be equal to 1 once the minimum requirements are met.

$$U(\{k\}, \mathcal{S}_{f(k)}) = \begin{cases} 1 & : p_k^i \geq P_k^i, \forall p_k^i \in \mathcal{P}_k \\ 0 & : \text{otherwise} \end{cases} \quad (5.3)$$

In the remaining cases, the application's performance varies with the values of the performance metrics. How the application performance depends on the metrics in practice is highly application-dependent. One approach to allow some flexibility for an application to express its particular utility is to use a weighted sum, with application-dependent weights  $\mathcal{W}_k = \{w_k^1, w_k^2, \dots, w_k^P\}$ , where  $P = |\mathcal{P}|$  is the total number of performance metrics, for each UE. To ensure the resulting utility will lie between 0 and 1, and can cover this whole interval, we need to have  $\sum_{i \in \mathcal{P}} w_k^i = 1$ . Similarly, the performance metrics need to be normalised so that they lie between 0 and 1, with 1 representing the best possible performance of the application, and 0 representing the application not functioning (at least with reference to that metric). This can be done by for example choosing minimum and maximum values, at which the application's performance will be very poor or very good, respectively, and interpolating between them. For metrics that are reversed (larger values are worse), the interpolation can go in the opposite direction (the higher value is mapped to 0 and the lower mapped to 1).

Then, the UE's utility is given by

$$U(\{k\}, \mathcal{S}_{f(k)}) = \sum_{i \in \mathcal{P}} w_k^i p_i \quad (5.4)$$

This gives a linear function for the utility, which to some extent limits the relationships that can be expressed between the application's performance and the performance metrics, but has the advantage of being simple, both to provide the information from the application, and to compute. In its traffic specification, the application only needs to provide a set of weights, one for each metric, and maximum and minimum values for each metric to perform the interpolation.

For a more general approach, the application could instead provide a function,  $u_k : \mathbb{R}^P \rightarrow [0, 1]$ , that maps the metric values to a utility, like so

$$U(\{k\}, \mathcal{S}_{f(k)}) = u_k(p_k^1, p_k^2, \dots, p_k^P) \quad (5.5)$$

however this would increase the signalling required as well as the complexity of calculating UE utilities, and it is not clear that such a general function would provide for better resource allocations than using a distribution of weights across the performance metrics as proposed above.

### 5.2.9.2 Federation and Deployment Levels

How UEs' utility values should be combined to obtain a utility for the whole federation is once again application-dependent, however there are a variety of aggregation functions that can be applied. Taking the mean of the UEs' utilities gives a good overall measure of the performance, but could result in very uneven performance between UEs being accepted, i.e. some UEs have very high utilities while others are low or even 0. If it is important for the application that all UEs have good performance, instead the minimum across the UEs could be used, or if the requirement is not as stringent, a more relaxed measure such as the tenth percentile or similar. Conversely, it is conceivable that some applications may only need one or a few well-functioning UEs, in which case a maximum or variants such as the 90-th percentile can be used. However, this is less likely to occur in practice for the use cases we consider.

Computing the federation-level utility is further complicated as some performance metrics vary with the number of UEs in the federation. This is the case for the achievable data rate and for the WPT received power. In these cases, when calculating the overall federation utility, individual UE utilities must be calculated using the performance metric values that the UE can obtain in the federation, not for each UE in isolation.

To assess the performance of the entire deployment, the main consideration is whether to prioritise overall utility, or fairness. If overall utility is more important, we can take a utilitarian social welfare function such as the sum or average of the federation utilities. If fairness is important, we can use an egalitarian measure of social welfare such as the minimum utility across the federations. We may also want to implement prioritisation between the different applications, so that critical applications are guaranteed some minimum utility, or are weighted more heavily in calculating the utility.

## 5.3 Federation Orchestration

Now that we have defined how to evaluate resource allocations, we turn to the problem of actually generating them. Unlike most previously considered fair allocation problems, we also have an extra aspect to consider, namely which agents (UEs) should be grouped together into federations. In most settings, as discussed in Section 5.1.3, agents act independently, as individuals, although some work has considered groups of agents.

We will first give a formulation of the overall problem, however given the difficulty of solving this problem directly, we will then focus on a two-stage approach, in which we first perform UE grouping and then resource allocation. We will give an overview of possible methods for solving each of these problems. Finally, we will discuss dynamic update of federations.

### 5.3.1 Federation Assignment: Problem Formulation

Given a set of UEs and applications running on them, as well as a RadioWeaves deployment, we can formulate the federation resource allocation problem as follows.

$$\max U(\mathcal{F}) \quad (5.6a)$$

$$(1 - z_f) + U(\mathcal{K}(f), \mathcal{S}(f)) > 0, \quad f \in \mathcal{F} \quad (5.6b)$$

$$\frac{1}{|\mathcal{K}|} \sum_{k \in \mathcal{K}} x_k^f \leq z_f, \quad f \in \mathcal{F} \quad (5.6c)$$

$$z_f \leq x_k^f, \quad k \in \mathcal{K}, f \in \mathcal{F} \quad (5.6d)$$

$$\sum_{f \in \mathcal{F}} x_k^f = 1, \quad k \in \mathcal{K} \quad (5.6e)$$

$$\sum_{f \in \mathcal{F}} y_s^f \leq 1, \quad s \in \mathcal{S} \quad (5.6f)$$

$$x_k^f \in \{0, 1\}, \quad k \in \mathcal{K}, f \in \mathcal{F} \quad (5.6g)$$

$$y_s^f \in \{0, 1\}, \quad s \in \mathcal{S}, f \in \mathcal{F} \quad (5.6h)$$

$$z_f \in \{0, 1\}, \quad f \in \mathcal{F} \quad (5.6i)$$

Here,  $\mathcal{F}$  is the set of federations to which UEs and CSPs can be assigned.  $|\mathcal{F}|$  should be sufficiently large that there will be enough federations to accommodate all UEs and CSPs, and in general not all federations  $f \in \mathcal{F}$  need to actually be used. A simple upper bound for  $|\mathcal{F}|$  is the number of UEs, since we will never need more federations than would be enough to place each UE in its own federation, but in practice  $|\mathcal{F}|$  can be much smaller than this.

$\mathcal{K}$  is the set of UEs, and  $\mathcal{S}$  is the set of CSPs.  $x_k^f$  and  $y_s^f$  are binary decision variables, indicating whether a UE  $k \in \mathcal{K}$  or a CSP  $s \in \mathcal{S}$  is assigned to a federation  $f \in \mathcal{F}$ , respectively, with 0 indicating not assigned and 1 indicating assigned.  $\mathcal{K}(f)$  is the set of UEs assigned to federation  $f \in \mathcal{F}$ , that is, those users for whom  $x_k^f = 1$ . Similarly,  $\mathcal{S}(f)$  is the set of CSPs assigned to  $f$ : those for whom  $y_s^f = 1$ .  $z_f$  is a binary decision variable indicating whether federation  $f \in \mathcal{F}$  is used in the solution, i.e. has at least one UE assigned to it. The objective function,  $U(\mathcal{F})$  gives the combined utility of the entire deployment, across all federations.

The first constraint, (5.6b), ensures that all federations should have at least some positive utility, i.e. that no federation will starve. The leftmost term cancels this constraint for federations that are not used. Constraints (5.6c) and (5.6d) provide lower and upper bounds on  $z_f$ , respectively, ensuring it correctly indicates whether federation  $f \in \mathcal{F}$  has had any UE assigned to it. Constraint (5.6e) makes sure that every UE is assigned to a federation, and no UE is assigned to more than one federation. Constraint (5.6f) prevents a CSP from being assigned to more than one federation, but we allow CSP to remain unassigned if they are not needed to maximise the objective. The last three constraints give the domains for the decision variables  $x_k^f$  and  $y_s^f$ .

There are two barriers to directly solving formulation (5.6). The first is the nonlinearity of the utility functions, which renders formulation (5.6) as not a proper integer programming problem, and the second is the difficulty of directly expressing the utility functions. Transforming the formulation into the configuration form (see Section 5.1.3 and [20]), and expressing the utilities as  $k$ -additive functions, could potentially solve these issues, but has its own pitfalls. In this case, we would instead have decision variables indicating the assignment of all possible bundles (sets of CSPs) to federations. The problem could then be approached by taking the relaxation, i.e. the configuration LP, and iteratively using column generation on its dual to generate bundles to assign to federations that progress towards the optimal solution of the configuration LP. The integer programming problem would then be solved for the generated bundles using for example branch-and-bound.

However, as discussed in [20], the configuration LP typically has a large integrality gap for arbitrary item utility values. Since we do not have well-behaved, additive utilities, it is therefore likely that such an approach will yield solutions far from the optimal of the integer problem taken across all possible bundles. It is also not straightforward to generate columns (i.e. bundles of CSPs) that would advance the optimisation, given the high level of synergy between the contributions of CSPs to UEs' utility functions.

Moreover, such methods typically result in long solution times, with exponential growth in the solution time as the problem size increases. Given that we aim to address use cases with potentially large numbers of UEs, in dense deployments with many CSPs, heuristic and approximation methods seem more promising for RadioWeaves. We further propose to simplify the problem by breaking it down into two stages: grouping of UEs into federations, and allocation of resources, primarily CSPs, to the resulting federations.

### 5.3.2 UE Grouping

The first stage in our proposed approach is to group the UEs into federations. The aim here is to create UE groups that share similar application requirements, and where the groups are of a reasonable size to be handled by a single federation. One way to perform user grouping is to simply use the use case categorisation done in [40], where we have four use case clusters, grouped by their key performance indicators (KPIs). This would require each user to indicate in advance which use-case it belongs to. However, a disadvantage of this approach is that the UE

grouping is static and inflexible. In the future, there may arise new use cases, or the categorisation may change. Nonetheless, this method is appealing because of its simplicity.

A more dynamic approach is to group the users based on their self-reported utilities derived from potential CSP allocations. This could be done using the following procedure.

1. Generate a set of sample CSP constellations. This can be done purely randomly, but a more tailored method would be to generate constellations that map well to the requirements of the expected applications. For example, constellations of CSPs close together would be likely to result in high utility for low latency applications, while constellations of CSPs with a high degree of spatial diversity would be suitable for positioning-focused applications. Generating a variety of different constellation types would then ensure there would be some constellations with high utility for all applications.
2. Calculate the individual utility for each UE, if it were served by each generated constellation.
3. Cluster UEs based on their utility values for the constellations.
4. If needed, divide large clusters into multiple clusters (for example to avoid having too many UEs in the same federation).
5. Each UE cluster then becomes a federation.

For step 3, in which UEs are clustered, there are many existing algorithms available in the literature. One that could be applicable here is  $K$ -means clustering [64, 65]. We already have a  $K$ -dimensional space, where  $K$  is the number of constellations generated, and users get a score (utility) from 0 to 1 in each dimension. Here, both the number of constellations and number of clusters would need to be tuned. Constellations could be generated iteratively, until the results are satisfactory. One suitable method for determining the number of clusters is the “jump” method [85], in which clustering is done for an increasing number of clusters, and a distortion curve generated. Large jumps in distortion indicate reasonable choices for the number of clusters. Alternatively, the number of federations may be decided by the available resources and topology of the infrastructure, for example creating one federation for each ECSP.

### 5.3.3 Allocation of CSPs to Federations

Once we have created federations and assigned UEs to them, we also need a method to allocate resources to the federations. As explained in Section 5.2.7, we will focus on CSPs as the resources to be allocated, since these pose a much greater challenge than the other resource types present in RadioWeaves. We further focus on heuristic and approximation methods as discussed in Section 5.3.1.

A typical heuristic approach to such problems is to start with some minimal assignment, e.g. each federation having no or only one CSP assigned to it, and then grow the federations iteratively based on the utility gained from adding each new CSP. However, for non-additive utility functions, this approach will not work, as utility does not necessarily increase with the addition of a new CSP. Multiple CSPs may need to be added to see an increase in utility, and adding CSPs may even reduce the utility, for example if the latency becomes too high.

As discussed in Section 5.1.3, many of the methods in the literature only apply to additive utilities, but there are nonetheless some methods that could be applicable to RadioWeaves, either directly or with modifications to adapt them to non-additive utilities and allocation to groups. In [81], a number of protocols are provided for fair allocation of indivisible goods to groups of agents, with

the aim of attaining  $h$ -democratic fairness guarantees. The simplest one, called Round Robin with Weighted Approval Voting (RWAV), works by each group choosing an item to add to its allocation following a round robin turn order. Each group chooses its next item by taking a weighted function over its agents' preferences, depending on their valuations of the remaining goods and the notion of fairness applied. For our case, this second part is actually easier, as we have a well-defined notion of group (federation) utility, depending on the application's specific requirements, and we do not necessarily seek to attain fairness but rather that all federations are at least satisfied (attain some minimal acceptable utility). Instead, a complication arises for us in using a round robin approach in that it may not be clear which item is most beneficial for a federation to pick next in cases where adding a single CSP is insufficient to improve the federation's utility.

Another protocol for group allocation of items given in [81] works as follows. The items are arranged on a line and are processed from left to right. Starting with an empty block, one item is added to the block at each step, then all groups are queried as to whether they find the current block acceptable. In [81], whether or not a group finds a block acceptable stems from the notion of democratic fairness applied, but in our case it could instead be whether the federation's minimum utility requirement is satisfied. Once at least one group finds the block acceptable, it is allocated to one such group, and then the algorithm repeats with the remaining items and groups.

A potential pitfall with this algorithm in our case is that, since we do not have additive or even monotonic utilities, adding items to the current block may never result in an acceptable block. Even in cases where it does, the allocations produced may be highly inefficient, as arbitrarily arranging CSPs in a line is likely to result in the block including many "irrelevant" CSPs for the federation it is eventually assigned to. This could even result in an allocation that, while fair, may not satisfy most or even any of the federations. One way to solve this problem could be to allow a federation to remove some CSPs from a block before accepting it. The removed CSPs would then be returned to the common pool (replaced on the line) to be allocated to other federations. This however adds complexity as each federation would need to evaluate the utility of not only the current block, but also subsets of it. This step could however be made more efficient by taking some shortcuts, using knowledge of the application's requirements in order to make intelligent guesses about which subsets will have the highest utility for the federation. For instance, a federation with a latency constraint could consider only subsets of CSPs that are close to one another and will thus have some maximum delay for coherent processing.

In [21], an algorithm is given for fair allocation of indivisible goods arranged on a tree, with the constraint that the allocation to each agent must be connected. First, the tree is rooted at an arbitrary vertex, then each agent can diminish the tree if a subtree would be enough to satisfy it (for maxmin share fairness, in their case). The last agent to diminish is allocated the subtree, and then the algorithm continues with the remaining tree and agents. In practice for RadioWeaves this would be similar to the algorithm above from [81], since as already explained we would need to add a similar diminishing there as well, however the tree version could be more directly applicable when the CSP topology already forms a tree, and may result in more natural and efficient allocations without as much extra processing to compute subsets of a proposed allocation. However, connected allocations are not suitable for some of our use cases, for example applications with positioning requirements that need a high degree of spatial diversity. As such, this algorithm would only provide a partial solution to compute allocations for some application types.

Another promising approach is the iterative algorithm given in [11], where the iteration step consists of over-allocating resources to the newly included agent, that is, the new agent is given more resources than are needed to satisfy it. This can result in conflicts with existing allocations,

which are then resolved. For our case, we could take a similar approach, creating too-large, conflicting allocations of CSPs in order to achieve satisfaction for each federation, and then resolving conflicts by removing or changing some of the CSPs allocated to some or all federations. With this method, care must be taken to avoid allocating a bundle of CSPs that have a too-large topological distance between them to applications requiring low latency, since large topological distances imply delays in collating CSI for coherent processing. If dynamic UE grouping is used (Section 5.3.2), the constellations generated for this could be used as initial seed allocations.

A further method that could, with some further development, be applied to RadioWeaves is that in [15]. Here, agents' have lexicographical preferences, that is there is a strict ordering of preferences for each resource expressed by each agent, although absolute utility values are not needed. A graph is then constructed showing which resources each agent is willing to swap its current resources for, given some existing allocation. By finding cycles in such a graph, a new allocation can be found that improves agents' utility. This can then be used as an iteration step in an overall allocation algorithm.

For RadioWeaves, this method would need to be extended to allow swaps of multiple CSPs, necessitating a larger graph where each vertex represents a set of related CSPs. Nonetheless, this could still be feasible if the size of sets of CSPs to be swapped can be limited. Further work is needed to determine the sizes of sets of CSPs displaying synergistic utilities in practice. A further problem remains in finding an initial allocation to start the algorithm. However, a benefit of this approach is that it also lends itself well to dynamic updating of allocations, not only to finding a full allocation.

### 5.3.4 Dynamic Update of Federations

Once an initial allocation of UEs and CSPs to federations has been performed, the federations need to be able to be changed as users move, new users arrive or existing users leave, or application needs change. This should be a more lightweight process than redoing the entire allocation from scratch. Dynamic updates can occur either periodically, or be triggered by some condition, such as a federation's utility dropping below some acceptable threshold. The dynamic update algorithm can be used not only in the actual event of a federation's utility dropping below the threshold, but also when this is predicted to happen, for example if a user's predicted mobility will cause it to have a lower channel quality and thus lower utility. Then the federations can adjust in advance, so that the lower utility is never actually experienced by UEs.

In terms of methods for performing dynamic updates, the lack of previous research on fair allocation with online agents (see Section 5.1.3) means there is little existing work to draw on. Moreover, it is proved in [29] that, for  $k$ -additive utility functions, a deal between agents involving all resources may be required in order to achieve maximum utilitarian social welfare, from a given starting allocation. While it is not clear whether this also applies to other social welfare functions, it suggests that a full re-allocation may be required to achieve optimal performance. We should therefore take an approach where a dynamic update is attempted, with a full re-allocation as a fallback if a sufficient improvement in performance cannot be found with more limited re-allocations.

One useful algorithm for dynamic updates could be that in [15], as discussed in Section 5.3.3. A simpler approach is to attempt deals between only pairs of agents, for which there are many more existing algorithms [7, 9], however with no guarantees of convergence. Guaranteed convergence is however not required if full re-allocation is available as a fallback mechanism in case dynamic update fails.

## 5.4 Conclusion

This chapter discussed the resource allocation in registered mode operation, primarily in terms of spatial resource allocation, that is, associating CSPs and other resources with the groups of UEs they will serve. The results from the other work packages will be included in the presented framework to extend the utility functions. It also provides a way to evaluate different design choices related to fairness and application performance. These will be later used in work package 5 to assess the employed algorithms in a real environment.

Next to resource allocations, devices need to access the network which is discussed in Chapter 3.

## Chapter 6

# Grant-Based and Grant-Free Access Techniques

### 6.1 Introduction

As clarified in Chapter 1, prior to grant-free and grant-based access methods, initial access needs to be performed to retrieve necessary information of the network, *i.e.*, to go from de-registered to registered mode. In the remainder of this chapter, we assume that the devices are in registered mode. This can be done through the initial access scheme as described above or this part can be skipped by configuring *a-priori* both the UEs and the network. The latter is commonly done for (unlicensed) low-power wide-area networks (LPWANs) to mitigate, *e.g.*, exhaustive cell searches or synchronisation procedures thereby reducing the energy consumption and simplifying the protocol design.

In a grant-free scheme, devices use preassigned pilot sequences or preambles. It allows both the pilot and data to be sent in a single step. The disadvantage of this approach is that it is not possible to utilise orthogonal pilots because the number of devices is much larger than the preamble length. Hence, the additional challenge in grant-free random access is device activity detection, which is the focus of the second part of this chapter.

The chapter is further outlined as follows. First, the grant-based access techniques are explored for RadioWeaves with an emphasis on URLLC. Section 6.3 defines the metrics used by Sections 6.4, 6.5 and 6.6 to evaluate the performance of the proposed grant-free access schemes.

### 6.2 Grant-Based Access Techniques For URLLC Applications

The design of a URLLC system poses stringent requirements on both latency and reliability. We may consider it as a constrained optimisation problem, which is to maximise a chosen reliability metric subject to a maximum latency. Alternatively, we can minimise the latency subject to a minimum quality-of-service (QoS) requirement. During the random access phase in grant-based systems, a significant delay is caused by the four-step handshaking procedure between the users and the CSPs, whereas in the payload transmission phase, the latency occurs due to a multitude of factors such as packet encoding and transmission, receive processing, and packet decoding. We can expect failures in both these phases due to factors such as preamble collision, poor channel conditions, severe interference, etc. We can improve the reliability by reducing the events of

these failures. For example, by using efficient preamble allocation and selection, adaptive MCS, alternative waveforms, dynamic scheduling of CSPs and interfering users, channel quality prediction, and other pro-active measures. Further, we can devise efficient retransmission schemes for the failed preamble/packet such as hybrid automatic repeat request (HARQ). A main challenge for URLLC is the procedure as to how to apply these schemes to improve the reliability while satisfying the constraints on latency and radio resources. In this context, we summarise the fundamental limits and some recent advances in these related topics.

### 6.2.1 Limits – Channel Coding and Decoding Duration

The encoding time for 5G low-density parity-check (LDPC) codes is under  $1\ \mu\text{s}$  [76, 92], and the bottleneck is the decoding time. In [91], the authors investigate the 5G LDPC decoding capabilities of graphics processing units (GPUs). They use many GPU cores to decode one LDPC codeword for low-latency communications. They also use GPU cores to work on different codewords simultaneously for high throughput applications. They reach the best case decoding latency  $87\ \mu\text{s}$  with five iterations by using Titan RTX and compare their latency results with other works in the literature in [91, Table 4].

Polar codes are selected for the control channel coding in 3GPP standard [2]. In [45, 82], it is shown that polar codes decoding latency is lower than LDPC code decoding latency for specific scenarios.

### 6.2.2 Scheduling and Proactive Measures For URLLC and enhanced Mobile Broadband (eMBB)

In this section, we give answers for the following two items:

- Scheduling of ‘interfering’ users, to improve received signal quality of devices in ‘bad’ locations
- Learning channel quality and take proactive measures when approaching ‘bad’ locations

We summarise some solutions from the literature that could be used in this project for URLLC and eMBB traffic types. We first present the solutions for URLLC and eMBB. After that, the interference management techniques are given for the URLLC traffic type. In addition, we present proactive measures based on channel quality to achieve reliability and latency requirements.

