Techniques for Decentralized and Dynamic Resource Allocation

by

Lorenzo Ferrari

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved October 2017 by the
Graduate Supervisory Committee:

Anna Scaglione, Chair
Daniel Bliss
Lei Ying
Martin Reisslein

ARIZONA STATE UNIVERSITY

December 2017

ABSTRACT

This thesis investigates three different resource allocation problems, aiming to achieve two common goals: i) adaptivity to a fast-changing environment, ii) distribution of the computation tasks to achieve a favorable solution. The motivation for this work relies on the modern-era proliferation of sensors and devices, in the Data Acquisition Systems (DAS) layer of the Internet of Things (IoT) architecture. To avoid congestion and enable low-latency services, limits have to be imposed on the amount of decisions that can be centralized (i.e. solved in the "cloud") and/or amount of control information that devices can exchange. This has been the motivation to develop i) a lightweight PHY Layer protocol for time synchronization and scheduling in Wireless Sensor Networks (WSNs), ii) an adaptive receiver that enables Sub-Nyquist sampling, for efficient spectrum sensing at high frequencies, and iii) an SDN-scheme for resource-sharing across different technologies and operators, to harmoniously and holistically respond to fluctuations in demands at the eNodeB' s layer.

The proposed solution for time synchronization and scheduling is a new protocol, called PulseSS, which is completely event-driven and is inspired by biological networks. The results on convergence and accuracy for locally connected networks, presented in this thesis, constitute the theoretical foundation for the protocol in terms of performance guarantee. The derived limits provided guidelines for ad-hoc solutions in the actual implementation of the protocol.

The proposed receiver for Compressive Spectrum Sensing (CSS) aims at tackling the noise folding phenomenon, e.g., the accumulation of noise from different subbands that are folded, prior to sampling and baseband processing, when an analog front-end aliasing mixer is utilized. The sensing phase design has been conducted via a utility maximization approach, thus the scheme derived has been called Cognitive Utility Maximization Multiple Access (CUMMA). The framework described in the last part of the thesis is inspired by stochastic network optimization tools and dynamics. While convergence of the proposed approach remains an open problem, the numerical results here presented suggest the capability of the algorithm to handle traffic fluctuations across operators, while respecting different time and economic constraints. The scheme has been named Decomposition of Infrastructure-based Dynamic Resource Allocation (DIDRA).

DEDICATION

to my father Luciano and my mother Cosetta

ACKNOWLEDGMENTS

This dissertation would have not been possible without the tireless and constant support of my advisor Anna Scaglione. You have always believed in me, and your combination of passion and knowledge has been an invaluable inspiration, throughout these years together. I also want to thank my committee for their constructive feedbacks and the instructive conversations we shared. During my PhD, I experienced the excitement for learning and trying to solve a research problem or, in some cases, create a new problem and help my advisor writing a research proposal. Some days however, this excitement was held down by doubts surrounding my future, homesickness and a variety of negative feelings. In this bittersweet mix of emotions, my labmates (Eran, Kari, Mahdi, Reinhard, To, Raksha, Bita, Sissi, Mahnoosh, Saeed, Gina, Masood, Marko) played a determinant role in keeping me grounded and always being there for a laugh, a beer after work or a word of wisdom during birthday celebrations (thanks Eran!). A special thank goes to my longtime friends in Italy (Lo, Andre, Vi, Sci, Le e Aba): every single accomplishment in my life has been made unforgettable by your presence. This thesis is dedicated to my parents for their love and their constant surrounding presence, even when we are thousands miles away: no emotion in my life can be compared to hearing from you ”Siamo orgogliosi di te!” (We’re proud of you!). Finally, to the woman who makes me fall asleep with a smile on my face, excited for the journey of life we will be able to take together. I cannot wait to be called your husband, I love you Maia.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

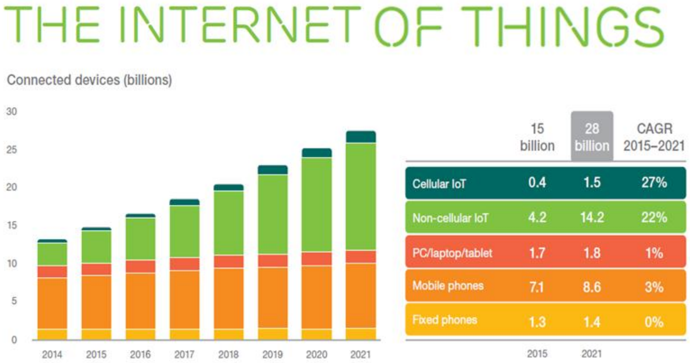
Figure Page

LIST OF ALGORITHMS

Algorithm Page

CHAPTER 1
INTRODUCTION

This thesis discusses new techniques to address the problem of scheduling in wireless networks. In 2013, the number of devices with a wireless connection has overcome the world population and it is indisputable such a dramatic increase will continue in the next years (see Fig.1.0.1). The broadly used, and often inflated, term to represent the proliferation of connectivity is "Internet of Things" (IoT). The main benefit of the IoT are tied with making it easy to overcome the limits of myopic decisions in everyday life activities and services, by creating an ecosystem of data that relate to many interdependent human activities and processes, that can be used to compute socially optimum decisions. Naturally, these two trends feed each other success: the easier it is to gather and process data collected over wide areas by a myriad of disparate sensors, the greater and broader are the benefits of deploying sensors and controllers everywhere. What is still not clear is how the IoT, introducing the human-to-thing and thing-to-thing communications, will change human behaviour and social interactions: how deep the Internet of Things will change our habits and everyday lives? The answer for this question is probably going to be revealed in the next ten



	15 billion	28 billion	CAGR 2015–2021
Cellular IoT	0.4	1.5	27%
Non-cellular IoT	4.2	14.2	22%
PC/laptop/tablet	1.7	1.8	1%
Mobile phones	7.1	8.6	3%
Fixed phones	1.3	1.4	0%
	2015	2021	

Figure 1.0.1: Prediction on the Number of Connected Devices in the early future (Source: Forbes)

years and many experts expect the success of *Internet of Things* to be application driven but, as for any other technology, the standardization process is mandatory to enable any kind of investment in Research and Development. Oleg Logvinov, chair, IEEE P2413 WG and director, special assignments, Industrial and Power Conversion Division, ST Microelectronics:

> “This is one of those rare watershed moments, where the world gets to witness the dawn of a new age of technology innovation. The growing intersection of smart technologies and high-speed communications will produce profound, positive changes in nearly every aspect of our daily lives. Smart cities, homes, and workplaces, e-health, resilient, self-healing power grids, digital factories, cleaner transportation, immersive entertainment: these are just a few areas of economic opportunity that would benefit from the increased interoperability and portability that a standardized IoT architecture brings.”

We can view the IoT network services as being partitioned into two relatively flat layers (Fig.1.0.2): 1) a top layer including the cloud computing systems; 2) beneath, a layer that includes what we refer to as Data Acquisition Systems (DAS).

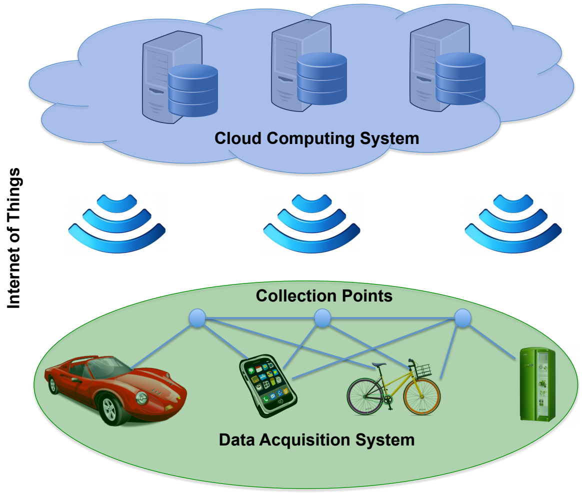


Figure 1.0.2: Layer structure of future Internet of Things

Both layers are distributed: the cloud comprises servers farms and a high speed network backbone, while DAS either rely on cellular or WiFi services or are essentially instantiations of Sensor Networks (SNs). The DAS are primarily based on IEEE.11 or IEEE.15 standards, and leverage on over thirty years of research on wireless packet-switched ad-hoc networks. Another useful medium for IoT applications is the often overlooked power-line communication medium, which also has a variety of standards available as G3-PLC, PRIME, HomePlug Green PHY, HomePlug AV2, the standards IEEE 1901/1901.2 and ITU-T G.hn/G.hnem. The question for communication and network researchers is if the SN models available for DAS today cover well the DAS needs of all possible applications and what else can fall in the IoT model. Specifically, what are the key performance metrics that could expand the applicability of the IoTs so that data processing can migrate to the cloud instead of being limited to a single, possibly complex and multi-modal sensor platform. As the number of "requests" to access a communication resource (whether a spatial frequency in massive MIMO, a time block in a TDMA system, a frequency band in an FDMA system or any combination of these) increases, it becomes of fundamental importance to solve the scheduling problem of finite resources as efficiently as possible. For cellular communications, the emerging interest in Cloud Radio Access Network (C-RAN) to centralize scheduling decisions is motivated by a potential lower deployment cost of base stations and a gain in degrees of freedom for the scheduling optimization: however moving the processing entirely in the cloud becomes unfeasible in dense areas and the communication overload in the front-haul (from the base station to the C-RAN) unsustainable. Additionally, there could still be scenarios were a C-RAN is not accessible for prohibiting costs or for complete absence of the C-RAN itself. Furthermore, Ultra-Reliable Low-Latency Communications (URLLC) would still prefer the processing and the computation to occur closer to the UE's and this motivates the research in decentralized solutions for allocating resources.

In Wireless SNs (WSNs), the two most widely used protocol are WiFi and Zigbee, which are inherently asynchronous and resolve medium access conflicts either through centralized management or through decentralized Carrier Sensing Multiple Access (CSMA) methods. The need for network synchronization, on the other hand, is typically addressed through out-of-band control channels, like the Global Positioning System (GPS), or through an application layer protocol such as the the Precision Time Protocol (PTP). While GPS and PTP can provide time information for synchronous sensing, they do not solve the communication scheduling problem which can be difficult, especially in large mesh networks of sensors. This motivates the study of a protocol that can simultaneously provide Synchronization and TDMA Scheduling of communication activities in clustered WSNs. The first chapter of this thesis presents the PulseSS protocol, where synchronization and scheduling are attained via updates of local variables: the main contribution of this thesis is the study of convergence properties of such updates, whereas practical implementation details and a testbed evaluation can be found in my former colleague Reinhard Gentz's PhD thesis [Gentz(2017)] and publications [Gentz *et al.*(2016), Gentz *et al.*(2015)].

The second chapter is dedicated to a Cognitive scheme to opportunistically exploit transmission opportunities in the spectrum, in an FDMA-TDMA based system, namely the Cognitive Utility Maximization Multiple Access (CUMMA) Receiver. Mobile devices are increasingly been used for entertainment (gaming, video streaming). The coexistence of these networks with the IoT and Machine-to-Machine (M2M) communications means that wireless applications may quickly become starved for bandwidth. Increasing the spectrum to millimeter waves can provide the much needed linear increase in throughput, but it poses the challenge of reversing the trend of bringing high speed sampling closer and closer to the antenna. Also, for IoT, a better match would be decentralized cognitive opportunistic spectrum access rather than scheduling in a Cloud Radio Access Network. But that requires somehow overcoming the bottleneck of Nyquist sampling when sensing the large spectrum available. The problem is formulated in terms of maximizing the utility a decision-maker can accrue by exploiting transmission opportunities, while being penalized for interfering on preexistent communications. Since the number of sub-band potentially accessible can be very large, especially in the mm-Wave spectrum where 5G is going to land, it would be desirable to obtain informations from multiple sub-bands at the same time, and the infeasibility of sampling at such high frequencies calls for the use of Sub-Nyquist sampling, for which however the noise folding creates a severe limitation for the *non-coherent* detection scheme proposed. To overcome this we look at the performances of a group-testing inspired strategy, which can be seen as a very low-density sensing matrix.

In the last chapter of this thesis we look at a new optimization framework for a holistic resource allocation, across different operators and different technologies, in the backhaul layer of the network architecture. A bottleneck of modern networks is the lack, or limited presence, of statistical multiplexing (sharing) of network resources among wireless operators and technologies. The new generation of cellular communications, currently under standardization by 3GPP, has been named 5G NR-SS which stands for New Radio Shared Spectrum, to emphasize the importance of developing methods to effectively enable an harmonious coexistence of different radio access schemes and/or different Mobile Network Operators (MNOs). Although extensive research effort has been done at the PHY layer to mitigate interference, here we argue that a complimentary solution could be offered by optimization of the site where the different wireless service streams meet: the network backhaul. We built our optimization framework on top of a newly defined SDN architecture, named LayBack, and proposed by our collaborators Prof. Martin Reisslein and his former PhD student Akhilesh Thyagaturu. We combine a multi time-scale decomposition, inspired by the extensive work on Network Utility Maximization (NUM) approaches to solve resource allocation problems, with a Lyapunov-drift-plus-penalty relaxation to enforce an economic-fairness among operators and prevent them from gaming the system in the long run. The scheme proposed is named Decomposition of Infrastructure-based Dynamic Resource Allocation (DIDRA). We show, numerically, that under certain conditions, the suggested optimization can benefit the different operators minimizing the end-to-end delay experienced in uplink traffic management, with respect to a static allocation. In our approach, we explicitly consider the network latencies and the temporal constraints the resource redistributions at different levels of the architecture (Operator core network, Gateways and eNBs) needs to abide. The work here presented is a preliminary investigation: convergence of the proposed approach and incorporation of multiple constraints at different layers of the protocol stack is beyond the scope of this thesis.

CHAPTER 2
DECENTRALIZED SYNCHRONIZATION AND SCHEDULING- THE PULSESS PROTOCOL

2.1 Chapter Outline

In this chapter, we propose a new protocol, PulseSS to attain Synchronization and Scheduling in a decentralized fashion, leveraging the bio-inspired dynamics of Pulse Coupled Oscillators (PCOs). We start by giving a background on the solutions for network synchronization and research in PCOs. We then describe the two basic components of PulseSS, namely the scheduling and the synchronization rule and updates, simplifying some of the protocol details that pertain the signaling and the topology formation. This description allows to shed light on the protocols fixed points and its expected trends. We then revisit the analysis of PCO based synchronization and scheduling over locally connected networks and contribute to establish performance guarantees for the convergence of the algorithm and design guidelines for implementation of the protocol. This work constitutes the analytical foundation for the design and implementation of the PCO-based distributed scheduling protocol presented in [Gentz *et al.*(2016), Gentz(2017)] for multi-hop networks. The key-strength of the protocol is that it offers a decentralized solution for two major problems in network communications: a) clock network distribution and b) channel resource allocation. It is then natural to study the attainable clock distribution accuracy when propagation delays come into play (see Section 2.4.2) and the equivalent capacity available to each node in the network for our scheduling mechanism. In Remark 2.5.10 we also discuss the connection between our algorithm's achievable schedule and the solution of the minimum coloring graph problem.

Compared to previous research, we want to highlight that: (a) the dynamics we chose to analyze simplify the implementation both through digital as well as analog circuits [Wang *et al.*(2011), Wang and Apsel(2007b), Wang *et al.*(2009)]; (b) we do not need to assume that nodes can separate the signals fired by different nodes when the firing occurs at unison, which may happen when the nodes are close to synchrony (since the absorption property treats multiple interfering pulses as one) (c) our results hold irrespective of the initial conditions. Furthermore, the absorption property allows perfect synchronization to occur after some time t when no delay exists; and, when there are delays, it allows (thanks to the refractory period) the nodes phase difference to remain fixed and converge to a value that is bounded by the maximum sum of the propagation delays over any path in the network after some time t. In most of the competing models mentioned above, convergence occurs only asymptotically, as time goes to infinity.

The main contributions are as follows: 1) we show that, for any positive coupling strength, the synchronous state is the unique fixed point for our model (see Section 2.4) and for a 3 nodes locally-connected network, convergence to the synchronous state occurs almost surely; 2) we study the effect of propagation delays, and extrapolate the synchronization accuracy expected for more complex topologies and random delays by characterizing the set of fixed points for our model in the presence of delays; 3) we adapt the PCO-based scheduling scheme introduced in [Pagliari *et al.*(2010)] to locally connected networks and analyze its convergence in a general class of these networks, which includes both star and line networks (see Section 2.5). The simulations in Section 2.6 corroborate our claims.

2.2 Background on Network Synchronization and Scheduling

We can divide sensor network synchronization algorithms in two main classes. The first class of methods is master slave; a good example is represented by the Network Time Protocol (NTP) that requires the flooding of a message from a master node with accurate time information over the entire network [Su and Akyildiz(2005), Maróti *et al.*(2004)]. Typically the transmission time of the master is measured with an absolute time reference coming from a GPS receiver and a timestamp is contained in the payload of the message. Every node is aware of the delays due to multihop transmissions and medium access waiting time, so it is able to modify and correct properly its own lock at the reception of the message from the master. The second class of algorithms is decentralized: a good representative is the Reference Broadcast Synchronization (RBS) [Elson *et al.*(2002)] protocol, where the transmission of a reference signal to the neighborhood initializes the synchronization process, then the nodes exchange a timestamp of their reference signal reception time and compute their relative clock difference (improvements of RBS are in [Sichitiu and Veerarittiphan(2003), Römer(2001), Mock *et al.*(2000)]). An iterative average consensus protocol [Li(2004)] suggests how to calculate the clock average and perform a decentralized computation of the clock skew. The idea of establishing synchronization in a decentralized way without the presence of an external input reference signal makes the system safe from spoofing and other illegitimate intrusion tries. Providing common timing is often necessary but not sufficient to attain TDM scheduling. Basically we can divide the schemes to access the communication medium in two main categories:

- Random Access (ALOHA, CSMA and their variants)
- Centralized Scheduling (TDMA, CDMA)

Usually, due to their decentralized nature, random access schemes are suitable for environments where:

- the size of the network is **unknown**,
- the topology is **dynamically** changing,
- the traffic demands are **bursty**.

The advantage of centralized scheduling is the reliability to transmit over the medium at a high and constant rate: the downside is the need of a master node to take charge of the process and this might result in a weakness to face dynamic changes of the environment or the failure of the master node itself. These considerations have prompted research on decentralized approaches for scheduling that would merge the benefits of the two schemes. In applications with large number of sensors with slow duty-cycles Time Division Multiplexing (TDM) has advantages over other multiplexing methods, as nodes can have both transmitter and receiver in sleep mode while their wait their turn, conserving energy, therefore especially useful for battery powered devices. TDM solutions attaining an optimal allocations are NP-hard [Ramanathan(1997)] but there are several heuristic solutions for TDM scheduling, aimed at allocating regularly a portion of a time frame to each node while meeting a given criterion of fairness [Huang and Bensaou(2001)] or maximizing data throughput. Protocols such as the USAP [Young(1996)], DTSAP [Pond and Li(1989)] or FLUSH [Kim *et al.*(2007)] use a message-passing approach, while DRAND [Rhee *et al.*(2006)] and the method in [Herman and Tixeuil(2004)] formulate the time scheduling problem as an instance of graph-coloring problem. These TDM scheduling algorithms typically give the availability of global synchronization for granted, and require updating global control information (such as the nodes' ID, destination, neighbors, data rates, routing information etc.), prior to eventually assigning a portion of transmission time (or a non-conflicting color) to each user in the transmission channel. This entails overhead and reduced resilience compared to random access protocols, which are often preferred over their synchronous alternatives. A popular centralized protocol for synchronization and scheduling is WirelessHART [Foundation(2007),Lennvall *et al.*(2008)]. Published in 2007, it has archived widespread acceptance for Wireless Sensor Networks (WSNs). However there are two main drawbacks which we believe would make WirelessHART less competitive, as seen in the comparison with our protocol in Table 2.1: first scheduling in WirelessHART is centrally managed by a dedicated node called Network Manager, limiting the size of the application and introducing a single point of failure. In contrast, in our proposed protocol each cluster is managed locally, thus our solution is naturally scalable. Second, WirelessHART requires global knowledge of the network topology, whereas in our proposed protocol the nodes of each cluster will assign themselves a fair share by communicating locally within their cluster. The protocol ISE100.11a is very similar to WirelessHART and has its key differences in the network layer and above. Therefore ISE100.11a suffers from the same problems as WirelessHART: central management requiring global knowledge of the network topology.

Protocol	WirelessHART	PulseSS
Medium Access Control	Central, by the Network Manager	Decentral, each cluster operates independently from other clusters
Knowledge of global network required	Yes	No
Source of Timing	Built in, but only basic mechanics defined	Build in
Timing Provided to Sensors	Yes	Yes
Timing Accuracy	few μs per hop have been shown [Kim *et al.*(2008)]	Simulations show errors of <5ns per hop depending on Bandwidth and Transmission power
Network Layer Defined	Yes	No

Table 2.1: Comparison between WirelessHART and PulseSS

We propose an easy to deploy protocol for clustered ad-hoc networks, combining decentralized synchronization and medium access control, named the Pulse coupled Synchronization and Scheduling (PulseSS) protocol. PulseSS works in an *ad-hoc* mesh network scenario, where clusters coexist and contend for the same spectrum resources adaptively. Each cluster has a special node acting as cluster head (CH), with similar properties to the IEEE 802.11 standard: 1) transmissions are only allowed from and to the CH; 2) CH's acknowledge the reception of signals from nodes in their range, so that hidden terminals can learn about conflicts.

Previous Work on Pulse Coupled Oscillators Synchronization

In 1975 Charles Peskin introduced the pulse coupled oscillator (PCO) model to explain the synchronization of pacemaker cells in heart tissues [Peskin(1975)]. Prior to that, swarm synchronization among pulsing agents, such as pacemaker cells, was observed frequently in nature [Buck and Jology(1988)] but could not be well explained mathematically. Fifteen years later Mirollo and Strogatz in [Mirollo and Strogatz(1990)] proved that fully connected networks of PCOs with excitatory coupling and convex dynamics always converge to fire at unison, except for a measure-zero set of initial conditions. They also exhamined the case of inhibitory coupling, in which the oscillators emergent behavior turns into a uniformly spaced daisy-chain of pulsing activities among the agents, which can be viewed as a conflict-free schedule of the pulsing activities [Degesys *et al.*(2007)]. In the early 2000s several groups recognized the applicability of PCO models for network synchronization [Hong and Scaglione(2005), Campbell *et al.*(1999), Mathar and Mattfeldt(1996), Timme *et al.*(2004), Motter *et al.*(2005), Izhikevich(1999), Frigui and Rhouma(2000), Lee and Chen(2008), Barbarossa and Celano(2005), Torikai and Saito(2004), Nakano and Saito(2002)] as well as scheduling [Degesys *et al.*(2007), Pagliari *et al.*(2010)], albeit less directly for the latter. The key difference of this work is that the PulseSS interlaces the PCO signaling with the scheduling signals, allowing to naturally separate the control traffic from the data traffic. Typically, protocols using these models have a fairly simple signaling mechanism that couple the dynamics of the nodes transmission activities. In turn, these protocols help integrate the physical and the medium access control layers with network synchronization (typically application layer) activities. However, PCO synchronization does not work well if it is merged with CSMA protocols [Werner-Allen *et al.*(2005)]. Efficient implementations of the PCO protocol disable CSMA [Pagliari and Scaglione(2011)] or use a separate radio band all-together [Wang and Apsel(2007a)]. Additionally, while the convergence of PCOs to the synchronous state has been studied extensively in the literature, e.g., [Mirollo and Strogatz(1990),Kuramoto(1991),Ernst *et al.*(1995),Vreeswijk(1996), O'Keeffe *et al.*(2015)], little is known for the convergence in locally connected networks [F.Nunez *et al.*(2015), Rothkegel and Lehnertz(2014), Wang *et al.*(2012b)], especially when propagation delays come into play. The problem of establishing almost sure convergence for locally connected networks remains open, and has only been partially addressed in recent works by imposing additional assumptions on the update dynamics and the initial conditions of the oscillators' phases (see e.g [Proskurnikov and Cao(2015)] which extends the analysis in [Wang *et al.*(2012b), Nez *et al.*(2015)] for Phase Response Curves (PRC) maps of the *delay-advanced type* [Izhikevich(2007)] and references therein). In [Tyrrell *et al.*(2008)], a claim on the convergence of the synchronization for a line network was provided, but was only verified through numerical simulations. Other works, such as [Lucarelli and Wang(2004), Werner-Allen *et al.*(2005),Degesys and Nagpal(2008),Degesys *et al.*(2007),Patel *et al.*(2007)], looked at the asymptotic behavior considering very small coupling between oscillators and focused on the effect of different functions modeling the dynamics of the oscillators, which can be approximated by a continuous-time Kuramoto model [Kuramoto(1991)]. These models do not apply for scheduling algorithms [Degesys *et al.*(2007), Pagliari *et al.*(2010)] that in all regimes are known to not produce the desired emergent behavior, unless the network is fully connected.

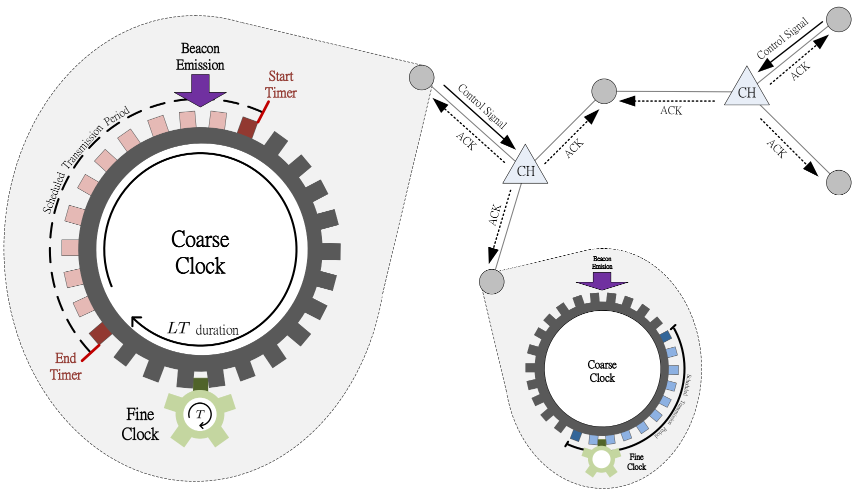


Figure 2.3.1: Illustration of the PulseSS coarse and the fine clocks maintained by each node.

Notation

Unless specified otherwise, we represent matrices and vectors with capital boldface letters (i.e., $\mathbf{B}$), we use calligraphic letters to indicate sets of nodes (i.e., $\mathcal{A}$), and the greek letters Φ and Ψ for the internal clocks of the nodes. The letter π is used to refer to an index after a permutation and the lowercase letters α, β, δ are positive parameters of the coupling equations. When not used as a node index, $j = \sqrt{-1}$. The notation t^+ refers to an instant after time t when an event triggered a clock update. We use suffixes to refer to a specific node, pair of nodes or a specific clique in Section 2.5.

2.3 Overview of the PulseSS Protocol

Let the WSN be described by the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of stationary sensor nodes and $\mathcal{E}$ (i.e., the set of edges) captures the pairs of nodes that are in range of each other. The network consists of a set of cluster heads (CHs) denoted by the set $\mathcal{C} \subset \mathcal{V}$ and a set of regular nodes $\mathcal{N} \triangleq \mathcal{V} - \mathcal{C}$ that communicates only with the CHs. For each $c \in \mathcal{C}$, we define $\mathcal{N}_c \subset \mathcal{N}$ as the set of regular (non-CH) nodes that lie within the transmission range of CH c; and, for each $v \in \mathcal{N}$, we define $\mathcal{C}_v \triangleq \{c \in \mathcal{C} : v \in \mathcal{N}_c\}$ as the set of CHs that are within the transmission range of v.

We shall assume that CHs are preassigned, such that each node has at least one CH in communication range. Nodes that have multiple CHs in communication range are referred to as *shared* (or *gateway*) *nodes.* The management of these nodes is crucial to ensure that all neighboring clusters can self-organize and attain conflict free schedules.

In PulseSS, each node maintains two local clocks, namely a *fine clock* with period T and a *coarse clock* with period LT, as illustrated in Fig. 2.3.1. Each cycle of the coarse clock is advanced by the expiration of L cycles of the fine clock and each cycle of the fine clock represents a transmission time slot of duration T. The PulseSS signaling is used to locally update the phases of both clocks, as will be explained mathematically next. These updates synchronize the phases of the fine clocks at all nodes at the slot level (see Section 2.4) and set the phases of the coarse clocks apart so as to schedule for each node a portion of the L time slots available in the frame, enabling proportional fairness and spatial reuse (see Section 2.5). These goals are achieved by having each node transmit two control signals, a preamble which we call the *start beacon* and a post-amble, called *end beacon*, meant to reach neighboring CHs. As in the 802.15.4 MAC these two signals delimit the period allotted for the two way data transmission between a node and is CH. However the beacons emissions are controlled by the regular nodes and not the CHs and there is no contention in this interval. The times of emission of these beacons governed by the local coarse clock expirations (every frame); the reception of such beacons by other nodes triggers adjustments of their own coarse clocks (and, thus, their schedules) as the CHs' corresponding acknowledgment is received. The notion of being *coupled* through an acknowledgment is new in PCO based protocols, and it is the key ingredient to attain collision avoidance.

Mathematically, let the state of the local fine clock at node $v \in \mathcal{V}$ be described by the phase variable

$$\Phi_v(t) = \frac{t}{T} + \phi_v \pmod{1}, \tag{2.3.1}$$

where t is the absolute time and $\phi_v \in [0, 1)$ is the offset of the clock relative to the absolute time origin. The phase variable increases from 0 to 1 linearly in each period and marks the portion of time that has elapsed within each time slot. Moreover, to determine its transmission schedule, node v maintains not one but two ascending timers for the coarse clock, i.e., a *start timer* and an *end timer*, as depicted in Fig. 2.3.1. The state of the start and the end timers can be described by the phase variables

$$\Psi_v^{(s)}(t) \triangleq \frac{t}{T} + \phi_v + \psi_v^{(s)} \pmod{L} = s_v(t) + \Phi_v(t), \tag{2.3.2}$$

$$\Psi_v^{(e)}(t) \triangleq \frac{t}{T} + \phi_v + \psi_v^{(e)} \pmod{L} = e_v(t) + \Phi_v(t), \tag{2.3.3}$$

where $\psi_v^{(s)}$ and $\psi_v^{(e)}$ are integer offsets of the timers and $s_v(t) \triangleq \lfloor \Psi_v^{(s)}(t) \rfloor$ and $e_v(t) \triangleq \lfloor \Psi_v^{(e)}(t) \rfloor$ are the indices of the start and end time slots. The timers expire when their respective phase variables reach the value L and are reset to 0 afterwards. As mentioned before, the expiration of the start and end timers marks the first and last time slots that node v is scheduled to transmit (for a duration of $[\Psi_v^{(s)}(t) - \Psi_v^{(e)}(t) \pmod{L}] = [s_v(t) - e_v(t) \pmod{L}]$) and the transmission begins with the start beacon and ends with the end beacon. The two control signals inform the CH that a node v in range is transmitting for that time. The corresponding acknowledgments by the CHs, called *start* and *end acknowledgments*, warn other nodes in range that the channel towards the CH is busy. By having each node u, that hears the acknowledgements, update the discrete portions of its own start and end timers to avoid overlap (i.e., $s_u(t)$ and $e_u(t)$), nodes avoid conflicts (c.f. Section 2.5). At the same time, synchronization is achieved by using the estimated emission times of these beacons modulo T to update the fine-clock phase $\Phi_v(t)$ (c.f. Section 2.4).

Note that, even though each node update is based only on the acknowledgment of its CHs' in range, the synchronization information will eventually propagate through the whole network via the updates and firings of shared nodes. It is important to remark that a node v updates its start and end timers based on the acknowledgements that occur right before and right after its start and end beacons, respectively. These acknowledgements may belong to different CHs. In fact, as time elapses, these acknowledgements will most likely come from CHs of the densest clusters. This will be made clearer in later sections.