#### 6.2.2.1 URLLC and eMBB Solutions

It is challenging to satisfy both requirements of URLLC and eMBB traffic types. In [10], the authors use a mini slot structure for URLLC data and send the data by superposition or puncturing. They propose a joint scheduling algorithm for URLLC and eMBB. They aim to maximize utility for eMBB traffic while satisfying the string requirements of URLLC traffic. There are also other papers that investigate the joint optimization of URLLC and eMBB users [38, 39, 56]. In [38], authors propose an artificial intelligence enabled approach that uses a reinforcement learning-based algorithm to jointly optimise URLLC and eMBB users’ requirements.

In [56], the authors propose a novel resource allocation method to minimise URLLC users’ latency and enhance the throughput for the eMBB users. Their algorithm considers the following parameters: channel quality, payload size, traffic type, hybrid automatic repeat request (HARQ),

control channel overhead, and latency requirements. They aim to maximise the number of served URLLC users while satisfying their reliability and latency requirements. For example, the algorithm selects the required number of physical resource blocks for URLLC packets' payload based on the channel quality indicator (CQI).

In [39], authors propose a null-space based preemptive scheduler for joint eMBB and URLLC traffics. They maximize the eMBB ergodic capacity while satisfying the URLLC users' demands. The proposed scheduler decreases the amount of interference for a URLLC user by de-orienting its decoding vector into almost the nullspace of an eMBB user that uses the same resources with the URLLC user.

### 6.2.2.2 URLLC Solutions

The URLLC traffic demand for both scheduled and non-scheduled devices need to be satisfied for 5G and 6G applications [13]. The critical device-to-device communications and critical data are examples of non-scheduled URLLC traffic. In [13], the authors propose a distributed and risk-aware machine learning algorithm for radio resource management (RRM) for both scheduled and non-scheduled devices in the network. The intelligent RRM is fed by measurements in the radio environment such as users, mobility maps, interference, and channel quality measurements. The proposed solution uses both orthogonal and non-orthogonal resource slicing to satisfy the delay and reliability demands of the URLLC traffic type. In their model, base stations are also coordinated and share information such as the handover of a device that demands URLLC traffic type. They also classify users based on their latency and reliability requirements.

In [88], the authors consider a multi-channel method to achieve low latency communication in the unlicensed band. They show that the duration between two successful consecutive frames decreases by the increasing number of unlicensed channels used for the multi-channel method.

### 6.2.2.3 Interference Management

In [93], the authors propose a method called suppressing alignment to decrease peak-to-average power ratio (PAPR) and out-of-band interference that affects adjacent channels. The proposed method uses channel state information to align the suppressing signal with the cyclic prefix (CP) duration of the OFDM signal at the receiver.

In [35], the authors propose a non-orthogonal scheme to reduce interference between two users. The first latency-critical user uses OFDM and grant-free (GF) access. The second latency-tolerant user uses OFDM with index modulation (OFDM-IM) and grant-based (GB) access. OFDM-IM uses both classical modulation scheme and subcarrier indices to transmit information. In OFDM-IM, there are some empty subcarriers that could help to control the interference. To control the interference that affects the first (latency-critical) user, they reduce the number of active subcarriers in the second user transmission that uses OFDM-IM. As a result, frequency flat interference becomes frequency-selective interference as seen in Fig. 4 in [35]. This method improves the bit error rate performance of the first user by controlling the interference.

### 6.2.2.4 Proactive Measures Based on Channel Quality

It is possible to increase the system performance in terms of reliability and latency by utilizing channel state information and applying proactive measures. In [62], the authors consider a smart factory environment with several sensors that send data periodically to an access point. They

assume that there is channel time correlation, and propose a method for a dynamic pilot allocation scheme to update channel state information. They also propose a graph-based resource allocation algorithm that considers latency and reliability requirements and uses channel state information. The algorithm prioritizes the sensors that are close to the access point. The authors also propose a fairer version of their algorithm.

In [70], the authors propose a method for the OFDM-IM system to increase the BER performance by utilizing channel information. In the OFDM-IM system, there are some empty subcarriers that carry information in the space domain. They assume that the channel is perfectly known both in the receiver and transmitter, and they propose a method to align empty subcarriers with deep fading sub-channels. They use a circular shift block both in the transmitter and the receiver to align empty subcarriers. As a result, there is a BER performance increase in the system.

### 6.2.3 Retransmission Schemes for URLLC

Retransmission techniques, especially HARQ, were widely investigated in wireless systems to improve the reliability of communication, at the cost of additional radio resources and delay. In URLLC, however, the additional delay induced by the packet retransmissions in HARQ could become a key bottleneck. To satisfy the contradictory objectives in URLLC, some extensions to the retransmission schemes are considered. A recent survey paper on HARQ is [6]. An earlier paper [74] summarised some potential enhancements.<sup>1</sup>

#### 6.2.3.1 Adaptive HARQ with Enriched Feedback

In conventional HARQ schemes, the receiver uses a binary feedback message (ACK/NACK) to inform the transmitter of the success/failure of a packet transmission. Once a NACK is received, or an ACK has not been received within a pre-determined time window, the retransmission will be initiated. To improve the retransmission process, some information, including channel state information (CSI), channel quality indicator (CQI), accumulated mutual information (AMI), and decoder state information (DSI), can be combined with the NACK message. Based on the received information, MCS, packet length, number of retransmissions (e.g., in K-repetition), power control, and resource allocation can be updated adaptively.

#### 6.2.3.2 Early/Proactive HARQ

As indicated in [74], approximately 60 percent of the processing time are usually spent for channel decoding in LTE. Therefore, it is beneficial to consider some methods to reduce the decoding time, especially when the decoding is unsuccessful (since the computational resources spent on the decoding process is mostly wasted). Early stop (ES) and deferred iterations (DI) can be considered in this case. The former terminates the decoding process once some stopping criteria is satisfied, and the latter postpones the decoding iterations until it is convinced that sufficient information is available for successful decoding, otherwise the retransmission request is sent immediately.

#### 6.2.3.3 Blind/Autonomous HARQ

When some relevant information is available at the transmitter, e.g., CSI, it may retransmit immediately without waiting for any feedback under scenarios such as deep fade. More simply, if

---

<sup>1</sup>Some relevant papers can be found in these survey papers, and they will not be cited separately here.

the transmitter repeatedly retransmits the packets without relying on any other information, the scheme reduces to K-repetition.

#### 6.2.3.4 Others

Non-orthogonal multiple access (NOMA) can be combined with HARQ (e.g., in [60, 73]) to improve the spectral efficiency. Some machine learning techniques were investigated for early feedback prediction [84]. Actually, machine learning can be applied in most of the aforementioned problems for potential performance improvement. A network coding based scheme with a sliding window to reduce the latency is proposed and compared with K-repetition in [30]. A theoretical performance characterization with discrete-time finite state Markov model can be found in [78]. By leveraging the recent results on the achievable rates of finite-length channel coding, the authors in [66] proposed a fast HARQ protocol and provided some theoretical results, including the message decoding probability, throughout, expected delay, and error probability.

### 6.2.4 Alternative Waveforms for URLLC

Some alternative waveforms, including GFDM, UFMC, filtered-OFDM, Generalized DFT-s-OFDM, and flexible DFT-s-OFDM, were summarized and compared in [89, Table 3], in terms of OOB and latency characteristics. However, waveforms can be as well an important tool to achieve ultra reliable services.

OFDM based waveforms have typically high sensitivity to any frequency mismatches, Doppler spread and phase noise. As an alternative to OFDM based waveform, it is proposed a constant envelope frequency modulated multicarrier waveform which can improve BER performance in that conditions in order to reach the most stringent reliability requirements. Additionally, thanks to the constant amplitude of this waveform and its low PAPR, devices will become more energy efficient as their power amplifiers can operate in their linear region. This energy efficiency becomes very important, especially in the use cases where simpler devices (up to class 3 devices) are used, as these require minimum energy consumption.

### 6.2.5 Learning-Based Random Access

Existing communication protocols are usually designed and optimized over simplified assumptions and models. Although this makes the analysis more tractable and leads to efficient implementation, the real-world scenarios are much more complicated, and a perfect model never exists. In terms of random access, the activities of users are usually uncoordinated and are triggered by random events. Additionally, due to the overcomplicated temporal and spatial correlation of events and the uncontrollable channel environment, it becomes exceedingly challenging to find a sufficiently expressive model to characterize the behaviors of users while maintaining its tractability.

Benefiting from the advances in deep learning, learning-based design of communication protocol has been extensively explored in recent years. The key advantage of these learning-based methods is that they do not require an explicit system model. Due to the universal function approximation capability of neural networks, the complex underlying model can be efficiently learnt from data. For random access, a straightforward application of deep learning include traffic prediction and real-time optimization of the system parameters (for example, the parameters of access class

barring and backoff) [33, 54]. Deep learning can also be integrated into physical-layer processings, for example, in [52], the authors proposed a learning-based random access framework to detect and resolve preamble collisions. This method is in contrast to the conventional random access schemes which are restricted to binary preamble detections.

Among different learning categories, reinforcement learning has shown its ability to improve communication protocols in various application scenarios. In principle, reinforcement learning is a tool to find the optimal decision policy in Markov decision processes. It has been widely investigated for partially-observable environments and Markov games, which yield the field of multi-agent reinforcement learning (MARL). Different from supervised learning which requires labeled data, reinforcement learning requires only a reward function to evaluate the decisions (actions), and the policy can be learned through the interaction with the environment. The application of MARL for random access has been investigated in [16, 18, 32, 51, 55, 96]. In [51], the authors considered pilot selection to avoid collision in random pilot selection, and their MARL-based scheme was shown to achieve a throughput within 85% of the optimum in a heavily-loaded system. The authors of [16] further considered a system with delay-constrained random traffics and proposed a model-based MARL method to accelerate training. A tiny state-space R-learning random access (TSRA) method was introduced in [32] for random access and showed superior performance compared with some existing baselines. A transmission tax-based decoupled MARL approach was introduced in [55] to dynamically adjust the transmission probabilities in a  $p$ -persistent carrier sensing multiple access system. In [96], the authors considered the application of reinforcement learning in NOMA systems. The authors of [18] considered a super-preamble structure, such that a device can send multiple preambles in an access attempt to improve the success probability, and the number of preambles to transmit is dynamically adjusted based on the status.

## 6.3 System Model and Evaluation Metrics of Grant-Free Access Techniques

### 6.3.1 System Model

Consider a RadioWeaves network with  $N$  CSPs each equipped with  $M$  antennas and serving  $K$  arbitrarily distributed single antenna users. Due to the sporadic nature of the traffic in the massive access scenario of massive machine-typed communication (mMTC), only a small fraction of the  $K$  users are active at any given time instant. We assume that each device transmits independently with an activation probability  $\epsilon \ll 1$ . Let  $a_k \in \{0, 1\}$  where  $a_k = 1$  denotes that the  $k^{\text{th}}$  device is active and  $a_k = 0$  that it is inactive and  $\Pr(a_k = 1) = \epsilon$  and  $\Pr(a_k = 0) = 1 - \epsilon$ . Let  $\mathbf{a} = (a_1, a_2, \dots, a_K)$  denote the activity of  $K$  users at any time instant. Due to the sporadic nature of mMTC traffic, the vector  $\mathbf{a}$  will be sparse. The set of active users is denoted by  $\mathcal{K}_a$  i.e.,  $\mathcal{K}_a = \{k : a_k = 1\}$ .

The channel between antenna  $m$  in CSP  $n$  to device  $k$  is given by

$$g_{nmk} = \sqrt{\beta_{mk}} h_{nmk}, \quad (6.1)$$

where  $\beta_{mk}$  is the large-scale fading coefficient between the  $n^{\text{th}}$  CSP and the user  $k$  and  $h_{nmk} \sim \mathcal{CN}(0, 1)$  is the small-scale fading coefficient.

Due to a large number of users, typically  $K \gg \tau_c$ , assigning orthogonal pilot sequences to each user is not feasible. Instead we assign non-orthogonal unique signature sequence,  $\mathbf{s}_k \in \mathbb{C}^{L \times 1}$

to each user  $k$ , where  $L \leq \tau_c$  is the signature sequence length. We assume that the signature sequences of all the users are known at the ECSP.

The signal  $\mathbf{y}_{nm} \in \mathbb{C}^{L \times 1}$  received at antenna  $m$  of CSP  $n$  is given by

$$\begin{aligned} \mathbf{y}_{nm} &= \sum_{k=1}^K a_k \sqrt{\rho_k} g_{nmk} \mathbf{s}_k + \mathbf{w}_{nm} \\ &= \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho \mathbf{g}_{nm} + \mathbf{w}_{nm}, \end{aligned} \quad (6.2)$$

where  $\mathbf{S} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_K] \in \mathbb{C}^{L \times K}$  is the collection of all signature sequences,  $\rho_k$  is the power transmitted by user  $k$ ,  $\mathbf{D}_a = \text{diag}(\mathbf{a})$ ,  $\mathbf{D}_\rho = \text{diag}(\rho_1, \rho_2, \dots, \rho_K)$ ,  $\mathbf{g}_{nm} = [g_{nm1} \ g_{nm2} \ \dots \ g_{nmK}]^T \in \mathbb{C}^{K \times 1}$  is the channel vector from all  $K$  users to the  $m^{\text{th}}$  antenna of the  $n^{\text{th}}$  CSP and  $\mathbf{w}_{nm} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I}_L)$  is the independent additive white Gaussian noise vector.

### 6.3.2 Evaluation Metrics

The performance of any detection scheme is characterised predominantly by two metrics namely, the probability of miss detection and the probability of false alarm. In the context of grant-free random access, miss detection means that, when the activity of a device goes undetected, while it is actually transmitting. Likewise, a false alarm occurs if a device is considered active by the detection algorithm whereas it is not transmitting any information. We define the probability of miss detection as the average ratio of undetected devices to the number of active devices

$$P_{\text{md}} = 1 - \mathbb{E} \left\{ \frac{|\mathcal{K}_a \cap \hat{\mathcal{K}}_a|}{|\mathcal{K}_a|} \right\},$$

where  $\hat{\mathcal{K}}_a = \{k | \hat{a}_k = 1, \forall k \in [1, K]\}$  denotes the estimated set of active devices. Note that on average  $|\mathcal{K}_a| = K_a$ . Similarly, the probability of false alarm is the ratio of inactive devices considered active to the number of inactive devices and is given by

$$P_{\text{fa}} = \mathbb{E} \left\{ \frac{|\hat{\mathcal{K}}_a \setminus \mathcal{K}_a|}{K - |\mathcal{K}_a|} \right\}.$$

Any carefully designed detection algorithm aims to decrease both these probabilities. However, any procedure that improves the detection performance invariably degrades the probability of false alarm and vice versa. A parameter  $v$  is introduced to choose a desired probability of false alarm and miss detection. A lower  $v$  yields a lower activity threshold, resulting in more devices considered active. This in turn lowers the probability of miss detection, while increasing the probability of generating a false alarm.

## 6.4 New Distributed AMP Algorithm for Activity Detection in Grant-Free Access

In this section, we develop a new algorithm for activity detection for grant-free multiple access in D-MIMO. The algorithm is a distributed version of the approximate message passing (AMP)-based on a soft combination of likelihood ratios computed independently at multiple CSP. The underpinning theoretical basis of our algorithm is a new observation that we made about the state evolution in the AMP. Specifically, with a minimum mean-square error denoiser, the state maintains a block-diagonal structure whenever the covariance matrices of the signals have such a structure. We show by numerical examples that the algorithm outperforms competing schemes from the literature. The materials in this section were originally presented in [17].

### 6.4.1 System Model and Power Allocation

We consider the uplink of a D-MIMO system, where  $N$  CSPs jointly serve  $K$  single-antenna devices. Each CSP has  $M$  receive antennas, and the total number of antennas in the system is denoted by  $M_{\text{tot}} = NM$ . Each device,  $k \in \mathcal{K}$ , is pre-allocated a pilot sequence  $\mathbf{s}_k = [s_{1k}, \dots, s_{Lk}]^T \in \mathbb{C}^L$  with unit energy, i.e.,  $\|\mathbf{s}_k\|^2 = 1$ . In each time slot, the activity of device  $k$  is modelled by a binary random variable,  $a_k \sim \text{Bernoulli}(\epsilon_k)$ .

The received signal,  $\mathbf{Y}_n \in \mathbb{C}^{L \times M}$ , at the  $n$ -th CSP can be expressed as

$$\mathbf{Y}_n = \sum_{k \in \mathcal{K}} \sqrt{L\rho_k} a_k \mathbf{s}_k \tilde{\mathbf{g}}_{nk}^T + \mathbf{W}_n, \quad (6.3)$$

where  $\rho_k \in [0, p_{\max}]$  is the transmit power of device  $k$ . The channel between CSP  $n$  and device  $k$  is modeled by  $\tilde{\mathbf{g}}_{nk} \sim \mathcal{CN}(\mathbf{0}, \tilde{\mathbf{R}}_{nk})$ , where  $\tilde{\mathbf{R}}_{nk} \in \mathbb{C}^{M \times M}$  is the spatial correlation matrix, and  $\beta_{nk} = \text{tr}(\tilde{\mathbf{R}}_{nk})/M$  can be interpreted as the large-scale fading coefficient (LSFC). The channel is assumed to be uncorrelated between different CSPs and devices. The noise matrix  $\mathbf{W}_n \in \mathbb{C}^{L \times M}$  has i.i.d.  $\mathcal{CN}(0, \sigma^2)$  elements, where  $\sigma^2$  is the noise variance.

For notational brevity, we define the effective channel  $\mathbf{g}_{nk} \triangleq \sqrt{L\rho_k} \tilde{\mathbf{g}}_{nk}$ , which has the distribution  $\mathcal{CN}(\mathbf{0}, \mathbf{R}_{nk})$ , where  $\mathbf{R}_{nk} = L\rho_k \tilde{\mathbf{R}}_{nk}$ , and  $\kappa_{nk} = L\rho_k \beta_{nk}$  can be interpreted as the received signal strength of device  $k$  at CSP  $n$ .

Denoting the pilot matrix by  $\mathbf{S} = [\mathbf{s}_1, \dots, \mathbf{s}_K]$ , the effective channel matrix by  $\mathbf{G}_n = [\mathbf{g}_{n1}, \dots, \mathbf{g}_{nK}]^T$ , and the vector of device activities by  $\mathbf{a} = [a_1, \dots, a_K]^T$ , the received signal model in (6.3) can be written as

$$\mathbf{Y}_n = \mathbf{S} \mathbf{D}_a \mathbf{G}_n + \mathbf{W}_n. \quad (6.4)$$

By combining the received signal at all CSPs, we obtain

$$\mathbf{Y} = \mathbf{S} \mathbf{D}_a \underbrace{[\mathbf{G}_1, \dots, \mathbf{G}_N]}_{\triangleq \mathbf{G}} + \underbrace{[\mathbf{W}_1, \dots, \mathbf{W}_N]}_{\triangleq \mathbf{W}}, \quad (6.5)$$

where  $\mathbf{G} = [\mathbf{g}_1, \dots, \mathbf{g}_K]^T$  and  $\mathbf{g}_n = [\mathbf{g}_{n1}^T, \dots, \mathbf{g}_{nK}^T]^T$  is the channel from device  $n$  to all CSPs. Note that by assuming uncorrelated fading across different CSPs,  $\mathbf{g}_k$  has the distribution  $\mathcal{CN}(\mathbf{0}, \mathbf{R}_K)$ , where  $\mathbf{R}_K$  is block-diagonal:  $\mathbf{R}_k = \text{bdiag}(\mathbf{R}_{1k}, \dots, \mathbf{R}_{Nk})$ .

The system model in (6.5) is an instance of the linear measurement model  $\mathbf{Y} = \mathbf{S}\mathbf{X} + \mathbf{W}$ , where the unknown signal matrix  $\mathbf{X}$  is row sparse, and each row  $\mathbf{x}_k^T = a_k \mathbf{g}_k^T$  has a Bernoulli-Gaussian distribution. Therefore, activity detection becomes a support recovery problem in compressed sensing (CS), which can be solved using the AMP algorithm.

### 6.4.1.1 Dynamic Cooperative Clustering

We assumed that each device was served by all CSPs. This configuration is not scalable in complexity and resource requirements as  $N \rightarrow \infty$ . Meanwhile, the AMP algorithm, or more generally, CS techniques, are known to work in the regime where the measurement size (pilot length) is larger than or equal to the support size (number of active devices).

To address these issues, we consider a dynamic cooperation clustering (DCC) framework, such that a device is served only by the CSPs with indices in the set  $\mathcal{K}_n^d \subset \mathcal{K}$ . Conversely, a CSP only serves a subset of devices  $\mathcal{K}_n^d = \{k \in \mathcal{K} : n \in \mathcal{N}_k^d\}$ . There are two advantages of using the DCC framework: 1) the computational complexity is reduced; 2) the effective number of active devices served by a CSP decreases.

### 6.4.1.2 Power Allocation

In D-MIMO, since the CSPs are spread out, the channel gains from a device to different CSPs vary significantly. The signal strength from a device is generally larger at CSPs that are physically close to the device than at other CSPs.

We propose a user-centric power allocation scheme that comes in a few different variations. The details are as follows:

- 1) Each device  $k$  is associated with the subset of CSPs, say  $\mathcal{N}_k^p$ , for which the large-scale fading components (LSFCs) exceed a threshold  $\beta_k^{\text{th}}$ :

$$\mathcal{N}_k^p = \{n \in \mathcal{N} : \beta_{nk} > \beta_k^{\text{th}}\}. \quad (6.6)$$

If no CSP satisfies this requirement, we associate the device to the CSP with the largest LSFC, i.e.,

$$\overline{\mathcal{N}}_k^p = \mathcal{N}_k^p \cup \{\text{argmax}_{n \in \mathcal{N}} \beta_{nk}\}. \quad (6.7)$$

- 2) For each device, a coefficient  $s_n$  is calculated. We consider the three different choices:

$$u_k = \begin{cases} 1, & \text{FullPower} \\ \max_{n \in \overline{\mathcal{N}}_k^p} \beta_{nk}, & \text{MasterCSP} \\ \frac{1}{|\overline{\mathcal{N}}_k^p|} \sum_{n \in \overline{\mathcal{N}}_k^p} \beta_{nk}, & \text{AvgCSP} \end{cases}. \quad (6.8)$$

- 3) For each device, the transmit power is set to

$$\rho_k = \min \{u_{\min}/u_k, 1\} p_{\max}, \quad (6.9)$$

where  $u_{\min} = \min_{k' : |\mathcal{N}_{k'}^p| \geq 1} u_{k'}$  is the minimum coefficient among all devices for which at least one CSP satisfies the LSFC requirement, i.e.,  $\beta_{nk} > \beta_k^{\text{th}}$ .

## 6.4.2 Activity Detection in D-MIMO

We present a novel distributed AMP (dAMP) procedure in Algorithm 1 which utilises the DCC framework and the block-diagonal structure of the covariance matrices to implement the device activity detection at different CSP in a distributed manner. We directly provide the algorithm here for readability, and interested readers can find the detailed derivations provided in Section A.1.

---

**Algorithm 1** distributed AMP (dAMP)
 

---

**Require:**  $\mathbf{S}, \{\mathbf{Y}_n\}, \{\mathbf{R}_{nk}\}$ 
**Ensure:**  $\mathbf{Z}_n^0 = \mathbf{Y}_n, \hat{\mathbf{x}}_{nk}^t = \mathbf{0}$ , and  $\Sigma_n^0 = \frac{1}{L} \mathbf{Y}_n^T \mathbf{Y}_n^*$ ,  $\forall n, \forall k$ 

```

1: for each  $n \in \mathcal{N}$ , independently do
2:   for  $t = 0, 1, \dots$  do
3:     for each  $k \in \mathcal{K}_n^d$  do
4:        $\xi_{nk}^t = (\mathbf{Z}_n^t)^T \mathbf{s}_k^* + \mathbf{x}_{nk}^t$ 
5:        $\Psi_{nk}^t = \mathbf{R}_{nk} (\mathbf{R}_{nk} + \Sigma_n^t)^{-1}$ 
6:        $\Omega_{nk}^t = (\Sigma_n^t)^{-1} - (\mathbf{R}_{nk} + \Sigma_n^t)^{-1}$ 
7:        $\rho_{nk}^t = \frac{|\mathbf{R}_{nk} + \Sigma_n^t|}{|\Sigma_n^t|} \exp(-(\xi_{nk}^t)^H \Omega_{nk}^t \xi_{nk}^t)$ 
8:        $\theta_{nk}^t = \left(1 + \frac{1-\epsilon_k}{\epsilon_k} \rho_{nk}^t\right)^{-1}$ 
9:        $\hat{\mathbf{x}}_{nk}^t = \theta_{nk}^t \Psi_{nk}^t \xi_{nk}^t$ 
10:    end for
11:     $\mathbf{U}_n^t = \frac{1}{K} \sum_{k \in \mathcal{K}_n^d} \theta_{nk}^t \Psi_{nk}^t (\mathbf{I} + (1-\theta_{nk}^t) \xi_{nk}^t (\xi_{nk}^t)^H \Omega_{nk}^t)$ 
12:     $\mathbf{Z}_n^{t+1} = \mathbf{Y}_n - \sum_{k \in \mathcal{K}_n^d} \mathbf{s}_k (\hat{\mathbf{x}}_{nk}^t)^T + \frac{K}{L} \mathbf{Z}_n^t \mathbf{U}_n^t$ 
13:     $\Sigma_n^{t+1} = \frac{1}{L} (\mathbf{Z}_n^{t+1})^T (\mathbf{Z}_n^{t+1})^*$ 
14:  end for
15: end for

```

---

A centralised AMP (cAMP) is also developed in a similar way, while the step-wise details are omitted. The key distinction in cAMP is that the denoiser for device  $n$  is designed using  $\theta_k^t = \left(1 + \frac{1-\epsilon_k}{\epsilon_k} \prod_{n \in \mathcal{N}_k^d} \rho_{nk}^t\right)^{-1}$  by combining the local LLRs from all its serving CSPs in each iteration. These algorithms can be modified for other network structures. For example, multiple neighboring CSPs can coherently process the received signals. In this respect, cAMP (fully coherent) and dAMP (noncoherent) represent two extreme cases.

#### 6.4.2.1 Complexity Analysis

The computational complexity of dAMP with correlated fading is dominated by the calculation of matrix inversions and determinants in steps 5-7 of Algorithm 1 with complexity  $O(M^3)$ . Therefore, the overall complexity is  $O(NTKM^3)$ . For the i.i.d. Rayleigh case, the complexity of matrix-vector multiplications in steps 4, 7, and 9 is  $O(M^2)$ , and the matrix multiplications in steps 12 and 13 have complexity  $O(LM^2)$ . Since we are interested in the regime where  $L \ll N$ , the overall complexity becomes  $O(NTKM^2)$ . Notice that dAMP can be distributed, and the processing per CSP has complexity  $O(TKM^2)$ . Furthermore, by using the DCC framework, we can replace  $K$  by  $\max_n |\mathcal{K}_n^d|$ .

For comparison, the covariance-based method in [42] has overall complexity  $O(TK(N_{\text{dom}}^3 + NL^2))$ , where  $N_{\text{dom}}$  is the number of dominant CSPs; for the typical case  $N_{\text{dom}} < L$ , the complexity becomes  $O(NTKL^2)$ . Note, however, that the method of [42] is developed for the i.i.d. Rayleigh case and while extensions are possible, they are likely to incur higher complexity. Since the number of antennas is typically small on a CSP, we have  $M < L$ , and our algorithms have lower complexity than that of [42].

Table 6.1: Runtime comparison in seconds.

	cAMP		dAMP		Cov. Approach
	all CSPs	DCC	all CSPs	DCC	
$L = 40$	0.69	0.28	0.38	0.21	2.33
$L = 20$	0.70	0.29	0.37	0.21	1.34

### 6.4.3 Simulations

We consider a D-MIMO system with  $N = 20$  CSPs with  $M = 3$  antennas each. A total of  $K = 400$  devices are randomly dropped in a  $2 \text{ km} \times 2 \text{ km}$  squared area with activity probability  $\epsilon_k = 0.1, \forall k$ . By using a wrap-around technique, we approximate an infinitely large network with 15 antennas and 10 active devices per square km. The pilots are random Gaussian sequences normalised to unit energy. The maximum transmit power is 23 dBm. The bandwidth is 1 MHz. The noise power spectral density is  $-169 \text{ dBm/Hz}$ . The LSFC is generated by  $-140.6 - 36.7 \log_{10}(d_k) + \Upsilon_k$  in dB, where  $d_k$  is the distance from device  $n$  to the CSP in km, and  $\Upsilon_k$  is the shadow fading effect distributed as  $\mathcal{N}(0, \sigma_{\text{sf}}^2)$ , with standard deviation  $\sigma_{\text{sf}} = 4 \text{ dB}$ . The small-scale fading is modeled by i.i.d. Rayleigh for each pair of CSP and device. The LSFC threshold for power allocation is set to satisfy  $p_{\text{max}} \beta_k^{\text{th}} = 6 \text{ dB}, \forall k$ . For the DCC framework, we connect each device to the 10 CSPs with the largest LSFC.

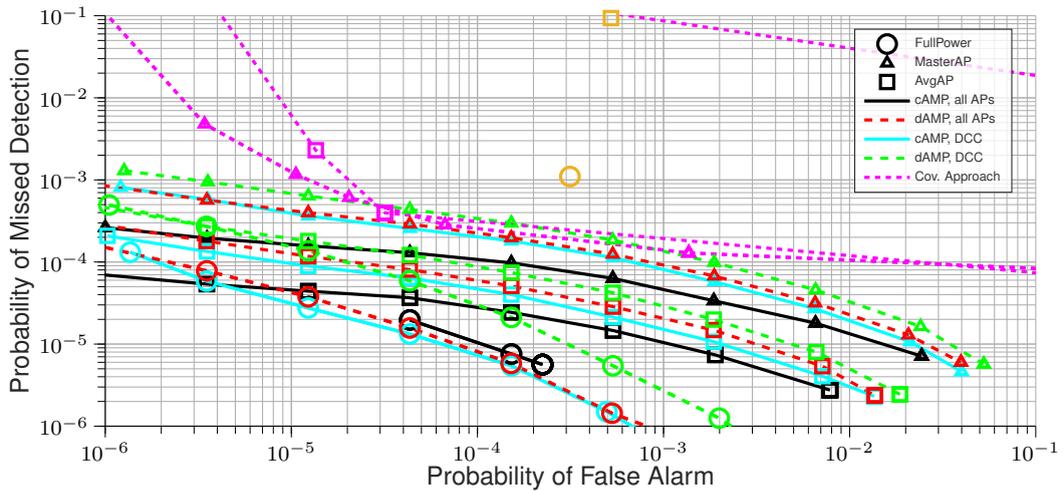
The performances of cAMP and dAMP are examined in Figure 6.1 with or without the DCC framework and with different power allocation schemes. The probability of false alarm (missed detection) was defined in Section 6.3. The covariance-based approach in [42] (with 3 dominant CSPs) and the hard-decision-and-fusion based AMP method<sup>2</sup> in [46] are used as baselines for comparison. A runtime comparison is provided in Table 6.1.<sup>3</sup>

The results for pilot length  $L = 40$  are shown in Fig. 6.1a. The following observations can be made: (i) When the pilot length is larger than or equal to the average number of active devices, AMP outperforms the covariance-based approach in almost all configurations since our AMP algorithms can coherently process received signals from more CSPs. (ii) AMP works better with full power. We hypothesise that this is because of the macro-diversity in distributed MIMO: for each device there are almost always some CSPs to which the path gain is high. Thus, although using full power usually works poorly for activity detection in co-located MIMO, it can be an option in distributed MIMO where it is difficult to obtain an explicit objective for optimising the power allocation. (iii) There is a performance gap between cAMP and dAMP due to the lack of coherent processing across different CSPs for dAMP.

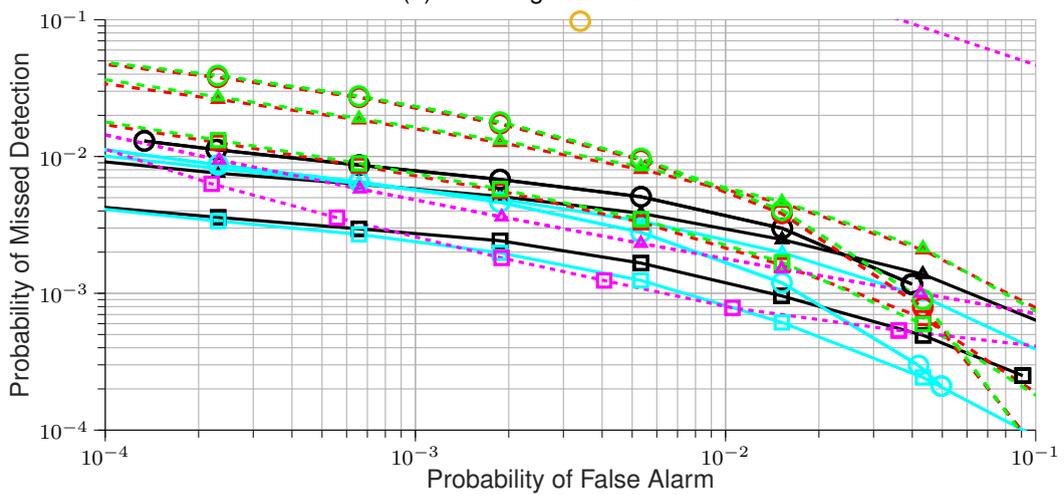
In Fig. 6.1b, the results are reproduced for pilot length  $L = 20$ , which is not a working regime for AMP in the co-located case. We observe: 1) the performance loss of AMP is more significant than for the covariance-based approach since the AMP is inherently restricted to scenarios where  $L \geq \sum a_n$ , while the covariance-based approach has a better scaling law [41]; 2) *AvgAP* becomes a better power allocation scheme, potentially due to its better control of interference power under increased pilot contamination; 3) DCC performs better than using all CSPs, since a CSP can

<sup>2</sup>Since [46] provided neither theoretical results nor algorithm details for the multi-antenna CSP case, we use the expressions of probabilities of missed detection and false alarm in [63] to perform the decision fusion. Notice that this method uses a minimum-probability-of-error criterion and does not produce receiver operating characteristic (ROC) curves.

<sup>3</sup>The simulations were performed on an Intel Xeon Gold 6130 Processor.



(a) Pilot length  $L = 40$



(b) Pilot length  $L = 20$

Figure 6.1: Performance of the cAMP, dAMP, and baseline algorithms with different power allocation schemes.

ignore the devices with bad channel conditions; this is particularly helpful when  $L$  is small relative to the average number of active users.

## 6.5 Exploiting Partial Channel State Information in Grant-Free Access

In this section, we will exploit the deterministic nature of static devices, e.g., in-place fixed Internet of Things (IoT) devices. In [26], the long-term behaviour of the channel for a large array is investigated. There, they conclude that the channel conditions are less time-variant than assumed in theoretical works, for fixed IoT devices. Moreover, even after the channel state had changed due to movement, e.g., a car passing by, the channel quickly recovers back to its previous state. They observed that 90% of the measured channels had a correlation of over 0.9 over a period of 8 h, with respect to the first channel measurement.

In Section 6.5.1, we extend the model introduced in Section 6.3 and present an iterative maximum likelihood estimator to detect the active devices for the co-located case. Following, Section 6.5.2 elaborates on the extension to the cell-free or RadioWeaves case.

### 6.5.1 Partial CSI in Co-Located Massive MIMO

#### 6.5.1.1 System Model

To access the network, each active device  $k \in \mathcal{K}_a$  sends a unique, non-orthogonal preamble of length  $T$ , known to the network. The pilot symbol of the preamble sent by device  $k$  at pilot symbol  $t$  is denoted by  $s_{k,t}$ . The channel vector between the receive antenna  $m$  and user  $k$  is denoted by  $h_{k,m} \in \mathbb{C}$ , and is considered fixed over the preamble duration. The received symbol at the  $m$ -th antenna at time  $t$  is

$$y_{t,m} = \sum_{k=0}^{K-1} h_{k,m} s_{k,t} \gamma_k + w_{t,m}, \quad (6.10)$$

where  $\gamma_k$  is an unknown complex scalar and  $w_{t,m} \in \mathbb{C}^M$  is i.i.d. zero mean circularly symmetric complex Gaussian (ZMCSCG) noise with variance  $\sigma^2$ . The unknown complex scalar  $\gamma_k$  can be developed as

$$\gamma_k = \sqrt{\rho_k} a_k e^{j\phi_k}, \quad (6.11)$$

$\rho_k \in \mathbb{R}^+$  is the transmit power of device  $k$ ,  $a_k \in \{0, 1\}$  is the device activity and  $\phi_k$  models a potential phase offset. This offset  $\phi_k$  can account for a carrier frequency offset (CFO), where the CFO is considered constant over the preamble duration. By assuming that all  $M$  antennas are perfectly synchronized, this offset is only dependent on the device. In case the device is inactive,  $\gamma_k$  will be zero. We will introduce the term *activity indicator* to denote  $\gamma_k$ .

Let us consider that the base station (BS) knows a part of the CSI, i.e.,  $g_{k,m}$  in

$$h_{k,m} = g_{k,m} + \lambda_k \epsilon_{k,m}, \quad (6.12)$$

where  $\epsilon_{k,m}$  are i.i.d. ZMCSCG variables with unit variance, and  $\lambda_k \in \mathbb{R}^+$  model the unknown part of the CSI. The large-scale fading coefficient of user  $k$  is  $\beta_k = \mathbb{E}(\|\mathbf{h}_k\|^2 / M) = \|\mathbf{g}_k\|^2 / M + \lambda_k^2$ . The factor  $\lambda_k$  models the quality of the known CSI. Hence, it quantifies the correlation of the actual channel  $h_{k,m}$  to the known CSI  $g_{k,m}$ . In the extreme case with  $\lambda_k = 0$ , the CSI is perfectly known and there is no uncertainty left, as was studied in [26]. This could be the case in practice in a fully static environment and if the CSI estimates are noiseless. However, for a realistic IoT

scenario, even for static devices, CSI is not perfect due to i) environment dynamics and ii) noisy estimates. The parameter  $\lambda_k$  then quantifies this imperfection. It is here assumed to be known<sup>4</sup>. Another extreme case, as considered in [41, 42], is obtained when  $g_{k,m} = 0, \forall m$ , implying that only the large scale fading coefficients of user  $k$  is known, *i.e.*,  $\lambda_k$ .

### 6.5.1.2 Device Activity Detection

This section describes the proposed activity detection algorithm. In the first subsection, the log-likelihood of the received preamble is derived. Then, given its nonconvex expression, an iterative approach is proposed to estimate the parameters  $\gamma_k \forall k$ . Finally, activity detection is performed.

### 6.5.1.3 Log-Likelihood of the Received Symbols

Combining (6.10) and (6.12), the symbol, received at antenna  $m$  and for pilot symbol  $t$ , is given by

$$y_{t,m} = \sum_{k=0}^{K-1} (g_{k,m} + \epsilon_{k,m} \lambda_k) s_{t,k} \gamma_k + w_{t,m}.$$

Stacking the observations at antenna  $m$  gives

$$\mathbf{y}_m = \sum_{k=0}^{K-1} g_{k,m} \mathbf{s}_k \gamma_k + \sum_{k=0}^{K-1} \epsilon_{k,m} \lambda_k \mathbf{s}_k \gamma_k + \mathbf{w}_m,$$

where

$$\mathbf{y}_m = \begin{pmatrix} y_{0,m} \\ \vdots \\ y_{t,m-1} \end{pmatrix}, \mathbf{s}_k = \begin{pmatrix} s_{0,k} \\ \vdots \\ s_{t,k-1} \end{pmatrix}, \mathbf{w}_m = \begin{pmatrix} w_{m,0} \\ \vdots \\ w_{t,m-1} \end{pmatrix}.$$

For a given value of  $\gamma_k$ ,  $\mathbf{y}_m | \gamma_k$  has a circularly symmetric Gaussian distribution with mean  $\sum_{k=0}^{K-1} g_{k,m} \mathbf{s}_k \gamma_k$ . After defining the vector  $\boldsymbol{\theta}_m = \sum_{k=0}^{K-1} \epsilon_{k,m} \lambda_k \mathbf{s}_k \gamma_k + \mathbf{w}_m$ , the covariance matrix is

$$\mathbf{C} = \mathbb{E} (\boldsymbol{\theta}_m \boldsymbol{\theta}_m^H) \quad (6.13)$$

$$= \sum_{k=0}^{K-1} \lambda_k^2 |\gamma_k|^2 \mathbf{s}_k \mathbf{s}_k^H + \sigma^2 \mathbf{I}_T, \quad (6.14)$$

where we used the fact that  $\epsilon_{k,m}$  were assumed to be i.i.d. and the additive noise is white. Note that this covariance matrix does not depend on the antenna index  $m$  and is thus valid for all  $\mathbf{y}_m$ .

Defining the vector  $\boldsymbol{\gamma} = (\gamma_0, \dots, \gamma_{K-1})^T$ , the log-likelihood of the observation vector  $\mathbf{y}_m$  is

$$\log p(\mathbf{y}_m | \boldsymbol{\gamma}) = -\ln(|\mathbf{C}|) - T \ln(\pi) - \boldsymbol{\theta}_m^H \mathbf{C}^{-1} \boldsymbol{\theta}_m.$$

Given the independence of  $\epsilon_{k,m}$  and  $w_{t,m}$  over the antennas, the different  $\mathbf{y}_m$  are independent as well. Hence, the likelihood of the aggregated observations at all antennas  $\mathbf{y} = (\mathbf{y}_0^T, \dots, \mathbf{y}_{M-1}^T)^T$

<sup>4</sup>It could be set to a certain value depending on the user activity profile and/or tracked for each user over time.

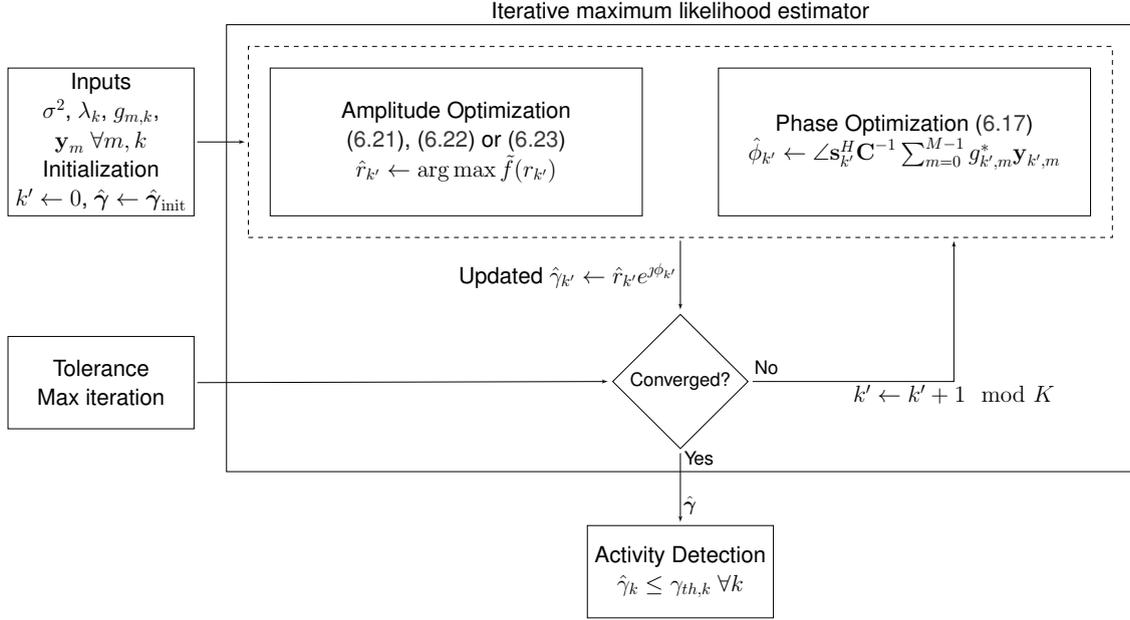


Figure 6.2: Block diagram of the iterative maximum likelihood estimator and activity detection.

becomes the product of the likelihood and the log-likelihood becomes the sum

$$\begin{aligned} \log p(\mathbf{y}|\boldsymbol{\gamma}) &= \sum_{m=0}^{M-1} \log p(\mathbf{y}_m|\boldsymbol{\gamma}) \\ &= -M \ln(|\mathbf{C}|) - MT \ln(\pi) - \sum_{m=0}^{M-1} \boldsymbol{\theta}_m^H \mathbf{C}^{-1} \boldsymbol{\theta}_m. \end{aligned} \quad (6.15)$$

**Iterative Algorithm for Maximizing Likelihood** The maximum likelihood estimator of  $\boldsymbol{\gamma}$  is obtained by maximizing

$$\hat{\boldsymbol{\gamma}}_{\text{ML}} = \arg \max_{\boldsymbol{\gamma}} \log p(\mathbf{y}|\boldsymbol{\gamma}).$$

Unfortunately, this problem is not trivial to solve given the nonlinear and nonconvex dependence of the log-likelihood, more specifically the covariance matrix  $\mathbf{C}$ , in  $\boldsymbol{\gamma}$ .