PulseSS exhibits the following three main features:

- **Synchronization** – The network is synchronized at the slot level, i.e., $\Phi_u(t) = \Phi_v(t)$, for all $u, v \in \mathcal{V}$.
- **Collision Avoidance:** The transmission schedules of all nodes within the neighborhood of the same CH are disjoint, i.e., for any $c \in \mathcal{C}$ and $u, v \in \mathcal{N}_c$,

  $$\Psi_v^{(s)}(t) - \Psi_v^{(e)}(t) \leq \Psi_v^{(s)}(t) - \Psi_u^{(s)}(t),$$

  where the above operations are modulo L.
- **Proportional Fair Scheduling:** Transmission schedules of nodes that are in the same cluster, with no other conflicts and that are transmitting in succession of each other, are proportional to their demands.

Moreover, in each time slot, we divide the duration T into uplink and downlink transmission periods with durations $T_u = \lambda T$ and $T_d = (1 - \lambda)T$, respectively, where $\lambda \in (0, 1)$. This approach not only enables two-way communication between each node and its CH, but also avoids conflict between the uplink and downlink transmissions of two exposed nodes, i.e., two nodes that are in range of each other but are transmitting

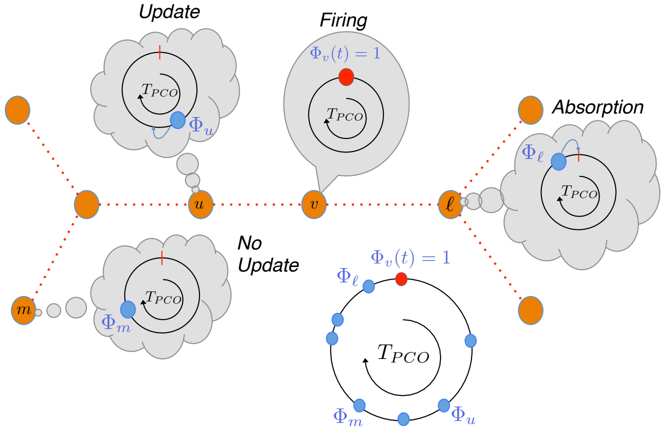


Figure 2.4.1: Example of a locally connected network of PCOs at the time node $v$fires, triggering the phase update of nodes ℓ and j. The location of the ball relative to the red mark on the top of each circle indicates the phase of each node, and a ball reaching the red mark indicates the occurrence of a firing.

and receiving simultaneously with different CH partners. Note that in the setup we assume that there is no direct CH to CH communications. However, they could occur if CHs were to operate also as normal nodes for a neighboring cluster.

2.4 PulseSS Synchronization

All PCO based algorithms rely on two common features: 1) the emission of beacon signals (or pulses) by each agent in the network and, 2) on the agents updates of their local timers (i.e., PCO clocks) upon reception of beacon signals from their neighbors. The emission of a beacon signal is referred to as the event of firing. An agent fires each time its local timer expires and, in this way, triggers its neighbors to adjust their local PCO clocks ahead, reducing the time until their next firing. The preamble signals commonly defined at the physical layer of communication systems can be used as the firing signals of PCO based algorithms, without additional overhead at any layer to support these protocols. The timer at node v can be modeled by the phase variable introduced in (2.3.1), which is normalized to 1 without loss of generality. When placed within the transmission range of each other, the firing of each node will trigger a phase update at any node that receives the firing. In Peskin's leaky integrate-and-fire model, upon hearing node v's firing at time t_v, node u updates its local auxiliary state variable $X_u(t_v) = g(\Phi_u(t_v))$ by the amount ϵ. The inverse mapping of the updated state variable then leads to a jump in the phase of the timer as follows:

$$\Phi_u(t_v^+) = \min\left\{g^{-1}(X_u(t_v) + \epsilon), 1\right\}, \tag{2.4.1}$$

where the constant ϵ is called the *coupling strength* and t_v^+ represents the time immediately following t_v. The function g is called the *PCO dynamic* and governs the behavior of the PCO network. It has been shown in [Mirollo and Strogatz(1990)] that, if g is smooth, monotonically increasing, and concave down, then synchronization in a fully connected network of PCOs is guaranteed to occur, except for a set of initial conditions with measure zero. More specifically, by choosing g such that $g(x) = \log x$ and $\epsilon = \log(1 + \alpha)$, (2.4.1) equals [Buck and Jology(1988)]:

$$\Phi_u(t_v^+) = \min\left\{(1 + \alpha)\Phi_u(t_v), 1\right\} \tag{2.4.2}$$

where $\alpha > 0$ is the *excitatory* coupling factor. Such choice for $g(x)$ is motivated by the convenience in the implementation of (2.4.2), while convergence for the fully connected network is guaranteed in [Mirollo and Strogatz(1990)]. If the phase of node u falls between $\frac{1}{1+\alpha}$ and 1 at the time of firing by node v(i.e., if $\Phi_u(t_v) \in (\frac{1}{1+\alpha}, 1])$, then the phase of node u will become 1 upon detection of the firing event of node v(i.e., $\Phi_u(t_v^+) = 1$) and will be triggered to fire immediately as well. The event is called the *absorption* of node u by node v. In a fully connected network, the absorption between two nodes remains permanent and will continue to occur progressively between clusters of nodes until synchrony is attained. In the next subsection we analyze the PCO synchronization with local connectivity of the network neglecting the propagation delays that will be included in 2.4.2.

2.4.1 PCO Synchronization with Local Connectivity

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph that represents the network topology, not necessarily fully connected, and let $e_{uv} = 1$, if $uv \in \mathcal{E}$, and $e_{uv} = 0$ otherwise. We set $e_{vv} = 0$ $\forall v \in \mathcal{V}$. The update equation in (2.4.2) can be modified as

$$\Phi_u(t_v^+) = \min\{(1 + \alpha e_{uv})\Phi_u(t_v), 1\} \tag{2.4.3}$$

Let us define the vector $\mathbf{\Delta}(t)$ with entries:

$$\Delta_{uv}(t) = \min\{(\Phi_v(t) - \Phi_u(t))\text{mod } 1, (\Phi_u(t) - \Phi_v(t))\text{mod } 1\} \tag{2.4.4}$$

for all uv such that $e_{uv} = 1$ (by definition we have $\Delta_{uv}(t) = \Delta_{vu}(t)$, $\forall t$). We then introduce the following:

Definition 2.4.1. *A network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of PCOs is said to reach a fixed point at time t^* if $\forall t > t^*$ we have $\mathbf{\Delta}(t) = \mathbf{\Delta}(t^*)$. If, in addition, we have that $\mathbf{\Delta}(t^*) = \mathbf{0}$ we say the network is synchronized (or has reached the synchronous state).*

We can now show:

Proposition 2.4.2. *For a locally connected network of PCOs that follow the dynamics in* (2.4.3) *with $\alpha > 0$, the synchronous state is the unique fixed point (as per Definition 2.4.1) i.e.,*

$$\forall t > t^*, \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*) \Leftrightarrow \mathbf{\Delta}(t^*) = \mathbf{0}. \tag{2.4.5}$$

The proof is in Appendix A. We also have the following proposition, proven in Appendix A:

Proposition 2.4.3. *On any connected network of $|\mathcal{V}| = 3$ PCOs following the dynamics in* (2.4.2) *with $\alpha > 0$, convergence to the synchronous state ($\mathbf{\Delta} = \mathbf{0}$) occurs* *almost surely from any initial condition. In addition, if we attach a node with a random initial phase to a synchronized network with an arbitrary topology, the overall network will convergence to the synchronous state almost surely.*

The second part of our proposition provides the practical insight that if protocol that allows nodes to join one by one during system setup starting from an arbitrary group of three nodes, then almost sure convergence is guaranteed. Note that we define almost sure convergence for the case $|\mathcal{V}| = 3$ as done in [Mirollo and Strogatz(1990)], where we have convergence to synchronization (i.e. to the fixed point $\boldsymbol{\Delta} = 0$) except for a measure zero set of initial conditions. The proof focuses on the case not covered by [Mirollo and Strogatz(1990)], of the line network with nodes $\{1, 2, 3\}$ and edges $\{(1, 2), (1, 3)\}$.

2.4.2 PCO Synchronization with Local Connectivity and Delays

To account for the propagation delays, which include the signal duration, travel time, processing time etc., we define:

$$r_{uv} = t_v + \tau_{uv} \tag{2.4.6}$$

where t_v is the time node v fires and τ_{uv} is the delay (expressed in time units equal to the $T = 1$). If node u is not in node v's neighborhood (i.e. $e_{uv} = 0$), $\tau_{uv} = 0$ and $r_{uv} = t_v$, otherwise r_{uv} represents the time node u is aware of node v's firing. We assume that all propagation delays are shorter than the PCO period, i.e. $\forall uv \in \mathcal{E}, \tau_{uv} < 1$ and that the delays are symmetric, i.e. $\tau_{uv} = \tau_{vu}$. In the presence of these propagation delays, PCO protocols cannot converge unless they include a *refractory period* [Peskin(1975)], i.e. a portion of the cycle, right after their firing event, during which the node does not update its phase. Notice that, in the absence of the refractory period, the firing of a node may trigger the neighboring node to fire right after the propagation delay, causing the node that originally fired to update after a roundtrip propagation delay from its initial firing event. Let ρ be the duration of the refractory period; this is the so-called *echo* effect which can be avoided if:

$$\rho > 2 \max \tau_{uv}. \tag{2.4.7}$$

The update equation in the presence of delays is:

$$\Phi_u(r_{uv}^+) = \begin{cases} \min\{(1+\alpha e_{uv})\Phi_u(r_{uv}), 1\}, & \rho < \Phi_u(r_{uv}) \bmod 1 \\ \Phi_u(r_{uv}) \bmod 1, & \text{else.} \end{cases} \tag{2.4.8}$$

Note that, if $e_{uv} = 1$:

$$\Phi_u(r_{uv}) = \Phi_u(t_v) + \tau_{uv} \pmod 1 \tag{2.4.9}$$

is the value of the clock phase of node u at the time it detects node v's firing. In this case, τ_{uv} can be viewed as an additive timing error and the update can be written as follows:

$$\Phi_u(r_{uv}^+) = \min\{(1+\alpha\hat{e}_{uv}(t_v))(|\Phi_u(t_v) + \tau_{uv}| \mod 1), 1\} \tag{2.4.10}$$

where $\hat{e}_{uv}(t_v)$ is defined as:

$$\hat{e}_{uv}(t_v) = \begin{cases} 1 & \text{if } e_{uv} = 1 \text{ and } |\Phi_u(t_v) + \tau_{uv}| \mod 1 > \rho \\ 0 & \text{else} \end{cases} \tag{2.4.11}$$

and can be seen as the element of a time varying adjacency matrix. We then can prove the following:

Proposition 2.4.4. *For deterministic $\tau_{uv} < +\infty$ $\forall u, v$, if we include a refractory period $2 \max \tau_{uv} \leq \rho < \frac{1}{2} + \min \tau_{uv}$, we have that for any locally connected network of PCOs following the dynamics in* (2.4.3)*:*

$$\forall t > t^*, \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*) \Leftrightarrow \mathbf{\Delta}(t^*) \in \mathcal{F} \tag{2.4.12}$$

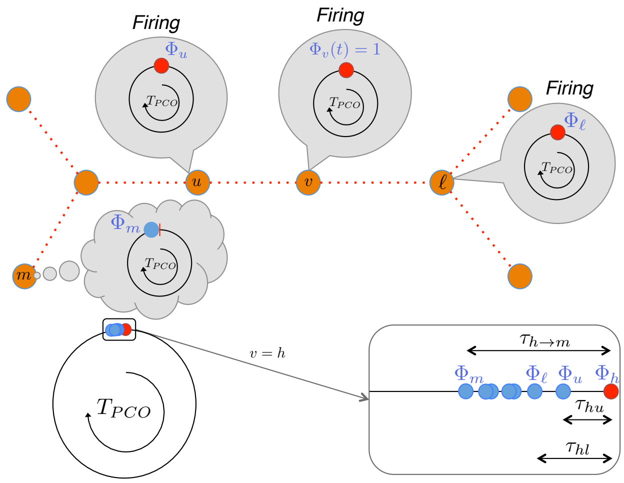


Figure 2.4.2: Convergence of the PCO protocol as stated in Proposition 2.4.4 for the network topology of Fig.2.4.1 with v as the *head* node.

where

$$\mathcal{F} \triangleq \{\boldsymbol{\Delta} : 0 \leq \Delta_{uv} \leq \tau_{uv} \quad \forall u, v \quad s.t. \quad e_{uv} = 1\} \tag{2.4.13}$$

represents the set of possible fixed points for the algorithm.

The proof can be found in Appendix A where the upper-bound for ρ is also discussed. A direct consequence of this proposition is that in order to have a possible choice for the refractory period ρ we need $\max_{uv} \tau_{uv} < \frac{1}{2} + \min_{uv} \tau_{uv}$ The convergence of the protocol is represented in Fig. 2.4.2 for the same topology as in Fig. 2.4.1. Let $\mathcal{P}_{uv}$ be the set of edges forming the shortest path between node u and node v. We can define the accumulated propagation delay on the path from v to u:

$$\tau_{u\to v} = \sum_{\ell m \in \mathcal{P}_{uv}} \tau_{\ell m} \tag{2.4.14}$$

where $\tau_{\ell m}$ is the time that has elapsed between the actual firing by node ℓ and the observation of the firing by node m. Clearly $\tau_{u\to v} = \tau_{uv}$ if $e_{uv} = 1$, i.e., if node u can directly hear the firing of node v. Then we notice that, as long as $\max_{uv} \tau_{u\to v} < \frac{1}{2}$ for any fixed points in $\mathcal{F}$ it is possible to consider a node h we name the *head* (not necessarily unique) such that its minimum distance Δ_{hu} defined in (2.4.4) $\forall u \in \mathcal{V}$ is:

$$\Delta_{hu} = (\Phi_h - \Phi_u) \mod 1 \tag{2.4.15}$$

since the node is ahead of every other node. We indicate this condition by saying the *head* node "preceeds all the other nodes". Since we have $\boldsymbol{\Delta} \in \mathcal{F}$ it is clear we have the following upper-bound:

$$\boldsymbol{\Delta}_{hu} \leq \tau_{h \to u}, \tag{2.4.16}$$

Our simulation shows that the bound in (2.4.16) is actually tight, and that is due to the fact that when the initial phases are spread around the PCO cycle, absorptions tend to occur in a cascade and each u node that is absorbed by node v remains at distance τ_{uv}. The reason why Proposition 2.4.4 has an inequality instead of an equality is if the initial conditions are such that two nodes are closer than their propagation delay they can remain at that closer distance relative to their propagation delay, due to the presence of the refractory period that makes all these cases fixed points. The residual synchronization error can be defined as:

$$\Delta_{\max} = \max_{uv} \Delta_{uv}. \tag{2.4.17}$$

At this point, as a direct consequence of the bound (2.4.16) and the property of the head-node in (2.4.15) we can derive the following expression for the expected residual synchronization error:

$$\mathbb{E}\{\Delta_{\max}\} \leq \sum_{h=1}^{N} p_h \left(\max_u \tau_{h \to u} \right) \tag{2.4.18}$$

where p_h indicates the probability that h is the *head* node which depends on the topology and the initial conditions. Although a general characterization of p_h is complex, the expression in (2.4.18) allows us to bound the residual synchronization error considering the best and the worst case scenario for the term $\max_u \tau_{h \to u}$, which is immediately derivable from the topology. A note of caution is that the term *best* *case* indicates the smallest possible bound over all possible choices of the head node, and not the best attainable synchronization error, since in principle the synchronous state $\mathbf{\Delta} = \mathbf{0}$ is a fixed point also for the model with delays. $\mathbb{E}\{\Delta_{\max}\}$ is the metric we consider in the simulations to evaluate the performances of a line and a star network of fixed length and increasing density of nodes. The expected value $\mathbb{E}\{\Delta_{\max}\}$ is an interesting metric to characterize the protocol performance and may be more insightful compared to the worst and the best case scenario, directly computable through our analysis in this section, because the latter remain identical over a wide variety of networks while $\mathbb{E}\{\Delta_{\max}\}$ changes.

Remark 2.4.5. If the propagation delays are random and bounded by τ_{max}, the result continues to apply as long as $\rho > 2\tau_{max}$. The fixed point is still compatible with the last realization of random delays characterizing all absorptions until the last.

The direct consequence of Proposition 2.4.4 is that in multi-hop networks, propagation delays tend to accumulate worsening the overall synchronization accuracy. This limits the application of PCO as a clock distribution mechanism in very large networks. To overcome this problem, we propose in [Gentz *et al.*(2016)] to couple the synchronization and scheduling with the purpose of separating firing events to give each node the possibility to estimate the propagation delays τ_{uv} and compensate for them in their updates, thus improving the final synchronization accuracy which will be bounded by the cumulative error in these estimates.

2.5 PulseSS Scheduling

Different from the synchronization updates, the scheduling of PulseSS for collision avoidance and proportional fair scheduling is achieved through the update of the discrete portions of the nodes' start and end timers (i.e., $s_v(t)$ and $e_v(t)$) in each cycle of the coarse clock. The updates and messaging mechanisms can be viewed as an extension of the theory of PCO desynchronization studied in [Pagliari *et al.*(2010)] and are described as follows. We use this section to describe the general structure of the scheduling update, represented by the block named *update function and dithering* in Fig. 2.5.3. Within this section we assume that a node has infinite amounts of data to transmit and its queue never runs empty. Activating the update is an option in the actual reference architecture, because the amount of data to transmit is finite and the resources available may be in excess of some or all nodes individual needs. This means that in the actual implementation of the solution the system has to decide if the update is necessary and how to pick the parameter that defines the desired demand. Specifically, suppose that the initial state of the start and end timers already satisfy the collision avoidance criterion, that is, for any $c \in \mathcal{C}$ and $u, v \in \mathcal{N}_c$,

$$\Psi_v^{(s)}(t) - \Psi_v^{(e)}(t) \leq \Psi_v^{(s)}(t) - \Psi_u^{(s)}(t) \pmod{L}. \tag{2.5.1}$$

This can be achieved by letting the initial difference of the start and end timers at each node be sufficiently small. If this holds true, we can denote by $\pi_k^c(t)$ the kth index at time t of the permutation of the nodes' indices that sorts the phase variables of the nodes in $\mathcal{N}_c$ in descending order at time t i.e., in the order such that

$$\Psi^{(s)}_{\pi_1^c(t)}(t) > \Psi^{(s)}_{\pi_2^c(t)}(t) > \cdots > \Psi^{(s)}_{\pi^c_{|\mathcal{N}_c|}(t)}(t).$$

In the following, we shall omit the time index t in $\Psi^{(s)}_{\pi_k(t)}(t)$ whenever its dependence on t is clear. For this algorithm, as it will be clear later, the firing order does not change over time. To simplify the notation, let us consider two functions pre, suc : $\mathcal{N} \times \mathcal{C} \to \mathcal{N}$, defined by:

$$\mathrm{pre}(v, c) = \pi^c_{k-1} \quad \in \mathcal{N}_c \tag{2.5.2}$$

$$\mathrm{suc}(v, c) = \pi^c_{k+1} \quad \in \mathcal{N}_c \tag{2.5.3}$$

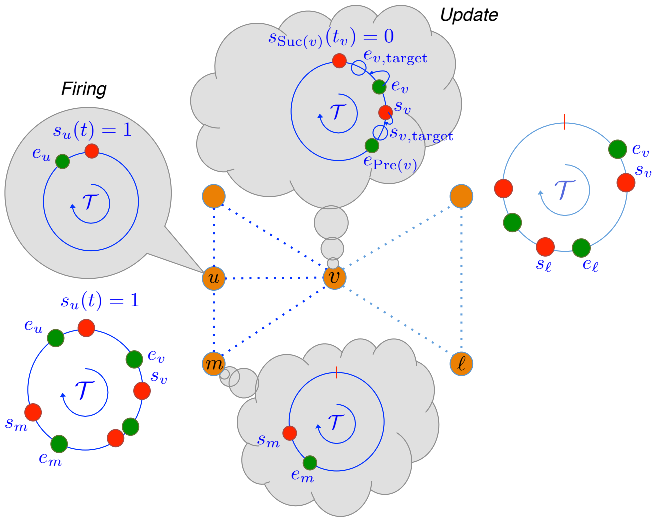


Figure 2.5.1: Update procedure for the scheduling algorithm when s_u reaches the firing point and node v identifies it as its successor.

for all $c \in \mathcal{C}$ and for all $v \in \mathcal{N}_c$ such that $i = \pi_k^c$, where $\pi_0^c = \pi_{|\mathcal{N}_c|}^c$ and $\pi_{|\mathcal{N}_c|+1}^c = \pi_1^c$ (the above quantities are not defined if $i \notin \mathcal{N}_c$). Here, $\mathrm{pre}(i,c)$ and $\mathrm{suc}(i,c)$ represent the nodes in $\mathcal{N}_c$ that produce a firing event (the expiration of one of the two timers) immediately before and after the firing of the start and the end timers of node v. The update procedure for node v at the time of firing of node $u = \mathrm{Suc}(v)$ is illustrated in Fig.2.5.1 for a topology with two cliques. Introducing:

$$\Gamma_{\pi_k^c}(t) = \frac{s_{\pi_k^c}(t) - e_{\pi_k^c}(t) \pmod L}{L} \tag{2.5.4}$$

$$\Theta_{\pi_k^c}(t) = \frac{e_{\mathrm{pre}(\pi_k^c,c)}(t) - s_{\pi_k^c}(t) \pmod L}{L} \tag{2.5.5}$$

for $k = 1, \ldots, |\mathcal{N}_c|$, it is possible to describe mathematically the evolution of the schedule for every cluster c by describing the dynamics of the vector:

$$\boldsymbol{\Upsilon}_c(t) \triangleq [\Theta_{\pi_1^c}(t), \Gamma_{\pi_1^c}(t), \ldots, \Theta_{\pi_{|\mathcal{V}_c|}^c}(t), \Gamma_{\pi_{|\mathcal{V}_c|}^c}(t)]^T. \tag{2.5.6}$$

Notice that the entries of this vector are the portions of the frame allocated to each node at time t and the corresponding intermediate guard-spaces. Therefore the fixed points of the algorithm represent the final schedule, assuming the demands remain unchanged for a sufficiently long period. It is of interest to understand if the schedule will correspond to an efficient use of the bandwidth and this is the aim of the analysis in the next sections. Our solution tries to implement, in a decentralized fashion, a linear update of the vector $\boldsymbol{\Upsilon}_c(t)$ whose dynamics converge to a schedule that has favorable properties as *fairness* and *efficiency*.

Definition 2.5.1. *We define a PulseSS Scheduling primitive any linear update of the vector in* (2.5.6) *which can be written as*

$$\boldsymbol{\Upsilon}(t^+) = \mathbf{J}\mathbf{U}\boldsymbol{\Upsilon}(t) \tag{2.5.7}$$

where $\boldsymbol{\Upsilon} = [\boldsymbol{\Upsilon}_1^T, \boldsymbol{\Upsilon}_2^T, \ldots, \boldsymbol{\Upsilon}_{|\mathcal{C}|}^T]^T$, $\mathbf{J}$ *is a permutation matrix and* $\mathbf{U}$ *is any matrix that has the following block structure*

$$\mathbf{U} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{B} & \mathbf{I} \end{bmatrix} \tag{2.5.8}$$

with $\mathbf{A}$ *being a stochastic matrix and* $\mathbf{I}$ *the identity matrix.*

Such update could be triggered by any CH acknowledgment, broadcasted to all nodes in the communication range. The PulseSS Primitive which is described in the next subsection has been named PulseSS Pairwise Update but similar analysis could be extended to different sequences of primitives that respect Definition 2.5.1.

2.5.1 PulseSS Pairwise Update

Since node v may be in the range of more than one CH, it is necessary to define

$$\mathrm{Pre}(v,t) = \mathrm{pre}\left(v, \arg\min_{c' \in \mathcal{C}_v}\{\Psi^{(e)}_{\mathrm{pre}(v,c')}(t) - \Psi^{(s)}_v(t)\}\right) \tag{2.5.9}$$

$$\mathrm{Suc}(v,t) = \mathrm{suc}\left(v, \arg\min_{c' \in \mathcal{C}_v}\{\Psi^{(e)}_v(t) - \Psi^{(s)}_{\mathrm{suc}(v,c')}(t)\}\right) \tag{2.5.10}$$

as the two nodes (not necessarily in the same cluster) which transmit immediately before and after node v among all the ones in clusters node v belongs to. Notice that we need node v to have no conflicts with all the nodes in the clusters that it belongs to, which can be more than one. The identification of these two nodes is fundamental for the bio-inspired procedure we are going to introduce in the next subsection[1] . To discriminate the firing times of $\mathrm{Pre}(\pi_k^c)$ from the others, node π_k^c can prepare an update for any firing it hears and discard the update if a more recent firing event is registered. $\mathrm{Suc}(\pi_k^c)$ is easy to identify since is the first to fire after the expiration of node π_k^c's own end clock. The value of $\Psi_{\mathrm{Pre}(\pi_k^c)}(t)$, which is the reference for node π_k^c to make its update, can be calculated simply measuring the time it elapsed between the firing of $Pre(\pi_k^c)$ and the local clock. Hence all the information needed to advance the protocol is implicitly available and firing beacons do not need to carry data, but rather can be special preambles that are easy to detect at the PHY layer. The expiration of the start and end timers marks the start and end of a node's transmission period in each cycle. Once the start (or the end) timer of a node, say node v, expires, a start (or an end) beacon is emitted by it in the UL period of the time slot. The beacon emitted by node v will then be acknowledged by all CHs in range, to inform all other nodes in the neighborhood of the CHs of the beacon emission. In case the collision avoidance criterion is violated, admission control at CHs would not acknowledge, i.e., may deny

[1] To discriminate the firing times of $\mathrm{Pre}(\pi_k^c)$ from the others, node π_k^c can prepare an update for any firing it hears and discard the update if a more recent firing event is registered. $\mathrm{Suc}(\pi_k^c)$ is easy to identify since is the first to fire after the expiration of node π_k^c's own end clock. The value of $\Psi_{\mathrm{Pre}(\pi_k^c)}(t)$, which is the reference for node π_k^c to make its update, can be calculated simply measuring the time it elapsed between the firing of $Pre(\pi_k^c)$ and the local clock. Hence all the information needed to advance the protocol is implicitly available and firing beacons do not need to carry data, but rather can be special preambles that are easy to detect at the PHY layer [Gentz *et al.*(2016), Gentz(2017)].

a second start beacon, before an end beacon is received, so that only one node at a time has channel access. We assume that all CHs in range acknowledge at unison with an identical beacon signal, such that acknowledgments are processed at the receiving node as a single signal affected by multi-path (we view this as a *cooperative-channel* acknowledgement). Let $\mathrm{pre}(v) \in \cup_{c\in\mathcal{C}_v}\mathcal{N}_c$ and $\mathrm{suc}(v) \in \cup_{c\in\mathcal{C}_v}\mathcal{N}_c$ be the nodes that transmit immediately before and after node v, i.e., the *predecessor* and *successor* of node v. Node v adjusts its local timers in each cycle based on the expiration times of the end and start timers of nodes $\mathrm{pre}(v)$ and $\mathrm{suc}(v)$ respectively. Therefore the fixed points of the algorithm represent the final schedule, assuming the demands remain unchanged for a sufficiently long period. It is of interest to understand if the schedule will correspond to an efficient use of the bandwidth and this is the aim of the analysis in the next sections.

Let $t_v^{(s)} \in \{t : \Psi_v^{(s)}(t) = L\}$ be the expiration time instant of the start timer of node v in a given cycle of the coarse clock and let $t_v^{(e)} = \min\{t > t_v^{(s)} : \Psi_v^{(e)}(t) = L\}$ be that of the end timer of node v that follows immediately after. Moreover, let $t_{\mathrm{pre}(v)}^{(e)} = \max\{t < t_v^{(s)} : \Psi_{\mathrm{pre}(v)}^{(e)}(t) = L\}$ be the most recent expiration time instant of the predecessor's end timer and let $t_{\mathrm{suc}(v)}^{(s)} = \min\{t > t_v^{(e)} : \Psi_{\mathrm{suc}(v)}^{(s)}(t) = L\}$ be that of the successor's start timer. The corresponding time estimates[2] at node v are denoted by $\hat{t}_{\mathrm{pre}(v),v}^{(e)}$ and $\hat{t}_{\mathrm{suc}(v),v}^{(s)}$.

Immediately after receiving the acknowledgment to the start timer of $\mathrm{suc}(v)$, node v, at time $t_{\mathrm{suc}(v)}^{(s)+}$, updates its local timers in an attempt to move the discrete portion of the clocks phases (i.e., the time slot index) towards the target values

$$s_{v,\mathrm{target}} = \frac{D_v+\delta}{D_v+2\delta} e_{\mathrm{pre}(v)}\left(t_{\mathrm{suc}(v)}^{(s)+}\right) + \frac{\delta}{D_v+2\delta} s_{\mathrm{suc}(v)}\left(t_{\mathrm{suc}(v)}^{(s)+}\right) \tag{2.5.11}$$

[2] Note that the time instants $t_{\mathrm{pre}(v)}^{(e)}$ and $t_{\mathrm{suc}(v)}^{(s)}$ can be estimated by node v through the reception time of CH's acknowledgments to these beacon signals, but the accuracy may be affected by synchronization errors and propagation delays, as described in the previous section.

$$e_{v,\text{target}} = \frac{\delta}{D_v+2\delta} e_{\text{pre}(v)}\left(t^{(s)+}_{\text{suc}(v)}\right) + \frac{D_v+\delta}{D_v+2\delta} s_{\text{suc}(v)}\left(t^{(s)+}_{\text{suc}(v)}\right) \tag{2.5.12}$$

where D_v is a parameter capturing the demand of node v, δ is the portion of time slots reserved as guard period in between transmissions. If the target values are achieved, a portion of $D_v/(D_v + 2\delta)$ of the time between the transmissions of its predecessor and successor is left for node v's transmission of its payload data and $\delta/(D_v + 2\delta)$ portion of the time is left before and after its own transmission as guard intervals.

When $t^{(s)}_{\text{suc}(v)}$ and $t^{(e)}_{\text{pre}(v)}$ are perfectly known and that no updates have been made to predecessor's phase before time $t^{(s)}_{\text{suc}(v),v}$, node v can infer that $s_{\text{suc}(v)}(t^{(s)+}_{\text{suc}(v)}) = 0$, since the timer must have reset to 0 after it has expired and that $e_{\text{pre}(v)}(t^{(s)+}_{\text{suc}(v)}) = (t^{(s)+}_{\text{suc}(v)} - t^{(e)}_{\text{pre}(v)})/T$, which is the time that has elapsed after the expiration of the end timer of node pre(v). However, in reality, these target values cannot be obtained precisely since only the estimates $\hat{t}^{(s)}_{\text{suc}(v),v}$ and $\hat{t}^{(e)}_{\text{pre}(v),v}$ are known at node v and also since the phase of the predecessor may in fact have been updated before time $\hat{t}^{(s)}_{\text{suc}(v),v}$ due to the beacon emission of node v. In this case, node v can only obtain the estimated target values

$$\hat{s}_{v,\text{target}} = \frac{D_v+\delta}{D_v+2\delta} \frac{\hat{t}^{(s)+}_{\text{suc}(v),v} - \hat{t}^{(e)}_{\text{pre}(v),v}}{T} \tag{2.5.13}$$

$$\hat{e}_{v,\text{target}} = \frac{\delta}{D_v+2\delta} \frac{\hat{t}^{(s)+}_{\text{suc}(v),v} - \hat{t}^{(e)}_{\text{pre}(v),v}}{T}. \tag{2.5.14}$$

Since the target values are not precise[3], it is necessary to further limit the adjustment of the timers at node v so that the relative order of its timers and the timers of its predecessor and successor are not altered, causing overlap in the schedules. This is

[3] The time values used are based on possibly outdated information about the predecessor node's state at the time it last fired.

achieved by further modifying their target values as

$$\tilde{s}_{v,\text{target}} = \min\left\{ \hat{s}_{v,\text{target}},\, \frac{s_v(\hat{t}^{(s)+}_{\text{suc}(v),v}) + \frac{\hat{t}^{(s)+}_{\text{suc}(v),v} - \hat{t}^{(e)}_{\text{pre}(v),v}}{T}}{2} \right\}$$

$$\tilde{e}_{v,\text{target}} = \max\left\{ e_{v,\text{target}},\, \frac{e_v(\hat{t}^{(s)+}_{\text{suc}(v),v})}{2} \right\}.$$

Finally, the local timers at node v are updated as

$$s_v(\hat{t}^{(s)+}_{\text{suc}(v),v}) = \mathcal{Q}\left[(1-\beta)s_v(\hat{t}^{(s)}_{v,\text{suc}(v)}) + \beta\tilde{s}_{v,\text{target}}\right] \tag{2.5.15}$$

$$e_v(\hat{t}^{(s)+}_{\text{suc}(v),v}) = \mathcal{Q}\left[(1-\beta)e_v(\hat{t}^{(s)}_{\text{suc}(v),v}) + \beta\tilde{e}_{v,\text{target}}\right] \tag{2.5.16}$$

where $\beta \in (0,1)$ and $\mathcal{Q}(\cdot)$ is a dithered quantization function [Wannamaker *et al.*(2000)] that maps the phase to the integer set $\{0, 1, \dots, L\}$ defined as $\mathcal{Q}(x) = \text{round}(x + v)$, where $v \sim \mathcal{U}(-1/2, 1/2)$. As shown in [Aysal *et al.*(2008)], the dithering operation ensures the convergence of the quantized consensus policy and has similar effects on PulseSS. In fact, as time elapses and synchronization is achieved, the dithered quantized desynchronization protocol mentioned above has been shown to converge for all-to-all networks in [Ashkiani and Scaglione(2012)]. Its properties in a locally connected networks are discussed in the next subsection.

The scheduling primitive just described only relies on the signaling of a node's respective pre- and successor while all other nodes are ignored for scheduling purposes. As per Definition 2.5.1 multiple nodes could update at the same time with knowledge, obtained via physical signaling, of the other state variables and this could potentially give a faster convergence.

2.5.2 Convergence of the Single Clique Scheduling Algorithm

In this section we recall the convergence result in [Pagliari *et al.*(2010)] and analyze the convergence rate of the algorithm. The analysis in the next sections is conducted considering the node local timers $s_v(t)$ and $e_v(t)$ as continuous variables that evolve between 0 and 1 (for L large enough, this approximates the normalized values over the number of cycles L quite accurately), and considering the linear mapping in (2.5.15)-(2.5.16) without dithering and the target values in (2.5.11)-(2.5.12). In the case of a single clique c in our graph, due to the updates, the state vector $\boldsymbol{\Upsilon}_c(t)$ evolves linearly with system matrix $\mathbf{M}^c$ is defined as:

$$\mathbf{M}^c = \prod_{k=1}^{|\mathcal{V}_c|} \boldsymbol{M}_{\pi_k^c} \tag{2.5.17}$$

and each $\boldsymbol{M}_{\pi_k^c}$ is the matrix for the update of node π_k^c in the clique c. This matrix has the following form:

$$\boldsymbol{M}_{\pi_k^c} = \mathbf{J}^{(2k-2)} \cdot \begin{bmatrix} \mathbf{U}_{\pi_k^c} & \mathbf{0}_{3\times(2|\mathcal{V}_c|-3)} \\ \mathbf{0}_{(2|\mathcal{V}_c|-3)\times 3} & \mathbf{I}_{(2|\mathcal{V}_c|-3)} \end{bmatrix} \cdot \mathbf{J}^{T^{(2k-2)}} \tag{2.5.18}$$

where $\mathbf{J}$ represents the circular shift matrix:

$$\mathbf{J} \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \ddots & \ddots & 0 & 0 \\ 0 & \cdots & 0 & 1 & 0 \end{bmatrix}. \tag{2.5.19}$$

and

$$\mathbf{U}_v \triangleq \begin{bmatrix} 1-\beta\frac{D_v+\delta}{D_v+2\delta} & \beta\frac{\delta}{D_v+2\delta} & \beta\frac{\delta}{D_v+2\delta} \\ \beta\frac{D_v}{D_v+2\delta} & 1-\beta\frac{2\delta}{D_v+2\delta} & \beta\frac{D_v}{D_v+2\delta} \\ \beta\frac{\delta}{D_v+2\delta} & \beta\frac{\delta}{D_v+2\delta} & 1-\beta\frac{D_v+\delta}{D_v+2\delta} \end{bmatrix}. \tag{2.5.20}$$

The proof in [Pagliari *et al.*(2010)] shows that for this configuration there exists a unique fixed point:

$$\boldsymbol{\Upsilon}_c^\star = \frac{\gamma^c}{\boldsymbol{D}^c}(\delta, D_{\pi_1^c}, \delta, D_{\pi_2^c}, \ldots, \delta, D_{\pi_{|\mathcal{V}_c|}^c})^T \tag{2.5.21}$$

where $\boldsymbol{D}^c = \sum_{k=1}^{|\mathcal{V}_c|} D_{\pi_k^c}$ and $\gamma^c = \dfrac{\boldsymbol{D}^c}{\boldsymbol{D}^c + |\mathcal{V}_c|\delta}$.

For the specific case of a single clique c with all nodes having the same demand ($D_{\pi_k^c} = D \ \forall k = 1, 2, \ldots, |\mathcal{V}_c|$) it is actually possible to complement the result with an estimate for the rate of convergence. In fact, when the evolution of the system can be modeled with a linear update, as in average consensus algorithms [Dimakis *et al.*(2010)], it is well known that the rate of convergence can be estimated via the second largest eigenvalue of the system matrix (i.e., convergence towards a fixed point is guaranteed if the highest eigenvalue is equal to 1 and the others are strictly smaller). Assuming equal demand for all nodes, it is possible to rewrite the system matrix in (2.5.17) as:

$$\mathbf{M}^c = \left(\begin{pmatrix} \mathbf{U} & \mathbf{0}_{3\times(2|\mathcal{V}_c|-3)} \\ \mathbf{0}_{(2|\mathcal{V}_c|-3)\times 3} & \mathbf{I}_{(2|\mathcal{V}_c|-3)} \end{pmatrix} \cdot \mathbf{J}^2 \right)^{|\mathcal{V}_c|} \tag{2.5.22}$$

where the dependence of the block matrix $\mathbf{U}$ on the node $v$has been lost setting an equal demand D for all nodes. At this point it is possible to derive the exact $2|\mathcal{V}_c|$-th degree characteristic equation for the product matrix inside the brackets, find an approximation for the second highest solution and then take the $|\mathcal{V}_c|$-th power of that value to find the second highest eigenvalue for the system matrix $\mathbf{M}^c$. We claim:

Proposition 2.5.2. *The second largest eigenvalue of* $\mathbf{M}^c$ *if all the nodes have the same demand D is:*

$$|\lambda_2^c| \approx 1 - \frac{2\beta\mu\pi^2}{|\mathcal{V}_c|^2} \tag{2.5.23}$$

where $0 < \beta < 1$ *is the coupling factor in the update equation in* (2.5.15),(2.5.16) *and* $\mu = \dfrac{\delta}{D + 2\delta}$.

The proof of this proposition is in Appendix A. Clearly, the convergence time increases with the number of nodes in the clique and decreases with the values of β and μ. However, augmenting μ by increasing the guard time δ relative to the demand

D lowers its efficiency. Furthermore, in non-ideal conditions in which in the measurement of $\Psi_{\mathrm{Pre}(i)}, \Phi_{\mathrm{Suc}(i)}$ are not precise (see footnote 1) or the local timers are quantized (for more detailed discussion we refer to [Gentz *et al.*(2016)]), aggressively increasing β may result in lack of convergence. While the rate of convergence is indicative of the trends we found in locally connected networks, the study of the fixed points requires appropriate changes, since the presence of shared nodes changes the structure of $\mathbf{M}_{\pi_k^c}$, introducing coupling among the sub-cliques. To describe these changes next we need to introduce new quantities, definitions, assumptions and notations, which precede our main convergence result. Nevertheless, we wish to remark that the result of Proposition 2.5.2 has been found, via simulation, to be a good approximation also for the behaviour of multiclique networks, if we consider for $\mathcal{V}_c$ the largest clique of the graph.

2.5.3 Convergence of the Multi-Cliques Scheduling Algorithm

In this section we analyze what are the possible schedules that are fixed points for the algorithm.

Definition 2.5.3 (Partial proportional fairness criterion)**.** *We say a schedule meets a partial proportional fairness criterion if, once convergence is reached (i.e.* $\forall t > t^*$ *for some* t^**),* $\forall (v, u) \quad v \neq u$ *if* $v, u \in \mathcal{L}_c$ *and* $u = \mathrm{pre}(v, c)$ *the following condition is met :*

$$\frac{(s_v(t) - e_v(t)) \bmod L}{D_v} = \frac{(s_u(t) - e_u(t)) \bmod L}{D_u}$$

Definition 2.5.4 (Global proportional fairness criterion)**.** *We say a schedule meets a global proportional fairness criterion if the two following properties are satisfied once convergence is reached (i.e.* $\forall t > t^*$ *for some* t^**):*

1. $\forall (v,u) \quad v \neq u$ *if* $v, u \in \mathcal{L}_c$*, then:*

$$\frac{(s_v(t) - e_v(t)) \bmod L}{D_v} = \frac{(s_u(t) - e_u(t)) \bmod L}{D_u}$$

2. $\forall u \quad \frac{[(s_u(t) - e_u(t)) \bmod L]}{L} \geq \min\limits_{c : u \in \mathcal{V}_c} \frac{D_u}{\sum_{v \in \mathcal{V}_c} (D_v + \delta)}$

The second property indicates that the solution guarantees that every node gets the minimum possible duration among all cliques in its range. Let us first introduce the following:

Assumption 2.5.5. *For every clique $\mathcal{V}_c$, all the local nodes (i.e. the set $\mathcal{L}_c$), occupy consecutive portions of the frame.*

We can then claim the following:

Theorem 2.5.6. *For a network with two cliques, the update rule in* (2.5.15)-(2.5.16) *will converge to a unique fixed point $\boldsymbol{\Upsilon}_c^\star$ $\forall c \in \mathcal{C}$ that respects the partial proprtional fairness criterion in Definition 2.5.3 , irrespective of the initial phases of the timers. If Assumption 2.5.5 holds, the resulting schedule will also respect the global proportional fairness criterion in Definition 2.5.4.*

The proof can be found in Appendix A.

In Fig. 2.5.2 we can see the convergence of the scheduling algorithm to a fixed point where the target $(s_{v,\text{target}}, e_{v,\text{target}}) = (s_v(t_v), e_v(t_v))$. Therefore, the timers of node v (as well of those of every other node) will no longer change from that update on.

Following a similar argument as in the proof of Theorem 2.5.6 we can claim

Proposition 2.5.7. *For topologies with more than two cliques we have, in general, fixed points for* (2.5.15)-(2.5.16) *form sets with measure greater than zero. All these points respect the partial proportional fairness criterion in Definition 2.5.3.*

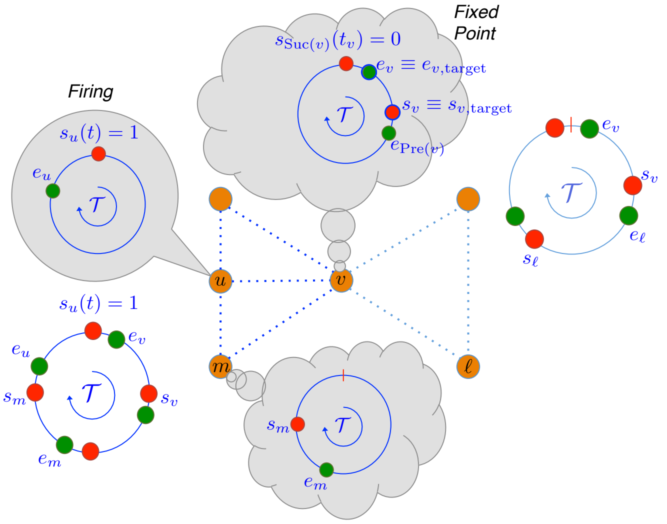


Figure 2.5.2: Convergence of the scheduling for the topology in Fig.2.5.1.

See Proof in Appendix A. In a nutshell, for general topologies we do not have enough contraints on the attainable schedule to guarantee a unique fixed point as in the two-clique case. Nevertheless, additional definitions and assumptions allow to characterize very peculiar cases in which a unique fixed point is attainable with more than two cliques. Let us introduce the partition $\mathcal{A}_c$ of the nodes set $\mathcal{V}$ as:

$$\mathcal{A}_c = \left\{ v \,\middle|\, c = \arg\max_{c' \in \mathcal{C}_v} \sum_{v \in \mathcal{V}_{c'}} (D_v + \delta) \right\} \tag{2.5.24}$$

It is clear that $\forall c$, $\mathcal{A}_c \subseteq \mathcal{V}_c$ and, with distinct overall demands for cliques that have shared nodes (i.e., if $\mathcal{S}_{cc'} \neq \emptyset$ then $\sum_{v \in \mathcal{V}_c} (D_v + \delta) \neq \sum_{v \in \mathcal{V}_{c'}} (D_v + \delta)$), the sets $\mathcal{A}_c$ form a proper partition of $\mathcal{V}$. Then we order these sets in decreasing order of demand size:

$$\sum_{v \in \mathcal{A}_1} (D_v + \delta) \geq \sum_{v \in \mathcal{A}_2} (D_v + \delta) \geq \ldots \sum_{v \in \mathcal{A}_{|\mathcal{C}|}} (D_v + \delta) \tag{2.5.25}$$

Let us introduce:

Assumption 2.5.8. *All the nodes in a clique are at most in two partitions* $\mathcal{A}_c$ *as defined in* (2.5.24). *Mathematically,* $\forall c \in \mathcal{C}$*, there is only a single* $c' \in \mathcal{C}$ *such that*

$$\mathcal{V}_c \subset \mathcal{A}_c \cup \mathcal{A}_{c'} \tag{2.5.26}$$

The following claim is proven in Appendix A.

Proposition 2.5.9. *If Assumptions 2.5.5-2.5.8 are met, the scheduling algorithm and its global proportional fairness property can be extended to topologies with an arbitrary number of cliques.*

Remark 2.5.10. *In the limit for $\delta \to 0$, if the schedule meets Property 2 in Definition 2.5.4, then it is also one of the possible solutions of the minimum coloring graph problem for that conflict graph.*

To meet Assumption 2.5.5 the topology of the conflict graph has to allow an assignment which leaves a portion available in any frame for nodes that belong only to one clique. However, this assumption may be violated in dense networks, as reported in [Gentz *et al.*(2016)] and a version of Assumption 2.5.8 that explains what conflict graphs can possibly meet Assumption 2.5.8 is elusive. Nonetheless, the presentation in this work should give the reader the necessary tools to analyze the possible attainable schedules on a case by case basis, given that a general treatment remains elusive. An example where this assumption is violated and the trend is still predictable is discussed in the proof of Proposition 2.5.7 and in the simulation results. In the next section we provide a description of the specific fixed point $\boldsymbol{\Upsilon}_c^\star$ $\forall c \in \mathcal{C}$ for the case where the demand is equal. For the treatment is advantageous to explicitly indicate the dependence on the frame duration $\mathscr{T} = LT$. Note that star and line networks are multi-clique graphs with maximum clique size equal to 2. For them we can state:

Corollary 2.5.11. *The line and the star networks have always a unique fixed point for the schedule consistent with the description in subsection 2.5.3.1.*

The proof is in Appendix A. If all the nodes have equal demand D, under both of these topologies, $\forall i$:

$$(s_v - e_v) \bmod L \cdot \mathscr{T} = \frac{D}{D+\delta} \frac{\mathscr{T}}{2} \tag{2.5.27}$$

which is about half of the resources as expected. From Proposition 2.5.9 we can also infer the following:

Corollary 2.5.12. *In the tree network, if parents have always higher demand than the children, there is a unique fixed point consistent with the description in Subsection 2.5.3.1.*

This is due to the fact that the condition in this corollary can be seen as an alternative way to state Assumption 2.5.8 for this topology, where we recall that the assignment to every partition $\mathcal{A}_c$ is based considering the overall demand of nodes (see (2.5.24)). This is also a pleasing result, because if the tree is used for data aggregation, it would be natural to have higher demand at the higher level of the tree.

2.5.3.1 Fixed Points

Next we describe what are the unique fixed points attainable under Proposition 2.5.9 (for brevity we assume a single demand value $D_v = D, \forall v \in \mathcal{V}$). Let us call $\mathscr{T}_c$ the portion of the frame available for the nodes in $\mathcal{A}_c$. We have:

$$\mathscr{T}_c = \begin{cases} \mathscr{T} & \text{if } \mathcal{A}_c = \mathcal{V}_c \\ \mathscr{T} - |\mathcal{S}_{cc'}|(T_{c'} + \delta_{c'}) + \delta_{c'} & \text{if } \mathcal{A}_c \subset \mathcal{V}_c, \mathcal{A}_{c'} \supset \mathcal{V}_c \setminus \mathcal{L}_c \end{cases} \tag{2.5.28}$$

and Assumption 2.5.8 guarantees the existence of such unique cluster c'. In (2.5.28), $T_{c'}$ and δ_c' represent respectively the time slot and the guard space before and after every node $v \in \mathcal{A}_{c'}$. They can be computed recursively following the decreasing order in (2.5.25) as follows:

$$T_c = \begin{cases} \dfrac{D}{D+\delta}\dfrac{\mathscr{T}_c}{|\mathcal{A}_c|} & \text{if } \mathscr{T}_c = \mathscr{T} \\ \dfrac{D}{D+\delta}\dfrac{\mathscr{T}_c}{|\mathcal{A}_c| + \frac{\delta}{D+\delta}} & \text{if } \mathscr{T}_c < \mathscr{T} \end{cases} \tag{2.5.29}$$

$$\delta_c = \frac{\delta}{D} T_c \tag{2.5.30}$$

At the fixed point, $\forall i \in \mathcal{A}_c$, we will have

$$(\Phi_v - \Psi_v) \bmod L \cdot T = T_c \tag{2.5.31}$$

To prove this represents a fixed point we will evaluate Φ_v^* and Ψ_v^* at time t_v (the time when node $v$makes its update) with (2.5.13),(2.5.14) (see also Fig.A.0.1):

$$\begin{aligned} s_{v,\text{target}}(t_v)\mathscr{T} &= \frac{D+\delta}{D+2\delta} \frac{D+2\delta}{D} T_c = \frac{\delta}{D} T_c + T_c = s_v(t_v)\mathscr{T} \\ e_{v,\text{target}}(t_v)\mathscr{T} &= \frac{\delta}{D+2\delta} \frac{D+2\delta}{D} T_c = \frac{\delta}{D} T_c = e_v(t_v)\mathscr{T} \end{aligned}$$

So the update procedure will be:

$$\begin{aligned} s_v(t_v^+) &= (1-\beta)s_v(t_v) + \beta s_{v,\text{target}}(t_v) = s_{v,\text{target}}(t_v) \\ e_v(t_v^+) &= (1-\beta)e_v(t_v) + \beta e_{v,\text{target}}(t_v) = e_v(t_v) \end{aligned}$$

which shows that the timers will keep their positions unchanged. Here we have assumed that $v \in \mathcal{A}_c$ where $\mathscr{T}_c < \mathscr{T}$ but the derivation would be exactly the same if we remove the additional term $\frac{\delta}{D+\delta}$ in (2.5.29). The interest in the derivation of the fixed points is given by their natural connection with the portion of the frame made available to each node. Before presenting the numerical results, we illustrate the complete diagram of a PulseSS protocol implementation in Figure 2.5.3.

2.6 Simulation Results

2.6.1 PCO Synchronization

We first show an example in Fig. 2.6.2 of convergence to the fixed point with and without propagation delays for the PCOs network in Fig. 2.6.1, where we plot only the components of $\boldsymbol{\Delta}$ relative to an arbitrary node in $\mathcal{L}_1$. We considered all

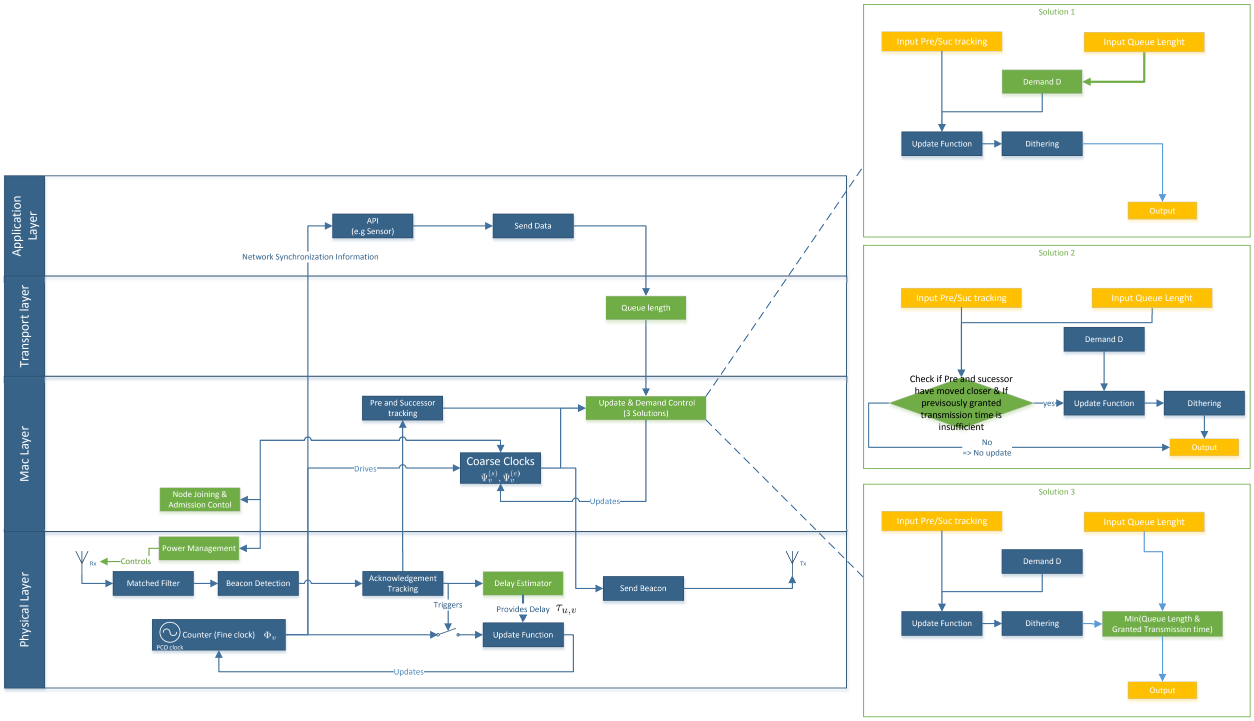


Figure 2.5.3: Overview of the components in the system for implementation of the PulseSS Protocol. Additional implementation details can be found in [Gentz *et al.*(2016), Gentz(2017)]

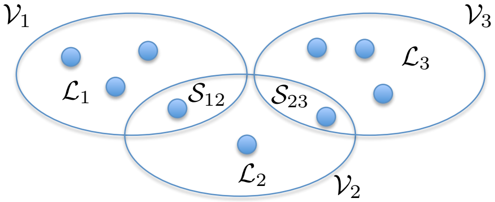
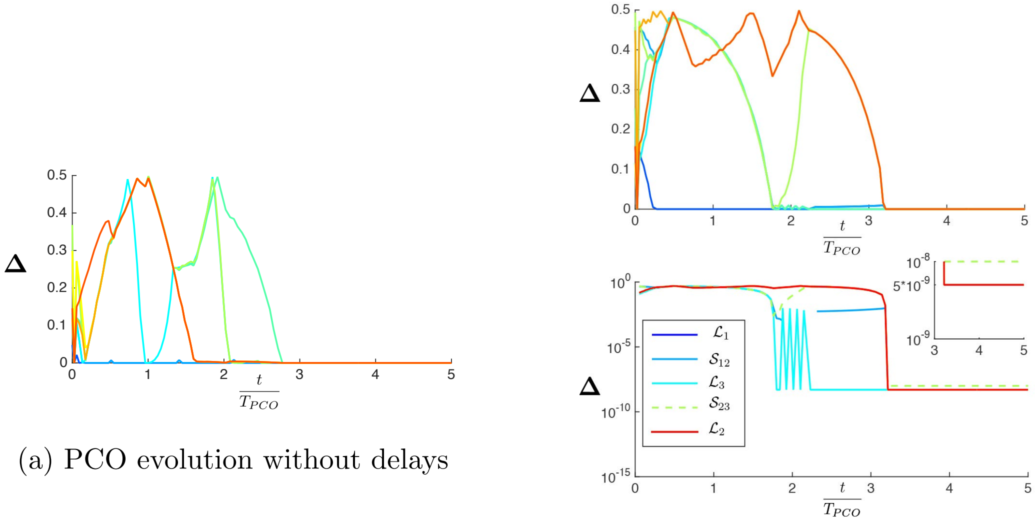


Figure 2.6.1: Topology for Experiments in Fig.2.6.2-2.6.6-2.6.7

the distances between the nodes to be equal such that $\tau_{uv} = \tau, \forall v \neq u$. We can see that without delays in Fig.2.6.2a we achieve perfect synchronization, while with delay the nodes remain separated by their propagation delays (see Fig.2.6.2b, where we also plot the components in log-scale.). In fact, observing the plot in log-scale in Fig. 2.6.2b we can see that there is a component of Δ equal to 2τ for the node in $\mathcal{S}_{23}$, the components for the nodes in $\mathcal{L}_2 \cup \mathcal{L}_3 \cup \mathcal{S}_{12}$ are equal to τ and the components for the other nodes in $\mathcal{L}_1$ are equal to 0, from which we can



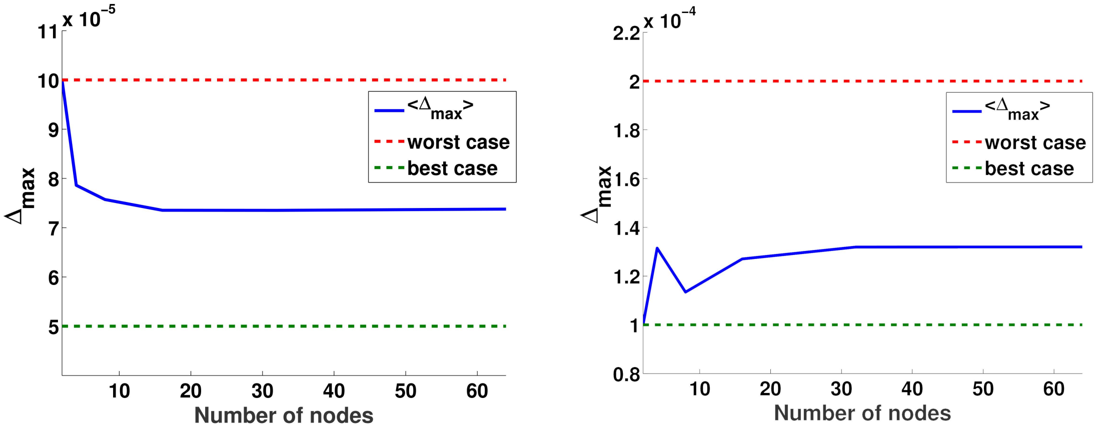
(a) PCO evolution without delays

(b) PCO evolution with delay

(linear and log scale)

Figure 2.6.2: The evolution of the synchronization with and without delays given the topology in Fig. 2.6.1, and the light blue node as arbitrary chosen reference. Parameters: $T = 1s$, $\alpha = 1e-2$, distance between each connected node $1m$, signal travel-speed $2e8m/s$, refractory period $1e-2$, uniform random initialization.

conclude that the *head* node is the one node in $\mathcal{S}_{23}$ and all the differences are in perfect agreement with our analysis. Furthermore, in light of Proposition 2.4.4 and equation (2.4.18), we simulated the average residual synchronization error $\mathbb{E}\{\Delta_{\max}\}$ for the line and the star networks with a variable number of nodes. The probabilities p_h for $h = 1, 2, \ldots, N$ are not known, however for any given topology it is possible to bound the residual synchronization error considering the best and the worst case. In Fig. 2.6.3a we show the synchronization accuracy averaged over random initial conditions for **line-networks** with an increasing number of nodes but a constant end to end delay between the nodes at the two network edges. The worst case is represented by $\tau_{\max}$ (which is the only possible case for $N = 2$), while the best case for a generic $N > 2$ is $\tau_{\max}/2$ with the *head* node being at the middle of the line. From Fig. 2.6.3a it is possible to notice the saturation of $\mathbb{E}\{\Delta_{\max}\} \approx \frac{3}{4}\tau_{\max}$, half way between the two extremes.



(a) Line network

(b) Star network

Figure 2.6.3: Average maximum displacement vs. network size.

In Fig. 2.6.3b, assuming that the delay is proportional to the relative distance, we plot the results for a **star topology** where the nodes are uniformly distributed over a disc with equal radius, leading to equal delay $\tau_{\max}$, except two nodes that are kept fixed at the center and at the edge of the disc. In this case, as expected, increasing the number of nodes degrades the performances. The initial case with $N = 2$ is the best case with $\mathbb{E}\{\Delta_{\max}\} = \tau_{\max}$. For this topology the worst case is represented by $2\tau_{\max}$. We notice an oscillation first (due to $\mathbf{p}_h$) and then, again, a saturation to $\mathbb{E}\{\Delta_{\max}\} \approx \frac{4}{3}\tau_{\max}$.

2.6.2 Scheduling Convergence

In Fig. 2.6.4 we show the accuracy of the approximate eigenvalues derived in Proposition 2.5.2 for the single clique update matrix $\mathbf{M}^c$. The circles correspond to our approximations in (A.0.2) and (A.0.3) (see Appendix A) and the crosses correspond to the numerical computed values. As expected, the accuracy of our estimate of the second largest eigenvalue grows with $\mathcal{V}_c$. In Fig. 2.6.5 we present the attainable TDMA scheduling by a two clique topology with $|\mathcal{L}_1| = 5$, $|\mathcal{S}_{12}| = 2$, $|\mathcal{L}_2| = 2$ ($D_v = D = 4$ $\forall v \in \mathcal{V}$ and $\delta = 1$). The local nodes in both $\mathcal{L}_1$ and $\mathcal{L}_2$ occupy

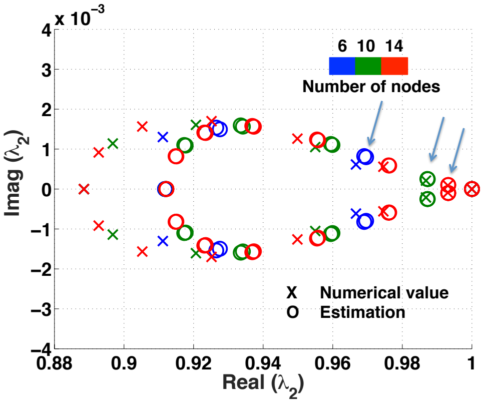
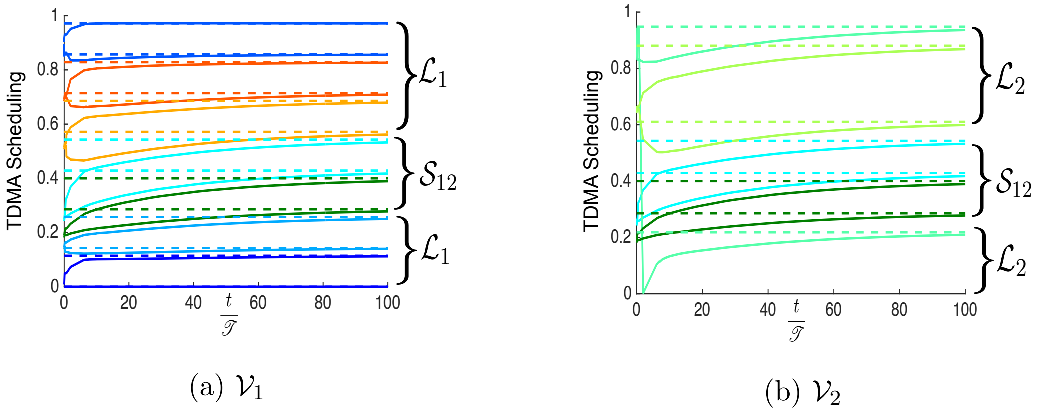


Figure 2.6.4: Numerical evaluation of the matrix $\mathbf{M}^c$ eigenvalues compared with the approximation in (A.0.4).