An idea to maximize the likelihood is to use an iterative approach, similarly as [41, 42]: at each iteration, all  $\gamma_k$  are kept fixed but one, which is optimized and updated. This way, they get updated one by one until convergence is attained, *i.e.*, a maximum number of iterations or a certain tolerance is reached. A block diagram of the algorithm is given in Fig. 6.2 and the pseudocode is summarized in Algorithm 2.

Let us consider that  $\gamma_{k'}$  needs to be updated. Note that  $\gamma_{k'}$  is complex-valued and the optimization needs to be done carefully. Using the definition introduced in (6.11), we can rewrite  $\gamma_{k'}$  with a phase-amplitude decomposition:  $\gamma_{k'} = r_{k'} e^{j\phi_{k'}}$ , with  $r_{k'} = |\gamma_{k'}| = \sqrt{\rho_k} a_k$ . The optimization with respect to  $\gamma_{k'}$  is done in the following in several steps: i) optimizing the phase  $\phi_{k'}$  for a fixed value of  $r_{k'}$ , ii) re-inserting this expression in the objective function to remove the dependence in  $\phi_{k'}$  and iii) optimizing the amplitude.

**Phase optimization** To highlight the dependence of the objective function in  $\gamma_{k'}$  for constant values of other  $\gamma_k$ ,  $k \neq k'$ , let us define the vector

$$\mathbf{y}_{k',m} = \mathbf{y}_m - \sum_{k=0, k \neq k'}^{K-1} g_{k,m} \mathbf{s}_k \gamma_k, \quad (6.16)$$

which can be seen as a cancellation of device interference to isolate the contribution from device  $k'$ . Hence, the objective function to maximize can be written as<sup>5</sup>

$$f(r_{k'}, \phi_{k'}) = -M \ln(|\mathbf{C}|) - \sum_{m=0}^{M-1} (\mathbf{y}_{k',m} - g_{k',m} \mathbf{s}_{k'} r_{k'} e^{j\phi_{k'}})^H \mathbf{C}^{-1} (\mathbf{y}_{k',m} - g_{k',m} \mathbf{s}_{k'} r_{k'} e^{j\phi_{k'}}),$$

where we explicitly express the dependence in  $(r_{k'}, \phi_{k'})$  while the other  $(r_k, \phi_k)$ ,  $k \neq k'$  do not appear since they are considered constant. Note that the matrix  $\mathbf{C}$ , defined in (6.13), does not depend on  $\phi_{k'}$  but only  $r_{k'}$ . In the extreme case of no prior CSI, *i.e.*,  $g_{k',m} = 0 \forall m$ , the dependence of  $f(r_{k'}, \phi_{k'})$  in  $\phi_{k'}$  disappears and there is an underdetermination and no estimate of the phase offset can be obtained. In other cases, we can find that, after some manipulations,

$$\frac{df}{d\phi_{k'}} = 0 \Leftrightarrow \hat{\phi}_{k'} = \angle \mathbf{s}_{k'}^H \mathbf{C}^{-1} \sum_{m=0}^{M-1} g_{k',m}^* \mathbf{y}_{k',m}. \quad (6.17)$$

This result has an intuitive understanding: the optimal phase  $\phi_{k'}$  tends to align the partial CSI with the observations due to device  $k'$ .

**Removing the phase dependence** Inserting this optimal value in the objective function  $f(r_{k'}, \phi_{k'})$  makes the dependence in  $\phi_{k'}$  vanish and gives

$$f(r_{k'}) = -M \ln(|\mathbf{C}|) - \sum_{m=0}^{M-1} \mathbf{y}_{k',m}^H \mathbf{C}^{-1} \mathbf{y}_{k',m} - r_{k'}^2 \mathbf{s}_{k'}^H \mathbf{C}^{-1} \mathbf{s}_{k'} \sum_{m=0}^{M-1} |g_{k',m}|^2 + 2 \left| \sum_{m=0}^{M-1} \mathbf{y}_{k',m}^H \mathbf{C}^{-1} \mathbf{s}_{k'} g_{k',m} \right| r_{k'}. \quad (6.18)$$

**Amplitude optimization** In the expression of  $f(r_{k'})$ , a complex modulus of an expression that depends on  $r_{k'}$  has appeared, which complicates differentiation. After some manipulations, we get

$$\tilde{f}(r_{k'}) = -M \ln(|\mathbf{C}|) + \frac{\alpha r_{k'}^2 + \beta r_{k'}}{1 + \delta r_{k'}^2}, \quad (6.19)$$

where we defined the constants (independent of  $r_{k'}$ )  $\alpha$ ,  $\beta$  and  $\delta$ , as

$$\begin{aligned} \alpha &= \sum_{m=0}^{M-1} |\mathbf{y}_{k',m}^H \mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'}|^2 \lambda_{k'}^2 - \mathbf{s}_{k'}^H \mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'} \sum_{m=0}^{M-1} |g_{k',m}|^2 \\ \beta &= 2 \left| \sum_{m=0}^{M-1} \mathbf{y}_{k',m}^H \mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'} g_{k',m} \right| \\ \delta &= \mathbf{s}_{k'}^H \mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'} \lambda_{k'}^2. \end{aligned} \quad (6.20)$$

<sup>5</sup>For clarity, we omit in the following the constant term  $MT \ln(\pi)$  which does not affect optimization as it does not depend on  $\gamma$  and vanishes after differentiation.

One can note that  $\tilde{f}(r_{k'})$  in (6.19) can now be differentiated with respect to  $r_{k'}$ . Setting the derivative to zero gives, noting that the denominator is always strictly positive,

$$\begin{aligned} \frac{d\tilde{f}}{dr_{k'}} &= 0 \\ \Leftrightarrow 0 &= -r_{k'}^3 2M\delta^2 - r_{k'}^2 \beta \delta + r_{k'}(-2M\delta + 2\alpha) + \beta, \end{aligned} \quad (6.21)$$

which is a polynomial of degree 3 in  $r_{k'}$ . There are closed-form solutions for the roots of such polynomials. One should still check for the best solutions among them, restricted to be positive and real.

The algorithm is summarized in the pseudocode **Algorithm 2**. At each iteration, the constants  $\alpha$ ,  $\beta$  and  $\delta$  can be easily re-evaluated based on (6.20).

---

**Algorithm 2** Iterative maximum likelihood device activity detector
 

---

**Require:**  $\sigma^2, \lambda_k, \mathbf{y}_m, g_{k,m}, \hat{\gamma}_{\text{init}} \forall k, m$

$k' \leftarrow 0$

$\hat{\gamma} \leftarrow \hat{\gamma}_{\text{init}}$

$\mathbf{C}^{-1} \leftarrow \left( \sum_{k=0}^{K-1} \lambda_k^2 |\hat{\gamma}_k|^2 \mathbf{s}_k \mathbf{s}_k^H + \sigma^2 \mathbf{I}_T \right)^{-1}$

**while** Not converged **do**

    Compute  $\mathbf{y}_{k',m}, \mathbf{C}_{-k'}^{-1}, \alpha, \beta$  and  $\delta$  based on (6.16) and (6.20)

$\hat{r}_{k'} \leftarrow \arg \max_{r_{k'}} \tilde{f}(r_{k'})$  ▷ Update amplitude based on (6.21), (6.22) or (6.23)

$\hat{\phi}_{k'} \leftarrow \angle \mathbf{s}_{k'}^H \mathbf{C}^{-1} \sum_{m=0}^{M-1} g_{k',m}^* \mathbf{y}_{k',m}$  ▷ Update phase

$\hat{\gamma}_{k'} \leftarrow \hat{r}_{k'} e^{j\hat{\phi}_{k'}}$

$\mathbf{C}^{-1} \leftarrow \mathbf{C}_{-k'}^{-1} - \frac{\mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'} \mathbf{s}_{k'}^H \mathbf{C}_{-k'}^{-1} r_{k'}^2 \lambda_{k'}^2}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{-k'}^{-1} \mathbf{s}_{k'} r_{k'}^2 \lambda_{k'}^2}$

$k' \leftarrow k' + 1 \pmod K$

**end while**

---

**Particular case: device with no CSI** Now consider that, for a given  $k'$ ,  $g_{k',m} = 0 \forall m$ . This could be because this device is new or moving a lot such that its CSI is outdated. Only, its parameter  $\lambda_{k'}$  is known. At iteration of user  $k'$ , evaluating (6.20) for  $\lambda_{k'} = 0$  implies that  $\beta = 0$ . Hence, (6.21) simplifies to

$$0 = 2r_{k'}(-r_{k'}^2 M\delta^2 - M\delta + \alpha),$$

which has a trivial solution in  $r_{k'} = 0$ . One of the other root is always negative. Keeping only the positive one, we find the amplitude update

$$\hat{r}_{k'} = \sqrt{\frac{\alpha - M\delta}{M\delta^2}}. \quad (6.22)$$

If this root is imaginary, we set  $\hat{r}_{k'}$  to zero. As discussed before introducing the phase update equation (6.17), in the case of no prior CSI, the phase ambiguity cannot be resolved. This particular case gives an update relatively similar to the maximum likelihood estimator derived in [41].

**Particular case: device with complete prior CSI** Now, consider that, for a given  $k'$ ,  $\lambda_{k'} = 0$ , so that the CSI is perfectly known. Only the phase shift and the transmit power are unknown. At

iteration of user  $k'$ , evaluating (6.20) for  $\lambda_{k'} = 0$  implies that  $\delta = 0$ . Hence, (6.21) simplifies to a linear equation  $0 = r_{k'}2\alpha + \beta$ , which gives the following amplitude update

$$\hat{r}_{k'} = \frac{-\beta}{2\alpha} = \frac{|\mathbf{s}_{k'}^H \mathbf{C}^{-1} \sum_{m=0}^{M-1} g_{k',m}^* \mathbf{y}_{k',m}|}{\mathbf{s}_{k'}^H \mathbf{C}^{-1} \mathbf{s}_{k'} \sum_{m'} |g_{k',m'}|^2}, \quad (6.23)$$

while the phase is updated according to (6.17).

**Initialization** To start the iterative algorithm, we consider different choices to initialize  $\hat{\gamma}_{\text{init}}$ . A simple choice is to initialize to zero, *i.e.*,  $\hat{\gamma}_{\text{init}}^0 = \mathbf{0}$ . Another choice is to initialize solely based on the available prior CSI, considering that  $\lambda_k \approx 0$ ,  $\forall k$ . The estimator is similar to [26], except that, here, prior CSI is used instead of complete CSI.

If  $\lambda_k \approx 0$ ,  $\forall k$ , the covariance matrix  $\mathbf{C}$ , defined in (6.13), simplifies to  $\mathbf{C} = \sigma^{-2} \mathbf{I}_T$ , which is independent of  $\gamma_k$ . Hence, many terms of the log-likelihood in (6.15) becomes independent of  $\gamma$ . Maximizing (6.15) becomes equivalent to the following minimization

$$\begin{aligned} \max_{\gamma} \log p(\mathbf{y}|\gamma) &= \min_{\gamma} \sum_{m=0}^{M-1} \left\| \mathbf{y}_m - \sum_{k=0}^{K-1} g_{k,m} \mathbf{s}_k \gamma_k \right\|^2 \\ &= \min_{\gamma} \|\mathbf{y} - \mathbf{\Gamma} \gamma\|^2, \end{aligned}$$

where we defined the vector and matrix notations

$$\mathbf{y} = \begin{pmatrix} \mathbf{y}_0 \\ \vdots \\ \mathbf{y}_{M-1} \end{pmatrix}, \quad \mathbf{\Gamma} = \begin{pmatrix} \mathbf{\Gamma}_0 \\ \vdots \\ \mathbf{\Gamma}_{M-1} \end{pmatrix}, \quad \mathbf{\Gamma}_m = (\mathbf{s}_0 \quad \dots \quad \mathbf{s}_{K-1}).$$

This minimization problem is a quadratic function of  $\gamma$ , which is convex, and can be easily solved by setting the Wirtinger derivative to zero. It can also be seen as a least squares problem. The estimate has the following closed-form expression

$$\hat{\gamma}_{\text{init}}^{\text{ZF}} = (\mathbf{\Gamma}^H \mathbf{\Gamma})^{-1} \mathbf{\Gamma}^H \mathbf{y}. \quad (6.24)$$

This last estimator can be seen as a ZF estimator, which requires a matrix inversion. To avoid ill-conditioning, a first necessary condition is that  $K \leq MT$ . This condition is not sufficient as the channel and preamble of two devices could be correlated, especially when  $K$  is on the order of  $MT$ . Moreover, if no prior information  $g_{k,m}$  is available for a given user  $k'$ , *i.e.*,  $g_{k',m} = 0$ ,  $\forall m$ , the inverse will also be ill-conditioned. Indeed, this implies that the  $k'$ -th column of  $\mathbf{\Gamma}$  becomes null and thus  $\mathbf{\Gamma}$  is rank deficient. Moreover, the prior CSI might be noisy, leading to unstable results.

To make initialization more robust, we can use a least minimum mean square error (LMMSE) criterion. To do this, some prior knowledge must be assumed on the statistics of  $\gamma$ , more specifically, its first and second order moments. We here make the following assumptions: i) the activity of each device is independent of one another, ii) the average activity and average transmit power of each device is known and iii) no prior information is known on the phase offset so that  $\phi_k$  is considered uniformly distributed between 0 and  $2\pi$ . Under these assumptions, we have  $\mathbb{E}(\gamma) = \mathbf{0}$  and  $\mathbf{D} = \mathbb{E}(\gamma \gamma^H) = \text{diag}(\mathbb{E}(a_0) \mathbb{E}(\rho_0), \dots, \mathbb{E}(a_{K-1}) \mathbb{E}(\rho_{K-1}))$ . Hence, for the linear observation model  $\mathbf{y} = \mathbf{\Gamma} \gamma + \mathbf{w}$ , still considering that  $\lambda_k \approx 0$ ,  $\forall k$ , the LMMSE estimator of  $\gamma$  is then given by [59]

$$\hat{\gamma}_{\text{init}}^{\text{LMMSE}} = (\mathbf{\Gamma}^H \mathbf{\Gamma} + \sigma^2 \mathbf{D}^{-1})^{-1} \mathbf{\Gamma}^H \mathbf{y}. \quad (6.25)$$

Note that the matrix to be inverted is always well-conditioned. Finally, a matched filter (MF) estimator could be used to avoid the need for matrix inversion.

$$\hat{\gamma}_{\text{init}}^{\text{MF}} = (\text{diag}(\mathbf{\Gamma}^H \mathbf{\Gamma}) + \sigma^2 \mathbf{D}^{-1})^{-1} \mathbf{\Gamma}^H \mathbf{y}. \quad (6.26)$$

**Activity Detection** A non-negative activity threshold  $\gamma_{th,k}$  is applied for each device  $k$ . A device is considered active if  $|\hat{\gamma}_k| \geq \gamma_{th,k}$ . The real-valued threshold is defined as,

$$\gamma_{th,k} = v \sqrt{\text{SNR}_k}^{-1}, \quad (6.27)$$

where  $v$  is chosen to have a desired probability of false alarm and miss detection performance and with  $\text{SNR}_k = M\beta_k/\sigma^2 = \mathbb{E}(\|\mathbf{h}_k\|^2)/\sigma^2 = (\|\mathbf{g}_k\|^2 + M\lambda_k^2)/\sigma^2$ .

#### 6.5.1.4 Numerical Assessment

Table 6.2: Simulation parameter set with default values.

Parameter	Symbol	Default value
Number of devices	$K$	500
Number of total antennas	$M$	64
Signal-to-noise ratio	SNR	20 dB
Device activity probability	$\epsilon_a$	0.1
Pilot sequence	$s_k$	$\sim \mathcal{CN}(0, 1)$
Pilot length	$\tau_p$	10 symbols
Phase offset	$\phi_k$	$\sim \mathcal{U}_{[0, 2\pi]}$
Number of simulations	$N_{\text{sim}}$	>10 000
Number of algorithm iterations	$N_{\text{iter}}$	$K \cdot 4$
Initialization vector	$\hat{\gamma}_{\text{init}}$	$\hat{\gamma}_{\text{init}}^{\text{LMMSE}}$ (6.25)
Unknown part of the CSI	$\lambda$	0.3

The default simulation configurations are summarized in Table 6.2. The device activity profile is generated randomly and independently for each device with a probability  $\epsilon_a = 0.1$ , meaning that on average  $\epsilon_a K = 50$  devices are active simultaneously. Or equivalently, the devices have an average duty cycle of 10%, which is high for typical IoT applications [24]. The channel between the BS and device  $k$  is modelled as in (6.12). The pilot sequence is randomly generated from a complex Gaussian distribution  $s_k \sim \mathcal{CN}(0, 1)$ , and is assumed to be known by the BS. Each device uses a pilot sequence of 10 symbols. A random phase offset  $\phi_k \sim \mathcal{U}_{[0, 2\pi]}$  is generated to simulate a carrier frequency offset (considered time-invariant over the preamble duration). The source code for all simulations can be accessed online<sup>6</sup>.

**Convergence of different initialization vectors** The convergence of different initializations is evaluated with respect to the genie-aided approach. In the genie-aided case, the algorithm is initialized with the real activity indicators, i.e.,  $\gamma$ . The convergence is assessed via the likelihood (6.15) and the mean square error (MSE). The former should monotonically increase with each iteration, while the MSE can vary as it can not directly be minimized. The performance of the different initialization vectors for  $\hat{\gamma}_{\text{init}}$  are depicted in Figure 6.3. The bottom figures zoom in on a

<sup>6</sup><https://github.com/wavecore-research/grant-free-random-access-partial-csi>

smaller region to distinguish the performance of the initialization vectors when converging closer to the genie-aided case. While all initialization methods approximate the genie-aided case, the initialization vector has a non-negligible impact on the performance of the algorithm. An intuitive approach is to initialize with  $\mathbf{0}$  because the activity probability is low and hence, on average, 90% of the devices are expected to be inactive. As illustrated in Figure 6.3,  $\hat{\gamma}_{\text{init}} = \mathbf{0}$  requires considerably more iterations to approach the other initialization methods.

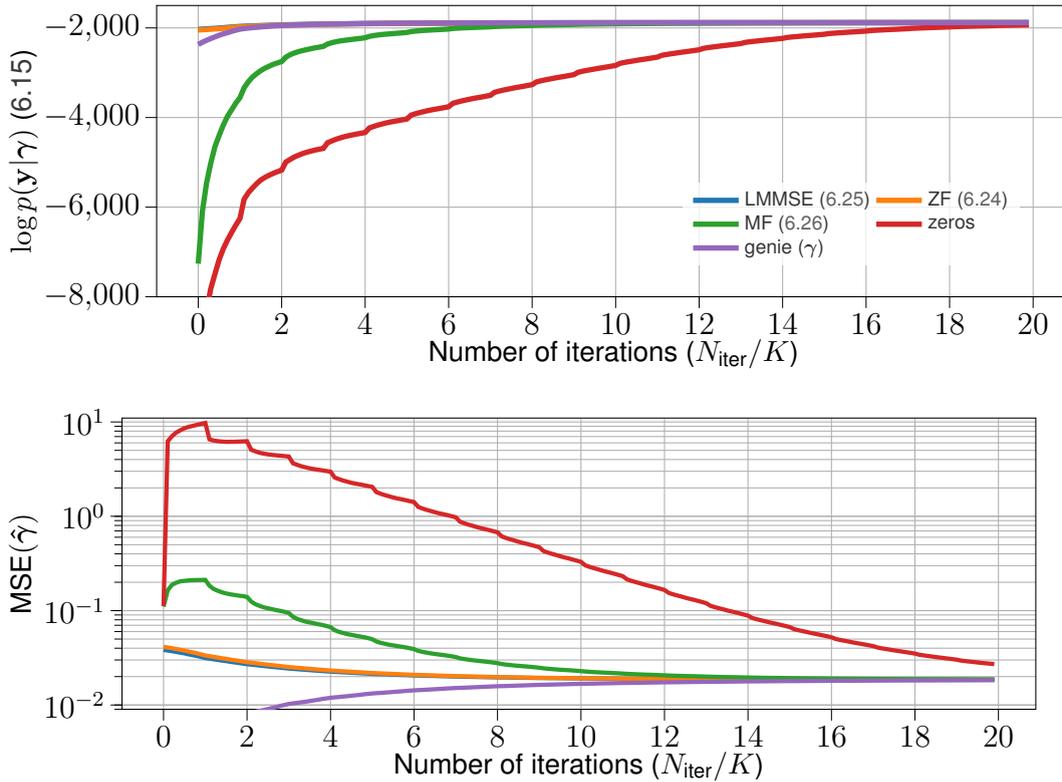


Figure 6.3: The log-likelihood (6.15) and MSE of the estimated activity indicators for different initialization vectors. While with all initialization vectors the genie-aided case is approximated, different number of iterations are required.

**Impact of the quality of prior CSI** Figure 6.4 illustrates the performance of the detector algorithms for different correlations between the actual channel and the known CSI, i.e.,  $\lambda$ . With increased  $\lambda$ , and thus decreased channel knowledge, both the LMMSE estimator and the proposed algorithm have an increased probability of miss detection. The figure also demonstrates the gain of the proposed algorithm with respect to the LMMSE estimator. The algorithm outperforms the LMMSE estimator for all  $\lambda$  and is most effective when the prior CSI has a strong correlation with the actual channel, and diminishes with decreased channel knowledge.