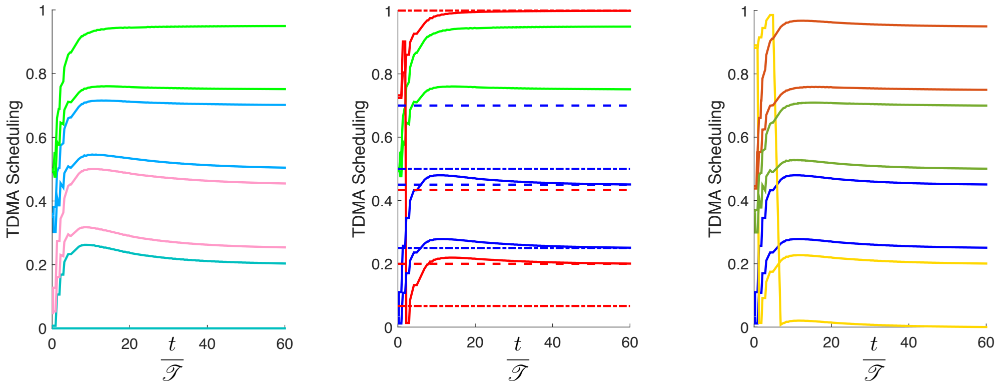


(a) $\mathcal{V}_1$

(b) $\mathcal{V}_2$

Figure 2.6.5: TDMA Scheduling for a two cliques topology

consecutive portions of the frame, therefore Assumption 2.5.5 is satisfied and in light of Theorem 2.5.6 we have convergence to the unique fixed point that satisfies the *global proportional fairness* criterion. In the plot, the start and end timers of each node are shown with the same color, solid lines represent the evolution of the timers and dashed lines represent the predicted fixed point from our analysis in 2.5.3.1. In Fig. 2.6.6 we present the case discussed in Appendix A where we have a set of possible fixed points, i.e. a set of attainable schedules. While each node in $\mathcal{V}_1$ and in $\mathcal{V}_3$ reaches its one and only possible schedule, the guard-space between the two shared nodes in $\mathcal{V}_2$ allows a range of fixed pointsfor the local node in $\mathcal{V}_2$. The range is limited on



(a) $\mathcal{V}_1$ (b) $\mathcal{V}_2$ (c) $\mathcal{V}3$

Figure 2.6.6: TDMA scheduling of the topology in Fig. 2.6.1. Note that the plots a-b-c are circular, thus '1' is adjacent to '0', and thus the 'jump' of the yellow node in $\mathcal{V}_3$.

both sides such that the local node in $\mathcal{V}_2$ is never the predecessor or successor of any of the two shared nodes. In Fig. 2.6.6b we show the range of possible start beacons Φ with two dashed lines and the range of end beacons Ψ with a dash-dotted line of the same color in the simulation. In Fig. 2.6.7 we plot the histogram of different shares obtained by the node in $\mathcal{L}_2$ obtained by MonteCarlo simulations. We can see the range is the one predicted by our equations in Appendix A. A pleasant result is that in larger number of cases ($\sim 43\%$) the local node gets the maximum share possible, i.e. *global proportional fairness* is often obtained, even though the conditions in Assumption 2.5.8 are not met.

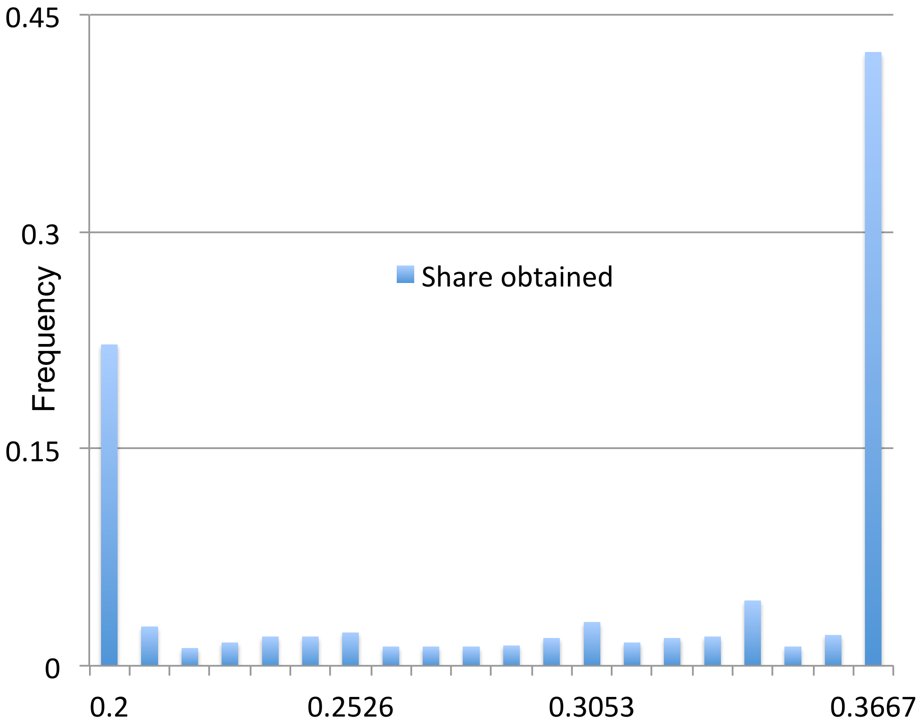


Figure 2.6.7: Transmission share obtained by the local node in $\mathcal{L}_2$ at convergence, in the range predicted in Appendix A.

CHAPTER 3
COGNITIVE UTILITY MAXIMIZATION MULTIPLE ACCESS-
THE CUMMA RECEIVER

3.1 Chapter Outline

In this chapter we discuss the design of a Cognitive Utility Maximization for Multiple Access (CUMMA) Receiver for opportunistic transmission. Our receiver combines the theory on sequential sensing with the principles of Sub-Nyquist sampling, and therefore this chapter starts by giving a background on the research in these two fields. In Section 3.3, we consider single-band spectrum sensing and formulate the optimization of the receiver as a partially observable Markov decision process (POMDP). To extend the application of our framework beyond spectrum sensing, we formulate the problem abstracting the PHY Layer: the connection will then be tightened in Section 3.5. In this abstraction, we can refer to each sub-band as a "resource" and indicate the spectrum occupancy with a binary vector $\mathbf{s}$ of states, that can be declared empty ($s_i = 0$, absence of communication over this sub-band) or busy ($s_i = 1$ presence of communication over this sub-band). If a sub-band is correctly identified as "empty" the decision-maker can accrue a reward, vice versa if the decision-maker declared as empty a busy sub-band, then a penalty is charged. A policy of the formulated POMDP consists of three components: 1) a set of selection rules indicating which resources to sense at each time, 2) a set of stopping rules, and the induced stopping times, indicating when to terminate the sensing phase of each resource, and 3) a set of decision rules governing whether to exploit or discard a resource after sensing. In 3.3.1 we explicitly characterize the optimal decision rules and show that the optimal stopping rule is given by two time-varying thresholds that also depend on the resources with pending decision and, since characterizing the time-varying thresholds and the optimal resource selection rules is analytically intractable, we develop a low-complexity suboptimal policy (described in 3.3.2): based on insights gained from the monotone properties of the thresholds in the single-resource case, we develop a recursive algorithm for computing approximate thresholds. A discussion on the asymptotical regret can be found in 3.3.3. Since the number of sub-band potentially accessible can be very large, especially in the mm-Wave spectrum where 5G is going to land, it would be desirable to obtain informations from multiple sub-bands at the same time, and the infeasibility of sampling at such high frequencies calls for the use of Sub-Nyquist sampling. In Section 3.4 we then investigate the possibility of mixing different sub-bands. Unfortunately, the study of a similar POMDP approach would suffer from a curse of dimensionality in the action space: to gain insight on the opportunity of mixing different sub-bands, a group-testing inspired strategy is proposed and set functions optimization tools are used to optimize the sensing phase. First, we connect this different strategy with the Direct Inspection case (i.e. same hardware limitation of 3.3) in 3.4.1, then we move to consider the mixing of the sub-bands in 3.4.2 starting from a pairwise test, i.e. test that mix only two sub-bands in 3.4.2.1, to then move to consider extension to a higher number of sub-bands in 3.4.2.2. Under the constraints on the strategy we are able to develop a constant factor approximation greedy algorithm, whose complexity remains polynomial in the number of resources. We also discuss the impact of noise in the *non-coherent* detection scheme proposed, and derive a Compressive-Sensing maximum likelihood estimate in 3.4.2.4. In Section 3.5 the analog front-end sampling of our *non-coherent* receiver is described and the parameter of the value function for the proposed POMDP and group testing strategy are mapped into the physics of the real application scenario. Finally in Section 3.6 we present simulation results to provide numerical evidence of the advantages of our proposed strategies.

3.2 Background

3.2.1 Sequential Sensing

A majority of existing work on sequential sensing focuses on typical detection metrics: minimizing the expected detection delay subject to constraints on detection accuracy in terms of the probabilities of false alarm and miss detection. This body of work can be partitioned into two general categories: passive sequential hypothesis testing and active sequential hypothesis testing. The former was pioneered by Wald in 1947 [Wald(1947)], which introduced the procedure known as Sequential Probability Ratio Test (SPRT). Under this formulation, the observation model under each hypothesis is predetermined, and a test only needs to determine when to stop taking observations and which hypothesis to declare at the time of stopping. The latter was pioneered by Chernoff in 1959 [Chernoff(1959)]. Active hypothesis testing has a control aspect that allows the decision maker to choose the experiment to be conducted at each time. Different experiments generate observations from different distributions under each hypothesis. An active test thus includes a selection rule (i.e., which experiment to carry out at each given time) in addition to a stopping rule and a declaration rule. Following these seminal results, there has been an extensive body of work on both passive (see, for example, [Arrow *et al.*(1949), Woodroofe(1976), Siegmund(1985), Veeravalli *et al.*(1993), Ganesan *et al.*(2008), Sahu and Kar(2016)] and references therein) and active (see [Bessler(1960), Lai *et al.*(2011), Nitinawarat *et al.*(2013), Naghshvar and Javidi(2013a), Naghshvar and Javidi(2013b), Cohen and Zhao(2015a)] and references therein) hypothesis testing. of delay and miss detection and false alarm probabilities (see, for example, [Wald and Wolfowitz(1948), Arrow *et al.*(1949), Woodroofe(1976), Siegmund(1985), Veeravalli *et al.*(1993), Ganesan *et al.*(2008), Lai *et al.*(2011)]). In these models a decision maker designs a set of tests for over K instants in order to determine the state of certain resources [Caromi *et al.*(2013)]. It is well known this problem has a dual unconstrained formulation, where the objective that has to be minimized is the sum of a sensing cost and a so-called "Bayesian-risk" term. However there is no closed form result that indicates how to compute the multipliers in the Bayesian-risk for given error probabilities target, and the previous literature provides different approximations to address this duality. In our framework, no hard constraints are placed on the error probabilities and the decision market strategy is exclusively devoted to maximize a utility function which includes a sensing cost and a reward/penalty depending on the action and the real state of the resource. The main novelty of our model relies on a "time-dependent" utility after the decision. This is motivated by many applications, where a delay cost associated to the time devoted for testing can have different physical reasons than the utility a right or wrong action (i.e. declaration) can provide for the remaining time.

The key difference between this work and the vast body of results on hypothesis testing is that the design objective in the problem studied here is the utility maximization that directly addresses the trade-off between exploration (i.e., detecting the state of each resource) and the time that remains for exploitation (of the resources, based on the information gathered during the sensing phase). Hypothesis testing problems, passive or active, are pure exploration problems. There are a couple of studies on sequential sensing for anomaly detection under the objective of minimizing operational cost [Cohen *et al.*(2014), Cohen and Zhao(2015b)] which can be considered as a utility function. Different from our work, these studies either restrict admissible sensing strategies to those that declare the state of one process before starting sensing another process (i.e., no switching across processes) [Cohen *et al.*(2014)] or focus on asymptotically (when the horizon length approaches infinity) optimal policies. The problem considered in this thesis allows switching across resources and focuses on a finite horizon, resulting in a full-blown POMDP that is significantly more difficult to tackle (for a discussion on the general complexity of POMDP see [Mundhenk *et al.*(1997)]).

In the Cognitive Radio literature our utility maximization problem can be seen as a parametric formulation of the energy efficiency metric maximization [Wu *et al.*(2014), Pei *et al.*(2011)]. In [Wu *et al.*(2014)] the presence of PU communication is inferred via energy detection (known to be optimal when no prior information is available) and a single PU channel was considered. Multiple PU channels and the capability of switching between different channels, known as *spectrum handoff*, is investigated in [Wang *et al.*(2012a)] and an efficient convex optimization procedure is developed to solve for the optimal values of the sensing slot duration and the channel switching probability in order to minimize energy consumption while guaranteeing satisfying throughput. In [Pei *et al.*(2011)] prior knowledge (the vector $\boldsymbol{\omega}$ in our model) over the state of different channels is considered, but the sequential decision process terminates when the SU decides to transmit over one single channel. The problem is formulated as DP but no heuristic is provided to tackle the combinatorial complexity and no further insight on the threshold structure of the decision is proposed. The threshold structure for the channel probing stopping rule that has been proved in [Chang and Liu(2009)] and that [Pei *et al.*(2011)] refers to, also considers only one possible transmission and a constant data time scenario, i.e. the transmission time is not affected by the time spent in sensing the channels. Moreover, in [Wu *et al.*(2014)]- [Wang *et al.*(2012a)] no prior information is available and all the channels are equal. Therefore, there is no ordering of the channels to take into account and no SPRT procedure to be optimized for the sensing (performed via energy detection with a deterministic sensing time), whereas our model tackles both aspects.

More general utility maximization approaches for cognitive networks can be found in [Zheng *et al.*(2013)] and [Zheng *et al.*(2016)], which leverage the class of utility maximization functions for optimal congestion and contention control introduced in [Lee *et al.*(2006)]. A censored truncated sequential spectrum sensing procedure is presented in [Maleki and Leus(2013)], where different cognitive radios sense the same channel and decide whether to send their estimates to a fusion center that then performs the final decision over the presence of a PU. We instead consider a single cognitive radio that can sense different channels (but only one at a time), which could be seen as using only one sensor per channel, and therefore each sensor is sensing a different state. The limit of one sensing operation at a time, in our formulation, could be seen as a rigid censoring constraint, with the possibility of suspending the decision over a channel, while continuing to sense and/or exploit others, and then potentially reconsider whether to sense or exploit that channel after some time instants. A concatenation of SPRT is proposed as a low-complexity, asymptotically optimal policy in [Caromi *et al.*(2013)].

Additional relevant works can be found in [Liang *et al.*(2008), Zhao *et al.*(2007), Yucek and Arslan(2009), Oksanen *et al.*(2010), Quan *et al.*(2008), Akyildiz *et al.*(2008), Letaief and Zhang(2009), Kim and Shin(2008), Zeng and Liang(2009), Bagheri and Scaglione(2015)]. The vast majority considers a rigid separation between exploration and exploitation phase, while in our framework we enable a combination of the two over different resources, by accessing some channels and simultaneously sensing a different resource at each time. Several works have adopted a Multi-Armed Bandit (MAB) formulation (see for example [Bagheri and Scaglione(2015), Gai *et al.*(2010)]), whereas others (including this work) followed a Partially Observable Markov Decision Process (POMDP) framework [Zhao *et al.*(2007), Krishnamurthy *et al.*(2016)]. It is important to highlight the following difference: in the MAB formulation the utility obtained by the player that selects a certain "arm" (i.e. an action) is the only information that is used to optimize the future choices. The concept of POMDP, instead, can be used to cast a wider class of decision problems where at each time epoch a certain action needs to be designed, that provides indirect information on what strategy the player should use to harness the maximum utility. In other words a 'sensing' action informs what the player should do. The MAB formulation is a special case of the POMDP in which the sensing action and the action that brings the utility to the player are the same thing. Therefore, the POMDP formulation allows in principle for a richer action space than a MAB problem and a POMDP cannot necessarily be mapped into a MAB problem. Other optimal sequential sensing methods to address the problem of optimally choosing the band to explore can be found in [Zhao *et al.*(2008), Unnikrishnan and Veeravalli(2010), Wang and Chen(2012)].

3.2.2 Xampling Architectures

In the second part of the Chapter, we look at the opportunity of using a sub-Nyquist sampling front-end which, in principle, can scan the entire spectrum at once, reducing significantly the sensor analog to digital conversion hardware complexity and its energy cost, at the expenses of increased complexity in the reconstruction of the underlying signal. The theoretical foundation for the design of sub-Nyquist sampling front-end lies in the representation of the signal as a nonlinear Union of Subspaces (UoS) [Lu and Do(2008)]. A common framework to cover several acquisition and reconstruction approaches under the umbrella of the UoS model was defined in [Mishali *et al.*(2011)], which named these analog to digital conversion techniques *Xampling architectures*. Xampling architectures preprocess the signal in the analog domain, and then sample at a lower rate compared what the Nyquist theorem dictates. The aim is to reduce the complexity and energy cost for the analog-to-digital conversion hardware. The downside is the increased complexity in the reconstruction of the underlying signal. Our work focuses on *multiband signals*, whose UoS representation is a *finite* union of subspaces with *infinite*, but countable dimensions, in the space spanned by orthogonal *sinc* functions. Examples of Xampling architectures for multiband signals are in e.g. [Fudge *et al.*(2008), Mishali and Eldar(2009), Mishali and Eldar(2010), Venkataramani and Bresler(2000)]. There are other CSS algorithms in the literature that are related. Typically, they start directly from a discrete time model (see e.g. [Zeng *et al.*(2011), Zeinalkhani and Banihashemi(2012)]) where the receiver has a fixed number of measurements, forming an underdetermined system of equations, whose solution is a sparse vector with support equal to the spectrum occupancy. We provide two arguments to study alternatives in the cognitive spectrum access problem. First, in the spectrum sensing problem, the objective is the detection of the idle channels, not the signal reconstruction: this suggests that the Xampling complexity may still exceed what is really necessary for this task, as previously discussed in [Cohen and Eldar(2014)]. Note also that, when considering the detection problem, most of the *standard* results in Compressive Sensing (CS), that bound the ℓ_2-norm of the estimation error, do not directly express the detection performance. The architecture studied in this thesis has the advantage of being sequential, requiring incoherent observations and being robust to time inaccuracies in the sampling hardware, as opposed to e.g. the multi-coset approach in [Venkataramani and Bresler(2000)]. For the spectrum sensing detection problem, the additive noise at the receiver plays an important role in the performance of interest. Hence, rather than focusing on reconstruction in *noiseless* scenarios, in our work we directly tackle the so called *noise folding* problem in the design [Arias-Castro and Eldar(2011)]. Noise folding gives an SNR deterioration approximately linear in the number of bands that are aliased prior to sampling [Arias-Castro and Eldar(2011)]. This can cause poor performance for several Xampling approaches at low SNR. As discussed in [Baron *et al.*(2010)], low density measurement matrices represent an effective countermeasure to noise folding. Additionally sparse matrices enable belief propagation techniques (i.e. message passing) for signal recovery (or detection in our context). These advantages compensate the drawbacks of using matrices that are generally worse conditioned than dense matrices. Second, in the cognitive sensing problem the direct application of Xampling architectures (see e.g. [Fudge *et al.*(2008), Mishali and Eldar(2009), Mishali and Eldar(2010), Venkataramani and Bresler(2000), Lexa *et al.*(2011), Ariananda and Leus(2012), Cohen and Eldar(2014)]), or any receiver that does not adapt its measurement strategy, poses a limitation: if the spectrum is not sufficiently sparse, neither the signal reconstruction, nor the detection of its presence in a certain band, can be accurate, even in the absence of noise. In fact, for general non-sparse signals, in a *noiseless* setting, [Cohen and Eldar(2014)] proved that half the Nyquist rate is necessary (see also [Lexa *et al.*(2011), Ariananda and Leus(2012)] for related discussions). This inability to adapt to the ever changing spectrum occupancy rate, makes its real application to cognitive spectrum sensing impractical. More recently, other authors have proposed and studied sensing strategies that would reduce the required number of measurements adaptively using compressive sensing, optimizing the sensing actions based on previous observations [Bagheri and Scaglione(2015), Hao *et al.*(2012), Malioutov *et al.*(2010), Zhao *et al.*(2015), Braun *et al.*(2015)]. The common goal in these works is the recovery of the full support of a given vector. Typically, the techniques proposed are shown to be able to cope with lower SNR in the signal reconstruction with low complexity. What these optimization do not capture is the fact that in cognitive spectrum sensing applications it is also desirable to have enough time to exploit the spectrum by making a timely decision. A cross-layer framework to jointly optimize spectrum sensing and scheduling is presented in [Michelusi and Mitra(2015)], where however the sensing phase duration is pre-determined and the policy optimizes the average *sensing traffic*, whereas in our model a dynamic optimization on the number of measurement is discussed based on recovery guarantees. In fact, the receiver collects energy measurements sequentially (sampling at a fraction of the Nyquist rate) under the assumption that the activity of the Primary Users (PUs) in a certain spectrum will persist for several sampling periods (which is a reasonable assumption in this context). The novelty in the proposed architecture stems from the idea of selecting *opportunistically* the signal used for aliasing the wide-spectrum input, so as to mix selective portions of the spectrum differently each time we perform an energy measurement.

Notation

We use bold lower-case to represent vectors, bold upper-case for matrices and calligraphic letters to indicate sets. For vectors, we use the same letter upper-case to indicate the $||\cdot||_1$ norm (i.e. $\Omega = ||\boldsymbol{\omega}||_1$) and with the notation $\mathbf{s}_{\mathcal{A}}$ we select the entries $i \in \mathcal{A}$ of vector $\mathbf{s}$. With $\|\mathbf{y}\|^2_{\mathbf{A}}$ we indicate the weighted ℓ_2 norm $\mathbf{y}^T\mathbf{A}\mathbf{y}$. For any set function $f(\mathcal{A})$ we define the marginal increment for adding element a, as $\partial_a f(\mathcal{A}) = f(\mathcal{A} + a) - f(\mathcal{A})$.

3.3 A POMDP Formulation

We consider the problem of optimally utilizing N resources over a horizon of length K. The state of each resource is either 0 ("good") or 1 ("bad") and is unknown *a priori*. Utilizing resource i results in either a reward r_i or a penalty ρ_i per unit time, depending on its state. To infer the states of the resources, only one of them can be sensed at each given time, accruing a random measurement drawn from a

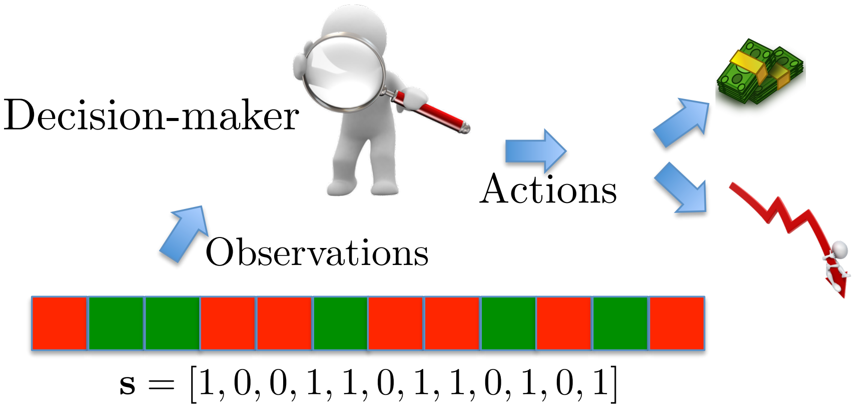


Figure 3.3.1: The decision-maker needs to determine the true state s_i of the resources in order to accrue the maximum utility.

distribution determined by the state of the chosen resource. We study the optimal sequential strategy governing the decision of sensing and exploitation at each time, that maximizes the expected utility (i.e., total reward minus total penalty) over the finite horizon (see Fig.3.3.1). Since sensing reduces the available time for utilization, the essence of the problem is in balancing the overhead associated with sensing and the penalty of utilizing resources in a bad state. Due to the limited sensing capability (i.e., only one resource can be sensed at a given time), it is also crucial to choose judiciously which resource to sense at each time. This problem arises in cognitive radio systems where the resources are channels that can be either busy or idle. Assume an agent has K instants of time available for the sensing and exploitation of a set of resources $\mathcal{N} = \{1, 2, \dots, N\}$. The agent accrues a reward that is a function of an underlying state vector

$$\mathbf{s} \triangleq (s_1, \dots, s_N)^T \in \mathbf{S} \equiv \{0, 1\}^N, \tag{3.3.1}$$

where the entries $s_i \in \{0, 1\}$ and $\mathbf{S}$ is the set of all possible states. We consider the s_i as indicators of good (0) or bad (1) state of a resource. For instance, the "idle" or "busy" state of a sub-band, where the decision maker has to explore/sense the channels and gets a reward for utilizing an idle channel and a penalty for utilizing a busy channel. We assume the states s_i are mutually independent Bernoulli random variables with known prior probabilities given by:

$$\boldsymbol{\omega}[1] = \{\omega_i[1] : i = 1, 2, \ldots, N\} \tag{3.3.2}$$

$$\omega_i[1] \triangleq P(s_i = 0). \tag{3.3.3}$$

Let $\mathcal{A}_k$ denote the set of resources for which a final decision of utilizing or discarding has not been reached at time k ($k = 1, \ldots, K$). Clearly, $\mathcal{A}_0$ includes all N resources. Our model then allows to access multiple resources (removed from $\mathcal{A}_k$) and to sense one resource (ϕ_k) at each time instant k. Underneath the decision there is a sequential binary hypotheses testing problem where a sample $y[k]$ from the selected resource ϕ_k is collected and the conditional probability density functions of the observations are assumed to be known:

$$H_0^{\phi_k} : s_{\phi_k} = 0, \quad y[k] \overset{i.i.d}{\sim} f_0^{\phi_k}(y) \tag{3.3.4}$$

$$H_1^{\phi_k} : s_{\phi_k} = 1, \quad y[k] \overset{i.i.d}{\sim} f_1^{\phi_k}(y) \tag{3.3.5}$$

We want to maximize a utility function that strikes the best trade-off between the need of acquiring information on the state $\mathbf{s}$ of the channels and the desire of exploiting good resources as early as possible. The decision maker needs to design: 1) a set of N stopping rules for the stopping times $\boldsymbol{\tau} \triangleq \{\tau_i : i = 1, 2, \ldots, N\}$, one for each resource, indicating when a final decision of utilizing or discarding the resource can be made; 2) a set of N decision rules $\boldsymbol{\delta} \triangleq \{\delta_i \in \{0(\text{utilize}), 1(\text{discard})\} : i = 1, 2, \ldots, N\}$, one for each resource, indicating the final decision at the time τ_i of stopping; 3) a sequence of selection rules $\boldsymbol{\phi} \triangleq \{\phi_k : k = 1, \ldots, K\}$ indicating which resource to sense at time k (if $\mathcal{A}_k$ is not empty). Let τ_i denote the time instant at which a final decision on whether to utilize or discard resource i is made and $\tilde{\tau} = \max_i \tau_i$ be the total sensing time, all depending on the two sets of rules $(\boldsymbol{\tau}, \boldsymbol{\phi})$. The decision maker's actions are the solutions to the following optimization problem:

$$\begin{aligned} &\underset{\boldsymbol{\tau},\boldsymbol{\delta},\boldsymbol{\phi}}{\text{max.}}\ \ \mathbb{E}[U(\mathbf{s},\mathcal{N},K,c,\boldsymbol{\tau},\boldsymbol{\delta},\boldsymbol{\phi})] \triangleq \mathbb{E}\left[-c\tilde{\tau}+\sum_{i=1}^{N}(K-\tau_i)R_i(s_i;\delta_i)\right] \\ &\text{subject to}\quad \tilde{\tau}\leq K. \end{aligned} \tag{3.3.6}$$

where the objective function has two terms:

- $-c\tilde{\tau}$ represents an effective sensing cost, and c is the sensing cost per unit of time;

- the function $R_i(s_i;\delta_i)$ is the utility per unit time for exploiting resource i:

$$R_i(s_i;\delta_i) = \begin{cases} r_i & \text{if}\ \ \delta_i = s_i = 0 \\ -\rho_i & \text{if}\ \ \delta_i = 0, s_i = 1 \\ 0 & \text{if}\ \ \delta_i = 1 \end{cases} \tag{3.3.7}$$

  where $r_i, \rho_i > 0$ indicate, respectively, the reward and the penalty for utilizing a *good* resource and utilizing a *bad* resource.

We can immediately notice an asymmetry in the function corresponding to the *exploitation reward* in (3.3.7) since, if one decides the resource is in *bad* state (i.e. that $s_i = 1$), the utility accrued is the same regardless of the real state of the channel.

Note that, since c can be moved out of the expectation in (3.3.6), $\mathbb{E}\left[U^*(\mathbf{s},\mathcal{N},K,c)\right] = \max\limits_{\tau,\delta,\phi} \mathbb{E}\left[U(\mathbf{s},\mathcal{N},K,c,\boldsymbol{\tau},\boldsymbol{\delta},\boldsymbol{\phi})\right]$ decreases monotonically with c, therefore the value of c is typically limited by an effective cost (i.e. the energy required for the receiver to continue sensing a channel in a spectrum access problem) and is not chosen to further optimize the achievable expected utility. We now procede to show how the optimization in (3.3.6) can be modeled as a Partially Observable Markov Decision Problem (POMDP), and use dynamic programming tools to describe the optimal $\boldsymbol{\tau}^*, \boldsymbol{\delta}^*, \boldsymbol{\phi}^*$. The state vector $\mathbf{s}$ is not directly observable, therefore the decision maker has to rely on her belief regarding the occupancy of the resources in order to make a decision. Let $\boldsymbol{\omega}[k]$ denote the belief vector at instant k (i.e., the vector of posterior probabilities that a resource is in state 0 given all past observations). We can use Bayes rule to derive the belief update after a new observation and write

$$\boldsymbol{\omega}[k+1] = \boldsymbol{\Pi}(\boldsymbol{\omega}[k], y[k], \phi_k) \tag{3.3.8}$$

$$\omega_i[k+1] = \Pi_i(\omega_i[k], y[k], \phi_k) \tag{3.3.9}$$

$$\Pi_i(\omega_i, o, \phi) \triangleq \begin{cases} \dfrac{\omega_i f_0^i(y)}{\omega_i f_0^i(y) + (1-\omega_i) f_1^i(y)} & \text{if } \; i = \phi \\ \omega_i & \text{if } \; i \neq \phi \end{cases} \tag{3.3.10}$$

The optimization problem in (3.3.6) can therefore be formulated as a POMDP where the belief vector $\boldsymbol{\omega}[k]$ represents the *state* for the decision maker and the state transitions equations are given by the belief update rule in (3.3.8). At each time instant k, based on the current belief $\boldsymbol{\omega}[k]$, the decision maker first removes from $\mathcal{A}_k$ all those resources for whom a final decision of utilizing or discarding can be made (we refer to this set as $\mathcal{D}$), and then she chooses one of the remaining resources to sense at time k. The value function of the POMDP problem can be expressed as:

$$V(\boldsymbol{\omega}, \mathcal{A}_k, k) \triangleq \max_{\mathcal{D} \subseteq \mathcal{A}_k} \left\{ (K-k+1) V_d(\boldsymbol{\omega}, \mathcal{D}) + \max_{i \in \mathcal{A}_{k+1} \equiv \mathcal{A}_k \setminus \mathcal{D}} V_t^i(\boldsymbol{\omega}, \mathcal{A}_{k+1}, k) \right\} \tag{3.3.11}$$

where $V_d(\boldsymbol{\omega}, D)$ indicates the maximum expected reward given by the resources in $\mathcal{D}$ for each of the $(K-k+1)$ remaining instants and $V_t^i(\boldsymbol{\omega}, \mathcal{A}, k)$ represents the value accrued for deciding to sense the resource i at time k (i.e. $\phi_k = i$).

It is easy to see from the utility function (3.3.7) that V_d has an additive structure, i.e. $V_d(\boldsymbol{\omega}, \mathcal{D}) = \sum_{i \in \mathcal{D}} V_d^i(\omega_i)$. Mathematically the functions $V_d^i(\omega_i)$ and $V_t^i(\omega_i, \mathcal{A}, k)$ can be expressed as follows:

$$V_d^i(\omega_i) \triangleq \max_{\delta_i \in \{0,1\}} \mathbb{E}\left[R_i(s_i; \delta_i)\right] \tag{3.3.12}$$

$$V_t^i(\boldsymbol{\omega}, \mathcal{A}, k) \triangleq -c + \int V(\mathbf{\Pi}(\boldsymbol{\omega}, y, i), \mathcal{A}, k+1)\, f_{1-\omega_i}^i(y) dy \tag{3.3.13}$$

for $k = 1, \dots, K$ where we need a final condition $V(\boldsymbol{\omega}, \mathcal{A}, K+1) = 0, \forall \mathcal{A}, \boldsymbol{\omega}$, to encode the constraint in (3.3.6). In (3.3.13) we have defined

$$f_{1-\omega_i}^i(y) \triangleq \omega_i f_0^i(y) + (1 - \omega_i) f_1^i(y) \tag{3.3.14}$$

It is easy to see that

$$V_d^i(\omega_i) = \max\{(r_i + \rho_i)\omega_i - \rho_i, 0\}. \tag{3.3.15}$$

Thus the optimal final decision δ_i^* is given by

$$\delta_i^* = u\left(\frac{\rho_i}{\rho_i + r_i} - \omega_i\right) \tag{3.3.16}$$

where $u(\cdot)$ is the unit step function. The value function $V(\boldsymbol{\omega}, \mathcal{A}_k, k)$ can be seen as the result of the maximization of a set function under set constraints, that is:

$$V(\boldsymbol{\omega}, \mathcal{A}_k, k) = \max_{\mathcal{D} \subseteq \mathcal{A}_k} J(\mathcal{D}) \tag{3.3.17}$$

where

$$J(\mathcal{D}) \triangleq \left\{ (K - k + 1) V_d(\boldsymbol{\omega}, \mathcal{D}) + \max_{i \in \mathcal{A}_k \backslash \mathcal{D}} V_t^i(\boldsymbol{\omega}, \mathcal{A}_k \backslash \mathcal{D}, k) \right\} \tag{3.3.18}$$

and this formulation will be used in the remainder of this work to show the structure of the optimal policy. We would like to point out that the formulation of the POMDP in (3.3.11) gives no indication on the fact that the channel sensed at time k should continue to be sensed at time $k+1$ or included in the set $\mathcal{D}$, i.e. a concatenation of independent truncated SPRT over each channel represents a suboptimal strategy for our problem and is optimal only for $K \to \infty$. It is useful to remark that, by considering a *time-dependent* utility after the decision, in our model the constraints on the detection metrics vary between channels and over time, whereas in the majority of other works the detection metrics constraint are typically constant over time (see [Wald(1947)] for analysis of truncated sequential hypothesis). To be more precise, our model can be seen as a Bayesian-risk formulation, where our utility terms can be seen as the Lagrangian multiplier of the constraints associated to the detection metrics, that change over time in light of the *time-dependent* utility function.

Remark 3.3.1. *Our formulation can be modified to account for correlation between group of resources. If we keep the condition that it is possible to sense only one resource at each time, the belief update rule in* (3.3.8) *should be modified to update also the ω_i's of resources correlated with the sensed one. The subset of resources, that can be accessed at each time, should contain all the ones correlated with each other, i.e. if two resources are correlated, they should be added to $\mathcal{D}$ at the same time, due to the possibility of gaining knowledge on a resource for which a terminal decision has already been made. It follows that, for each group of correlated resources, we have to consider a sub multi-hypothesis problem for all the possible combinations of binary states, and find a similar approximation as the one presented in the next section for the decision regions in the plane of the belief vector over these resources (for details on the geometry of such structure see [Baum and Veeravalli(1994)]). This in principle can be handled for small groups of correlated resources, but makes the problem even more complex.*

3.3.1 The Optimal Stopping Rule and Decision Rule

In this section we first describe the correspondence between the actions of our decision process as solution of (3.3.11) and the optimal rules $(\boldsymbol{\tau}^*, \boldsymbol{\phi}^*)$ introduced above, that the decision maker needs to determine. Then we introduce a low-complexity policy to approximate the optimal action. Let us start by considering the solution of

(3.3.17) at time k, i.e.:

$$\mathcal{D}^* = \arg\max_{\mathcal{D}\subseteq\mathcal{A}_k} J(\mathcal{D}). \tag{3.3.19}$$

By definition of the set of optimal stopping times $\boldsymbol{\tau}$ (see (3.3.6)), for each resource i, the optimal stopping time τ_i^* is:

$$\tau_i^* = \begin{cases} k-1, & \text{for } \ i \in \mathcal{D}^* \\ k' > k-1 & \text{for } \ i \in \mathcal{A}_k \setminus \mathcal{D}^*. \end{cases} \tag{3.3.20}$$

Notice that since we start from instant $k = 1$ if we decide to immediately stop the sensing we have $\tau = 0$ and this is consistent with the formulation in (3.3.6) and the following mapping to the value function. In light of the outer maximization over $\mathcal{D}$ in (3.3.11), the optimal selection rule for the decision maker will be:

$$\phi_k^* = \arg\max_{i\in\mathcal{A}_{k+1}\equiv\mathcal{A}_k\setminus\mathcal{D}^*} V_t^i(\boldsymbol{\omega}, \mathcal{A}_{k+1}, k) \tag{3.3.21}$$

From (3.3.20) and (3.3.21) it emerges how the optimal selection rules $\boldsymbol{\phi}^*$ and the optimal stopping times $\boldsymbol{\tau}^*$ are coupled. We then introduce the following lemma:

Lemma 3.3.2. *$\forall\mathcal{A}_k \subseteq \mathcal{N}$, $i, j \in \mathcal{A}_k$ and for $k = 1, \ldots, K$, both $V_t^i(\boldsymbol{\omega}, \mathcal{A}_{k+1}, k)$, $V(\boldsymbol{\omega}, \mathcal{A}_k, k)$ are convex functions of ω_j.*

Proof. The proof is in Appendix B. It is similar to the approach used in [Zhao and Ye(2010)] to prove the convexity of the Q-functions. ■

Lemma 3.3.2 induces the structure of the optimal stopping rules, which describe the optimal stopping times τ_i^*, $i = 1, 2, \ldots, N$ that we formally present in the following theorem:

Theorem 3.3.3. *The optimal stopping time $\tau_i^*, \forall i \in \mathcal{N}$ is described by two thresholds (ν_1^i, ν_0^i) that depend on the remaining time and channels to be explored, i.e. $(\nu_1^i = \nu_1^i(\mathcal{A}_k, k), \nu_0^i = \nu_0^i(\mathcal{A}_k, k))$ such that at any k the optimum action for the resource i is*

- *take a final decision over resource i (i.e. $\tau_i = k$) if*

  $\omega_i[k] \geq \nu_0^i(\mathcal{A}_k, k) \;\vee\; \omega_i[k] \leq \nu_1^i(\mathcal{A}_k, k)$

- *postponing the decision on resource i (i.e. $\tau_i > k$), if*

  $\nu_1^i(\mathcal{A}_k, k) < \omega_i[k] < \nu_0^i(\mathcal{A}_k, k)$

Furthermore $\forall i, \mathcal{A}_k, k$

$\nu_1^i(\mathcal{A}_k, k) \leq \frac{\rho_i}{\rho_i + r_i} \leq \nu_0^i(\mathcal{A}_k, k)$ *and from* (3.3.16) *it follows that:*

$$\delta_i^* = \begin{cases} 1 & \text{if } \omega_i[k] \leq \nu_1^i(\mathcal{A}_k, k) \\ 0 & \text{if } \omega_i[k] \geq \nu_0^i(\mathcal{A}_k, k). \end{cases} \tag{3.3.22}$$

Proof. Following the same approach we used to show the convexity of $V(\boldsymbol{\omega}, \mathcal{A}_k, k)$ in Appendix B, we rewrite (3.3.11) for an arbitrary resource i as follows:

$$V(\boldsymbol{\omega}, \mathcal{A}_k, k) = \max \left\{ \max_{\{i\} \subseteq \mathcal{D} \subseteq \mathcal{A}_k} J(\mathcal{D}), \max_{\mathcal{D} \subseteq \mathcal{A}_k \setminus \{i\}} J(\mathcal{D}) \right\} \tag{3.3.23}$$

where the function $J(\mathcal{D})$ has been defined in (3.3.18), and then we refer to the two terms of the outer maximization as $f_1 = \max\limits_{\{i\} \subseteq \mathcal{D} \subseteq \mathcal{A}_k} J(\mathcal{D})$ and $f_2 = \max_{\mathcal{D} \subseteq \mathcal{A}_k \setminus \{i\}} J(\mathcal{D})$. Note that f_1 corresponds to the maximum value from the actions that immediately decide on resource i, whereas f_2 indicates the maximum value from the actions that do not decide on resource i at time k. The convexity of f_1 and f_2 is proven in Appendix B. Clearly $f_1 \geq f_2$ for $\omega_i = 0$ and for $\omega_i = 1$ since, when the state of the channel is known, is clearly preferable (or equivalent) to immediately decide on it, and also f_1 is a piece-wise linear function of ω_i with only two segments that intersect in $\omega_i = \frac{\rho_i}{\rho_i + r_i}$ (see definition of $V_d^i(\omega_i)$ in (3.3.15)). Hence, there are only two possibilities:

- $f_1 < f_2$ for $\nu_1^i(\mathcal{A}_k, k) < \omega_i[k] < \nu_0^i(\mathcal{A}_k, k)$ where

  $0 \leq \nu_1^i(\mathcal{A}_k, k) \leq \frac{\rho_i}{\rho_i + r_i} \leq \nu_0^i(\mathcal{A}_k, k)$;

- $f_1 \geq f_2$ $\forall \omega_i \in [0, 1]$.

In this second case, we will say the two functions do not intersect, i.e. there is no region where the decision maker should prefer to not include the resource in $\mathcal{D}$; therefore we set $\nu_1^i(\mathcal{A}_k, k) = \nu_0^i(\mathcal{A}_k, k) = \frac{\rho_i}{\rho_i + r_i}$ to indicate the decision maker should immediately decide on resource i and this concludes the proof. $\blacksquare$

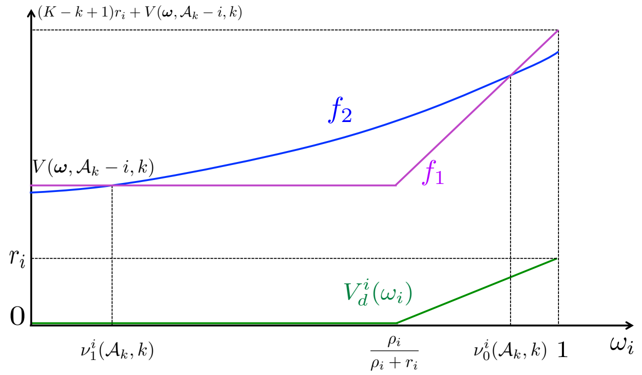


Figure 3.3.2: Representation of the functions of ω_i: f_1,f_2 from (3.3.23) at time k

It is not entirely surprising that the problem we defined leads to an optimal policy with a two thresholds structure, since analogous policies have been found to be optimal for a general truncated SPRT [Wald and Wolfowitz(1948)] with a deadline constraint either deterministic [Tartakovsky *et al.*(2015)] or stochastic [Frazier and Yu(2008)] and in general many different sequential decisions schemes. However, as described in the Introduction, our formulation is different in light of the time dependence of the Bayesian term, i.e. the different utility that a decision over a certain resource can produce at different times.

Theorem 3.3.3 is a description of the optimal stopping rules but does not indicate how to compute the two thresholds nor how to select the resource ϕ_k^* to be sensed at time k. Due to the recursive nature of the function $V_t^i(\boldsymbol{\omega}, \mathcal{A}_{k+1}, k)$ in (3.3.13) and the dependence on the rest of the system, the exact computation of these optimal thresholds remains elusive. In order to provide a suboptimal strategy that is computationally manageable in Section 3.3.2, let us look at the situation where we have only one resource and we have to choose between testing the resource or taking a final decision, either $\delta_i = 1$ or $\delta_i = 0$. We will refer to the two single-resource decision thresholds as $\nu_1^i[k] = \nu_1^i(i,k),\ \nu_0^i[k] = \nu_0^i(i,k)$, where we introduce this short notation for convenience. We then introduce the following lemmas.

Lemma 3.3.4. $\forall i \in \mathcal{N},\ k = 1,\ldots,K-1$ the two single-resource thresholds will monotonically contract, i.e.

$$\nu_1^i[k] \leq \nu_1^i[k+1] \tag{3.3.24}$$

$$\nu_0^i[k] \geq \nu_0^i[k+1] \tag{3.3.25}$$

Proof. The intuition behind this lemma is that, as the decision deadline approaches, the urge to decide whether to utilize or discard the resource increases and the decision thresholds shrink. For a rigorous proof see Appendix B. ■

Lemma 3.3.5. $\forall i \in \mathcal{N},\ k = 1,\ldots,K,\ \mathcal{A}' \in 2^{\mathcal{N}-i}$, the following inequalities hold

$$\nu_1^i[k] \leq \nu_1^i(\mathcal{A}'+i,k) \tag{3.3.26}$$

$$\nu_0^i[k] \geq \nu_0^i(\mathcal{A}'+i,k) \tag{3.3.27}$$

Proof. The intuition behind this lemma is that adding resources to the state $\mathcal{A}$ produces a similar effect to removing time dedicated to each resource. Thus a similar intuition as for Lemma 3.3.4 applies. The full proof is in Appendix B. ■

Lemma 3.3.6. $\forall k, k' = 0,\ldots,K,\ k' \geq k,\ \mathcal{A} \in 2^{\mathcal{N}}, i \in \mathcal{A}$

$$\nu_1^i[k] \leq \nu_1^i(\mathcal{A},k') \tag{3.3.28}$$

$$\nu_0^i[k] \geq \nu_0^i(\mathcal{A},k') \tag{3.3.29}$$

Proof. Follows from Lemmas 3.3.4-3.3.5 and setting $\mathcal{A} = \mathcal{A}' + i$. ■

Lemma 3.3.6 gives us insight on the behavior of the thresholds $(\nu_1^i(\mathcal{A}_k, k), \nu_0^i(\mathcal{A}_k, k))$ that motivates our heuristic strategy in Section 3.3.2, which uses a single-channel thresholds approximation to replace the optimal thresholds. In the next section we will first discuss how to approximate the thresholds $\nu_1^i[k]$, $\nu_0^i[k]$ and then we will introduce a manageable strategy to sense the resources and decide which action to take based on the approximate decision thresholds.

3.3.2 A Low Complexity Approximation Algorithm

For the rest of this subsection, we will use underline to indicate a lower bound and overline for upper bound of a given quantity or function, i.e. $\underline{a} \leq a \leq \overline{a}$. For our heuristic, as we will motivate in the following, we will use the bounds $\underline{\nu_1^i}[k]$ and $\overline{\nu_0^i}[k]$ that correspond to the single-resource decision thresholds, derived from the formulation previously introduced. In principle, the tighter are the bounds we can provide for the thresholds, the better our heuristic will perform. Let us start by finding simple bounds. The decision thresholds $(\nu_1^i[k], \nu_0^i[k])$ at time k are the solutions with respect to ω_i of

$$(K - k + 1)V_d^i(\omega_i) = V_t^i(\omega_i, i, k). \tag{3.3.30}$$

From the convexity of $V_t^i(\omega_i, i, k)$ (see Lemma 3.3.2) and the values of the function for $\omega_i = 0, 1$ we know that (see Fig.3.3.3)

$$V_t^i(\omega_i, i, k) \leq -c + (K - k)r_i\omega_i. \tag{3.3.31}$$

Furthermore, $V_d^i(\omega_i)$ is piece-wise linear with only two segments that intersect in $\omega_i = \frac{\rho_i}{\rho_i + r_i}$.

Remark 3.3.7. *A more general formulation with* 4 *different rewards/penalties for* (3.3.7) *would not alter the structure of the problem nor invalidate our results. Our*

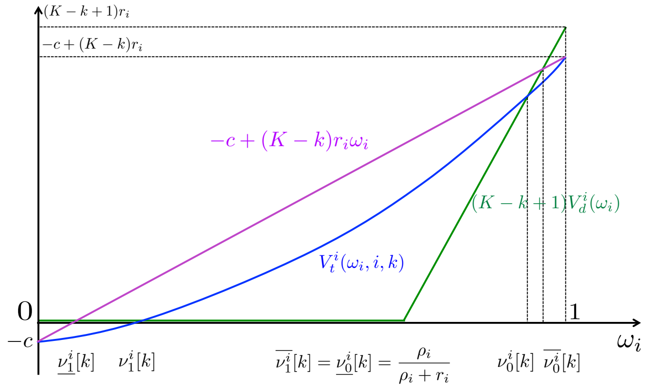


Figure 3.3.3: Representation of the two functions of ω_i: $V_t^i(\boldsymbol{\omega},i,k)$ associated to the test of resource i and $(K-k+1)V_d^i(\omega_i)$ for the immediate decision on resource i.

definition has been motivated by the emphasis we want to put on the utilization of the resources, which we assume occurs only if the detected state is 0*, but if one had defined*

$$R_i(s_i;\delta_i) \triangleq \begin{cases} C_{00}^i > 0 & \textit{if } s_i=0,\delta_i=0 \\ C_{10}^i < 0 & \textit{if } s_i=0,\delta_i=1 \\ C_{11}^i > 0 & \textit{if } s_i=1,\delta_i=1 \\ C_{01}^i < 0 & \textit{if } s_i=1,\delta_i=0 \end{cases} \tag{3.3.32}$$

then with some manipulations, one can find the strategy would have the same structure, where Fig.3.3.3 would be "rotated".

By intersecting the upper bound for $V_t^i(\omega_i,i,k)$ in (3.3.31) with $(K-k+1)V_d^i(\omega_i)$ it follows:

$$\underline{\nu_1^i}[k] = \min\left\{\frac{c}{(K-k)r_i},\frac{\rho_i}{\rho_i+r_i}\right\} \tag{3.3.33}$$

$$\overline{\nu_0^i}[k] = \max\left\{\frac{(K-k+1)\rho_i-c}{(K-k+1)\rho_i+r_i},\frac{\rho_i}{\rho_i+r_i}\right\} \tag{3.3.34}$$

$$\overline{\nu_1^i}[k] = \underline{\nu_0^i}[k] = \frac{\rho_i}{\rho_i+r_i} \tag{3.3.35}$$

where we consider $\nu_1^i[k]=\nu_0^i[k]=\frac{\rho_i}{\rho_i+r_i}$ if (3.3.30) has no solutions for $\omega_i\in[0,1]$. The bounds above can be very loose depending on the parameters of our problem and lead to a poor approximation. Potentially tighter upper and lower bounds than the ones directly obtainable from (3.3.33)-(3.3.34)-(3.3.35) can be found using a probabilistic approach to write the value function of our dynamic program. The general idea is to consider a worst and a best case scenario any time the belief over a certain resource has not crossed one of the two thresholds. This is possible since the belief update (for a given observation) and the value function are monotonically increasing functions of ω_i (details are reported in Appendix B). Let us define the following functions for $s = 0, 1$

$$\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, s) \triangleq P\{\omega_i[k+\ell] \geq \omega|\omega_i[k] = \varphi \cap s_i = s\} \tag{3.3.36}$$

$$\overline{\mu_i}[k|s] \triangleq \bar{F}^{1}_{\omega_i[k+1]}(\underline{\nu^i_1}[k]|\overline{\nu^i_0}[k-1], s) - \bar{F}^{1}_{\omega_i[k+1]}(\overline{\nu^i_0}[k]|\underline{\nu^i_1}[k-1], s) \tag{3.3.37}$$

$$\underline{\mu_i}[k|s] \triangleq \bar{F}^{1}_{\omega_i[k+1]}(\overline{\nu^i_1}[k]|\underline{\nu^i_1}[k-1], s) - \bar{F}^{1}_{\omega_i[k+1]}(\underline{\nu^i_0}[k]|\overline{\nu^i_0}[k-1], s) \tag{3.3.38}$$

where (3.3.36) is the complementary CDF of the updated belief, which is a random variable while (3.3.37)-(3.3.38) refer to the probability of not overcoming the belief thresholds.

We will then prove

Lemma 3.3.8. *$\forall i \in \mathcal{N}, k = 1, \ldots, K$ we can derive the following upper and lower bound for the function $V^i_t(\omega_i, i, k)$*

$$\begin{aligned} \overline{V^i_t}(\omega_i, i, k) &= -c \\ &+ \min\left\{\sum_{\ell=k}^{K-1}\left[\omega_i \prod_{m=k+1}^{\ell} \overline{\mu_i}[m|0] + (1-\omega_i)\prod_{m=k+1}^{\ell}\overline{\mu_i}[m|1]\right]\right. \\ &\max\Big\{-u[\ell-k]c + (K-\ell)\left[\overline{\nu^i_0}[\ell] r_i \bar{F}^1_i\Big(\underline{\nu^i_0}[\ell+1]|\overline{\nu^i_0}[\ell], 0\Big)\right. \\ &\left.\left. -(1-\overline{\nu^i_0}[\ell])\rho_i \bar{F}^1_i\Big(\overline{\nu^i_0}[\ell+1]|\underline{\nu^i_1}[\ell], 1\Big)\right], 0\Big\}, (K-k) r_i \omega_i\right\} \end{aligned} \tag{3.3.39}$$

$$\underline{V^i_t}(\omega_i, i, k) = -c + \sum_{\ell=k}^{K-1}\left[\omega_i \prod_{m=k+1}^{\ell}\underline{\mu_i}[m|0] + (1-\omega_i)\prod_{m=k+1}^{\ell}\underline{\mu_i}[m|1]\right]$$

$$\max\Big\{-u[\ell-k]c+(K-\ell)\left[\underline{\nu_1^i}[\ell]r_i\bar{F}_i^1\Big(\overline{\nu_0^i}[\ell+1]|\underline{\nu_1^i}[\ell],0\Big)\right.$$
$$\left.-(1-\underline{\nu_1^i}[\ell])\rho_i\bar{F}_i^1\left(\underline{\nu_0^i}[\ell+1]|\overline{\nu_0^i}[\ell],1\right)\right],0\Big\} \tag{3.3.40}$$

where we used the step function $u[k]=1$ *for* $k>0$ *and* $u[1]=0$ *and the short notation* $\bar{F}_i^1(\omega|\varphi,s)$ *for* $\bar{F}^1_{\omega_i[\ell+1]}(\omega|\varphi,s)$*.*

Proof. See Appendix B. Notice that, since we evaluate these bounds at time k in ω_i, the bounds for the thresholds at time $\ell=k$ can be replaced with ω_i. ■

Gathering these results, we can now present our thresholds approximation in Algorithm 3.1. Note that our approach is valid irrespective of the sensing model. In fact, in order to express in closed form the bounds in (3.3.39)-(3.3.40) one only needs to evaluate the functions $\bar{F}^1_{\omega_i[k+1]}(\omega|\varphi,s)$ in (3.3.36).

We highlight it is important to run the algorithm for decreasing value of k in order to use potentially tighter bounds when we compute (3.3.39)-(3.3.40) for lower values of k. The last step in the algorithm is a direct consequence of Lemma 3.3.4. In Section 3.6 we will numerically illustrate how the tightness of the bounds from Algorithm 3.1 is highly dependent on the value of the parameters of our utility function and on the quality of the test (i.e the sensing SNR ζ_i for our application). Having introduced two methods to approximate the thresholds, we now introduce our low-complexity strategy. The pseudocode for our heuristic is presented in Algorithm 3.2. Interestingly, the distance between the two bounds can be used to find an on-line (not computable *a priori*) non trivial upper-bound for the cumulative utility loss of Algorithm 2.

Lemma 3.3.9. *The expected sensing time for resource* i*, given we continue to sense it from time* k *until we make a final decision at time* τ_i*, can be upper-bounded as* *follows:*

$$\mathbb{E}[\tau_i - k|k] \leq \min\left\{\omega_i[k]\frac{\varsigma\left(\overline{\nu_0^i}[k], \omega_i[k]\right) + \hat{D}(f_0^i||f_1^i)}{D(f_0^i||f_1^i)}\right.$$
$$\left.+(1-\omega_i[k])\frac{\varsigma\left(\omega_i[k], \underline{\nu_1^i}[k]\right) + \hat{D}(f_1^i||f_0^i)}{D(f_1^i||f_0^i)}, K-k\right\} \tag{3.3.41}$$

where $\varsigma(x,y) \triangleq \log\left(\frac{x(1-y)}{(1-x)y}\right)$ *and*

$$\hat{D}(f_0^i||f_1^i) \triangleq \mathbb{E}\left\{\log\left(\frac{f_0^i(y)}{f_1^i(y)}\right)|s_i = 0, f_1^i(y) \leq f_0^i(y)\right\} \tag{3.3.42}$$

$$\hat{D}(f_1^i||f_0^i) \triangleq \mathbb{E}\left\{\log\left(\frac{f_1^i(y)}{f_0^i(y)}\right)|s_i = 1, f_1^i(y) \geq f_0^i(y)\right\} \tag{3.3.43}$$

Proof. See Appendix B ■

The reason for immediately deciding on the resources whose belief has crossed one of the two single-channel thresholds follows from Lemma 3.3.6, which implies that at any time instant k' after k the decision maker will always decide not to test resource i adding $kV_d^i(\omega_i)$ to the overall utility. Therefore postponing that decision after k will give a lower or equal expected utility. In other words, if $V_d^i(\omega_i) > 0$ and we expect to select to exploit resource i at some point, we will have a lower utility postponing that decision; if, instead, $V_d^i(\omega_i) = 0$ and we discard channel i we accrue zero utility irrespective on when we take that action. Using larger thresholds than the optimal ones appears a legitimate choice for applications where the accuracy of the test has a larger impact on the performances than the time devoted to sensing. We then approximate the optimal selection rule $\boldsymbol{\phi}^*$ with the index $\frac{\omega_i[k]r_i}{\mathbb{E}[\tau_i - k|k]}$ which is computed by using the upper-bound introduced in Lemma 3.3.9. The motivation for the index and its asymptotic optimality are discussed in Appendix B.

3.3.3 Asymptotic Regret

Even if this work focuses on finite horizon, it is useful to briefly discuss the asymptotic regret. Let us define the regret $\Psi(L)$ as the *cumulative* loss over the entire horizon K

$$\Psi(K) = K \sum_{i \in \mathcal{N}} \omega_i[1] r_i - \mathbb{E}\left[U(\mathbf{s}, \mathcal{N}, K, c)\right] \tag{3.3.45}$$

where $K \sum_{i \in \mathcal{N}} \omega_i[1] r_i$ represents the maximum achievable expected utility by a genie that knows exactly the state of each resource. To find an asymptotically optimal strategy in term of regret per slot, i.e. a strategy that achieves $\lim_{K \to \infty} \frac{\Psi(K)}{K} = 0$, one can simply consider a strategy that senses for a fixed amount of time growing with $\log K$, that would asymptotically achieve zero probability to declare the wrong status for each resource, and then directly derive the limit. This proposed static strategy would give a *cumulative* regret with order $O\left(\log K\right)$. In our simulation, we will then consider, for the different proposed strategies, the growth rate of the regret with $\log K$, specifically $\lim_{K \to \infty} \frac{\Psi(K)}{\log K}$ to highlight the importance of the resource sorting in reducing this quantity. The static approach, however, is clearly not feasible in the regime of small K and this motivates our study of the optimum policy structure and our heuristic for such scenarios.

3.4 A Group Testing Inspired Strategy

In this Section we want to investigate the opportunity of texting multiple bands at the same time, i.e. with one test. The first consideration is that a combined test would introduce correlation between resources and as we highlighted in Remark 3.3.1, this would significantly complicate the formulation of the problem. We then take a different route with respect to the previous section and add the limitation that resources can only be sensed once, even directly inspected or in a combined test with other sub-bands. In light of this assumption we then need to explicitly take into account the false alarm and missed detection probabilities in our utility function, whereas, without loss of generality we can consider the sensing cost $c = 0$. We consider the same vector state $\mathbf{s}$ and $\boldsymbol{\omega}$ as in the previous section. However, to be consistent with general notation rules, e.g., calligraphic letters to indicate sets, we will slightly modify the notation: instead of the sets of stopping and selection rules and$\boldsymbol{\tau}, \boldsymbol{\phi}$ we will use κ for the total number of tests and $\mathcal{A}_{||}$ for the set of sub-bands mixed in the test at time k. Differently from the previous section, we assume the player acquires information about the entries via random observations coming from a known pdf parameterized by an **unknown** vector. In each measurement at time $k = 1, 2, \ldots, \kappa < K$ the player has the possibility to dynamically and adaptively select a subset of entries through a sensing vector $\mathbf{b}_k = [b_{k1}, b_{k2}, \ldots, b_{kN}]$, i.e. we assume the observation $y[k]$ is drawn from $f_{\theta[k]}(y)$ where $\theta[k] = \theta(\mathbf{b}_k, \mathbf{s})$. The decision maker needs to design:

1. the $\kappa \times N$ measurement matrix $\mathbf{B}$ whose rows are the measurement vectors $\mathbf{b}_k$ for $k = 1, 2, \ldots, \kappa$ for each test and κ indicates the sensing (exploration) time to acquire information on the states s_i via the observations $y[k]$,

2. a set of N decision rules $\boldsymbol{\delta} = \{\delta_i \in \{0, 1\} : i = 1, 2, \ldots, N\}$ over the unknown states s_i of the resources at the end of the exploration phase

The total utility for the player will be proportional to the time left for exploitation $(K - \kappa)$. Let us consider a reward $r_i > 0$ for correctly detecting an empty/busy resource and a penalty $\rho_i < 0$ for failing to detect a busy/empty resource, the utility can be written as

$$U(\mathbf{s}, \mathcal{N}, K, \mathbf{B}, \boldsymbol{\delta}) \triangleq \begin{cases} (K-\kappa)\sum_{i=1}^{N} \omega_i r_i \left(1-\alpha_i\right) + (1-\omega_i)\rho_i\beta_i & \text{case } 0 \\ (K-\kappa)\sum_{i=1}^{N} (1-\omega_i)(1-\beta_i) r_i + \omega_i \alpha_i \rho_i & \text{case } 1 \end{cases} \tag{3.4.1}$$

where α_i, β_i denote the type I and type II errors probability respectively, i.e. $\alpha_i = P(\delta_i = 1 | s_i = 0)$ and $\beta_i = P(\delta_i = 0 | s_i = 1)$. To differentiate between case 0/1 allows to consider applications where the utility comes from an action on the entries detected as empty/busy: i.e., in a spectrum sensing application, the utility would come from the decision on transmitting over frequency bands found empty, whereas for a radar application, it makes more sense to consider the utility comes from taking action on the frequency (spatial directions) found busy. Finding the optimal policy corresponds to solve the following optimization problem

$$\underset{\mathbf{B}, \boldsymbol{\delta}}{\text{maximize}} \quad \mathbb{E}\left[U(\mathbf{s}, \mathcal{N}, K, \mathbf{B}, \boldsymbol{\delta})\right] \tag{3.4.2}$$

The theory presented in this thesis could be easily modified to fit different observation models and assumptions, but in light of our application we will limit to consider the following form for θ:

$$\theta[k] = \theta(\mathbf{b}_k, \mathbf{s}) = \mathbf{b}_k \left(\boldsymbol{\varphi}^T + \mathbf{w}^T\right) \tag{3.4.3}$$

where $\varphi = [\varphi_1, \varphi_2, \ldots, \varphi_N]$ is a non-negative vector, such that the state variable $s_i = 1$ when $\varphi_i > 0$ and 0 when $\varphi_i = 0$, $\mathbf{w} = [w_1, w_2, \ldots, w_N]$, in the context of spectrum sensing, represents the background noise on each frequency band. The theory developed in this chapter concerns the detection of the non-negative entries of $\boldsymbol{\varphi}$ and consequently the maximization of the utility accruable from the resources declared to be in the empty/busy state (3.4.1). The observation model also assumes that $f_{\theta[k]}(y) \equiv Exp(\theta[k])$, i.e.

$$y[k] \sim f_{\theta[k]}(y) = \frac{1}{\theta[k]} e^{-\frac{y}{\theta[k]}} \tag{3.4.4}$$

where, for convenience, we use the alternative parameterization for the exponential distribution.