**Impact of the signal-to-noise ratio** Figure 6.5 shows the false alarm and miss detection probability of the LMMSE estimator and the iterative maximum likelihood device activity detector for different device SNRs. The full CSI case is included as a baseline for comparison, where the full CSI is known instead of only a portion ( $\lambda$ ). Fig. 6.5 demonstrates the large performance gain of the proposed algorithm with respect to the LMMSE estimator. The graph demonstrates that the iterative algorithm lowers the probability of miss detection by a factor of 21 for the same probability of false alarm. The performance is only marginally increased for very low SNRs (below zero).

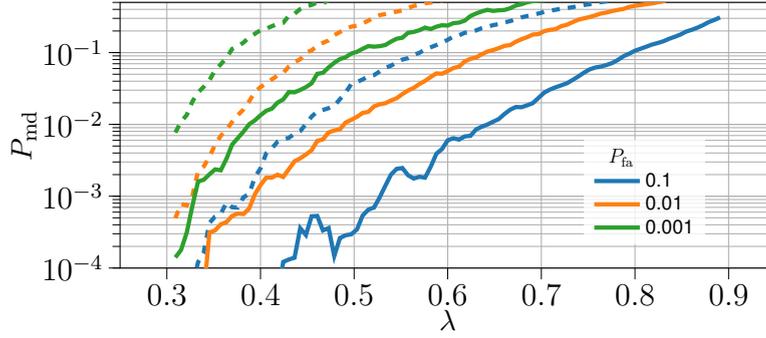


Figure 6.4: Performance of the proposed algorithm (—) versus the LMMSE estimator (---) for different values of channel knowledge. The probability of miss detection is shown for different values of  $P_{fa}$ .

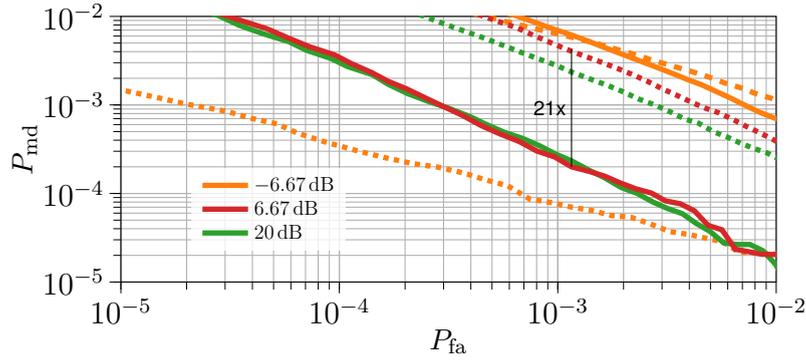


Figure 6.5: The probability of false alarm and miss detection for different device SNRs for the LMMSE estimator when having full CSI (.....) and partial CSI (---), and the proposed iterative algorithm with partial CSI (—). No miss detection or false alarm occurred in the full CSI case for SNR values of 6.67 dB and 20 dB.

## 6.5.2 Partial CSI in RadioWeaves

The previous work is extended to the RadioWeaves system, where we have geographically distributed CSPs each having one or more antennas.

### 6.5.2.1 System Model Extension to RadioWeaves

The model (6.10) of the co-located case can be extended by considering that are  $N$  CSPs equipped with  $M_n$  antennas. The UE index is denoted as  $k$ . Let us consider that the CSPs know a part of the CSI, *i.e.*,  $\mathbf{g}_{k,n}$  in

$$\mathbf{h}_{n,k} = \gamma_k (\mathbf{g}_{n,k} + \lambda_{n,k} \boldsymbol{\epsilon}_{n,k}) e^{j\phi_{n,k}},$$

where  $\gamma_k$  is a scalar given by the square root of the transmitted power,  $\lambda_{n,k}$  is a factor that quantifies the correlation of the channel to the known CSI. If  $\lambda_{n,k} = 0$ , we are in the fully static case while if  $\mathbf{g}_{n,k} = \mathbf{0}$ , we are back to the case of only knowing large scale fading coefficient (cf. [41, 42]). We can check both cases of knowing or not knowing  $\lambda_{n,k}$ . We consider that  $\boldsymbol{\epsilon}_{n,k}$  is composed of ZMCSCG i.i.d. elements with unit variance. The phase shift  $\phi_{n,k}$  comes from the presence of CFO, where the phase offset is assumed constant across the transmission duration. The CFO is assumed constant for the antennas of the same CSP but different across users and CSPs.

### 6.5.2.2 Device Activity Detection

Similarly to Section 6.5.1, the log-likelihood of the received preambles is derived, and an iterative algorithm is proposed to detect the activity of the UEs.

The derivations are included in Section A.2. Setting the derivative of (A.20) to zero gives

$$\frac{d\tilde{f}}{d\gamma_{k'}} = 0$$

$$0 = \sum_n \frac{-\gamma_{k'}^3 2M_n \delta_n^2 - \gamma_{k'}^2 \beta_n \delta_n + \gamma_{k'} (-2M_n \delta_n + 2\alpha_n) + \beta_n}{(1 + \delta_n \gamma_{k'}^2)^2}.$$

Solving this equation implies finding the root of a polynomial of degree with scales with  $4N$ , which can rapidly become prohibitive. To resolve this, the sum over  $n$  can be truncated and be restricted to the CSP with highest average channel power to user  $k$ . In that case, as in the centralised case, the problem can be simplified to finding the roots of a polynomial of degree 3 in  $\gamma_{k'}^7$ .

---

<sup>7</sup>The numerical evaluation of the proposed algorithm will be provided in another Reindeer deliverable.

## 6.6 Impact of Antenna Deployment Topology on Performance of Grant-Free Access Detection Methods

In this section, we study the impact of antenna topology on the activity detection performance. First, we consider the activity detection problem when the antennas are distributed and later, the activity detection algorithm is described.

### 6.6.1 Signal Model And Problem Formulation

Consider a RadioWeaves network with  $M$  uniform linear array (ULA) panels each equipped with  $N$  antennas and serving  $K$  arbitrarily distributed single antenna users. Each ULA in the RadioWeaves deployment is considered as a CSP and all CSPs are considered to be connected to an ECSP for joint processing of the signals. Due to the sporadic nature of the traffic in the massive access scenario of mMTC, only a small fraction of the  $K$  users are active at any given time instant. We assume that each device transmits independently with an activation probability  $\epsilon \ll 1$ . Let  $a_k \in \{0, 1\}$  where  $a_k = 1$  denotes that the  $k^{\text{th}}$  device is active and  $a_k = 0$  that it is inactive and  $\Pr(a_k = 1) = \epsilon$  and  $\Pr(a_k = 0) = 1 - \epsilon$ . Let  $\mathbf{a} = (a_1, a_2, \dots, a_K)$  denote the activity of  $K$  users at any time instant. Due to the sporadic nature of mMTC traffic, the vector  $\mathbf{a}$  will be sparse. The set of active users is denoted by  $\mathcal{K}_a$  i.e.,  $\mathcal{K}_a = \{k : a_k = 1\}$ .

The channel gain between the  $n^{\text{th}}$  antenna in the  $m^{\text{th}}$  AP to device  $k$  is given by

$$g_{mnk} = \beta_{mk}^{\frac{1}{2}} h_{mnk} \quad (6.28)$$

where  $\beta_{mk}$  is the large-scale fading coefficient between the  $m^{\text{th}}$  CSP and the user  $k$  and  $h_{mnk} \sim \mathcal{CN}(0, 1)$  is the small-scale fading coefficient. We assume that the large-scale fading coefficient parameters  $\{\beta_{mk}\}$  are known at the ECSP [41, 94]. We consider a block fading scenario where each channel remains constant during a coherence interval [68, Ch.2] and all the channels are independently distributed. Let  $\tau_c$  be the number of channel uses per coherence interval. Due to the large number of users, typically  $K \gg \tau_c$ , assigning orthogonal pilot sequences to each user is not feasible. Instead we assign non-orthogonal unique signature sequence,  $\mathbf{s}_k \in \mathbb{C}^{L \times 1}$  to each user  $k$ , where  $L \leq \tau_c$  is the signature sequence length. We assume that the signature sequences of all the users are known at the ECSP.

The signal  $\mathbf{y}_{mn} \in \mathbb{C}^{L \times 1}$  received at the  $n^{\text{th}}$  antenna of the  $m^{\text{th}}$  CSP is given by

$$\begin{aligned} \mathbf{y}_{mn} &= \sum_{k=1}^K a_k \rho_k^{\frac{1}{2}} g_{mnk} \mathbf{s}_k + \mathbf{w}_{mn} \\ &= \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \mathbf{g}_{mn} + \mathbf{w}_{mn}, \end{aligned} \quad (6.29)$$

where  $\mathbf{S} = [\mathbf{s}_1 \ \mathbf{s}_2 \ \dots \ \mathbf{s}_K] \in \mathbb{C}^{L \times K}$  is the collection of all signature sequences,  $\rho_k$  is the power transmitted by user  $k$ ,  $\mathbf{D}_a = \text{diag}(\mathbf{a})$ ,  $\mathbf{D}_\rho = \text{diag}(\rho_1, \rho_2, \dots, \rho_K)$ ,  $\mathbf{g}_{mn} = [g_{mn1} \ g_{mn2} \ \dots \ g_{mnK}]^T \in \mathbb{C}^{K \times 1}$  is the channel vector from all  $K$  users to the  $n^{\text{th}}$  antenna of the  $m^{\text{th}}$  CSP and  $\mathbf{w}_{mn} \sim \mathcal{CN}(\mathbf{0}, \sigma^2 \mathbf{I}_L)$  is the independent additive white Gaussian noise vector.

Thus, the signal  $\mathbf{Y}_m \in \mathbb{C}^{L \times N}$  received at the  $m^{\text{th}}$  CSP can be expressed as

$$\mathbf{Y}_m = \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \mathbf{G}_m + \mathbf{W}_m, \quad (6.30)$$

where  $\mathbf{G}_m = [\mathbf{g}_{m1} \ \mathbf{g}_{m2} \ \dots \ \mathbf{g}_{mN}] \in \mathbb{C}^{K \times N}$  is the channel matrix between the  $K$  users and the  $m^{\text{th}}$  CSP and  $\mathbf{W}_m = [\mathbf{w}_{m1} \ \mathbf{w}_{m2} \ \dots \ \mathbf{w}_{mN}] \in \mathbb{C}^{L \times N}$  is the noise matrix.

Let the collection of signals be

$$\begin{aligned} \mathbf{Y} &= \begin{bmatrix} \mathbf{Y}_1 \\ \mathbf{Y}_2 \\ \vdots \\ \mathbf{Y}_M \end{bmatrix} = \begin{bmatrix} \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \mathbf{G}_1 \\ \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \mathbf{G}_2 \\ \vdots \\ \mathbf{S} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \mathbf{G}_M \end{bmatrix} + \mathbf{W} \\ &= \begin{bmatrix} \mathbf{S} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{S} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{S} \end{bmatrix} \begin{bmatrix} \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{D}_a \mathbf{D}_\rho^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \\ \vdots \\ \mathbf{G}_M \end{bmatrix} + \mathbf{W}, \end{aligned} \quad (6.31)$$

where  $\mathbf{W} = [\mathbf{W}_1^T \ \mathbf{W}_2^T \ \dots \ \mathbf{W}_M^T]^T$ . From (6.31), it can be seen that the columns of  $\mathbf{Y}$  are independent and each column is distributed as  $\mathbf{Y}(:, i) \sim \mathcal{CN}(\mathbf{0}_{LM}, \mathbf{Q})$ ,  $\forall i = 1, 2, \dots, N$ , where  $\mathbf{Q}$  is the covariance matrix given by

$$\mathbf{Q} = \begin{bmatrix} \mathbf{S} \mathbf{D}_\gamma \mathbf{D}_{\beta_1} \mathbf{S}^H & \mathbf{0}_L & \dots & \mathbf{0}_L \\ \mathbf{0}_L & \mathbf{S} \mathbf{D}_\gamma \mathbf{D}_{\beta_2} \mathbf{S}^H & \dots & \mathbf{0}_L \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_L & \mathbf{0}_L & \dots & \mathbf{S} \mathbf{D}_\gamma \mathbf{D}_{\beta_M} \mathbf{S}^H \end{bmatrix} + \sigma^2 \mathbf{I}_{LM}, \quad (6.32)$$

where  $\mathbf{D}_{\beta_m}$  is a diagonal matrix with diagonal elements corresponding to the large-scale fading coefficient from  $K$  users to  $m^{\text{th}}$  CSP, i.e.,  $\mathbf{D}_{\beta_m} = \text{diag}(\beta_m)$  where  $\beta_m = (\beta_{m1}, \beta_{m2}, \dots, \beta_{mK})$  and  $\mathbf{D}_\gamma = \text{diag}(\gamma)$ , where  $\gamma = (a_1 \rho_1, a_2 \rho_2, \dots, a_K \rho_K)$ .

By utilizing the block-diagonal structure of the covariance matrix  $\mathbf{Q}$ , the likelihood of  $\mathbf{Y}$  given  $\gamma$  is

$$\begin{aligned} p(\mathbf{Y}|\gamma) &= \prod_{m=1}^M \prod_{n=1}^N \frac{1}{|\pi \mathbf{Q}_m|} \exp(-\mathbf{y}_{mn}^H \mathbf{Q}_m^{-1} \mathbf{y}_{mn}) \\ &= \prod_{m=1}^M \frac{1}{|\pi \mathbf{Q}_m|^N} \exp(-\text{tr}(\mathbf{Q}_m^{-1} \mathbf{Y}_m \mathbf{Y}_m^H)), \end{aligned} \quad (6.33)$$

where  $\mathbf{Q}_m = \mathbf{S} \mathbf{D}_\gamma \mathbf{D}_{\beta_m} \mathbf{S}^H + \sigma^2 \mathbf{I}_L$ . The maximum likelihood estimate of  $\gamma$  can be found by maximizing  $p(\mathbf{Y}|\gamma)$  or equivalently minimizing  $-\log(p(\mathbf{Y}|\gamma))$  which is given by

$$\begin{aligned} \gamma^* &= \underset{\gamma}{\text{argmin}} \sum_{m=1}^M \log |\mathbf{Q}_m| + \text{tr} \left( \mathbf{Q}_m^{-1} \frac{\mathbf{Y}_m \mathbf{Y}_m^H}{N} \right) \\ &\text{subject to } \gamma \geq \mathbf{0}_K. \end{aligned} \quad (6.34)$$

To perform the activity detection, all the received signals at the CSPs need to be passed to the ECSP for  $L \geq N$ . When  $L < N$ , CSP  $m$  sends the sample covariance  $\mathbf{Y}_m \mathbf{Y}_m^H$  to the ECSP to reduce the fronthaul usage. The CPU needs to solve the optimization problem in (6.34). For  $M = 1$ , the co-located architecture case, a covariance-based coordinate descent algorithm is proposed in [47] for device activity detection. However, for a cell-free architecture, due to the presence of  $M > 1$  summation terms in (6.34), the brute force approach to solve (6.34) requires huge complexity and the complexity increases exponentially with  $M$ .

## 6.6.2 Device Activity Detection

In this section, we study the cost function (6.34) and exploit the features of cell-free architecture to develop algorithms for activity detection in grant-free random access schemes.

### 6.6.2.1 Coordinate Descent Cost Function

Let

$$f(\gamma) = \sum_{m=1}^M \log |\mathbf{Q}_m| + \text{tr} \left( \mathbf{Q}_m^{-1} \frac{\mathbf{Y}_m \mathbf{Y}_m^H}{N} \right) \quad (6.35)$$

be the cost function which needs to be minimized in (6.34). Define

$$f^m(\gamma) = \log |\mathbf{Q}_m| + \text{tr} \left( \mathbf{Q}_m^{-1} \frac{\mathbf{Y}_m \mathbf{Y}_m^H}{N} \right) \quad (6.36)$$

be the cost function associated with the  $m^{\text{th}}$  block in (6.35). Then we can write  $f(\gamma) = \sum_{m=1}^M f^m(\gamma)$ . Setting  $\mathbf{Q}_m$  as a function of  $\gamma$ , i.e.,

$$\mathbf{Q}_m(\gamma) = \mathbf{S} \mathbf{D}_\gamma \mathbf{D}_{\beta_m} \mathbf{S}^H + \sigma^2 \mathbf{I}_L \quad (6.37)$$

$$= \sum_{k=1}^K \gamma_k \beta_{mk} \mathbf{s}_k \mathbf{s}_k^H + \sigma^2 \mathbf{I}_L, \quad (6.38)$$

we can see  $\mathbf{Q}_m$  as a sum of  $K$  rank-one updates to  $\sigma^2 \mathbf{I}_L$ . Thus, we can optimise  $f(\gamma)$  with respect to one argument  $\gamma_k$ ,  $k \in \{1, 2, \dots, K\}$  in one step and we iterate several times over the whole set of variables until the cost function cannot be further reduced. A random ordering is considered while optimising to avoid dependency during detection if any. For  $k \in \{1, 2, \dots, K\}$ , let us define  $f_k^m(d) = f^m(\gamma + d \mathbf{e}_k)$ , where  $\mathbf{e}_k$  is the  $k^{\text{th}}$  canonical basis with a single-1 at the  $k^{\text{th}}$  coordinate. By applying the Sherman-Morrison rank-1 update identity [83] on  $\mathbf{Q}_m$ , we obtain

$$(\mathbf{Q}_m + d \beta_{mk} \mathbf{s}_k \mathbf{s}_k^H)^{-1} = \mathbf{Q}_m^{-1} - d \beta_{mk} \frac{\mathbf{Q}_m^{-1} \mathbf{s}_k \mathbf{s}_k^H \mathbf{Q}_m^{-1}}{1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k}. \quad (6.39)$$

By applying the determinant identity [90], we can obtain

$$|\mathbf{Q}_m + d \beta_{mk} \mathbf{s}_k \mathbf{s}_k^H| = (1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k) |\mathbf{Q}_m|. \quad (6.40)$$

Now we can write the overall maximum likelihood (ML) cost function in (6.35) for each coordinate  $k$  as  $f_k(d) = \sum_{m=1}^M f_k^m(d)$ , given by

$$f_k(d) = c + \sum_{m=1}^M \log(1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k) - d \beta_{mk} \frac{\mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{Q}_{\mathbf{Y}_m} \mathbf{Q}_m^{-1} \mathbf{s}_k}{1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k}, \quad (6.41)$$

where  $c = \sum_{m=1}^M (\log |\mathbf{Q}_m| + \text{tr}(\mathbf{Q}_m^{-1} \mathbf{Q}_{\mathbf{Y}_m}))$  is a constant and  $\mathbf{Q}_{\mathbf{Y}_m} = \frac{\mathbf{Y}_m \mathbf{Y}_m^H}{N}$ . Taking the derivative of  $f_k(d)$  with respect to  $d$  and equating to zero gives

$$f_k'(d) = \sum_{m=1}^M \frac{\beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k}{(1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k)} - d \beta_{mk} \frac{\mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{Q}_{\mathbf{Y}_m} \mathbf{Q}_m^{-1} \mathbf{s}_k}{(1 + d \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k)^2} = 0, \quad (6.42)$$

which is a polynomial of degree  $2M - 1$ . Hence, finding the value of  $d$  which minimizes (6.41) requires a complexity of  $\mathcal{O}(M^4 L^2)$  and involves solving higher degree polynomial equations. This huge complexity calls for a low complexity design to ensure scalability of the device activity detection in cell-free massive MIMO networks.

### 6.6.2.2 Dominant CSP Based Activity Detection

For a device  $k$ , let

$$m' = \underset{m}{\operatorname{argmax}}\{\beta_{mk}\} \quad (6.43)$$

be the index of the CSP with which the device has the dominant large-scale fading coefficient and we call this CSP the most dominant CSP for the device  $k$ . In the proposed dominant CSP based activity detection, the updates for any device is given by its corresponding dominant CSP. Hence, at the CPU, we minimize the cost function with respect to the dominant CSP for device  $k$  and the soft information about the device  $k$  from this CSP is propagated to the other CSPs. The cost function of device  $k$  with respect to the dominant CSP  $m'$  is given by

$$f_{k,m'}(d) = \log(1 + d\beta_{m'k}\mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{s}_k) - d\beta_{m'k}\frac{\mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{Q}_{\mathbf{Y}_{m'}}\mathbf{Q}_{m'}^{-1}\mathbf{s}_k}{1 + d\beta_{m'k}\mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{s}_k}. \quad (6.44)$$

Taking the derivative of (6.44) and equating it to zero, we obtain

$$d^* = \frac{\mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{Q}_{\mathbf{Y}_{m'}}\mathbf{Q}_{m'}^{-1}\mathbf{s}_k - \mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{s}_k}{\beta_{m'k}(\mathbf{s}_k^H\mathbf{Q}_{m'}^{-1}\mathbf{s}_k)^2}. \quad (6.45)$$

Note that  $d^*$  is the minimizer of  $f_{k,m'}(d)$ , but need not be the minimizer of  $f_k(d)$ . To preserve the non-negativity of  $\gamma$  in (6.34), the optimal update step  $d$  is given by  $\delta = \max\{d^*, -\gamma_k\}$  and the coordinate is updated as  $\gamma_k \leftarrow \gamma_k + \delta$ . Using (6.39), the update step  $d$  is propagated to all the sub covariance matrices  $\mathbf{Q}_m$ ,  $\forall m = 1, 2, \dots, M$ . This procedure will be done over the whole set of random permutation of variables from the set  $\{1, 2, \dots, K\}$  and we iterate the entire procedure until the cost function cannot be further reduced. The proposed algorithm is summarized in Algorithm 3. The complexity of the proposed algorithm based on dominant CSP is  $\mathcal{O}(IKML^2)$ , where  $I$  is the maximum number of iterations. The term  $\mathcal{O}(L^2)$  considers the matrix-vector multiplications in Algorithm 3.

To perform activity detection, the output from Algorithm 3 is compared against a threshold  $\gamma_k^{th}$  for each device  $k$  and is given by

$$\hat{a}_k = \begin{cases} 1, & \text{if } \hat{\gamma}_k \geq \gamma_k^{th} \\ 0, & \text{otherwise.} \end{cases} \quad (6.46)$$

Let  $\hat{\mathcal{K}}_a = \{k \mid \hat{a}_k = 1, \forall k \in [1, K]\}$  be the estimate of the set of active devices. The threshold  $\gamma_k^{th}$  is chosen to have a desired probability of miss detection and probability of false alarm performance.