3.4.1 Direct Inspection (DI) Case

In the Direct Inspection (DI) case, we limit $\mathbf{b}_k$ to have only one non-zero entry i, i.e. $b_{ki} \neq 0$, $b_{kj} = 0\ \forall j \neq i$. This means that there is an underlying hypothesis testing:

$$\mathcal{H}_0 : y[k] \sim Exp\ (\theta_0[k])$$

$$\mathcal{H}_1 : y[k] \sim Exp\ (\theta[k])$$

with $\theta_0[k] = b_{ki}w_i$ and $\theta[k] = b_{ki}(\varphi_i + w_i) > \theta_0[k]$. In this context, it is know that the the signal energy is a sufficient statistic for the test and the energy detection is optimal. Assuming no prior knowledge over the φ_i's in case of existing communication, we only need to set the test threshold which we set in order to maximize the utility defined in (3.4.1). With few algebraic steps we get:

$$y[k] \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \frac{\ln\left(\gamma_i \frac{\theta^\star[k]}{\theta_0}\right)}{\frac{1}{\theta_0} - \frac{1}{\theta^\star[k]}} \tag{3.4.5}$$

where $\theta^\star[k] \triangleq \max\{y[k], b_{ki}(\varphi_{\min} + w_i)\}$ and

$$\gamma_i \triangleq \begin{cases} \frac{r_i\omega_i}{|\rho_i|(1-\omega_i)} & \text{case 0} \\ \frac{|\rho_i|\omega_i}{r_i(1-\omega_i)} & \text{case 1} \end{cases} \tag{3.4.6}$$

Notice that, assuming a minimum average received signal power $\varphi_{\min} > 0$ in case of existing transmission, makes the test meaningful also for values of $\gamma_i < 1$.

Assumption 3.4.1. *To simplify the decision problem, we will assume every resource has to be sensed before being declared empty/busy. This can be enforced as a standard/protocol rule or numerically guaranteed by setting* $\forall i \in \mathcal{N}, \omega_i < \frac{\rho_i}{\rho_i + r_i}$ *(case 0) /* $\omega_i > \frac{r_i}{|\rho_i| + r_i}$*(case 1).*

It is clear that the optimality [1] of the test completely characterizes the set of decision rules $\boldsymbol{\delta}$ for the sensed resources, while Assumption 3.4.1 gives us the decision rule for the non-sensed resources. This implies that for the DI case, the optimization in (3.4.2) can be expressed solely in terms of $\mathbf{B}$. It is also known that for this type of text, where there is uncertainty in a parameter of the alternative hypothesis, one does not know the exact miss probability β, thus we will use an upper-bound, which will reflect in a lower bound for the achievable utility. Since this test is the one used in the DI strategy, we add the superscript ^{DI} to the test error probabilities α_i and β_i and write

$$\alpha_i^{DI} = \min\left\{ \left(\frac{|\rho_i|(1-\omega_i)}{r_i\omega_i\left(1+\frac{\varphi_{\min}}{w_i}\right)} \right)^{\frac{1+\frac{\varphi_{\min}}{w_i}}{\frac{\varphi_{\min}}{w_i}}}, 1 \right\} \tag{3.4.7}$$

$$\beta_i^{DI} = 1 - \left(\alpha_i^{DI}\right)^{\frac{1}{1+\frac{\varphi_i}{w_i}}} \tag{3.4.8}$$

where as expected the false alarm probability is independent from the alternative hypothesis, whereas the detection improves with the true average transmitted power φ_i. What we can guarantee, since $\varphi_i \geq \varphi_{\min}$ is that

$$\beta_i^{DI} \leq 1 - \left(\frac{|\rho_i|(1-\omega_i)}{r_i\omega_i\left(1+\frac{\varphi_{\min}}{w_i}\right)} \right)^{\frac{w_i}{\varphi_{\min}}} = \beta_{i,\max}^{DI} \tag{3.4.9}$$

Remark 3.4.2. *The test performance for the DI case does not depend on b_{ki}, therefore, for the DI case no further optimization is needed over the sensing matrix $\mathbf{B}$, other than selecting the non-zero entries.*

[1] The threshold in (3.4.6) is the optimal threshold that minimizes the Bayesian risk (maximize our utility) for the case of binary hypothesis testing, where φ_i is known. It is of common practice to replace the MLE estimate for the unknown parameter φ_i (GLRT) and then reduce to the binary case, using the same threshold. A local most powerful test exists for $\theta \to \theta_0$ but the GLRT is preferred since we want to consider high SNR range.

The impossibility of knowing the true β_i of each test motivates us to optimize the minimum guaranteed achievable utility, using the bound in (3.4.9). This consideration together with Assumption 3.4.1 leads to rewrite the optimization problem in (3.4.2) for the DI case as

$$\underset{\mathcal{A}\subseteq\mathcal{N}}{\text{maximize}} \quad U^{DI}(\mathcal{A}) \tag{3.4.10}$$

where

$$U^{DI}\left(\mathcal{A}\right) \triangleq (K - |\mathcal{A}|) \sum_{i\in\mathcal{A}} u_i^{DI} \tag{3.4.11}$$

$$u_i^{DI} \triangleq \omega_i r_i (1 - \alpha_i^{DI}) + (1 - \omega_i)\rho_i \beta_{i,\max}^{DI} \tag{3.4.12}$$

We then introduce the following Lemma

Lemma 3.4.3. *$U^{DI}(\mathcal{A})$ is a normalized, non-monotone, non-negative sub-modular function of $\mathcal{A}$.*

Proof. See Appendix B ■

Lemma 3.4.3 implies that there are diminishing returns in augmenting sets by adding a certain action to bigger and bigger sets. The maximization of a non-monotonic sub-modular function is generally NP-hard, but the case of interest is not as difficult. In fact, it is clear that by sorting the resources i so that:

$$u_1^{DI} \geq u_2^{DI} \geq \ldots \geq u_N^{DI} \tag{3.4.13}$$

than the set of size i, $\mathcal{A}_i = \{1, \ldots, i\}$ will be such that for any set $\mathcal{X}$ of size $|\mathcal{X}| = i$

$$\sum_{k=1}^{i} u_k^{DI} \geq \sum_{k\in\mathcal{X}} u_k^{DI}$$

Therefore, what remains is to find the best set size i such that

$$U^{DI}(\mathcal{A}) \leq U^{DI}(\mathcal{A}_i) \leq \max_i \left((K - i) \sum_{k=1}^{i} u_k^{DI} \right) \tag{3.4.14}$$

The maximum in (3.4.14) is attained for

$$i^* = \inf_i \{i : \partial_{i+1} U^{DI}(\mathcal{A}_i) < 0\} \tag{3.4.15}$$

where $\partial_{i+1} U^{DI}(\mathcal{A}_i) = (K - i)u^{DI}_{i+1} - \sum_{k=1}^{i+1} u^{DI}_k$. In fact, given the function is submodular as soon as this condition is attained it is maintained for $i + 2, i + 3$ etc. given that the marginal returns continue to decrease. This maximization is *greedy* and stops when the marginal reward becomes negative.

3.4.2 Mixing Sub-bands

We now allow the test to mix different sub-bands, i.e. the vector $\mathbf{b}_k$ to have more than one non-zero entry. The main idea of this section is to develop a relatively simple dynamic strategy, to choose a sensing matrix that can be characterized, in closed form, and that outperforms the DI alternative. To mitigate the noise folding effects it is necessary to use low density measurement matrices. A common approach for recovery with low density measurement matrices is to use belief propagation via message passing[2], whose most well known application is Low Density Parity Check (LDPC) optimum error correction decoding. However, for LDPC methods (but also for CS methods) the design usually guarantees asymptotical bounds on the ℓ_2-norm and little is known for optimal design in the finite regime. The difficulty of such design arises from the inherent multi-hypothesis testing problem of sensing several resources at the same time. This is why, to develop our dynamic design, we look at a Group Testing (GT) approach, which allows us to have a binary hypothesis testing for each measurement. In this way, we keep the complexity of the analysis relatively low, and we can derive the expected performance for any sensing matrix, under mild assumptions. The main assumption to enable our analysis is that the sensing matrix

[2] In our model, an uninformative prior can be assigned to the φ_i's to run the belief propagation message-passing algorithm on the obtained measurements

must not have length-4 cycles, i.e. two different measurements do not mix more than one sub-band in common. Such condition is also favorable, and typically required, for belief propagation algorithms, e.g. message passing, which suffer from loopy networks with short cycles. Additionally, in the context of *group testing*, little is known in presence of measurement errors that depend on the group size, which is the scenario we are here considering, as the remainder of the chapter will detail. Asymptotic results on the targeting rate for measurement-dependent noise, using an information-theoretic approach, are given in [Kaspi *et al.*(2015)], where the noise is modeled as independent additive Bernoulli with bias dependent on the test size, giving therefore the same false-alarm and missed-detection probabilities, relative to the single test. An additional noise, called *dilution effect* was considered in [Atia and Saligrama(2012)], where each resource could independently flip from 1 to 0 before the grouped test, and information-theoretic bounds were provided. In our model the false alarm and miss-detection probabilities are dependent on the optimization of the test threshold, therefore the noise is not independently added (nor an independent dilution can be considered). Furthermore, the strategy derived depends on the finite horizon for K, i.e. our results are not asymptotic. The same considerations apply to similar information-theoretic approaches in [Scarlett and Cevher(2016), Chan *et al.*(2011), Sharma and Murthy(2015)]. Let us then start by considering a matrix $\mathbf{B}$ without length-4 cycles. From the sensing matrix $\mathbf{B}$, let us define the sets $\mathcal{A}_k = \{i \in \mathcal{N} : b_{ki} \neq 0\}$ and $\mathcal{B}_i = \{1 \leq k \leq \kappa : b_{ki} \neq 0\}$. Note that at times we use $\mathbf{B}$ as an argument in functions that, strictly speaking, are just functions of the sets $\mathcal{A}_k$ just defined. As outlined in the introduction, aliasing of the spectrum comes with an associated noise folding phenomenon. Its impact is particular severe in a non coherent scheme as ours. In fact, the samples are collected sequentially and not in parallel, which means that we do not have multiple observations of the same value but only sequential observations tied to the same underlying random process.For each test we define a binary *group testing-like* hypothesis testing where:

$$\begin{cases} \mathcal{H}_0: & \forall i \in \mathcal{A}_k \quad s_i = 0 \\ & \Rightarrow \theta_0[k] = \mathbf{b}_k \mathbf{w}^T \\ \mathcal{H}_1: & \exists i \in \mathcal{A}_k \quad s.t. \quad s_i = 1 \\ & \Rightarrow \theta[k] \geq \left(\min_i b_{ki}\right) \varphi_{\min} + \mathbf{b}_k \mathbf{w}^T = \theta_{\min}[k] \end{cases} \tag{3.4.16}$$

We envision that such test would be useful for a downlink transmission in which the Access Point may want to allow multiple communications at the same time and can alert the SUs over a narrowband signaling channel to access the spectrum. The test can be written as:

$$\frac{\max_{\theta[k] \geq \theta_{\min}[k]} f_{\theta[k]}(y[k])}{f_{\theta_0[k]}(y[k])} \underset{\mathcal{H}_0}{\overset{\mathcal{H}_1}{\gtrless}} \gamma_k \tag{3.4.17}$$

for which we can derive:

$$\alpha(\mathbf{b}_k, \gamma_k) = \left(\frac{1}{\gamma_k \left(1 + \frac{\theta_{\min}}{\theta_0}\right)} \right)^{\frac{1 + \frac{\theta_{\min}}{\theta_0}}{\frac{\theta_{\min}}{\theta_0}}} \tag{3.4.18}$$

$$\beta(\mathbf{b}_k, \gamma_k) = 1 - (\alpha(\mathbf{b}_k, \gamma_k))^{\frac{1}{1 + \frac{\theta}{\theta_0}}} \tag{3.4.19}$$

The decision declares that resource i is busy ($\mathcal{H}_1$ is true) if the majority of the tests where resource i is involved is positive, else it accepts the null hypothesis $\mathcal{H}_0$ for resource i. Thus:

$$\pi_0(i, \mathbf{b}, \gamma) = \left(1 - \prod_{j \in \mathcal{A}_k \setminus i} \omega_j\right) (1 - \beta_i(\mathbf{b}, \gamma; 0)) + \alpha(\mathbf{b}, \gamma) \prod_{j \in \mathcal{A}_k \setminus i} \omega_j \tag{3.4.20}$$

$$\pi_1(i, \mathbf{b}, \gamma) = 1 - \beta_i(\mathbf{b}, \gamma; 1) \tag{3.4.21}$$

where the functions $\pi_j(i, \mathbf{b}, \gamma)$, $j = 0, 1$ are only defined when $b_i \neq 0$. These functions represent the probabilities of declaring $\mathcal{H}_1$ in a grouped test defined by $\mathbf{b}$ with threshold γ and given $s_i = j,\ j = 0, 1$. Notice that the error probabilities α, β refer to each binary hypothesis testing defined in (3.4.16). The notation for $\beta_i(\mathbf{b}, \gamma; s_i)$ indicates the probability of having a missed-detection conditioned on the state s_i of one of the resources. It then follows that

$$\alpha_i^{GT}(\mathbf{B}, \boldsymbol{\gamma}) \triangleq 1 - F_{PBD}\left(\left\lceil \frac{|\mathcal{B}_i|}{2} \right\rceil - 1; |\mathcal{B}_i|, \{\pi_0(i, \mathbf{b}_k, \gamma_k) : k \in \mathcal{B}_i\}\right) \tag{3.4.22}$$

$$\beta_i^{GT}(\mathbf{B}, \boldsymbol{\gamma}) \triangleq F_{PBD}\left(\left\lceil \frac{|\mathcal{B}_i|}{2} \right\rceil - 1; |\mathcal{B}_i|, \{\pi_1(i, \mathbf{b}_k, \gamma_k) : k \in \mathcal{B}_i\}\right) \tag{3.4.23}$$

where $F_{PBD}(k; n, \mathbf{p})$ indicates the CDF of a Poisson Binomial Distribution parameterized by $\mathbf{p} \in [0, 1]^n$. One can then replace (3.4.22)-(3.4.23) in (3.4.1) to then solve the optimization in (3.4.2), where the equivalence between the decision rules $\boldsymbol{\delta}$ and the selection of the thresholds $\boldsymbol{\gamma}$ is essentially the same as for the DI case. Notice that the condition of no length-4 cycles for $\mathbf{B}$ allows to write (3.4.22)-(3.4.23), i.e. to consider each of the $\mathcal{B}_i$ tests independent, conditioned on the state of the resource i. The optimization remains extremely complex due to the complexity of the decision space for $\mathbf{B}$ and the sum of an exponentially growing number of terms for the probabilities defined in (3.4.22)-(3.4.23). Nevertheless, it gives a method to evaluate the objective of our optimization for any sensing matrix $\mathbf{B}$, where the optimization over $\boldsymbol{\gamma}$ can be numerically solved. Notice in fact that, (3.4.22)-(3.4.23) are monotonic functions of the probabilities π_0, π_1 defined in (3.4.20)-(3.4.21), which are monotonic in the γ_k's, and therefore a unique solution for $\boldsymbol{\gamma}$ exists.

Next, we introduce additional constraints to (3.4.2), in particular on the structure of $\mathbf{B}$, in order to evaluate whether a GT strategy could be superior to the DI approach.

Note that an ML or a MAP estimator, for a rank-deficient sensing matrix, do not provide optimality guarantees in terms of minimum error probability or minimum Bayesian risk. Nevertheless, for the same sensing matrix, we expect the MAP estimator to outperform the binary *group-testing* hypothesis in (3.4.16) by simply adding more degrees of freedom to the decision $\boldsymbol{\delta}$ in the κ-th dimensional space of the observations. Therefore the evaluation of the objective in (3.4.2) via (3.4.22)-(3.4.23) provides a benchmark for the utility obtainable with a more refined detection method.

3.4.2.1 The Pairwise Tests Case

We start by considering matrices $\mathbf{B}$ that have the following property: each resource is sensed only one time, either directly inspected or mixed with another resource, and no test mixes more than 2 resources, i.e. $|\mathcal{A}_k| \leq L = 2, |\mathcal{B}_i| \leq 1 \ \ \forall k = 1, \ldots, \kappa, \ i = 1, \ldots, N$. Let us discuss the test that mixes entries i and j. According to the strategy derived at the beginning of the section one can use (3.4.20)-(3.4.21)-(3.4.22)-(3.4.23) to write out the per-time instant utility obtainable after the decision. First, from (3.4.22), without prior knowledge over φ_i, φ_j other than the threshold φ_{min}, the best choice to minimize α is to set $b_i = b_j$ (we refer to this false alarm probability as α_{ij}). Therefore, as similarly derived for the previous DI case, one can then consider binary coefficients for $\mathbf{b}_k$, i.e. $b_k i \neq 0 \rightarrow b_{ki} = 1$. This will hold true also for the extension of $L > 2$ and will give implementation advantages discussed in 3.5.2. After that, a missed detection in (3.4.16) can occur for 3 different states of the resources i, j but we upper-bound these by always considering $\theta = \theta_{min}$ (we refer to this missed detection probability as $\beta_{ij,\max}$). We then obtain

$$\begin{aligned} u_{ij}^{GT} &\triangleq \omega_i\omega_j(r_i + r_j)(1 - \alpha_{ij}^{GT}) + [(\omega_i(1-\omega_j)(r_i + \rho_j) + \\ &\omega_j(1-\omega_i)(r_j + \rho_i) + (1-\omega_i)(1-\omega_j)(\rho_i + \rho_j))]\, \beta_{ij,\max}^{GT} \end{aligned} \tag{3.4.24}$$

where the threshold for this test γ_{ij} has been set to maximize (3.4.24), i.e.

$$\gamma_{ij} = \frac{\omega_i\omega_j(r_i + r_j)}{(1-\omega_i)(|\rho_i| - \omega_j r_j) + (1-\omega_j)(|\rho_j| - \omega_i r_i)}. \tag{3.4.25}$$

Let us then consider a graph where each resource is a vertex and the edge weight u_{ij} between two vertices ij is the utility (per time instant) u_{ij}^{GT} just defined (the weight of the loops u_{ii}^{GT} are given by u_i^{DI} in (3.4.12)). We can then translate our problem into a particular instance of a *max-cut problem*: picking a subset of the edges and form a subgraph, where each edge represents a test, to maximize the objective in (3.4.2). Formally, we can write

$$\begin{aligned} &\underset{\mathcal{E}}{\text{maximize}} \quad U^{GT}(\mathcal{E}) \\ &\text{subject to} \quad \deg_{\mathcal{E}}(i) \leq 1 \;\; \forall i \in \mathcal{N} \end{aligned} \tag{3.4.26}$$

where

$$U^{GT}(\mathcal{E}) \triangleq (K - |\mathcal{E}|)\left(\sum_{ij \in \mathcal{E}} u_{ij}^{GT}\right) \tag{3.4.27}$$

and $\deg_{\mathcal{E}}(i)$ is the nodal degree of node i induced by the undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$. It is possible to map the constraint on the nodal degree in the objective of (3.4.26) by adding a penalty for the violation of such constraint. This guarantees the optimal solution will be equivalent to (3.4.26), i.e. no set of edges that does not respect the constraint can improve the objective, and any feasible set of edges would have the same objective in the two problems. We rewrite our optimization as

$$\underset{\mathcal{E}}{\text{maximize}} \quad U^{GT}(\mathcal{E}) - M \sum_{i \in \mathcal{N}} \Upsilon(\deg_{\mathcal{E}}(i)) \tag{3.4.28}$$

where

$$\Upsilon(n) \triangleq \begin{cases} 0 & \text{for } n \leq 1 \\ n-1 & \text{for } n \geq 2 \end{cases} \tag{3.4.29}$$

and M is a positive constant.

Lemma 3.4.4. *For $M > 0$ the objective in* (3.4.28) *is a non-monotone sub-modular function of $\mathcal{E}$ and it is possible to find $M^* > 0$ such that for any $M > M^*$ the two optimizations* (3.4.26)-(3.4.28) *are equivalent.*

Proof. See Appendix B ■

We know discuss the extension of our approach for $L > 2$.

3.4.2.2 Extension to $L > 2$

The approach in the previous section can be extended to tests that mix more than 2 channels. However, instead of just edges or self loops to indicate the tests, we could have cycles of length up to L. The nodal degree in (3.4.28) will then be interpreted as the number of cycles a node is in, and the set of edges will be replaced with the set of cycles. The validity of Lemma 3.4.4 naturally extends to this case as well. We then replace the set $\mathcal{E}$ of edges with the set $\mathscr{C}$ of possible cycles, and use $\mathcal{C}$ to indicate the generic cycle (which could be a self-loop, an edge or a cycle with length 3 or greater). In light of the constraint $|\mathcal{B}_i| \leq 1$ we will have that no node can be in two cycles. In Fig.3.4.1 we show two possible set of cycles of length up to 4. On the right we have a set of tests that respect our constraint: there is a test that only considers one resource and three tests that combine 2, 3, and 4 resources respectively, but no resource is considered in two different tests. On the left, instead, a resource is considered in two tests: one where is combined with other 3 resources and one where is inspected directly, and such configuration is therefore not acceptable.

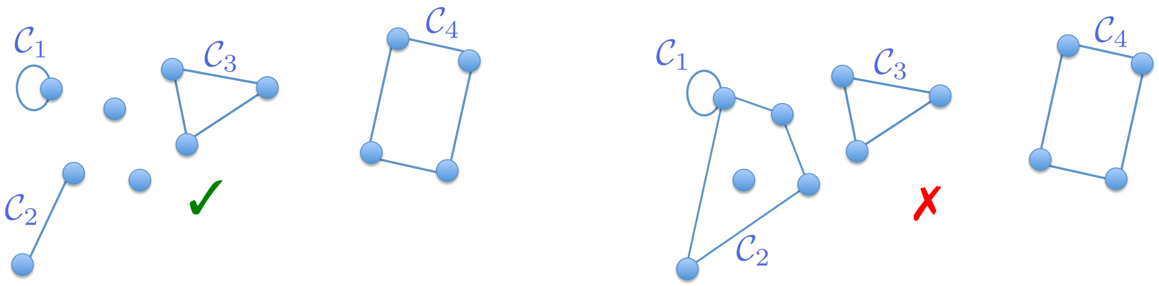


Figure 3.4.1: Example of two sets of tests. The right configuration has 4 tests and no resource is considered in two different tests therefore it respect our constraints, whereas the left configuration has a resource included in two tests and is not a feasible solution.

In the greedy procedure in Algorithm 3.3 there is a constant number of operations per query, which indicates the overall complexity of the algorithm is dominated by the sorting of the cycles utilities. The set $\bar{\mathscr{C}}$ indicates the set of cycles that are not adjacent (share a node) with any of the cycle in $\mathscr{C}$. Since in the worst case, sorting n values require $O(n^2)$ operations, the complexity will be given by the total number of possible tests $O\left(\left(\sum_{\ell=1}^{L}\binom{N}{\ell}\right)^2\right) = O\left(N^{2L}\right)$, i.e. polynomial in N and exponential in L.

3.4.2.3 The Factor Approximation of the Greedy Algorithm

Having proven the sub-modularity of (3.4.28) in Lemma 3.4.4, it is natural to resort to a greedy procedure, however it is important to highlight that the objective in (3.4.28) does not respect the non-negativity property. To the best of our knowledge, there is no known procedure in the literature on approaching the maximization of a general sub-modular non monotone function, if the minimum value is not known: no constant approximation factor guarantee can therefore be given in general. Nevertheless, due to the particular structure of our problem it is possible to find a factor approximation for the output of the greedy algorithm.

Lemma 3.4.5. *Algorithm 3.3 guarantees a α-constant factor approximation of the optimal solution for* (3.4.28)*, where* $\alpha = \frac{1}{\min\{L_{\mathit{eff}}, \frac{K}{2}\}} \frac{K-1}{K-\min\{L_{\mathit{eff}}, \frac{K}{2}\}}$.

Proof. See Appendix B. ■

Note that

$$\partial_{\mathcal{C}'} U^{GT}(\mathscr{C}) = -\sum_{\mathcal{C}\in\mathscr{C}} u_{\mathcal{C}} + (K - |\mathscr{C}| - 1)u_{\mathcal{C}'} \tag{3.4.30}$$

so, as long as the number of tests $|\mathscr{C}|$ added in the greedy maximization is less than the time horizon K, we have

$$\arg\max_{\mathcal{C}\in\bar{\mathscr{C}}} \partial_{\mathcal{C}} U^{GT}(\mathscr{C}) = \arg\max_{\mathcal{C}\in\bar{\mathscr{C}}} u_{\mathcal{C}}. \tag{3.4.31}$$

This relation indicates that, in the greedy procedure, edges are added in decreasing order of utility, respecting the constraint on the nodal degree in light of Lemma 3.4.4.

Notice, also, that from (3.4.30) it is easy to find that the optimal $|\mathscr{C}|$ will never be greater than $\lceil \frac{K-1}{2} \rceil$.

3.4.2.4 The Approximate ML Estimate for Mixed Tests

In the previous sections we have provided methods that find a low density measurement matrix. As will be apparent in our numerical results, the noise folding phenomenon justifies the use of sparse sensing matrices, which are ideal when one wants to use belief propagation to the decision problem. However, for the sake of comparison here we propose a possible alternative approach, which can be applied to any measurement matrix $\mathbf{B}$. Let us in fact consider κ measurements have been collected, that involve the mixing of a set $\mathcal{A} \subseteq \mathcal{N}$ of resources. One could ignore the prior ω_i and derive the ML estimate for φ. First, let us write the log-likelihood

$$\begin{aligned} \log\left(f\left(\boldsymbol{y}|\boldsymbol{\varphi}_{\mathcal{A}}\right)\right) &= -\sum_{k=1}^{\kappa} \log\theta[k] + \frac{y[k]}{\theta[k]} \\ &\overset{\theta[k]\to y[k]}{\approx} -\sum_{k=1}^{K} 1 + \log y[k] + \frac{1}{2}\left(\frac{y[k]-\theta[k]}{y[k]}\right)^2 \end{aligned} \tag{3.4.32}$$

where the linearization corresponds to the Taylor expansion of the likelihood for the observations around their mean (recall (3.4.3)-(3.4.4)). One could them aim at solving the following weighted ℓ_1 minimization in a LASSO fashion

$$\hat{\boldsymbol{\varphi}}_{\mathcal{A}} = \arg\min_{\boldsymbol{\varphi}_{\mathcal{A}}} \|\boldsymbol{\Gamma}_{\mathcal{A}}\boldsymbol{\varphi}_{\mathcal{A}}^T\|_1 + \frac{1}{2}\|\left(\mathbf{y} - \mathbf{B}(\boldsymbol{\varphi}_{\mathcal{A}}^T + \mathbf{w}_{\mathcal{A}}^T)\right)\|_{\mathbf{C}^{-1}}^2 \tag{3.4.33}$$

with $\boldsymbol{\Gamma} = \{\gamma_i \text{ from (3.4.6)} : i \in \mathcal{N}\}$ and $\mathbf{C} = \mathrm{diag}(\mathbf{y})$ being the covariance of the observations. The first penalty term in the objective enhances sparsity, favoring the entries with lower threshold γ_i to have $\varphi_i > 0$; the second term in the objective comes from the ML estimate in (3.4.32). Note that, compared to the non-sequential sampling models (i.e. those using a filterbank), the application of the LASSO in this context is an approximation. The random demodulator in [Tropp *et al.*(2010)], similar to our scheme in terms of architecture, is an Xampling ADC converter for signals that are sum of harmonics with constant amplitude, i.e. each subspace, in the UoS representation, has finite dimension. This is not the case we are interested in, as mentioned in the introduction. For our *multiband* signal model, instead, rather than having observations that are noisy linear combination of a sparse input, the samples p.d.f. depends on those linear combinations.

3.5 The PHY Layer of the CUMMA Receiver

We now map the sequential decision models in Section 3.3-3.4 to find a single/multiband spectrum sensing technique for cognitive radio systems. In the context of spectrum sensing for cognitive radio, since data transmission includes large amounts of control overhead in addition to the data payload, it is natural to assume that the activity of the Primary Users (PUs) in a certain spectrum will persist for several sampling periods (see Fig.3.5.1). However, assuming this interval lasts $T = KT_s$, the sensing mechanisms should be providing the fastest decision it can. In fact, the

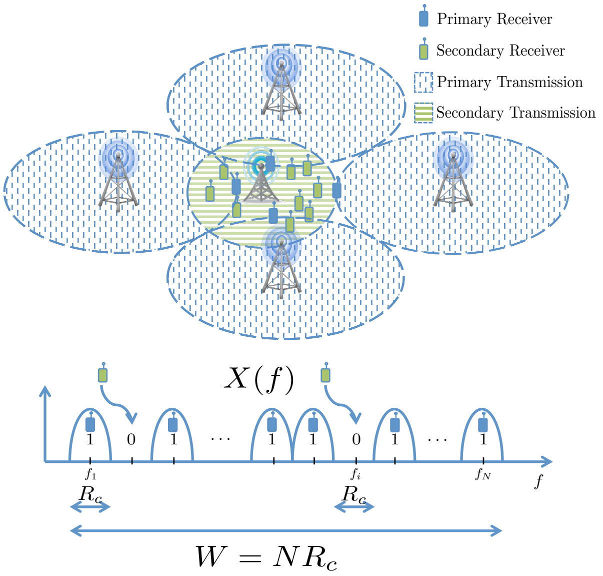


Figure 3.5.1: Cognitive radio scenario

objective of the cognitive receiver is to sequentially sense the spectrum for the first portion of the interval and transmit the most it can during the remaining time the sub-channels it found empty.

3.5.1 Analog Front-End Sampling

The aliasing sequence, folds the spectrum present in specific sub-bands onto the center frequency of the wake-up radio receiver. The samples are spaced by intervals of duration $T_s = 1/R_s$ and, as the diagram in in Fig. 3.5.2 shows, rather than having a filter bank architecture as in [Bagheri and Scaglione(2015)], our approach is to do sequential non-coherent tests, according to the strategy, i.e. the sensing matrix $\mathbf{B}$ designed via our greedy algorithm, to maximize the utility accruable from utilizing the sub-bands (channels) detected as free. We assume that the complex envelope of the analog signal we are exploring is a multicomponent signal, whose components are

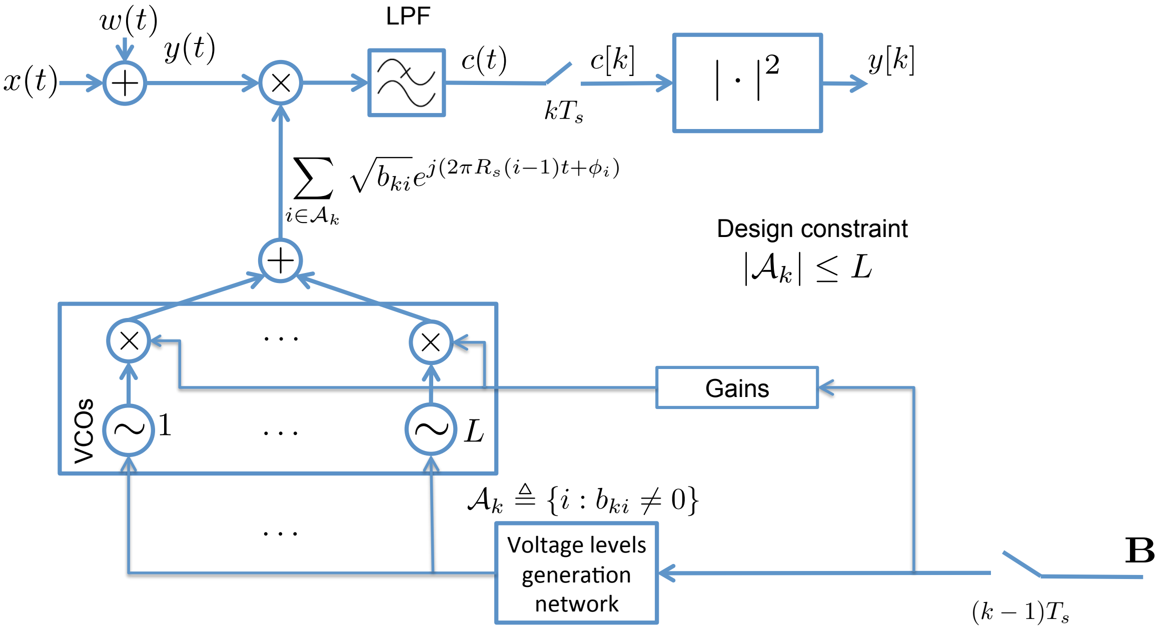

$w(t)$
$y(t)$
$x(t)$
LPF
$c(t)$
$c[k]$
kT_s
$|\cdot|^2$
$y[k]$
$\sum_{i\in\mathcal{A}_k}\sqrt{b_{ki}}e^{j(2\pi R_s(i-1)t+\phi_i)}$
Design constraint
$|\mathcal{A}_k| \leq L$
$\cdots$
Gains
VCOs
1
L
$\mathcal{A}_k \triangleq \{i : b_{ki} \neq 0\}$
Voltage levels generation network
$\mathbf{B}$
$(k-1)T_s$

Figure 3.5.2: The CUMMA Receiver diagram for cognitive sequential sensing: the circuit of VCO's select the different bands to mix at each time instants with weights controlled by the sensing matrix $\mathbf{B}$. For the DI scenario, $L = 1$.

a frequency band width equal to $W = NR_s$ and, hence, during the interval $0 \leq t < T$ the received signal is:

$$y(t) = x(t) + w(t) \tag{3.5.1}$$

$$x(t) = \sum_{i=1}^{N} s_i[m] x_i(t) e^{-j2\pi R_s(i-1)t} \tag{3.5.2}$$

where we already introduced the state variables $s_i[m]$ previously, $w(t) \sim \mathcal{N}(0, N_0\delta(\tau))$ is Additive White Gaussian Noise, and $x_i(t)$ are the transmitter signals modeled as band-limited random processes with bandwidth R_s, i.e. it is equal with probability one to:

$$x_i(t) = \sum_{k=1}^{K} x_i[k] sinc(\pi(R_s t - k + 1)). \tag{3.5.3}$$

The sequential receiver we propose first modulates the received signal at each antenna over the period $(k-1)T_s \leq t < kT_s$ with:

$$\beta_k(t) = \sum_{i=1}^{N} \sqrt{b_{ki}} e^{j(2\pi R_s(i-1)t + \phi_i)}, \tag{3.5.4}$$

where b_{ki} are the coefficients of the sensing matrix $\mathbf{B}$ previously introduced and the phase $\phi_i = -2\pi R_s(i-1)\tau_i$ accounts for the delay in generating the tone at the i-th frequency, i.e. our receiver can be implemented by combining different oscillators that do not require to be synchronized. Then, after convolving the modulated signal with an ideal low-pass filter with impulse response $sinc(\pi R_s t)$, it samples the output $c(t)$ at times $kT_s,\ k = 1, \ldots, \kappa$. This operation is equivalent[3] to an orthogonal projection, as shown below:

$$c[k] = [y(t)\beta_k(t)] \star R_s sinc(\pi R_s t)|_{t=kT_s} = \sum_{i=1}^{N} \sqrt{b_{ki}} e^{j\phi_i} Y_{ki} \tag{3.5.5}$$

[3] If the periodic signals where not truncated in time the relationship would be exact, in practice there will be some approximation error due to the windowing of the signal over the prescribed interval $[(k-1)T_s, kT_s]$. The effect of this can be mitigated by using raised cosine filtering and a non rectangular window to reduce the effect of side lobes.

where Y_{ki} represent the orthogonal projections over the period $(k-1)T_s \leq t < kT_s$ of $y(t)$ over the following signals:

$$Y_{ki} =< y(t), R_s e^{j2\pi R_s(i-1)t} sinc\left(\pi\left(R_s t - k + 1\right)\right) > \tag{3.5.6}$$

Considering that the signals

$$\left\{e^{j2\pi R_s(i-1)t} sinc\left(\pi\left(R_s t - k + 1\right)\right)\right\}_{i,k\in\mathbb{Z}}$$

form a orthogonal basis, and that (3.5.1) is equivalent to:

$$x(t) =\sum_{k=1}^{K}\sum_{i=1}^{N} s_i x_i[k] e^{j2\pi R_s(i-1)t} sinc(\pi(R_s t - k + 1)) \tag{3.5.7}$$

$$Y_{ki} = s_i x_i[k] + w[k] \tag{3.5.8}$$

where $w[k] \sim \mathcal{CN}(0, N_0)$. If we model $x_i[k]$ also as i.i.d. $x_i[k] \sim \mathcal{CN}(0, \varphi_i)$ we get that for a given state $\mathbf{s}$:

$$Y_{ki} \sim \mathcal{CN}(0, \varphi_i + N_0). \tag{3.5.9}$$

where $\boldsymbol{\varphi}$ (introduced in the presentation of our model) is a vector collecting the average, unknown *a priori*, power received from the existing communications. The receiver samples for $k = 1, \ldots, \kappa$ are:

$$c[k] = \sum_{i=1}^{N} \sqrt{b_{ki}} e^{j\phi_i} \left(s_i x_i[k] + w[k]\right) \tag{3.5.10}$$

and therefore (assuming the delays ϕ_i's are independent and uniformly distributed) they are also conditionally zero mean Gaussian random variables:

$$c[k] \sim \mathcal{CN}\left(0, \theta[k] \triangleq \mathbf{b}_k \left(\boldsymbol{\varphi}^T + \mathbf{w}^T\right)\right). \tag{3.5.11}$$

It follows that the information for the detection of Primary communications is in the variance and by considering as observations

$$y[k] \triangleq |c[k]|^2 \tag{3.5.12}$$

then one has $y[k] \sim Exp(\theta[k])$ in accordance with the hypothesis for the model developed in Section 3.4.

Remark 3.5.1. *Note that while modulated signals are discrete and non-Gaussian, here it is reasonable to assume that its distribution is well approximated by a Gaussian p.d.f since the receiver is not synchronized with the active source and the signal received, while remaining in its original band, is most likely subject to linear distortion due to a multi-path channel. It would be more appropriate potentially to include a certain correlation among the samples $x_i[\ell]$ and it is just for simplicity that we do not consider it, given the generalization is straightforward and does not impact the derivation of the opportunistic strategy.*

3.5.2 Hardware Limitations

Naturally, the mapping of the signal in general will be imperfect and, like in any ADC, calibration is necessary [Chen *et al.*(2010),Israeli *et al.*(2014)]. For most ADCs the assumption is that this calibration is done during an initial training phase, in which an known input signal can be used to estimate the equivalent matrix $\mathbf{B}$. As far as the proposed architecture is concerned, the circuit diagram of Fig.3.5.2 assumes a *settling time* for the VCOs much smaller than T_s, i.e. the sampling period for the single channel sub-band. If this assumption does not hold, one should use a LPF with a smaller bandwidth and collect the samples $c[k]$ at an even slower rate than R_s, to wait for the VCOs to settle. This modification would not alter the statistical characterization of the samples, derived in the previous subsection. The drawback of taking samples less often is that (assuming the same occupancy coherence time) one would have harnessed less information than what is available in the received signal, and would have less than K slots to decide. Given that our strategy is derived as a function of K, this would not invalidate our findings. Another possibility would be to replace the L tunable VCOs with N oscillators at constant frequencies, corresponding to the N possible bands of the signal. Using N oscillators would increase the power consumption and cost of the circuit but would significantly reduce the switching time between two measurements. Hence, this would be the natural choice if one wants to exploit a dense sensing matrix. Instead, the use of a bank of VCOs is preferable if the matrices are sparse because a small number of VCOs can synthesize the mixing signal. The switching would be in fact performed by a multiplexer, that would take the sum of the up to L tones selected by the vector $\mathbf{b}_k$. In general, since we focus on the detection of the signal, with reasonably good components we expect that calibration will either far less demanding or unnecessary, if one accepts loss in sensitivity. In fact, the binary coefficients for the vector $\mathbf{b}$ can be set to ones and zeros, as discussed in 3.4.2.1. Controlling the gains is unnecessary for the system to work and it is preferable to not add tunable gains as they can be another possible source of uncertainty and complexity in the system. Finally, imperfect tuning of the VCOs will reduce the SNR, either by spreading or misplacing the center frequency of the components of interest, but not fundamentally impair its detection.

3.5.3 Utility Parameters

In this section we derive the mapping between the utility function parameters and the PHY Layer of our CUMMA Receiver, in order to numerically evaluate the Algorithms introduced in Sections 3.3-3.4

The decision maker is a secondary transmitter and each resource i is associated with a “channel” to communicate with a secondary receiver. The term channel is used in a broad sense and it could represent a frequency band with certain bandwidth, a set of tones in an OFDM system or a collection of spreading codes in a Code Division Multiple Access (CDMA) system.

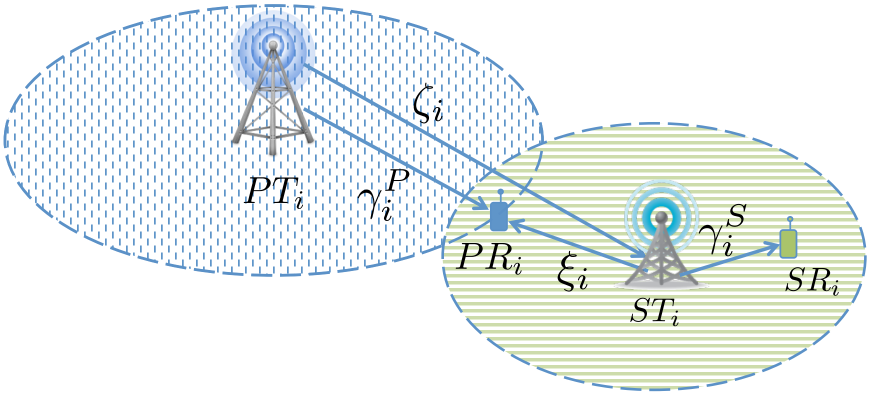


Figure 3.5.3: The CUMMA Receiver Cognitive Scenario

We denote with $\gamma_i^{P,S}$ the two instantaneous Signal-to-Noise Ratios at the receiver for the primary and secondary communication respectively. ξ_i represents the instantaneous SNR from the secondary transmitter that could interfere with the primary communication and ζ_i indicates the instantaneous SNR at the secondary transmitter for the primary transmitter, which is the information the secondary transmitter uses to detect the primary communication (see Fig.3.5.3). We consider all the communication channels of interest as Rayleigh fading channels, therefore all the SNR's are exponentially distributed, i.e.

$$p_\eta(x) = \frac{1}{\bar{\eta}} \exp\left(-\frac{x}{\bar{\eta}}\right) \quad \text{for } \eta \in \left\{\gamma_i^P, \gamma_i^S, \xi_i, \zeta_i\right\} \tag{3.5.13}$$

where $\bar{\eta}$ indicates the average value of η. The phase of the received signal is uniform.

If the secondary transmitter decides to transmit over the i-th channel ($\delta_i = 0$) and the Primary Transmitter is not transmitting ($s_i = 0$) then it accrues a utility per time instant which is a function of γ_i^S. For our model, we decide to choose the outage rate ($R_{i,out}^S$) as the reward per time instant left (r_i), since it is reasonable to assume a slowly varying channel and that the instantaneous CSI (Channel Side Information) is not known at the receiver, therefore the secondary transmitter uses a constant data rate to transmit. We refer to a design parameter $P_{i,out}^S$ that indicates the probability that the system can be in outage, i.e. the probability that the secondary transmitter cannot successfully decode the transmitted symbols. Since we assume Rayleigh fading, we can express our reward per time instant left

$$r_i = \left(1 - P^S_{i,out}\right) W_0 \log_2 \left[1 - \bar{\gamma}^S_i \ln\left(1 - P^S_{i,out}\right)\right] \tag{3.5.14}$$

where the derivation can be found in [Choudhury and Gibson(2007)]. If instead, the secondary transmitter decides to use the channel ($\delta_i = 0$), interfering with the Primary Transmitter communication ($s_i = 1$), then it receives a penalty equal to the loss in outage rate caused to the Primary User, considering its interference adds to the noise at the Primary Receiver. Let us in fact assume that the Primary Transmitter also transmits at a certain outage rate, given by its value of $P^P_{i,out}$. In presence of interference from the secondary user its effective transmission rate becomes

$$R^P_{i,\xi_i} = (1 - P^P_{i,\xi_i}) W_0 \log_2 \left[1 - \bar{\gamma}^P_i \ln\left(1 - P^P_{i,out}\right)\right] \tag{3.5.15}$$

where

$$P^P_{i,\bar{\xi}_i} = \frac{P^P_{i,out} - \bar{\xi}_i \ln\left(1 - P^P_{i,out}\right)}{1 - \bar{\xi}_i \ln\left(1 - P^P_{i,out}\right)} \tag{3.5.16}$$

(for details see Appendix B). The penalty ρ_i is therefore defined as the loss in rate caused by the interference

$$\rho_i = R^P_{i,out} - R^P_{i,\xi_i} = \left(P^P_{i,\bar{\xi}_i} - P^P_{i,out}\right) W_0 \log_2 \left[1 - \bar{\gamma}^P_i \ln\left(1 - P^P_{i,out}\right)\right] \tag{3.5.17}$$

In our model, we assume that cross-channel interference is negligible, i.e. the only possible interference for the primary communication over channel i is given by a secondary transmission over the same channel.

In the first part of this chapter we considered the decision maker has perfect knowledge of the averages $\bar{\gamma}^P_i, \bar{\gamma}^S_i, \bar{\xi}_i, \bar{\zeta}_i$ and the designed $P^P_{i,out}, P^S_{i,out}$ for all the N resources. For the POMDP, following the discussion on the signal model in Section 3.5.1, we then consider $f^i_0(y), f^i_1(y)$ exponential with parameter θ^i_0, θ^i_1 and the ratio

$\frac{\theta_1^i}{\theta_0^i} = 1 + \bar{\zeta}_i = 1 + \frac{\varphi_i}{w_i}$. For this particular sensing model, the KL distances are known and the two terms $\hat{D}(f_0^i||f_1^i), \hat{D}(f_1^i||f_0^i)$ can be computed directly from their definition in (3.3.42)-(3.3.43):

$$\hat{D}(f_0^i||f_1^i) = \frac{\log(1+\bar{\zeta}_i)}{1-\left(1+\bar{\zeta}\right)^{-\frac{\bar{\zeta}_i+1}{\bar{\zeta}_i}}} - \frac{\bar{\zeta}_i}{1+\bar{\zeta}_i}, \quad \hat{D}(f_1^i||f_0^i) = \bar{\zeta}_i.$$

We can then directly evaluate the upper-bound in (3.3.41) for $\mathbb{E}[\tau_i - k|k]$. We can also give a closed form expression for the complementary CDF $\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, s)$ in (3.3.36), that will be used for different results that we will present in the next section. For our specific sensing model, by simple algebra and use the belief update equation in (3.3.8) we can prove that:

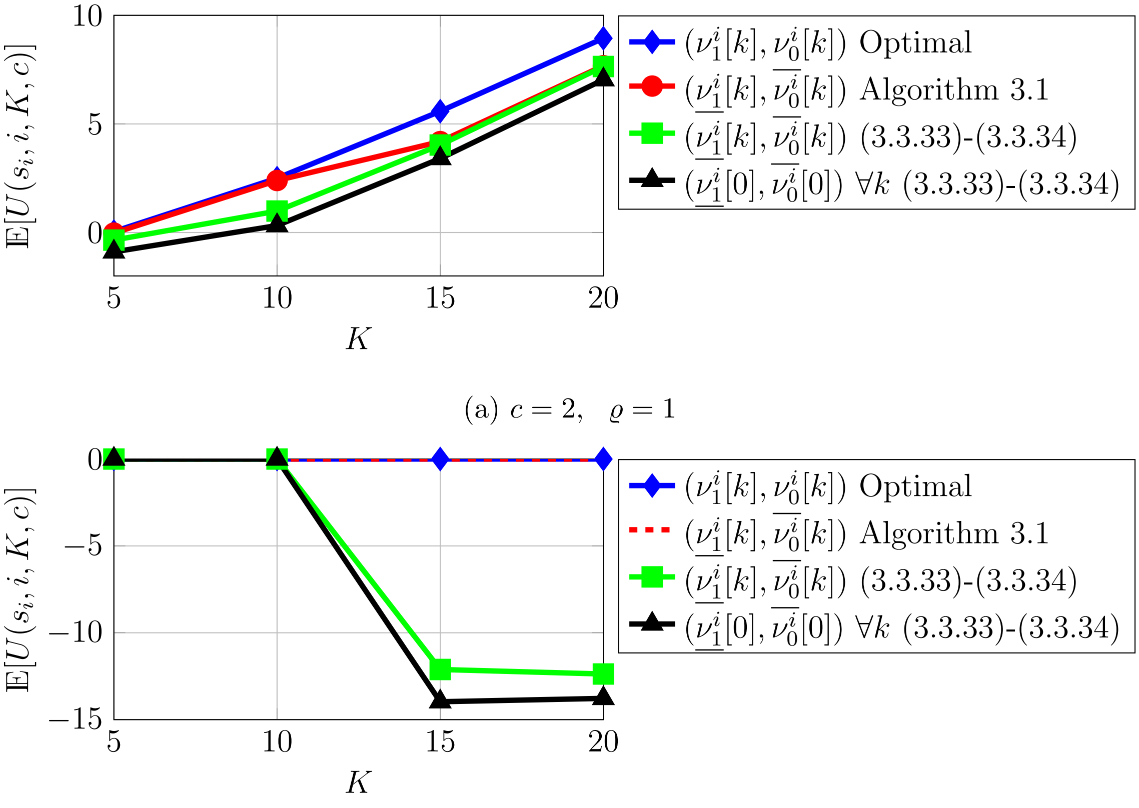
$$\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, 0) = F^{\ell}_{\Gamma}\left(\frac{(1+\bar{\zeta}_i)\varsigma(\varphi,\omega)}{\bar{\zeta}_i} + \ln\left(1+\bar{\zeta}_i\right)^{\ell}\right) \tag{3.5.18}$$

$$\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, 1) = F^{\ell}_{\Gamma}\left(\frac{\varsigma(\varphi,\omega)}{\bar{\zeta}_i} + \ln\left(1+\bar{\zeta}_i\right)^{\ell}\right) \tag{3.5.19}$$

where $F^{\ell}_{\Gamma}(x)$ is the CDF of the sum of ℓ exponential random variables with unitary mean. The key step to derive (3.5.18)-(3.5.19) is noting that, conditioned on the state s of the resource, we have that $\frac{y[k]}{\theta_s^i} \overset{i.i.d.}{\sim} Exp(1)$. Once we have a closed expression for $\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, s)$ for $s = 0, 1$ and expressions for r_i, ρ_i in (3.5.14)-(3.5.17) respectively, we can follow Algorithm 3.1 to find bounds on the thresholds and use Algorithm 3.2. In the second part of the chapter, we then moved to the more realistic assumption that φ_i is not known at the receiver. So in this context one can set $\rho_i \propto \varrho_i$, where ϱ_i is a parameter that accounts for different QoS over a certain band in the utility maximization, and use an estimate for the possible level of interference.

3.6 Simulation Results

We now present our simulation results for the two problems proposed.



(a) $c = 2, \quad \varrho = 1$

(b) $c = 10, \quad \varrho = 2$

Figure 3.6.1: Utility for the single-resource case with different decision thresholds in the regime of small K

3.6.1 POMDP

We highlight that, for the case of a single resource, we can compute the optimal decision thresholds $(\nu_1^i[k], \nu_0^i[k])$ obtained by MonteCarlo evaluation of the integral in (3.3.13). In our simulations we will show the improvement in the utility we can get by using the decision threshold bounds obtained with our approach in Algorithm 1 instead of using the easy bounds in (3.3.33)-(3.3.34), that are obtained by convexity of the function V_t. For the cognitive radio case we have the following parameters: $\omega_i[1] = 0.5, r_i = 2, \rho_i = 2, \bar{\zeta}_i = 3$.

In Fig. 3.6.1a we can see how for $K \leq 10$ our thresholds approximation in Algorithm 3.1 can reach the same utility of the optimal decision thresholds, while for higher values of K it acquires the same utility of the easy bounds obtainable from (3.3.33)-(3.3.34). The reason for this is that, when K increases, our upper-bound for V_t in (3.3.39), which is essentially a sum of upper-bounds, becomes looser. We also looked at the performance obtainable by constant decision thresholds (further referred to as CT strategy), i.e.

$$\underline{\nu_1^i}[k] = \underline{\nu_1^i}[1] = \min\left\{\frac{c}{(K-1)r_i}, \frac{\rho_i}{\rho_i + r_i}\right\} \tag{3.6.1}$$

$$\overline{\nu_0^i}[k] = \overline{\nu_0^i}[1] = \max\left\{\frac{K\rho_i - c}{K\rho_i + r_i}, \frac{\rho_i}{\rho_i + r_i}\right\} \tag{3.6.2}$$

$\forall k = 1, \ldots, K$ which always achieve worse utility than the others, being the looser approximation of the optimal thresholds. Notice that for higher value of c and ϱ, in Fig.3.6.1b, the optimal strategy is to not start sensing and simply accepting utility 0. Our thresholds approximation (Algorithm 3.1) is able to capture this and follows the optimal strategy, while using the bounds in (3.3.33)-(3.3.34) or constant thresholds gives a negative utility. We can notice how the difference in utility between the optimal strategy and our approximation is higher for the marketing strategy application and developing new bounds for this case will be the object of future research. For the second experiment we analyze the case with multiple channels to be sensed for the cognitive radio application. We consider a heterogeneous network with 4 Primary Transmitters located at the corners of a square with side $500m$. Our entire bandwidth goes from 800 to $900MHz$ (see Fig. 3.5.1). A secondary transmitter wants to find opportunities to communicate with $N = 20$ secondary receivers randomly spread around. The 20 channels have a bandwidth of $5MHz$ and we assume they are equally divided among the 4 Primary transmitter: observations are then collected at the Nyquist rate for the single channel, i.e. $2\mu s$. We assumed that the primary transmitter power is $10dBm$ and the height of the transmitter is $10m$, while for the secondary we chose power equal to $5dBm$ and a transmitter height of $3m$. The height of all the receivers is equal to $1m$. We assume the secondary transmitter has an estimate for $\bar{\gamma}_i^S$ (the average SNR at the secondary receiver from the secondary communication) and $\bar{\zeta}_i$ (the average SNR at the secondary transmitter from the primary communication), considering a deterministic path loss propagation model. We used a deterministic two-ray model to predict the average SNR received [Rappaport(2001)], i.e. the φ_i value introduced in Section 3.4 $\left(\text{i.e. } \varphi_i = P_t G \frac{h_t^2 h_r^2}{d_i^4}\right)$. The value of G has been set to $2 \cdot 10^{-4}$ to have average received SNR (ζ_i) at the secondary (in case of primary transmission) in the range of $5 - 10dB$, which is a reasonable range if we assume to only have thermal noise and no additional interference over the channel. The secondary transmitter does not know the position of the potential primary receiver but he estimates $\bar{\gamma}_i^P$ (the average SNR at the primary receiver from the primary communication, without interference) and $\overline{\xi}_i$ (the average SNR from the secondary transmitter to the primary receiver, that could interfere with an existing primary communication), by considering the closest point the primary receiver could be in the coverage area, i.e. the highest interference he could create (which does not necessarily correspond to the highest ρ_i). As explained in Section 3.3, the value of c is limited by the actual cost of testing. We want to study how the performances in terms of utility for different strategies (stopping rule and selection rule) change for different values of ϱ and c. We compared the performances of Algorithm 3.2 with three possible alternatives:

1. A selection rule that follows an initial arbitrary order (indicated in our plot with NS= "No Sorting"), i.e. $\phi_k = \arg\max_{i \in \mathcal{A}_{k+1}} i$.
2. Constant decision thresholds (indicated in our plot with CT= "Constant Thresholds"), as in (3.6.1)-(3.6.2) $\forall i \in \mathcal{N}$
3. both 1) and 2) (indicated with $CTNS$)

Notice that the NS procedure corresponds to conducting a sequence of concatenated truncated SPRT, which, as highlighted in Section 3.3, represents a suboptimal strategy for our problem (even with optimal thresholds). We will study the performances in three different regimes of K. For low values of K (generally $K \leq N$) the coupling of the problem becomes more relevant since there is no time for the decision maker to sense all the resources and the single-resource decision thresholds are a loose approximation of the actual optimal thresholds which are much tighter. In light of this, we add the following step in our Algorithm 3.2. After computing the quantities $\overline{\mathbb{E}}\left[\tau_i - k|k\right]$ for $i \in \mathcal{A}_k$ from (3.3.41) and sort the resources according to our index in (3.3.44), we keep in $\mathcal{A}_{k+1}$ all the resources with higher index as long as the following condition is satisfied:

$$\sum_{i \in \mathcal{A}_{k+1}} \overline{\mathbb{E}}[\tau_i - k|k] < (1+\epsilon)(K-k+1) \tag{3.6.3}$$

and add to $\mathcal{D}$ the remaining ones. The motivation for this additional step is that, since we expect to not have time to sense all of them, we might start acquiring positive utility in expectation from some of them. In Fig.3.6.2 we indicate this additional removal with "AR". We can see that for $K \leq 10$ the CT modification of Algorithm 2

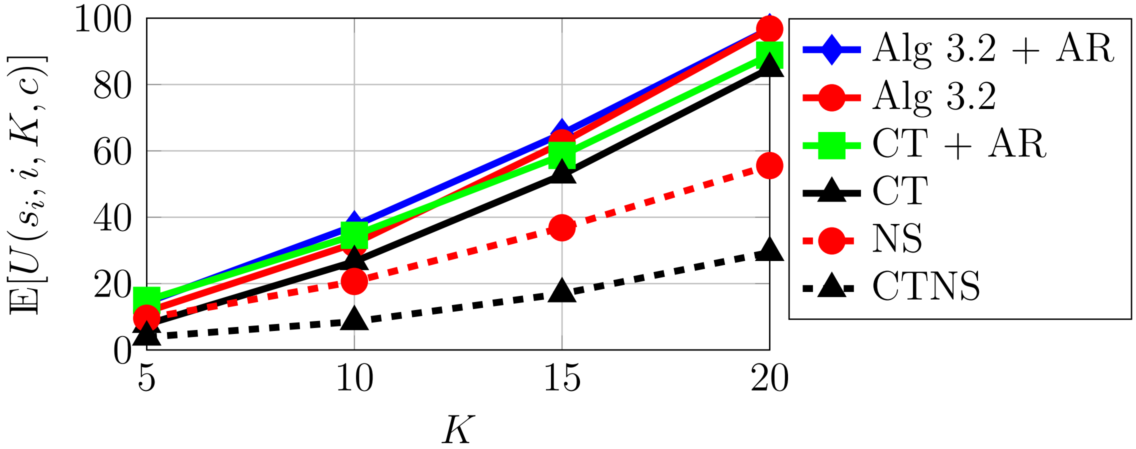
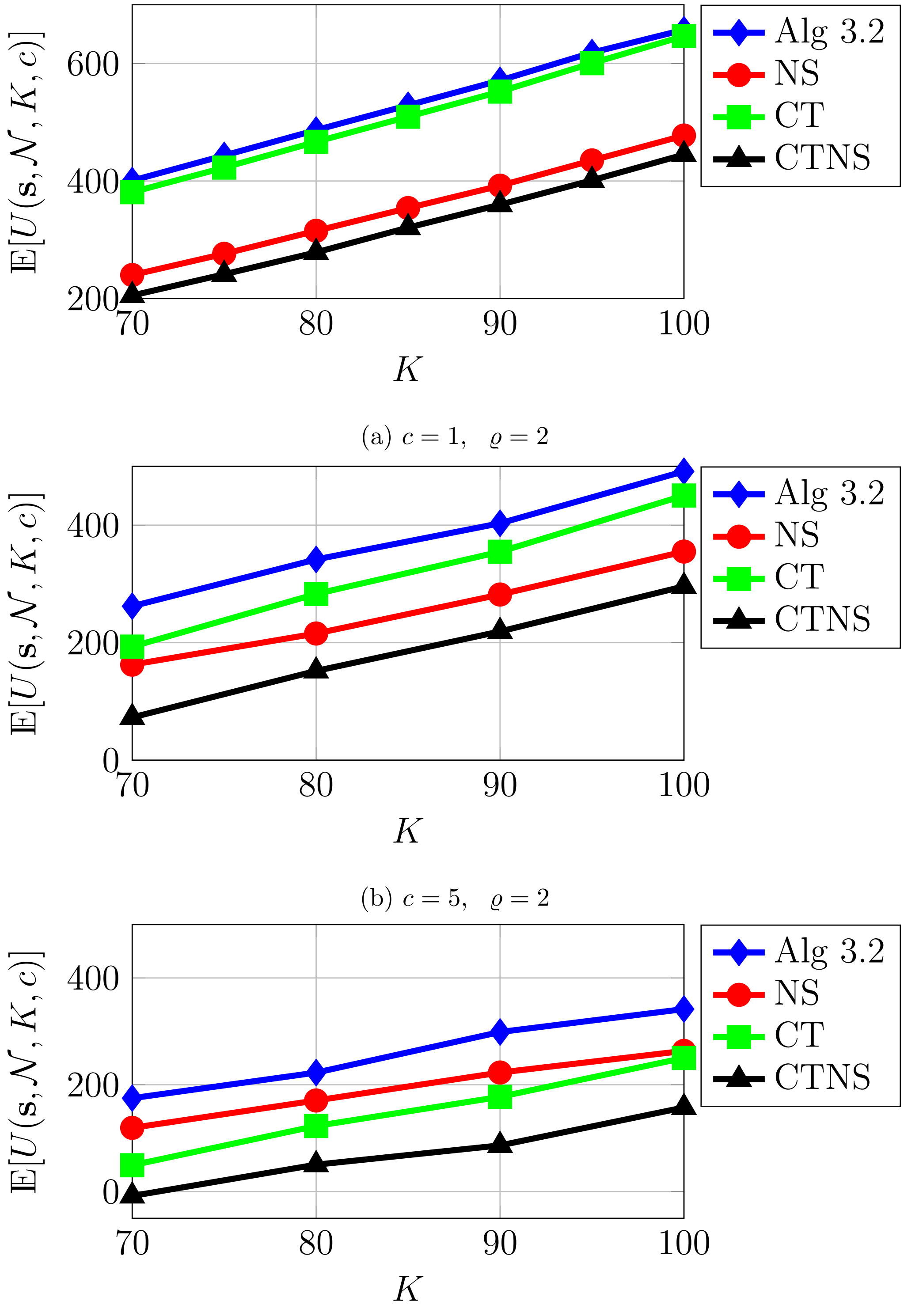


Figure 3.6.2: Utility of our heuristic in the regime of small K ($N = 20$, $c = 1$, $\varrho = 1$ and $\epsilon = 0.5$ in (3.6.3)).

can achieve higher utility, by including the additional removal of resources, than the original Algorithm 3.2. For $K = 20$ the importance of the additional removal becomes almost negligible for Algorithm 3.2, while it continues to have an impact on the CT modification. In this regime of K, we can see the importance of the selection rule by looking at the two strategies that do not sort the resources (dashed lines) and are clearly outperformed by the other strategies. In Fig. 3.6.3 we show the utility achieved by the 4 strategies previously introduced: Alg 3.2, CT, NS, CTNS for moderate values of K (i.e. $70 \leq K \leq 100$). We can see how Algorithm 3.2 outperforms the other strategies and also that when we increase the cost c, the performances of CT, initially close to our complete heuristic, get worse and considering time-varying threshold becomes more important than sorting, i.e. the NS utility is higher than CT. The CTNS approach is always the worse. The same trend was observed for different values of ϱ. Finally, we look at the regime for high K. Following the discussion in Section 3.3.3, in Fig.3.6.4 we plot the growth rate of the regret $\frac{\Psi(K)}{\log K}$ for the 4 different strategies. We can see this quantity is approximately constant, and the key factor to reduce this constant relies in the selection rule, other than the time-varying behavior of the decision thresholds. Similar trends were observed for different values of c and ϱ where both higher c and ϱ increase the regret $\Psi(K)$.