### 6.6.2.3 Clustering Based Activity Detection

The activity detection in Algorithm 3 uses data from one dominant CSP per device and the performance improves when more antennas are used at the CSP [28, 47]. However, for activity detection in a cell-free network, the optimal method would be if all CSPs are contributing to the activity detection for all users, but this is unnecessarily computationally complex as mentioned in Sec. 6.6.2.1. Since only a few CSPs are close to each user, we consider a cluster of CSPs with good channels to the user. In this subsection, we consider the minimization of the cost function in (6.41), by utilizing the received signals from a cluster of dominant CSPs for each device. Towards this, we define the function which returns the set of indices of the  $T$  maximum values from the set of real numbers  $\mathcal{T}$ , as

$$\underset{\cdot}{\operatorname{indmax}}_{\cdot, T}\{\mathcal{T}\}.$$

---

**Algorithm 3** Coordinate Descend Algorithm for estimating  $\gamma$ 


---

**Input:** Observations  $\mathbf{Y}_m, \forall m = 1, 2, \dots, M, \beta_{mk}, \forall m = 1, 2, \dots, M, k = 1, 2, \dots, K$

**Initialize:**  $\mathbf{Q}_m^{-1} = \sigma^{-2} \mathbf{I}_L, \forall m = 1, 2, \dots, M, \hat{\gamma}^0 = \mathbf{0}_K$

```

1: Compute  $\mathbf{Q}_{\mathbf{Y}_m} = \frac{1}{N} \mathbf{Y}_m \mathbf{Y}_m^H, \forall m = 1, 2, \dots, M$ 
2: for  $i = 1, 2, \dots, I$  do
3:   Select an index set  $\mathcal{K}$  from the random permutation of set  $\{1, 2, \dots, K\}$ 
4:   for  $k \in \mathcal{K}$  do
5:     Find the strongest link or CSP for device  $k$ , i.e.,
6:      $m' = \operatorname{argmax}_m \{\beta_{mk}\}$ 
7:      $\delta = \max \left\{ \frac{\mathbf{s}_k^H \mathbf{Q}_{m'}^{-1} \mathbf{Q}_{\mathbf{Y}_{m'}} \mathbf{Q}_{m'}^{-1} \mathbf{s}_k - \mathbf{s}_k^H \mathbf{Q}_{m'}^{-1} \mathbf{s}_k}{\beta_{m'k} (\mathbf{s}_k^H \mathbf{Q}_{m'}^{-1} \mathbf{s}_k)^2}, -\hat{\gamma}_k \right\}$ 
8:      $\hat{\gamma}_k^i = \hat{\gamma}_k^{i-1} + \delta$ 
9:     for  $m = 1, 2, \dots, M$  do
10:       $\mathbf{Q}_m^{-1} \leftarrow \mathbf{Q}_m^{-1} - \delta \frac{\beta_{mk} \mathbf{Q}_m^{-1} \mathbf{s}_k \mathbf{s}_k^H \mathbf{Q}_m^{-1}}{1 + \delta \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k}$ 
11:    end for
12:  end for
13:  if  $f(\hat{\gamma}^i) \geq f(\hat{\gamma}^{i-1})$  then
14:     $\hat{\gamma} = \hat{\gamma}^{i-1}$ 
15:    break
16:  end if
17:   $\hat{\gamma} = \hat{\gamma}^i$ 
18: end for
    return  $\hat{\gamma}$ 

```

---

Note that the above function reduces to  $\operatorname{argmax}$ , when  $T = 1$ .

Let

$$\mathcal{M}_k = \operatorname{indmax}_{m,T} \{\beta_{mk}\}, \quad (6.47)$$

be the cluster of  $T < M$  dominant CSPs of the device  $k$ . For  $m \in \mathcal{M}_k$ , define

$$a_m = \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k \quad (6.48)$$

$$b_m = \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{Q}_{\mathbf{Y}_m} \mathbf{Q}_m^{-1} \mathbf{s}_k. \quad (6.49)$$

To minimize the cost function (6.41) by utilizing the signals from the  $T$  dominant CSPs for the user  $k$ , we redefine the cost function as

$$f_{k,T}(d) = \sum_{m \in \mathcal{M}_k} \left( \log(1 + da_m) - \frac{db_m}{1 + da_m} \right). \quad (6.50)$$

Taking the derivative of (6.50) with respect to  $d$ , yields

$$f'_{k,T}(d) = \sum_{m \in \mathcal{M}_k} \frac{a_m}{1 + da_m} + \frac{b_m}{(1 + da_m)^2}. \quad (6.51)$$

Equating (6.51) to zero yields

$$\sum_{m \in \mathcal{M}_k} \left( ((a_m + b_m) + a_m^2 d) \prod_{m' \in \mathcal{M}_k \setminus \{m\}} (1 + 2a_{m'} d + a_{m'}^2 d^2) \right) = 0 \quad (6.52)$$

---

**Algorithm 4** Clustering based coordinate descend algorithm for estimating  $\gamma$ 


---

**Input:** Observations  $\mathbf{Y}_m, \forall m = 1, 2, \dots, M, \beta_{mk}, \forall m = 1, 2, \dots, M, k = 1, 2, \dots, K$

**Initialize:**  $\mathbf{Q}_m^{-1} = \sigma^{-2} \mathbf{I}_L, \forall m = 1, 2, \dots, M, \hat{\gamma}^0 = \mathbf{0}_K$

```

1: Compute  $\mathbf{Q}_{\mathbf{Y}_m} = \frac{1}{N} \mathbf{Y}_m \mathbf{Y}_m^H, \forall m = 1, 2, \dots, M.$ 
2: Compute  $\mathcal{M}_k = \text{indmax}_{m,T} \{\beta_{mk}\} \forall k = 1, 2, \dots, K.$ 
3: for  $i = 1, 2, \dots, I$  do
4:   Select an index set  $\mathcal{K}$  from the random permutation of set  $\{1, 2, \dots, K\}$ 
5:   for  $k \in \mathcal{K}$  do
6:     for  $m \in \mathcal{M}_k$  do
7:       Compute  $a_m = \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k$  and
8:  $b_m = \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{Q}_{\mathbf{Y}_m} \mathbf{Q}_m^{-1} \mathbf{s}_k$ 
9:     end for
10:    Solve the polynomial equation
11:  $f'_{k,T}(d) = \sum_{m \in \mathcal{M}_k} \left( (a_m + b_m) + a_m^2 d \right) \cdot \prod_{m' \in \mathcal{M}_k \setminus \{m\}} (1 + 2a_{m'} d + a_{m'}^2 d^2) = 0$ 
12:    Compute  $\mathcal{D} = \{d : f'_{k,T}(d) = 0, \Im(d) = 0, \Re(d) \geq -\gamma_k\} \cup \{-\gamma_k\}$ 
13:    Let  $f_{k,T}(d) = \sum_{m \in \mathcal{M}_k} \left( \log(1 + da_m) - \frac{db_m}{1+da_m} \right).$ 
14: Compute  $\delta = \text{argmin}_{d \in \mathcal{D}} f_{k,T}(d).$ 
15:  $\hat{\gamma}_k^i = \hat{\gamma}_k^{i-1} + \delta$ 
16:   for  $m = 1, 2, \dots, M$  do
17:      $\mathbf{Q}_m^{-1} \leftarrow \mathbf{Q}_m^{-1} - \delta \frac{\beta_{mk} \mathbf{Q}_m^{-1} \mathbf{s}_k \mathbf{s}_k^H \mathbf{Q}_m^{-1}}{1 + \delta \beta_{mk} \mathbf{s}_k^H \mathbf{Q}_m^{-1} \mathbf{s}_k}$ 
18:   end for
19: end for
20: if  $f(\hat{\gamma}^i) \geq f(\hat{\gamma}^{i-1})$  then
21:    $\hat{\gamma} = \hat{\gamma}^{i-1}$ 
22:   break
23: end if
24:  $\hat{\gamma} = \hat{\gamma}^i$ 
25: end for
   return  $\hat{\gamma}$ 

```

---

which is a polynomial equation in  $d$  of degree  $2T - 1$ . Let

$$\mathcal{D} = \{d : f'_{k,T}(d) = 0, \Im(d) = 0, \Re(d) \geq -\gamma_k\} \cup \{-\gamma_k\}, \quad (6.53)$$

be the set of real roots of (6.52) and compute

$$\delta = \text{argmin}_{d \in \mathcal{D}} f_{k,T}(d). \quad (6.54)$$

The value  $-\gamma_k$  is added to the set  $\mathcal{D}$  to preserve the positivity of  $\gamma$  in (6.34) and the coordinate is updated as  $\gamma_k \leftarrow \gamma_k + \delta$ . The updating of sub-covariance blocks is carried out as explained in Sec. 6.6.2.2. The proposed algorithm is outlined in Algorithm 4 and the activity detection can be performed using (6.46).

When  $T = 1$ , the clustering based algorithm reduces to Algorithm 3. For  $T = 2$ , we have degree 3 polynomial equation in (6.52) and the roots can be solved in closed form [48]. For  $T > 2$ , we have polynomials of degree 5 and higher and there exists no closed form solutions for the roots [8]. For  $T > 2$ , the approximate roots of the polynomial in (6.52) can be found by finding the eigen values

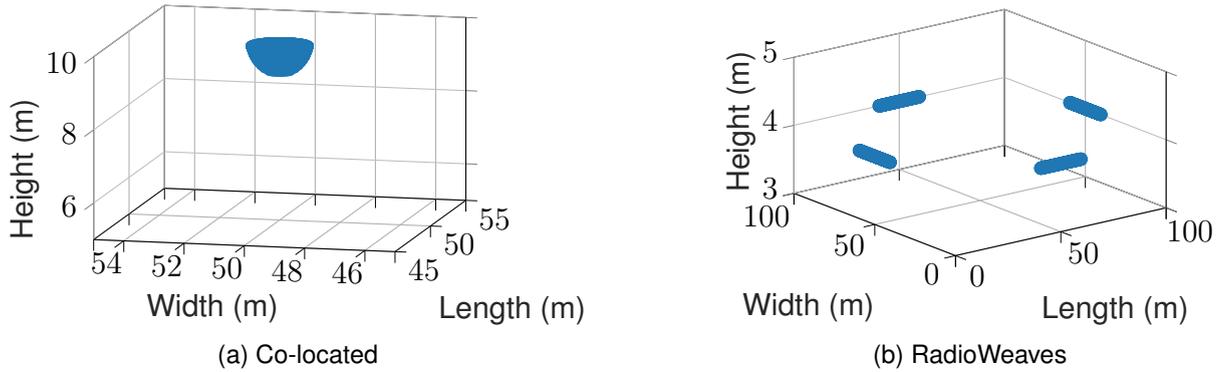


Figure 6.6: Two different antenna deployments: (a) co-located deployment in the shape of Candelabrum and (b) RadioWeaves deployment with 4 ULAs on the walls.

Table 6.3: Simulation Parameters

Parameter	Symbol	Value
Frequency of operation	$f$	2 GHz
Bandwidth		20 MHz
Temperature		300 K
Number of transmit antennas	$M$	64
Number of users	$K$	100
Transmit power		$10 \mu\text{W}$
Noise figure		7 dB
Activation probability	$\epsilon$	0.1
Signature sequence length	$L$	10
Room specifications		$100 \text{ m} \times 100 \text{ m} \times 10 \text{ m}$

of the companion matrix formed using the coefficients [69, Ch. 6] and the computation complexity is  $\mathcal{O}(T^3)$  [5]. Thus the overall complexity of the Algorithm 4 is  $\mathcal{O}(IK(TL^2 + T^3 + ML^2))$ . The term  $T^3$  corresponds to the complexity for finding the coefficients of (6.52), which can be computed using  $2T$  point convolution. The term  $TL^2$  and  $ML^2$  corresponds to the complexities associated with computation of coefficients  $a_m$ ,  $b_m$  and updating of covariance matrices, respectively.

### 6.6.3 Simulation Results

We consider two deployment strategies, co-located and RadioWeaves deployments. For co-located reference case, we consider a candelabrum type deployment on the ceiling of the wall as shown in Fig. 6.6a. The antennas are pointing in every corner in the room and thus, this shape guarantees that we have coverage at every part of the room. For a RadioWeaves deployment, we consider 4 ULAs on the walls of the room, as shown in Figure 6.6b.

The simulation parameters are given in Table 6.3. The impact of antenna deployment topology of the activity detection performance in a grant-free scenario is plotted in Figure 6.7. It can be seen that the RadioWeaves deployment performs much better than co-located deployment, owing to the fact that the spatial resolution between the users is improved by using RadioWeaves [43].

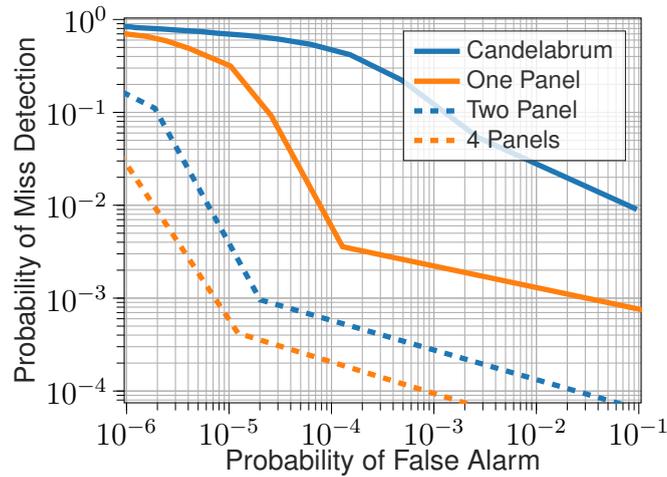


Figure 6.7: Grant Free activity detection performance in  $100 \text{ m} \times 100 \text{ m} \times 10 \text{ m}$  room with clustering based activity detection.

## 6.7 Conclusion

In this chapter grant-free and grant-based access techniques are investigated for RadioWeaves networks. The grant-based access methods are studied for URLLC, while several grant-free protocols are proposed to support a high number of devices accessing the network. One of the algorithms uses approximate message passing in a distributed setting. Another method exploits partially available CSI, obtained by a noisy channel estimate or from a previous channel estimate. Lastly, the impact of the antenna deployment on a grant-free access technique is studied, where we demonstrate that the RadioWeaves scenario performs much better than a co-located setup due to the high spatial diversity of the infrastructure.

# Chapter 7

## Conclusion

RadioWeaves considers an unprecedented number of resources embedded into the surrounding infrastructure. Consequently, enormous performance gains can be obtained, if the complexity of such a network can be managed. In this deliverable, the initial access stage –prior to and during communication– is studied for RadioWeaves.

The initial access procedures are categorised into three distinct phases, i.e., network set-up, de-registered and registered mode. We propose a frequency carrier synchronisation protocol to perform (partially) coherently processing of geographically distributed antenna arrays, or more general contact service points (CSPs). This must be done during the network-setup and later periodically to maintain coherent. After the network-setup procedure, energy neutral (EN) devices need to be powered for the first time and network-specific information needs to be acquired prior to any form of communication. To do so, we propose an initial powering protocol and explored ways to provide coverage during this de-register mode operation. Following this, the device is known to the network, i.e., in registered mode. As the RadioWeaves infrastructure has an abundance of resources available, clever scheduling of these resources and user equipment (UE) grouping must be considered. As a first study, we have derived a joint grouping and routing optimisation algorithm to route the data for downlink communication to sets of UEs. This was further generalised into a framework, which dynamically orchestrates federations serving different applications running on the UEs. This framework takes into account the application requirements, RadioWeaves infrastructure, current resource loads and fairness to optimise resource allocation. This framework will be extended during the progress of the other work packages based on their findings. After setting up the network, registration and resource allocation, medium resources need to be acquired in order to allow for communication. There, two approaches are considered: grant-based and grant-free access. Grant-based techniques are explored focusing on ultra-reliable low-latency communications (URLLC). To support a large amount of simultaneous random access requests, we proposed several grant-free access techniques.

The developed techniques, algorithms and frameworks will be i) further extended and improved based on the results of the other work packages and ii) adopted by works in other work packages.

# Bibliography

- [1] *GSM Recommendation 05.01: Physical Layer on Radio Path*. ETSI, 1998.
- [2] 3GPP Technical Specifications 38.212, Technical Specification Group Radio Access Network, NR (Release 17), v17.0.0, 2022.
- [3] Ericsson AB. R1-1720954, IMT-2020 self-evaluation: Radio Network Energy Performance, 3GPP TSG-RAN WG1 #91, Reno, US, December 2017.
- [4] Omid Abari, Hariharan Rahul, Dina Katabi, and Mondira Pant. Airshare: Distributed coherent transmission made seamless. In *IEEE Conference on Computer Communications (INFOCOM)*, pages 1742–1750. IEEE, 2015.
- [5] Oliver Aberth. Iteration methods for finding all zeros of a polynomial simultaneously. *Mathematics of computation*, 27(122):339–344, 1973.
- [6] Ashfaq Ahmed, Arafat Al-Dweik, Youssef Iraqi, Husameldin Mukhtar, Muhammad Naeem, and Ekram Hossain. Hybrid automatic repeat request (HARQ) in wireless communications systems and standards: A contemporary survey. *IEEE Communications Surveys & Tutorials*, 23(4):2711–2752, 2021.
- [7] Martin Aleksandrov and Toby Walsh. Online fair division: A survey. In *Proceedings of the AAI Conference on Artificial Intelligence*, volume 34, pages 13557–13562, 2020.
- [8] Valerij Borisovič Alekseev. *Abel's Theorem in Problems and Solutions: Based on the lectures of Professor VI Arnold*. Springer Science & Business Media, 2004.
- [9] Georgios Amanatidis, Georgios Birmpas, Aris Filos-Ratsikas, and Alexandros A Voudouris. Fair division of indivisible goods: A survey. *arXiv preprint arXiv:2202.07551*, 2022.
- [10] Arjun Anand, Gustavo De Veciana, and Sanjay Shakkottai. Joint scheduling of URLLC and eMBB traffic in 5G wireless networks. *IEEE/ACM Transactions on Networking*, 28(2):477–490, 2020.
- [11] Chidambaram Annamalai, Christos Kalaitzis, and Ola Svensson. Combinatorial algorithm for restricted max-min fair allocation. *ACM Transactions on Algorithms (TALG)*, 13(3):1–28, 2017.
- [12] Arash Asadpour and Amin Saberi. An approximation algorithm for max-min fair allocation of indivisible goods. *SIAM Journal on Computing*, 39(7):2970–2989, 2010.
- [13] Amin Azari, Mustafa Ozger, and Cicek Cavdar. Risk-aware resource allocation for URLLC: Challenges and strategies with machine learning. *IEEE Communications Magazine*, 57(3):42–48, 2019.
- [14] Haris Aziz. Developments in multi-agent fair allocation. In *Proceedings of the AAI Conference on Artificial Intelligence*, volume 34, pages 13563–13568, 2020.
- [15] Haris Aziz, Peter Biro, Jérôme Lang, Julien Lesca, and Jérôme Monnot. Optimal Reallocation under Additive and Ordinal Preferences. In *2016 International Conference on Autonomous Agents And Multiagent Systems*, pages 402–410, 2016.
- [16] Jianan Bai, Zheng Chen, and Erik G Larsson. Multi-agent policy optimization for pilot selection in delay-constrained grant-free multiple access. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pages 1477–1481. IEEE, 2021.

- [17] Jianan Bai and Erik G. Larsson. Activity Detection in Distributed MIMO: Distributed AMP via Likelihood Ratio Fusion. *IEEE Wireless Communications Letters*, to appear, 2022.
- [18] Jianan Bai, Hao Song, Yang Yi, and Lingjia Liu. Multiagent reinforcement learning meets random access in massive cellular internet of things. *IEEE Internet of Things Journal*, 8(24):17417–17428, 2021.
- [19] Horia Vlad Balan, Ryan Rogalin, Antonios Michaloliakos, Konstantinos Psounis, and Giuseppe Caire. AirSync: Enabling distributed multiuser MIMO with full spatial multiplexing. *IEEE/ACM Transactions on Networking*, 21(6):1681–1695, Dec. 2013.
- [20] Nikhil Bansal and Maxim Sviridenko. The santa claus problem. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 31–40, 2006.
- [21] S Bouveret, K Cechlárová, E Elkind, A Igarashi, and D Peters. Fair division of a graph. pages 135–141. AAAI Press/International Joint Conferences on Artificial Intelligence, 2017.
- [22] Sylvain Bouveret and Michel Lemaître. Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems*, 30(2):259–290, 2016.
- [23] Chesney Buyle, Bert Cox, Daan Delabie, Benjamin Deutschmann, Thomas Wilding, Maximilian Graber, Klaus Witrissal, and Ulrich Mühlmann. System design study for energy-neutral devices interacting with the RadioWeaves infrastructure. Deliverable ICT-52-2020 / D4.1, REINDEER project, 2022.
- [24] Gilles Callebaut, Guus Leenders, Jarne Van Mulders, Geoffrey Ottoy, Lieven De Strycker, and Liesbet Van der Perre. The Art of Designing Remote IoT Devices—Technologies and Strategies for a Long Battery Life. *Sensors*, 21(3), 2021.
- [25] Gilles Callebaut, William Tärneberg, Liesbet Van der Perre, and Emma Fitzgerald. Dynamic Federations for 6G Cell-Free Networking: Concepts and Terminology. In *2022 IEEE 23rd International Workshop on Signal Processing Advances in Wireless Communication (SPAWC) (IEEE SPAWC 2022)*, Oulu, Finland, July 2022.
- [26] Gilles Callebaut, Liesbet Van der Perre, and François Rottenberg. Grant-Free Random Access in Massive MIMO for Static Low-Power IoT Nodes. pages 46–51, 2021.
- [27] Tuğrul Çavdar and Erkan Güler. Hympro: a hybrid multi-path routing algorithm for cognitive radio ad hoc networks. *Telecommunication Systems*, 69(1):61–76, 2018.
- [28] Zhilin Chen, Foad Sahrabi, Ya-Feng Liu, and Wei Yu. Covariance based joint activity and data detection for massive random access with massive MIMO. In *IEEE Int. Conf. on Commun. (ICC)*, pages 1–6, 2019.
- [29] Yann Chevaleyre, Ulle Endriss, Sylvia Estivie, Nicolas Maudet, et al. Multiagent resource allocation with k-additive utility functions. In *Proceedings of the DIMACS-LAMSADE workshop on computer science and decision theory*, volume 3, pages 83–100, 2004.
- [30] Jinho Choi. Sliding Network Coding for URLLC. *IEEE Transactions on Wireless Communications*, 2021.
- [31] Erik Dahlman, Stefan Parkvall, and Johan Skold. *5G NR: The next generation wireless access technology*. Academic Press, 2020.
- [32] Lei Deng, Danzhou Wu, Zilong Liu, Yijin Zhang, and Yunghsiang S Han. Reinforcement learning for improved random access in delay-constrained heterogeneous wireless networks. *arXiv preprint arXiv:2205.02057*, 2022.
- [33] Apostolos Destounis, Dimitrios Tsilimantou, Mérouane Debbah, and Georgios S Paschos. Learn2mac: On-line learning multiple access for urllc applications. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 1–6. IEEE, 2019.
- [34] Benjamin J. B. Deutschmann, Thomas Wilding, Erik G. Larsson, and Klaus Witrissal. Location-based Initial Access for Wireless Power Transfer with Physically Large Arrays. In *WS08 IEEE ICC 2022 Workshop on Synergies of communication, localization, and sensing towards 6G (WS08 ICC'22 Workshop - ComLS-6G)*, Seoul, Korea (South), May 2022.

- [35] Seda Doğan, Armed Tusha, and Hüseyin Arslan. NOMA with index modulation for uplink URLLC through grant-free access. *IEEE Journal of Selected Topics in Signal Processing*, 13(6):1249–1257, 2019.
- [36] David L Donoho, Arian Maleki, and Andrea Montanari. Message-passing algorithms for compressed sensing. *Proceedings of the National Academy of Sciences*, 106(45):18914–18919, 2009.
- [37] Ove Edfors, Romulo Brazil, Herbert Petautschnig, Gilles Callebaut, Thomas Feys, Liesbet Van der Perre, Erik G. Larsson, Ove Edfors, Emma Fitzgerald, Liang Liu, Jesus Rodriguez Sanchez, William Tärneberg, Pal Frenger, Benjamin Deutschmann, Thomas Wilding, and Klaus Witrisal. Initial assessment of architectures and hardware resources for a RadioWeaves infrastructure, Reindeer deliverable D2.1. Technical report, January 2022.
- [38] Medhat Elsayed and Melike Erol-Kantarci. AI-enabled radio resource allocation in 5G for URLLC and eMBB users. In *2019 IEEE 2nd 5G World Forum (5GWF)*, pages 590–595. IEEE, 2019.
- [39] Ali A Esswie and Klaus I Pedersen. Opportunistic spatial preemptive scheduling for URLLC and eMBB coexistence in multi-user 5G networks. *Ieee Access*, 6:38451–38463, 2018.
- [40] Juan Francisco Esteban and Martina Truskaller. Use case-driven specifications and technical requirements and initial channel model. Deliverable ICT-52-2020 / D1.1, REINDEER project, Sep 2021. unpublished.
- [41] Alexander Fengler, Saeid Haghghatshoar, Peter Jung, and Giuseppe Caire. Non-Bayesian activity detection, large-scale fading coefficient estimation, and unsourced random access with a massive MIMO receiver. *IEEE Transactions on Information Theory*, 67(5):2925–2951, 2021.
- [42] Unnikrishnan Kunnath Ganesan, Emil Björnson, and Erik G Larsson. Clustering-Based Activity Detection Algorithms for Grant-Free Random Access in Cell-Free Massive MIMO. *IEEE Transactions on Communications*, 69(11):7520–7530, 2021.
- [43] Unnikrishnan Kunnath Ganesan, Emil Björnson, and Erik G. Larsson. RadioWeaves for Extreme Spatial Multiplexing in Indoor Environments. In *54th Asilomar Conference on Signals, Systems, and Computers*, pages 1007–1011. IEEE, 2020.
- [44] Mohammad Ghodsi, Mohamad Latifian, Arman Mohammadi, Sadra Moradian, and Masoud Seddighin. Rent division among groups. In *International Conference on Combinatorial Optimization and Applications*, pages 577–591. Springer, 2018.
- [45] Pascal Giard, Gabi Sarkis, Camille Leroux, Claude Thibeault, and Warren J Gross. Low-latency software polar decoders. *Journal of Signal Processing Systems*, 90(5):761–775, 2018.
- [46] Mangqing Guo and M. Cenk Gursoy. Joint Activity Detection and Channel Estimation in Cell-Free Massive MIMO Networks With Massive Connectivity. *IEEE Trans. on Commun.*, 70(1):317–331, 2022.
- [47] Saeid Haghghatshoar, Peter Jung, and Giuseppe Caire. Improved scaling law for activity detection in massive MIMO systems. In *IEEE International Symposium on Information Theory (ISIT)*, pages 381–385, 2018.
- [48] GC Holmes. The use of hyperbolic cosines in solving cubic polynomials. *The Mathematical Gazette*, 86(507):473–477, 2002.
- [49] Roger A Horn and Charles R Johnson. *Matrix analysis*. Cambridge University Press, 2012.
- [50] Defeng Huang and Khaled Ben Letaief. Carrier frequency offset estimation for OFDM systems using null subcarriers. *IEEE Transactions on Communications*, 54(5):813–823, May 2006.
- [51] Rui Huang, Vincent WS Wong, and Robert Schober. Throughput optimization for grant-free multiple access with multiagent deep reinforcement learning. *IEEE Transactions on Wireless Communications*, 20(1):228–242, 2020.
- [52] Han Seung Jang, Hoon Lee, Tony QS Quek, and Hyundong Shin. Deep learning-based cellular random access framework. *IEEE Transactions on Wireless Communications*, 20(11):7503–7518, 2021.
- [53] Andrew Jay. Santa Claus, Machine Scheduling and Bipartite Hypergraphs. Master’s thesis, University of Waterloo, 2020.

- [54] Nan Jiang, Yansha Deng, and Arumugam Nallanathan. Traffic prediction and random access control optimization: Learning and non-learning-based approaches. *IEEE Communications Magazine*, 59(3):16–22, 2021.
- [55] Zhiyuan Jiang, Andrei Marinescu, Luiz A DaSilva, Sheng Zhou, and Zhisheng Niu. Scalable multi-agent learning for situationally-aware multiple-access and grant-free transmissions. In *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2019.
- [56] Ali Karimi, Klaus I Pedersen, Nurul Huda Mahmood, Guillermo Pocovi, and Preben Mogensen. Efficient low complexity packet scheduling algorithm for mixed URLLC and eMBB traffic in 5G. In *2019 IEEE 89th Vehicular Technology Conference (VTC2019-Spring)*, pages 1–6. IEEE, 2019.
- [57] Ian Kash, Ariel D Procaccia, and Nisarg Shah. No agent left behind: Dynamic fair division of multiple resources. *Journal of Artificial Intelligence Research*, 51:579–603, 2014.
- [58] Salil Kashyap, Emil Bjornson, and Erik G. Larsson. On the Feasibility of Wireless Energy Transfer Using Massive Antenna Arrays. *IEEE Transactions on Wireless Communications*, 15(5):3466–3480, may 2016.
- [59] Steven M Kay. *Fundamentals of statistical signal processing*. Prentice Hall PTR, 1993.
- [60] Radosław Kotaba, Carles Navarro Manchón, Tommaso Balercia, and Petar Popovski. How URLLC can Benefit from NOMA-based Retransmissions. *IEEE Transactions on Wireless Communications*, 20(3):1684–1699, 2020.
- [61] Maria Kyropoulou, Warut Suksompong, and Alexandros A Voudouris. Almost envy-freeness in group resource allocation. *Theoretical Computer Science*, 841:110–123, 2020.
- [62] Federico Librino and Paolo Santi. The Complexity–Performance Tradeoff in Resource Allocation for URLLC Exploiting Dynamic CSI. *IEEE Internet of Things Journal*, 8(17):13266–13277, 2021.
- [63] Liang Liu and Wei Yu. Massive connectivity with massive MIMO-Part I: Device activity detection and channel estimation. *IEEE Trans. on Signal Processing*, 66(11):2933–2946, 2018.
- [64] Stuart Lloyd. Least squares quantization in PCM. *IEEE transactions on information theory*, 28(2):129–137, 1982.
- [65] J MacQueen. Classification and analysis of multivariate observations. In *5th Berkeley Symp. Math. Statist. Probability*, pages 281–297, 1967.
- [66] Behrooz Makki, Tommy Svensson, Giuseppe Caire, and Michele Zorzi. Fast harq over finite blocklength codes: A technique for low-latency reliable communication. *IEEE Transactions on Wireless Communications*, 18(1):194–209, 2018.
- [67] Pasin Manurangsi and Warut Suksompong. Asymptotic existence of fair divisions for groups. *Mathematical Social Sciences*, 89:100–108, 2017.
- [68] T. L. Marzetta, E. G. Larsson, H. Yang, and H. Q. Ngo. *Fundamentals of Massive MIMO*. Cambridge University Press, 2016.
- [69] JM McNamee. *Numerical Methods for Roots of Polynomials-Part I*. Elsevier, 2007.
- [70] Ebubekir Memisoglu, Ertugrul Basar, and Huseyin Arslan. Fading-aligned OFDM with index modulation for mMTC services. *Physical Communication*, 35:100680, 2019.
- [71] J Meredith. Study on channel model for frequency spectrum above 6 ghz. Technical report, 2016.
- [72] Paul H Moose. A technique for orthogonal frequency division multiplexing frequency offset correction. *IEEE Transactions on Communications*, 42(10):2908–2914, Oct. 1994.
- [73] Faisal Nadeem, Mahyar Shirvanimoghaddam, Yonghui Li, and Branka Vucetic. Delay-sensitive noma-harq for short packet communications. *Entropy*, 23(7):880, 2021.
- [74] Klaus I Pedersen, Saeed R Khosravirad, Gilberto Berardinelli, and Frank Frederiksen. Rethink hybrid automatic repeat request design for 5G: Five configurable enhancements. *IEEE Wireless Communications*, 24(6):154–160, 2017.

- [75] S. O. Petersson. Power-Efficient Beam Pattern Synthesis via Dual Polarization Beamforming. In *14th European Conference on Antennas and Propagation (EuCAP)*, 2020.
- [76] Vladimir L Petrović, Dragomir M El Mezeni, and Andreja Radošević. Flexible 5G new radio LDPC encoder optimized for high hardware usage efficiency. *Electronics*, 10(9):1106, 2021.
- [77] Ryan Rogalin, Ozgun Y Bursalioglu, Haralabos Papadopoulos, Giuseppe Caire, Andreas F Molisch, Antonios Michaloliakos, Vlad Balan, and Konstantinos Psounis. Scalable synchronization and reciprocity calibration for distributed multiuser MIMO. *IEEE Transactions on Wireless Communications*, 13(4):1815–1831, Apr. 2014.
- [78] Cenk Sahin, Lingjia Liu, Erik Perrins, and Liangping Ma. Delay-sensitive communications over IR-HARQ: Modulation, coding latency, and reliability. *IEEE Journal on Selected Areas in Communications*, 37(4):749–764, 2019.
- [79] Timothy M Schmidl and Donald C Cox. Robust frequency and timing synchronization for OFDM. *IEEE Transactions on Communications*, 45(12):1613–1621, Dec. 1997.
- [80] Erel Segal-Halevi and Shmuel Nitzan. Envy-free cake-cutting among families. *arXiv preprint arXiv:1607.01517*, 2016.
- [81] Erel Segal-Halevi and Warut Suksompong. Democratic fair allocation of indivisible goods. *Artificial Intelligence*, 277:103167, 2019.
- [82] Arti Sharma and Mohammad Salim. Polar Code Appropriateness for Ultra-Reliable and Low-Latency Use Cases of 5G Systems. *Int. J. Networked Distributed Comput.*, 7(3):93–99, 2019.
- [83] Jack Sherman and Winifred J Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *The Annals of Mathematical Statistics*, 21(1):124–127, 1950.
- [84] Nils Strodthoff, Barış Göktepe, Thomas Schierl, Cornelius Hellge, and Wojciech Samek. Enhanced machine learning techniques for early HARQ feedback prediction in 5G. *IEEE Journal on Selected Areas in Communications*, 37(11):2573–2587, 2019.
- [85] Catherine A Sugar and Gareth M James. Finding the number of clusters in a dataset: An information-theoretic approach. *Journal of the American Statistical Association*, 98(463):750–763, 2003.
- [86] Warut Suksompong. Approximate maximin shares for groups of agents. *Mathematical Social Sciences*, 92:40–47, 2018.
- [87] Warut Suksompong. Constraints in fair division. *ACM SIGecom Exchanges*, 19(2):46–61, 2021.
- [88] Gordon J Sutton, Jie Zeng, Ren Ping Liu, Wei Ni, Diep N Nguyen, Beeshanga A Jayawickrama, Xiaojing Huang, Mehran Abolhasan, and Zhang Zhang. Enabling ultra-reliable and low-latency communications through unlicensed spectrum. *IEEE Network*, 32(2):70–77, 2018.
- [89] Gordon J Sutton, Jie Zeng, Ren Ping Liu, Wei Ni, Diep N Nguyen, Beeshanga A Jayawickrama, Xiaojing Huang, Mehran Abolhasan, Zhang Zhang, Eryk Dutkiewicz, et al. Enabling technologies for ultra-reliable and low latency communications: From PHY and MAC layer perspectives. *IEEE Communications Surveys & Tutorials*, 21(3):2488–2524, 2019.
- [90] James Joseph Sylvester. On the relation between the minor determinants of linearly equivalent quadratic functions. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 1(4):295–305, 1851.
- [91] Chance Tarver, Matthew Tonnemacher, Hao Chen, Jianzhong Zhang, and Joseph R Cavallaro. GPU-Based, LDPC Decoding for 5G and Beyond. *IEEE Open Journal of Circuits and Systems*, 2:278–290, 2021.
- [92] Yunke Tian, Yong Bai, and Dake Liu. Low-Latency QC-LDPC Encoder Design for 5G NR. *Sensors*, 21(18):6266, 2021.
- [93] Anas Tom, Alphan Şahin, and Hüseyin Arslan. Suppressing alignment: Joint PAPR and out-of-band power leakage reduction for OFDM-based systems. *IEEE Transactions on Communications*, 64(3):1100–1109, 2015.

- 
- [94] Chenwei Wang, Ozgun Y Bursalioglu, Haralabos Papadopoulos, and Giuseppe Caire. On-the-fly large-scale channel-gain estimation for massive antenna-array base stations. In *IEEE International Conference on Communications (ICC)*, pages 1–6, 2018.
- [95] Di Wu, Yong Zeng, Shi Jin, and Rui Zhang. Environment-Aware and Training-Free Beam Alignment for mmWave Massive MIMO via Channel Knowledge Map. In *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–7. IEEE, jun 2021.
- [96] Jiazhen Zhang, Xiaofeng Tao, Huici Wu, Ning Zhang, and Xuefei Zhang. Deep reinforcement learning for throughput improvement of the uplink grant-free noma system. *IEEE Internet of Things Journal*, 7(7):6369–6379, 2020.

# Appendix A

## Appendix

## A.1 Derivation of Distributed approximate message passing (AMP) Algorithm

We include the detailed derivations associated with the distributed device activity detection procedure in Algorithm 1 here.

### A.1.1 AMP with Minimum mean square error (MMSE) Denoiser and Likelihood-Ratio Test

By initialising  $\mathbf{Z}^0 = \mathbf{Y}$  and  $\hat{\mathbf{X}}^0 = \mathbf{O}_{K \times M_{\text{tot}}}$ , the AMP iteration  $t \in \{0, 1, \dots\}$  for complex-valued signals is [63],

$$\hat{\mathbf{x}}_k^{t+1} = \mathbf{g}_t(\underbrace{(\mathbf{Z}^t)^T \boldsymbol{\phi}_k^* + \hat{\mathbf{x}}_k^t}_{\triangleq \boldsymbol{\xi}_k^t}), \quad \forall k \in \mathcal{K}, \quad (\text{A.1})$$

$$\mathbf{Z}^{t+1} = \mathbf{Y} - \boldsymbol{\Phi} \hat{\mathbf{X}}^{t+1} + \frac{1}{L} \mathbf{Z}^t \sum_{k \in \mathcal{K}} \mathbf{g}'_t(\boldsymbol{\xi}_k^t), \quad (\text{A.2})$$

where  $\hat{\mathbf{X}}^t = [\hat{\mathbf{x}}_1^t, \dots, \hat{\mathbf{x}}_K^t]^T$ . Here,  $\mathbf{g}_t(\cdot) : \mathbb{C}^{M_{\text{tot}}} \rightarrow \mathbb{C}^{M_{\text{tot}}}$  is the denoiser and  $\mathbf{g}'_t(\boldsymbol{\xi})$  represents its Jacobian at  $\boldsymbol{\xi}$ .

As demonstrated in the state evolution analysis [36], under some mild conditions and in the large-system limit,  $\boldsymbol{\xi}_k^t$  behaves like a Gaussian-noise corrupted version of  $\mathbf{x}_k$ , i.e.,

$$\boldsymbol{\xi}_k^t \sim \mathbf{x}_k + \mathcal{CN}(\mathbf{0}, \boldsymbol{\Sigma}^t). \quad (\text{A.3})$$

In (A.3),  $\boldsymbol{\Sigma}^t$  is referred to as the *state*; this state evolves by

$$\boldsymbol{\Sigma}^{t+1} = \sigma^2 \mathbf{I} + \frac{1}{L} \sum_{k \in \mathcal{K}} \mathbb{E}[(\mathbf{g}_t(\mathbf{x}_k + \mathbf{v}^t) - \mathbf{x}_k)(\mathbf{g}_t(\mathbf{x}_k + \mathbf{v}^t) - \mathbf{x}_k)^H], \quad (\text{A.4})$$

where  $\mathbf{v}^t$  is a random vector with distribution  $\mathcal{CN}(\mathbf{0}, \boldsymbol{\Sigma}^t)$  which is independent of  $\mathbf{x}_k$ , and the expectation is taken over the joint distribution of  $\mathbf{x}_k$  and  $\mathbf{v}^t$ . The initial state is given by

$$\boldsymbol{\Sigma}^0 = \sigma^2 \mathbf{I} + \frac{1}{L} \sum_{k \in \mathcal{K}} \mathbf{R}_k. \quad (\text{A.5})$$

The MMSE denoiser is given by the MMSE estimate of  $\mathbf{x}_k$  given  $\boldsymbol{\xi}_k^t$ ,

$$\mathbf{g}_t(\boldsymbol{\xi}_k^t) = \mathbb{E}[\mathbf{x}_k | \boldsymbol{\xi}_k^t] = \theta_k^t(\boldsymbol{\xi}_k^t) \cdot \boldsymbol{\Psi}_k^t \boldsymbol{\xi}_k^t, \quad (\text{A.6})$$

where

$$\theta_k^t(\boldsymbol{\xi}) = \left( 1 + \frac{1 - \epsilon_k}{\epsilon_k} \frac{|\mathbf{R}_k + \boldsymbol{\Sigma}^t|}{|\boldsymbol{\Sigma}^t|} \exp(-\boldsymbol{\xi}^H \boldsymbol{\Omega}_k^t \boldsymbol{\xi}) \right)^{-1}, \quad (\text{A.7})$$

$$\boldsymbol{\Psi}_k^t = \mathbf{R}_k (\mathbf{R}_k + \boldsymbol{\Sigma}^t)^{-1}, \quad (\text{A.8})$$

$$\boldsymbol{\Omega}_k^t = (\boldsymbol{\Sigma}^t)^{-1} - (\mathbf{R}_k + \boldsymbol{\Sigma}^t)^{-1}. \quad (\text{A.9})$$

The support recovery problem is equivalent to the detection of the non-zero entries in the binary vector  $\mathbf{a}$ . To determine the value of  $a_k$ , we consider the binary hypothesis test

$$\mathcal{H}_0 : a_k = 0 \quad \text{and} \quad \mathcal{H}_1 : a_k = 1. \quad (\text{A.10})$$

The likelihood-ratio test (LRT) is given by<sup>1</sup>

$$\ell_k \triangleq \frac{p(\boldsymbol{\xi}_k | a_k = 0)}{p(\boldsymbol{\xi}_k | a_k = 1)} \underset{\mathcal{H}_1}{\overset{\mathcal{H}_0}{\geq}} \gamma, \quad (\text{A.11})$$

where  $\gamma > 0$  is the decision threshold. According to (A.3), the likelihood-ratio can be written as

$$\ell_k = \frac{\mathcal{CN}(\boldsymbol{\xi}_k | \mathbf{0}, \boldsymbol{\Sigma})}{\mathcal{CN}(\boldsymbol{\xi}_k | \mathbf{0}, \mathbf{R}_k + \boldsymbol{\Sigma})} = \frac{|\mathbf{R}_k + \boldsymbol{\Sigma}|}{|\boldsymbol{\Sigma}|} \exp(-\boldsymbol{\xi}_k^H \boldsymbol{\Omega}_k \boldsymbol{\xi}_k). \quad (\text{A.12})$$

Notice that (A.7) can be rewritten as  $\theta_k^t = (1 + \frac{1-\epsilon_k}{\epsilon_k} \ell_k^t)^{-1}$ .

With a large number of antennas,  $M_{\text{tot}}$ , a naive implementation of the AMP algorithm has two major drawbacks: 1) calculating the determinants and inverting the  $M_{\text{tot}} \times M_{\text{tot}}$  matrices in (A.7), (A.8) and (A.9) can be computationally demanding; 2) sending the  $L \times M_{\text{tot}}$ -dimensional matrix  $\mathbf{Y}$  requires high fronthaul capacity.

## A.1.2 Covariance Structure in the AMP State Evolution

The received signal model in distributed MIMO, see (6.3), has a special property: the covariance matrices  $\{\mathbf{R}_k\}$  are block-diagonal. In the following theorem, we show that during the state evolution in AMP, the states maintain the same block-diagonal structure during all iterations.

**Theorem 1.** *Assume that  $\{\mathbf{R}_k\}$  have a block-diagonal structure:  $\mathbf{R}_n = \text{bdiag}(\mathbf{R}_{n1}, \dots, \mathbf{R}_{nK})$ . By using the MMSE denoiser in (A.6), the state  $\boldsymbol{\Sigma}^t$  in the state evolution (A.4) stays as a block-diagonal matrix with the same structure for each block, i.e.,  $\boldsymbol{\Sigma}^t = \text{bdiag}(\boldsymbol{\Sigma}_1^t, \dots, \boldsymbol{\Sigma}_N^t)$ , for all  $t$ .*

*Proof.* We prove this theorem by induction. First, when the covariance matrices  $\{\mathbf{R}_k\}$  share a block-diagonal structure, the initial state  $\boldsymbol{\Sigma}^0$  in (A.5) has the same block-diagonal structure. Then, assuming that  $\boldsymbol{\Sigma}^t$  stays in this structure, we show that  $\boldsymbol{\Sigma}^{t+1}$  has the same structure.

**Definition 1.** *(Partially Odd or Even Function) A function  $f : \mathbb{R}^M \rightarrow \mathbb{R}$  is partially odd or even in indices  $\mathcal{I} \subset \mathcal{M} = \{1, \dots, M\}$  if  $f(\eta_{\mathcal{I}}(\mathbf{x})) = -f(\mathbf{x})$  or  $f(\eta_{\mathcal{I}}(\mathbf{x})) = f(\mathbf{x})$ , respectively. Here,  $\eta_{\mathcal{I}}(\cdot)$  is an element-wise operator with  $[\eta_{\mathcal{I}}(\mathbf{x})]_i$  equals to  $-x_i$  for  $i \in \mathcal{I}$ , and  $x_i$  otherwise.*

An arbitrary expectation term in the summand of the second term in the state evolution (A.4) can be written as

$$\begin{aligned} & \mathbb{E} \left[ (\mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}) - \mathbf{x}) (\mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}) - \mathbf{x})^H \right] \\ &= \mathbb{E} \left[ \mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}) (\mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}))^H \right] + \mathbb{E} [\mathbf{x} \mathbf{x}^H] \\ & \quad - \mathbb{E} [\mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}) \mathbf{x}^H] - \mathbb{E} [\mathbf{x} (\mathbf{g}(\mathbf{x} + \mathbf{v}, \boldsymbol{\Sigma}))^H]. \end{aligned} \quad (\text{A.13})$$

According to (A.6), the first term equals

$$\boldsymbol{\Psi} \underbrace{\mathbb{E} [\theta(\mathbf{x} + \mathbf{v})^2 (\mathbf{x} + \mathbf{v}) (\mathbf{x} + \mathbf{v})^H]}_{\triangleq \mathbf{Q}} \boldsymbol{\Psi}^H, \quad (\text{A.14})$$

<sup>1</sup>For brevity, we henceforth omit the iteration index  $t$  in the superscripts.

where  $\theta(\cdot)$  is defined in (A.7). By denoting as  $p_x(\mathbf{x})$  and  $p_v(\mathbf{v})$  the density functions of  $\mathbf{x}$  and  $\mathbf{v}$ , respectively, the  $(i, j)$ -the element of  $\mathbf{Q}$  is given by

$$[\mathbf{Q}]_{i,j} = \int_{\mathbf{x}, \mathbf{v}} \underbrace{(x_i + v_i)(x_j + v_j)^* \theta(\mathbf{x} + \mathbf{v})^2 p_x(\mathbf{x}) p_v(\mathbf{v})}_{\triangleq f_{i,j}(\mathbf{x}, \mathbf{y})}. \quad (\text{A.15})$$

Denote by  $\mathcal{M}_k$  the row (column) indices corresponding to the  $k$ -th diagonal block. Then  $f_{i,j}(\mathbf{x}, \mathbf{y})$  is partially odd in  $\mathcal{M}_k$  if  $i \in \mathcal{M}_k$  and  $j \notin \mathcal{M}_k$ , or  $i \notin \mathcal{M}_k$  and  $j \in \mathcal{M}_k$ , and partially even in  $\mathcal{M}_k$  otherwise. That is,  $[\mathbf{Q}]_{i,j} = 0$  if the indices  $i$  and  $j$  are not in the same diagonal block. This means that  $\mathbf{Q}$  is also block-diagonal with the same structure as  $\{\mathbf{R}_n\}$ . Then, the first term in (A.13), which equals to  $\Psi \mathbf{Q} \Psi^H$ , keeps the same block-diagonal structure.

By using similar arguments, one can show that the remaining terms have the same structure. □

According to Theorem 1, the inversion of the  $M_{\text{tot}} \times M_{\text{tot}}$  matrices in (A.8) and (A.9) can be performed by inverting their diagonal blocks, which are of dimension  $M \times M$ .<sup>2</sup>

When the channel vector from device  $k$  to CSP  $n$  is modeled by i.i.d. Rayleigh fading, the channel covariance matrix becomes  $\tilde{\mathbf{R}}_{nk} = \beta_{nk} \mathbf{I}_M$ . Correspondingly, the effective channel  $\mathbf{h}_k$  from device  $k$  to all CSPs has the distribution  $\mathcal{CN}(\mathbf{0}, \mathbf{R}_k)$  with  $\mathbf{R}_k = \text{bdiag}(\kappa_{1k} \mathbf{I}_M, \dots, \kappa_{Nk} \mathbf{I}_M)$ . The following corollary can be viewed as a generalization of [63, Theorem 1] to the scenario of distributed MIMO.

**Corollary 1.** *Assume that  $\{\mathbf{R}_k\}$  have the diagonal structure  $\mathbf{R}_k = \text{bdiag}(\kappa_{1k} \mathbf{I}, \dots, \kappa_{Nk} \mathbf{I})$ . By using the MMSE denoiser in (A.6), the state  $\Sigma^t$  stays as a scaled identity matrix for each diagonal block, i.e.,  $\Sigma^t = \text{bdiag}(\tau_1^t \mathbf{I}, \dots, \tau_N^t \mathbf{I})$ , for all  $t$ .*

*Proof.* By setting the size of the diagonal blocks in Theorem 1 to one, we conclude that the state  $\Sigma^t$  stays as a diagonal matrix. Then, by using the symmetry, we conclude that the elements corresponding to the same CSP are equal. □

In the i.i.d. Rayleigh case, the calculations of all matrix inversions and determinants simplify to scalar operations.

### A.1.3 Distributed Activity Detection

Since by Theorem 1,  $\Sigma$  and  $\Omega_n$  are both block-diagonal, we can rewrite the likelihood-ratio in (A.12) as

$$\ell_k = \prod_{n \in \mathcal{N}} \underbrace{\frac{|\mathbf{R}_{nk} + \Sigma_n|}{|\Sigma_n|} \exp(-\boldsymbol{\xi}_{nk}^H \Omega_{nk} \boldsymbol{\xi}_{nk})}_{\triangleq \ell_{nk}}. \quad (\text{A.16})$$

Equivalently, the LLR is

$$\log \ell_k = \sum_{n \in \mathcal{N}} \log \ell_{nk}, \quad (\text{A.17})$$

<sup>2</sup>For simplicity, we assume that all CSPs have the same number of antennas. The algorithm, however, can be easily modified to support arbitrary numbers of antennas.

where

$$\log \ell_{nk} = \log \frac{|\mathbf{R}_{nk} + \Sigma_n|}{|\Sigma_n|} - \boldsymbol{\xi}_{nk}^H \boldsymbol{\Omega}_{nk} \boldsymbol{\xi}_{nk}. \quad (\text{A.18})$$

Here,  $\Sigma_n$  and  $\boldsymbol{\Omega}_{nk}$  are the  $n$ -th diagonal blocks of  $\Sigma$  and  $\boldsymbol{\Omega}_k$ , respectively, and  $\boldsymbol{\xi}_{nk}$  is the corresponding subvector of  $\boldsymbol{\xi}_k$ . This means that the LLR  $\log \ell_k$  can be written as the sum of  $\{\log \ell_{nk}\}$ , which can be interpreted as the *local* LLRs after *coherently* processing the received signals at each CSP.

In the special case of i.i.d. Rayleigh fading, the LLR can be further simplified into

$$\log \ell_{nk} = M \log \left( 1 + \frac{\kappa_{nk}}{\tau_n} \right) - \frac{\kappa_{nk} \|\boldsymbol{\xi}_{nk}\|^2}{\tau_n (\kappa_{nk} + \tau_n)}, \quad (\text{A.19})$$

where the quantity  $\kappa_{nk}/\tau_n$  can be interpreted as the signal-to-noise ratio (SNR).

Inspired by the factorization in (A.16), we propose a distributed approach to activity detection in distributed MIMO. The procedure is as follows: each CSP runs the AMP algorithm locally by using only the received signal  $\mathbf{Y}_k$  and sends the local LLR  $\log \hat{\ell}_{nk}$  to the aggregator. Then, the aggregator computes the LLR  $\log \hat{\ell}_k = \sum_{n \in \mathcal{N}} \log \hat{\ell}_{nk}$  for activity detection.

## A.2 Device Activity Detection with Partial CSI in RadioWeaves

The signal received at antenna  $m$  of CSP  $n$  and for pilot symbol  $t$ , is given by

$$y_{m,n,t} = \sum_{k=0}^{K-1} (g_{m,n,k} + \epsilon_{m,n,k} \lambda_{n,k}) e^{j\phi_{n,k}} s_{k,t} \gamma_k + w_{m,n,t},$$

where  $s_{k,t}$  is the preamble symbol of user  $k$  at time  $t$ ,  $w_{m,n,t}$  is additive white Gaussian noise, which is assumed to be independently and identically distributed (i.i.d.). Stacking the observations at antenna  $m$  of CSP  $n$  over time, gives

$$\mathbf{y}_{m,n} = \sum_{k=0}^{K-1} (g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}} + \epsilon_{m,n,k} \lambda_{n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}}) + \mathbf{w}_{m,n}.$$

One can note that, for a fixed/deterministic vector  $\boldsymbol{\gamma}$  and  $\boldsymbol{\phi}_n$ ,  $\mathbf{y}_{m,n} | \boldsymbol{\gamma}, \boldsymbol{\phi}_n$  has a circularly symmetric Gaussian distribution with mean  $\sum_{k=0}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}}$  and covariance matrix

$$\mathbf{C}_n = \sum_k \lambda_{n,k}^2 \gamma_k^2 \mathbf{s}_k \mathbf{s}_k^H + \sigma_w^2 \mathbf{I}.$$

Hence, the log-likelihood of this observation vector is

$$\begin{aligned} f_{m,n}(\boldsymbol{\gamma}, \boldsymbol{\phi}) &= -\ln(|\mathbf{C}_n|) - T \ln(\pi) \\ &\quad - \left( \mathbf{y}_{m,n} - \sum_{k=0}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}} \right)^H \mathbf{C}_n^{-1} \left( \mathbf{y}_{m,n} - \sum_{k=0}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}} \right). \end{aligned}$$

Note that the different  $\mathbf{y}_{m,n}$  are uncorrelated. Then the log-likelihood of the aggregated observations at all antennas becomes the sum

$$\begin{aligned} f(\boldsymbol{\gamma}, \boldsymbol{\phi}) &= p(\mathbf{y} | \boldsymbol{\gamma}, \boldsymbol{\phi}) = -\sum_n M_n \ln(|\mathbf{C}_n|) - \sum_n M_n T \ln(\pi) \\ &\quad - \sum_{m,n} \left( \mathbf{y}_{m,n} - \sum_{k=0}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}} \right)^H \mathbf{C}_n^{-1} \left( \mathbf{y}_{m,n} - \sum_{k=0}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}} \right). \end{aligned}$$

Building upon the work presented in Section 6.5.1, we here also maximise the likelihood by using an iterative approach and update the parameters related to each user one at a time. At a given iteration, only parameters related to user  $k'$  are updated:  $\gamma_{k'}$  and  $\phi_{n,k'}$ . We can define

$$\mathbf{y}_{m,n,k'} = \mathbf{y}_{m,n} - \sum_{k,k \neq k'}^{K-1} g_{m,n,k} \mathbf{s}_k \gamma_k e^{j\phi_{n,k}}.$$

The likelihood becomes

$$\begin{aligned} f(\gamma_{k'}, \phi_{k'}) &= - \sum_n M_n \ln(|\mathbf{C}_n|) - \sum_n M_n T \ln(\pi) \\ &\quad - \sum_{m,n} (\mathbf{y}_{m,n,k'} - g_{m,n,k'} \mathbf{s}_{k'} \gamma_{k'} e^{j\phi_{n,k'}})^H \mathbf{C}_n^{-1} (\mathbf{y}_{m,n,k'} - g_{m,n,k'} \mathbf{s}_{k'} \gamma_{k'} e^{j\phi_{n,k'}}), \end{aligned}$$

after some manipulations,

$$\begin{aligned} \frac{df}{d\phi_{n',k'}} = 0 &\leftrightarrow \phi_{n',k'} = \angle \sum_m g_{m,n',k'}^* \mathbf{s}_{k'}^H \gamma_{k'} \mathbf{C}_n^{-1} \mathbf{y}_{m,n',k'} \\ &= \angle \mathbf{s}_{k'}^H \mathbf{C}_n^{-1} \sum_m g_{m,n',k'}^* \mathbf{y}_{m,n',k'}. \end{aligned}$$

Inserting this optimal value in the objective function makes the dependence in  $\phi_{n',k'}$  vanish and gives

$$\begin{aligned} f(\gamma_{k'}) &= - \sum_n M_n \ln(|\mathbf{C}_n|) - \sum_n M_n T \ln(\pi) \\ &\quad - \sum_{m,n} (\mathbf{y}_{m,n,k'} - g_{m,n,k'} \mathbf{s}_{k'} \gamma_{k'} e^{j\phi_{n',k'}})^H \mathbf{C}_n^{-1} (\mathbf{y}_{m,n,k'} - g_{m,n,k'} \mathbf{s}_{k'} \gamma_{k'} e^{j\phi_{n',k'}}) \\ &= - \sum_n M_n \ln(|\mathbf{C}_n|) - \sum_n M_n T \ln(\pi) \\ &\quad - \sum_{m,n} \mathbf{y}_{m,n,k'}^H \mathbf{C}_n^{-1} \mathbf{y}_{m,n,k'} - \gamma_{k'}^2 \mathbf{s}_{k'}^H \sum_n \mathbf{C}_n^{-1} \mathbf{s}_{k'} \sum_m |g_{m,n,k'}|^2 \\ &\quad + 2\gamma_{k'} \sum_n \left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_n^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|. \end{aligned}$$

The dependence in  $\phi_{n',k'}$  disappears but we have a complex modulus of something that depends on  $\gamma_{k'}$ , which is not differentiable. Let us define

$$\mathbf{C}_{n,-k'} = \sum_{k \setminus k'} \lambda_{n,k}^2 \gamma_k^2 \mathbf{s}_k \mathbf{s}_k^H + \sigma_w^2 \mathbf{I},$$

which does not depend on  $\gamma_{k'}$ . Let us apply the Sherman-Morrison formula

$$\begin{aligned}
(\mathbf{A} + \mathbf{u}\mathbf{v}^\top)^{-1} &= \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^\top\mathbf{A}^{-1}}{1 + \mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}} \\
\mathbf{C}_n^{-1} &= \mathbf{C}_{n,-k'}^{-1} - \frac{\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}\mathbf{s}_{k'}^H\mathbf{C}_{n,-k'}^{-1}\gamma_{k'}^2\lambda_{n',k'}^2}{1 + \mathbf{s}_{k'}^H\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}\gamma_{k'}^2\lambda_{n',k'}^2} \\
(\mathbf{A} + \mathbf{u}\mathbf{v}^\top)^{-1}\mathbf{u} &= \mathbf{A}^{-1}\mathbf{u} - \frac{\mathbf{A}^{-1}\mathbf{u}\mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}}{1 + \mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}} \\
&= \frac{\mathbf{A}^{-1}\mathbf{u}}{1 + \mathbf{v}^\top\mathbf{A}^{-1}\mathbf{u}} \\
\mathbf{C}_n^{-1}\mathbf{s}_{k'}\gamma_{k'}\lambda_{n',k'} &= \frac{\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}\gamma_{k'}\lambda_{n',k'}}{1 + \mathbf{s}_{k'}^H\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}\gamma_{k'}^2\lambda_{n',k'}^2} \\
\mathbf{C}_n^{-1}\mathbf{s}_{k'} &= \frac{\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}}{1 + \mathbf{s}_{k'}^H\mathbf{C}_{n,-k'}^{-1}\mathbf{s}_{k'}\gamma_{k'}^2\lambda_{n',k'}^2}
\end{aligned}$$

We thus find

$$2\gamma_{k'} \sum_n \left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_n^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right| = 2\gamma_{k'} \sum_n \frac{\left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2},$$

and the objective function then becomes

$$\begin{aligned}
f(\gamma_{k'}) &= - \sum_n M_n \ln(|\mathbf{C}_n|) - \sum_n M_n T \ln(\pi) \\
&\quad - \sum_{m,n} \mathbf{y}_{m,n,k'}^H \mathbf{C}_n^{-1} \mathbf{y}_{m,n,k'} - \gamma_{k'}^2 \mathbf{s}_{k'}^H \sum_n \mathbf{C}_n^{-1} \mathbf{s}_{k'} \sum_m |g_{m,n,k'}|^2 \\
&\quad + 2\gamma_{k'} \sum_n \frac{\left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2}.
\end{aligned}$$

The differentiability problem is solved. Now we apply the Sherman-Morrison formula on each inverse to put in evidence the dependence in  $\gamma_{k'}$ . We omit terms that do not depend on it and will

vanish after taking the derivative.

$$\tilde{f}(\gamma_{k'}) = - \sum_n M_n \ln(|\mathbf{C}_n|) \quad (\text{A.20})$$

$$+ \sum_{m,n} \mathbf{y}_{m,n,k'}^H \frac{\mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'}^H \mathbf{s}_{k'} \mathbf{C}_{n,-k'}^{-1} \gamma_{k'}^2 \lambda_{n',k'}^2}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} \mathbf{y}_{m,n,k'} \quad (\text{A.21})$$

$$- \gamma_{k'}^2 \sum_n \frac{\mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'}}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} \sum_m |g_{m,n,k'}|^2 \quad (\text{A.22})$$

$$+ 2 \frac{\left| \sum_{m,n} \mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} \gamma_{k'} \quad (\text{A.23})$$

$$= - \sum_n M_n \ln(|\mathbf{C}_n|) \quad (\text{A.24})$$

$$+ \sum_{m,n} \frac{|\mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'}|^2 \lambda_{n',k'}^2 \gamma_{k'}^2}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} - \gamma_{k'}^2 \frac{\mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'}}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} \sum_{m,n} |g_{m,n,k'}|^2 \quad (\text{A.25})$$

$$+ 2 \gamma_{k'} \sum_n \frac{\left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|}{1 + \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \gamma_{k'}^2 \lambda_{n',k'}^2} \quad (\text{A.26})$$

$$= - \sum_n M_n \ln(|\mathbf{C}_n|) + \sum_n \frac{\alpha_n \gamma_{k'}^2 + \beta_n \gamma_{k'}}{1 + \delta_n \gamma_{k'}^2} \quad (\text{A.27})$$

where we defined

$$\alpha_n = \sum_m |\mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'}|^2 \lambda_{n',k'}^2 - \sum_n \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \sum_m |g_{m,n,k'}|^2$$

$$\beta_n = 2 \left| \sum_m \mathbf{y}_{m,n,k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} g_{m,n,k'} \right|$$

$$\delta_n = \mathbf{s}_{k'}^H \mathbf{C}_{n,-k'}^{-1} \mathbf{s}_{k'} \lambda_{n',k'}^2.$$

$$\begin{aligned} \frac{d\tilde{f}}{d\gamma_{k'}} &= \sum_n -2M_n \lambda_{n,k}^2 \mathbf{s}_k^H \mathbf{C}_n^{-1} \mathbf{s}_k \gamma_{k'} + \frac{(2\alpha_n \gamma_{k'} + \beta_n)(1 + \delta_n \gamma_{k'}^2) - (\alpha_n \gamma_{k'}^2 + \beta_n \gamma_{k'}) 2\delta_n \gamma_{k'}}{(1 + \delta_n \gamma_{k'}^2)^2} \\ &= \sum_n -2M_n \lambda_{n,k}^2 \frac{\mathbf{s}_k^H \mathbf{C}_n^{-1} \mathbf{s}_k}{1 + \mathbf{s}_k^H \mathbf{C}_n^{-1} \mathbf{s}_k \gamma_{k'}^2 \lambda_{n,k}^2} \gamma_{k'} + \frac{(2\alpha_n \gamma_{k'} + \beta_n)(1 + \delta_n \gamma_{k'}^2) - (\alpha_n \gamma_{k'}^2 + \beta_n \gamma_{k'}) 2\delta_n \gamma_{k'}}{(1 + \delta_n \gamma_{k'}^2)^2} \\ &= \sum_n -2M_n \frac{\delta_n \gamma_{k'}}{1 + \delta_n \gamma_{k'}^2} + \frac{(2\alpha_n \gamma_{k'} + \beta_n)(1 + \delta_n \gamma_{k'}^2) - (\alpha_n \gamma_{k'}^2 + \beta_n \gamma_{k'}) 2\delta_n \gamma_{k'}}{(1 + \delta_n \gamma_{k'}^2)^2} \end{aligned}$$

Setting the derivative to zero gives

$$\begin{aligned} \frac{d\tilde{f}}{d\gamma_{k'}} &= 0 \\ 0 &= \sum_n \frac{-2M_n \delta_n \gamma_{k'} (1 + \delta_n \gamma_{k'}^2) + (2\alpha_n \gamma_{k'} + \beta_n)(1 + \delta_n \gamma_{k'}^2) - (\alpha_n \gamma_{k'}^2 + \beta_n \gamma_{k'}) 2\delta_n \gamma_{k'}}{(1 + \delta_n \gamma_{k'}^2)^2} \\ 0 &= \sum_n \frac{-\gamma_{k'}^3 2M_n \delta_n^2 - \gamma_{k'}^2 \beta_n \delta_n + \gamma_{k'} (-2M_n \delta_n + 2\alpha_n) + \beta_n}{(1 + \delta_n \gamma_{k'}^2)^2}. \end{aligned}$$