3.6.2 Group Testing

We now look at the performances of the strategy derived in Section 3.4. In particular, we showcase the ability of our approach to dynamically switch between a DI (scanning receiver) and a GT approach, based on the expected occupancy (the vector of priors $\boldsymbol{\omega}$), the time available K, the minimum SNR threshold $SNR_{\min} = \frac{\varphi_{\min}}{w}$ and the number of resources N. For the cognitive radio application, the concept of *exploitation* of the resource is tied to the discussion in 3.5.3, for which the utility is



(a) $c = 1, \quad \varrho = 2$

(b) $c = 5, \quad \varrho = 2$

(c) $c = 10, \quad \varrho = 2$

Figure 3.6.3: Utility of our heuristic in the regime of moderate K

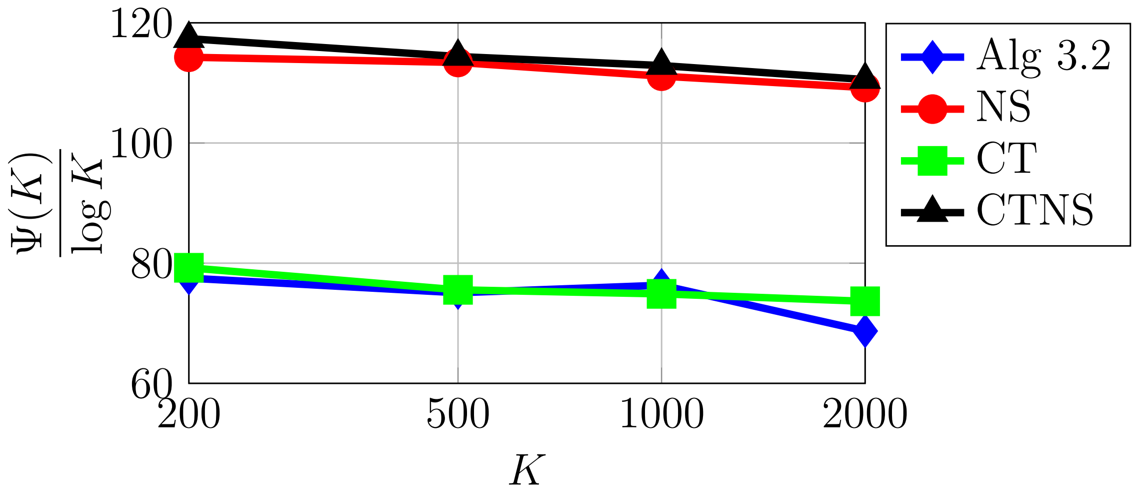
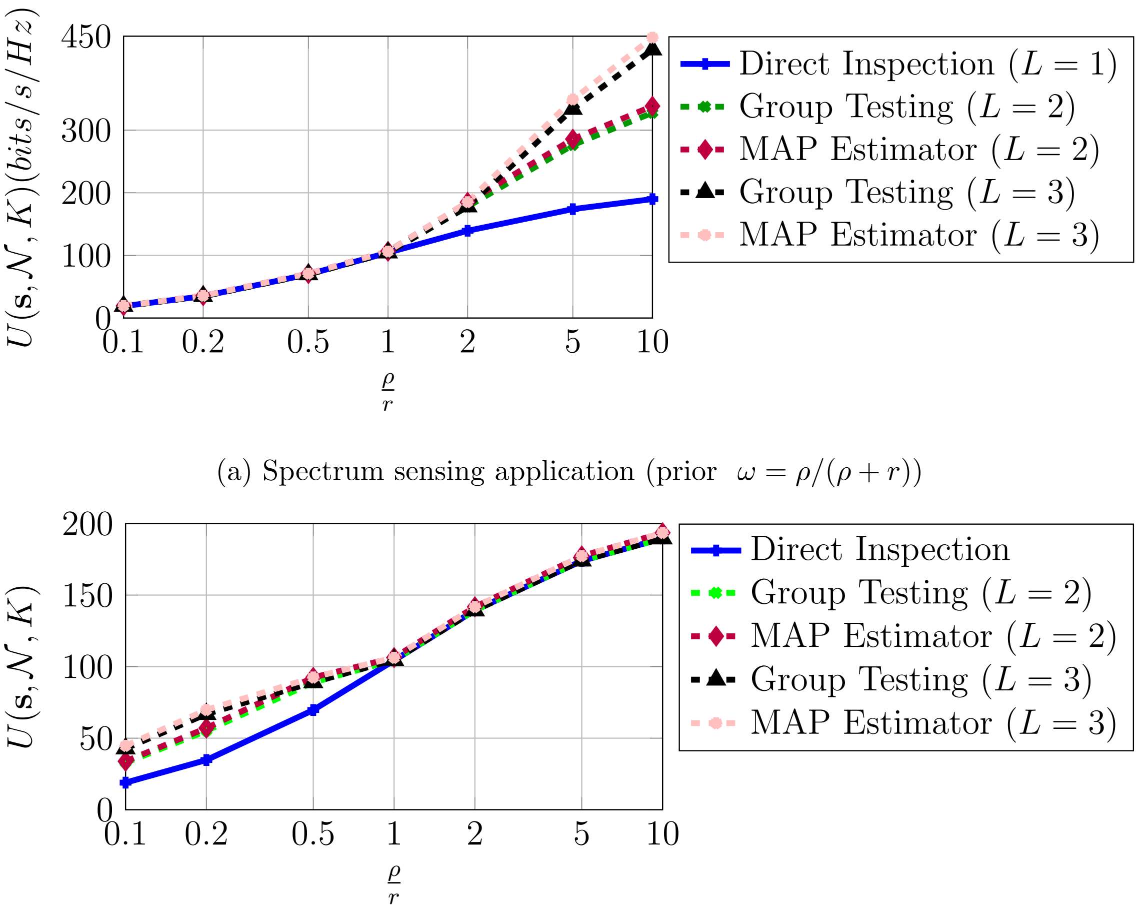


Figure 3.6.4: Asymptotic growth rate of the regret $\Psi(K)$ with $\log K$ for our heuristic $(c = 2, \varrho = 2)$

expressed in $bits/s/Hz$[4] The longer the time available to transmit the larger is number of bits that can be transmitted over that band. For the other case, i.e. when the reward comes from detecting correctly which resources that are busy (for example a RADAR application), it is not immediately clear why the utility would be proportional to the number of remaining time instants. To interpret this, we model the action upon declaration of $s_i = 1$ as a Bernoulli trial which accrues a reward r_i if such action is successful (i.e. the target is actually hit) and this happens with a certain probability p_i for each attempt. The number of attempts T_i necessary to hit the target will then be geometrically distributed. One can find then that the expected reward is equal to $r_i P(T_i \leq (K-\kappa)) = r_i \sum_{k=1}^{K-\kappa} p_i(1-p_i)^{k-1} = r_i(1-(1-p_i)^{K-\kappa}) \approx (K-\kappa)r_i p_i$ for small p_i, which would motivate having an expected utility which increases linearly with time. The ρ_i associated with this case would model an *intervention cost*, which main purpose is to limit the false alarm rate. It is important to highlight, however, the time dependency in the objective would prevent our formulation to return a stan-

[4] From (3.4.1)-(3.4.2), r_i's and ρ_i's can be normalized over the communication bandwidth without altering the optimization.

dard constant false alarm rate CFAR detection method. Nevertheless, our model can apply to electronic warfare (tentatives of create jamming), wake-up radio and other problems where the action (and the associated utility) is on the channels that are declared busy. Notice that, in light of the symmetry in the definition of the threshold γ_i, one can switch the r's and ρ's to go from case 0 (cognitive radio) to case 1 (RADAR) and find the same trends, even for the combined tests. However, we highlight the difference in the two scenarios in the first simulation we present. For this experiment we set $K = 30, N = 60$ and $r_i = r, \rho_i = \rho$ and $\omega_i = \omega, SNR_i = SNR_{\min}(10dB)\ \forall i \in \mathcal{N}$ we have that for ω equal to $\frac{\rho}{\rho+r}$or $\frac{r}{\rho+r}$ for case 0 and case 1 respectively. These are the threshold values given in Assumption 3.4.1, to guarantee no resource can give positive utility if not tested. As we can see, in both scenarios the utility increases with the ratio $\frac{\rho}{r}$, since the prior increases favorably with respect to the utility function. However, for the spectrum sensing application, the GT approach gives higher utility than the DI when $\frac{\rho}{r}$ increases, i.e in the same direction of increasing utility. For the RADAR application, instead, the GT is preferable when $\frac{\rho}{r}$ decreases, i.e. in opposed direction to the increase in utility. The motivation is that when the penalty increases with respect to the reward, the GT approach for spectrum sensing will be conservative by not transmitting in any of the channels in a *pool* if one is found busy. Nevertheless, the increasing prior ω allows, at the same time, to find multiple empty sub-bands with just one test and gain in utility. For the RADAR application, when the penalty increases with respect to the reward, there is a disadvantage in declaring as busy all the elements in the test, even if the prior ω decreases. Clearly this limits the benefit of combined tests, whereas when $\frac{\rho}{r}$ decreases, there is a gain since one element found busy in the pool guarantees higher reward. Apart from this asymmetry, both cases show the same trends in utility over number of available resources and the value of $SNR_{\min}$. For all the figures we refer to $L = 2, 3$ as the maximum number of



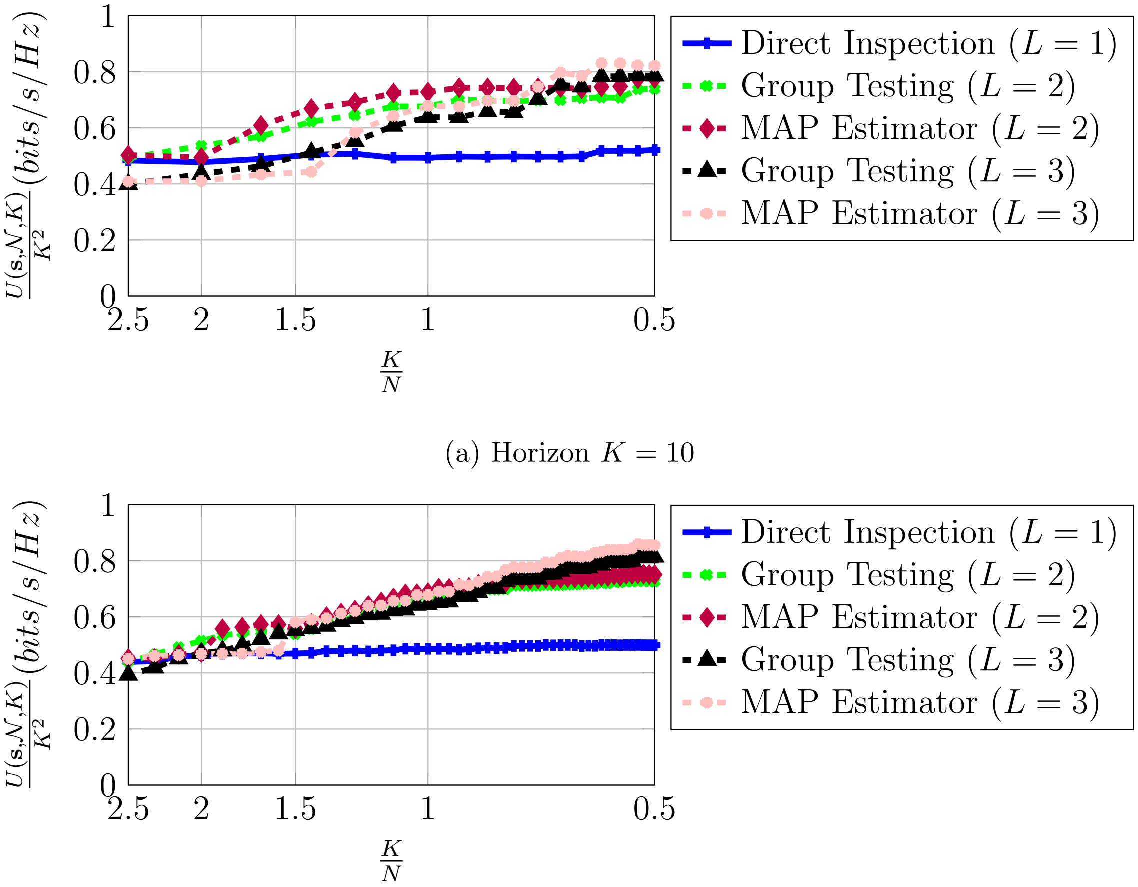
(a) Spectrum sensing application (prior $\omega = \rho/(\rho + r)$)

(b) Radar application (prior $\omega = r/(\rho + r)$)

Figure 3.6.5: Comparing utility for different approaches vs the ratio $\frac{\rho}{r}$ (horizon $K = 30$, number of resources $N = 60$, $SNR_{\min} = 10dB$). The utility for the radar application (case 1) is normalized over the unit measure of r_i and ρ_i

resources per test allowed in our greedy procedure in Algorithm 3.3. In principle, the optimal value for U^{GT} monotonically increases with L since increasing L introduces additional degrees of freedom. However, as proved in our Lemma 3.4.5, the approximation factor of the greedy maximization is potentially worse for higher values of L, as the following numerical results will show. We indicate with "Group Testing" the utility obtained with our GT approach. The "MAP Estimator" is the estimator that knows the true values φ_i, uses the same matrix $\mathbf{B}$ of the GT approach, but then decides on each resource, based on the posterior for ω_i, using belief-propagation. In

Fig.3.6.6 we plot the utility (normalized over K^2) over the ratio $\frac{K}{N}$ for two different horizons, i.e. $K = 10$ and $K = 30$ and $SNR_{\min} = 10dB$. We can see that, only for $\frac{K}{N} \lessapprox 0.75$, the GT approach outperforms whereas when the horizon increases, almost no benefit is given by mixing resources: roughly speaking, there is enough time to test them independently and have higher accuracy. For this experiment, we looked at case 0 and set $\omega_i \sim \left(0.7, \frac{\rho_i}{\rho_i + r_i}\right)$, where $r_i = \log(1 + SNR_{i,S})$ and $\rho_i = 5r_i$ with $SNR_{i,S_{dB}} \sim \mathcal{U}([10, 20])$. The SNR for the test, i.e. $\frac{\varphi}{w}$ is generated uniformly between 10 and 20 dB, but the only information used in our algorithm is the minimum value, i.e. in this case $10dB$. In the regime we show, the DI is approximately constant since it is easy to show $U^{DI,OPT} \leq \frac{K^2}{4} u_{\max}$: for a fixed K there is basically no benefit in increasing N over $\frac{K}{2}$ except for having $\frac{K}{2}$ higher rewards due to the random generation of the parameters of the utility function. We then looked at how the utility behaves versus the SNR of each test. In this case the SNR was drawn uniformly between $SNR_{\min_{dB}}$ and $SNR_{\min_{dB}} + 10$, and once again only the value of $SNR_{\min} = \frac{\varphi_{\min}}{w}$ was used in the algorithm, which is shown in the abscissa of the figures. Matching our intuition, we can see how the GT approach outperforms the DI only when $SNR_{\min}$ is high enough and also that the gain in utility is larger for $K = 10$ than for $K = 30$. In fact, for this experiment the number of resources has been set constant to $N = 20$ and as previously highlighted, increasing K for fixed N diminishes the advantages of combining resources in a test. In this case we also plotted the utility obtainable with the ML estimate via Compressive Sensing, described in Section 3.4.2.4, with a dense matrix that has the same aspect ratio of the one found via GT approach (i.e. that scans the same set of resources for the same number of tests). Only for the ML estimate via CS, we actually took the sample mean over 10 observations for each observation $y[k]$. We can see that, despite having more measurements, such approach gives a much lower utility than DI or the proposed GT. This illustrate the negative effect of noise folding. For $K = 30$, we also compared our approach with the belief propagation in a loopy network, obtained by using a LDPC matrix (see [Baron *et al.*(2010)] for details). With $N = 20$ resources and an expected sparsity approximately equal to 4, we chose a regular LDPC matrix with a row weight of 5 (20/4 as suggested in [Baron *et al.*(2010)]) and a column weight of 3 to have 12 tests. The LDPC has not been implemented for $K = 10$ since the constraints on the regularity would have given either a diagonal matrix (same as DI), or a relatively dense matrix. The absence of any optimization in the choice of which and how many resources to test produces a utility which, for low SNR, is lower than the DI approach proposed. For high enough SNR, LDPC method can outperform the DI approach, but still gives a utility lower than our GT strategy with $L = 2$. This highlights the benefit of having an active sub-Nyquist receiver compared to a static offline selection of the parameters.



(a) Horizon $K = 10$

(b) Horizon $K = 30$

Figure 3.6.6: Comparing utility for different approaches vs the ratio horizon K over the number of resources N for different horizons K and the prior $\omega_i \sim \mathcal{U}\left(0.7, \frac{\rho_i}{r_i+\rho_i}\right)$. The utility on the y axis is normalized by K^2 $(SNR_{\min} = 10dB)$

Algorithm 3.1: Thresholds Approximation Algorithm

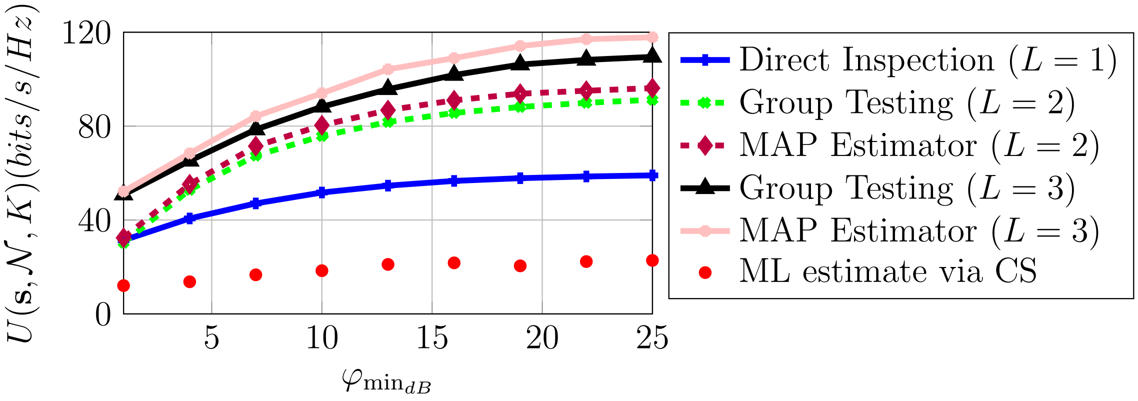
```
$\forall i \in \mathcal{N}$
1) set $\underline{\nu_1^i}[K] = \overline{\nu_1^i}[K] = \underline{\nu_0^i}[K] = \overline{\nu_0^i}[K] = \frac{\rho_i}{\rho_i + r_i}$
2) for $k = K - 1 : -1 : 1$ do
        • Compute $\overline{V_t^i}(\omega_i, i, k)$ from (3.3.39) and $\underline{V_t^i}(\omega_i, i, k)$ from (3.3.40)
        • find the numerical solutions $\omega_1 < \omega_2$ of
              $\overline{V_t^i}(\omega_i, i, k) = (K - k + 1)V_d^i(\omega_i)$
          where $\omega_1 = \omega_2 = \frac{\rho_i}{\rho_i + r_i}$ if no solutions.
          Set
              $\underline{\nu_1^i}[k] = \omega_1$
              $\overline{\nu_0^i}[k] = \omega_2$
        • find the two numerical solutions $\omega_1 < \omega_2$ of
              $\underline{V_t^i}(\omega_i, i, k) = (K - k + 1)V_d^i(\omega_i)$
          where $\omega_1 = \omega_2 = \frac{\rho_i}{\rho_i + r_i}$ if no solutions.
          Set
              $\underline{\nu_1^i}[k] = \min(\omega_1, \underline{\nu_1^i}[k + 1])$
              $\overline{\nu_0^i}[k] = \max(\omega_2, \overline{\nu_0^i}[k + 1])$
end
```

Algorithm 3.2: Heuristic for the joint design of $\boldsymbol{\tau}$ and $\boldsymbol{\phi}$

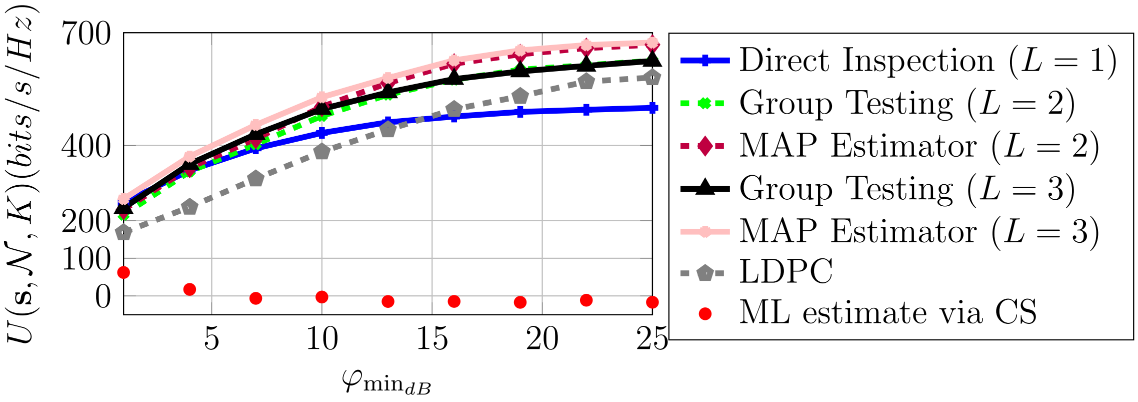
```
$k = 1$, $\mathcal{A}_k = \mathcal{N}$;
while $k \leq K$ and $\mathcal{A}_k \neq \emptyset$ do
    $\mathcal{D} = \emptyset$;
    1) Search for channels $i$ such that $\omega_i[k] < \underline{\nu_1^i}[k] \vee \omega_i[k] > \overline{\nu_0^i}[k]$;
    if there are such channels then
        for every channel $i$ that has been found do
            $\mathcal{D} = \mathcal{D} \cup \{i\}$;
        end
    end
    2) Take a decision over the resources in $\mathcal{D}$, accrue utility
      $(K - k + 1)V_d(\boldsymbol{\omega}, \mathcal{D})$ and remove $\mathcal{D}$ from the state ($\mathcal{A}_k - \mathcal{D} \rightarrow \mathcal{A}_{k+1}$).
    3) Test channel
      $$\phi_k = \arg\max_{i \in \mathcal{A}_{k+1}} \frac{\omega_i[k] r_i}{\mathbb{E}[\tau_i - k | k]} \tag{3.3.44}$$
      and update $\boldsymbol{\omega}[k]$ from (3.3.8).
    4) $k \rightarrow k + 1$;
end
```

Algorithm 3.3: Greedy Maximization of $U^{GT}(\mathscr{C})$

```
Initialize: $\mathscr{C} = \emptyset$.
While $\exists\ \mathcal{C} \in \bar{\mathscr{C}}$ such that $\partial_{\mathcal{C}} U^{GT}(\mathscr{C}) > 0$
    Find $c^* = \arg\max_{\mathcal{C} \in \bar{\mathscr{C}}} \partial_{\mathcal{C}} U^{GT}(\mathscr{C})$
    $\mathscr{C} \leftarrow \mathscr{C} \cup \mathcal{C}^*$
End
```



(a) Horizon $K = 10$



(b) Horizon $K = 30$

Figure 3.6.7: Comparing utility for different approaches vs SNR for different horizons K and the prior $\omega_i \sim \mathcal{U}\left(0.7, \frac{\rho_i}{\rho_i + r_i}\right)$ (number of resources $N = 20$)

CHAPTER 4

DECOMPOSITION OF INFRASTRUCTURE-BASED DYNAMIC RESOURCE ALLOCATION- THE DIDRA SCHEME

4.1 Chapter Outline

In this chapter we discuss a new multi-time scale optimization proposed to enable resource sharing in new generation backhaul operator networks. We start by providing the context for the application of our optimization: a holistic network architecture, named Layback, that has been proposed by our collaborators Akhilesh Thyagaturu and Prof. Martin Reisslein in [Thyagaturu *et al.*(2017)]. In their vision, Software Defined Networking (SDN) represents the key element to enable an integrated signalling infrastructure, which allows to develop the optimization framework here proposed. In Section 4.2, we discuss previous works that investigated the possibility of sharing resources across different technologies, thus enabling a more dynamic resource allocation intra operator. Due to the extensive body of work that could fit under this umbrella, we will focus, for this chapter, on works that have proposed an architectural framework and/or an optimization that followed a Network Utility Maximization (NUM) formulation. We then move to detail our optimization framework in Section 4.3 and give a general formulation for utility decomposition in networks, where agents have the capability of performing different tasks at different time-scales: the framework is then applied to minimize the end-to-end delay with a max projection scheduling policy. For the specific application of uplink traffic management in the backhaul, we combine the modified primal dual decomposition [Palomar and Chiang(2006)], to suit the different operating time-scales of the architecture, with a Lyapunov drift-plus-penalty relaxation [Georgiadis *et al.*(2006), Neely(2006)] to meet an *economic* constraint on the resource allocated to each operator in the long run. In Section 4.4, the simulation results showcase the effectiveness of the method propose: the natural oscillations of the system can harmoniously redistribute the communication resources along the network, thus minimizing the uplink end-to-end delay from the eNBs to the core network.

4.2 Background

Some of the communication bottlenecks in wireless access networks are not due to resource shortages but to the lack of resource sharing among the different wireless operators and wireless technologies. In fact, each wireless operator typically runs its own network. Similarly, each wireless technology, such as LTE or Wi-Fi, operates in its own radio access network and corresponding backhaul network, with a static subdivision of resources. The optimization is proposed as part of a holistic network architecture, named Layback, proposed in [Thyagaturu *et al.*(2017)]. The LayBack architecture is enabled by software defined networking (SDN) and consists of five main layers (see Fig. 4.2.1), namely: the devices layer, the radio node (e.g., eNB, WiFi AP) layer, the gateway (e.g., small cell gateways, CRAN), the SDN switching layer, and the SDN backhaul layer (e.g., legacy enhanced packet core (EPC) controlled by SDN applications). While there have been some efforts in wireless standards [Liu *et al.*(2016b)], [Taleb *et al.*(2015)] and in academic research to define a framework to share network resources across wireless technologies, the solutions available to date provide very limited flexibility, such as sharing only among individual LTE cells [Liu *et al.*(2016a)], [Samdanis *et al.*(2016)]. Thus, there is only very limited statistical multiplexing (sharing) of network resources among wireless operators and technologies [Niu *et al.*(2016)], [Biermann *et al.*(2012)]. The limited sharing of networking resources across wireless operators and technologies is to a large degree due to (i)

the lack of a convenient effective signaling infrastructure across the wireless access network, and (*ii*) the lack of an optimization framework that could accommodate signalling delays incurred between different network components. To address these issues we recently proposed in the LayBack network architecture. The LayBack architecture is enabled by software defined networking (SDN) and consists of five main layers (see Fig. 4.2.1), namely: the devices layer, the radio node (e.g., eNB, WiFi AP) layer, the gateway (e.g., small cell gateways, CRAN), the SDN switching layer, and the SDN backhaul layer (e.g., legacy enhanced packet core (EPC) controlled by SDN applications).

The seminal paper of Kelly et al. in 1997 [Kelly *et al.*(1998)] introduced the concept of NUM to solve the problem of resource allocation in a network. The benefit of the NUM formulation is that it can be readily decomposed, and this inspired extensive work in the last 20 years that generated new algorithms and cross-layer optimization protocols to solve a variety of network flow control problems. These problems often lie at the intersection between distributed optimization and stochastic network theory; comprehensive surveys can be found in [Lin *et al.*(2006), Chiang *et al.*(2007), Chiang(2008), Pham and Hwang(2017)]. This body of work also motivated a reverse engineering process over former network protocols, expressed often times as a deterministic set of rules, to cast them as NUMs and gain insights on their efficient performances, or lack thereof (see [Kelly(2001)] for NUM applications in TCP traffic). For instance, the work of L. Tassiulas, and A. Ephremides [Tassiulas and Ephremides(1992), Tassiulas and Ephremides(1993)] on Queue-length Maximum Weight (QMW) scheduling, paved the way for several other researchers who extended the condition under which throughput optimality can be established, or other performance guarantees can be met [Andrews *et al.*(2004), Kar *et al.*(2008), Ji *et al.*(2013)]. In particular, in terms of delay, QMW scheduling is not guaranteed to carry optimal performance [Cui and Yeh(2014)], leading to the investigation of variations of the algorithm that enhance its delay performances in general multi-hop networks [Cui *et al.*(2016), Birmiwal *et al.*(2012)] or provide better guarantees [Kar *et al.*(2012), Neely(2013)]. A common feature of these problems is that a centralized optimal scheduler can be, for several reason, impractical, hence the decomposition via the NUM formulation can provide the desired implementation scalability. In the decomposition of NUM problems, it is known that different decomposition techniques impose different constraints on the layers timescales [Palomar and Chiang(2006), Johansson *et al.*(2006)]: a common assumption, however, is the so-called *time-scale separation assumption*, which states that the session interval T_s is much larger than the convergence time T_r of the greedy resource allocation policy [Chiang(2008)]. In other words, under this assumption, the distributed optimization is abstracted from the dynamic control and they can be considered separately (hence the term *separation*). Under this principle, decentralized algorithms for link scheduling based on queue lengths have been proposed in [Gupta *et al.*(2009), Bui *et al.*(2009), Jiang and Walrand(2011)], and more recently in [Teng and Song(2017)]. In this work, we abstract the PHY layer and consider the management of uplink traffic in the backhaul, where the SDN operates by keeping the queues logically separated at each eNB, while the shared resources are granted from the orchestrator to the operator, from the operator to the GWs, and from the GWs to the eNBs: under this assumption, and in the simplified case of continuous flows and infinite queue backlogs, QMW is also delay optimal [Banirazi *et al.*(2014)]. In the decomposition of the associated max product utility over the layers of the architecture, we consider realistic network latencies, which make the *time-scales separation assumption* unrealistic. The works that remove the *time-scale separation assumption* are divided in two classes: 1) those that use intermediate iterates as decisions and assume continuous underlying flows [Lin *et al.*(2008),Srikant(2004)] and 2) those that propose a multi-time scale approach across different layers of the protocol stack [Altman *et al.*(2012), Van Nguyen *et al.*(2013), Pham *et al.*(2015)]. In [Lin *et al.*(2008)], the authors show that a β-fairness utility function can be maximized, while guaranteeing system stability, under the assumptions that the number of users per class follows a recurrent Markov Chain. We follow a similar rationale as that in [Lin *et al.*(2008)] for the intermediate decisions. However, since we are not considering a proper utility function, but rather the implicit one that corresponds to the optimal QMW policy, convergence remains an open issue. Like in the second class of works we consider multi-time scales but those correspond to different layers of the architecture rather than to different allocation problems that take place in different layers of the protocol stack. To illustrate what the Layback architecture could enable, our specific goal in this thesis chapter is to focus on the the benefits obtained by sharing the backhaul resources dynamically, rather than sharing all resources available at the different layers. In addition to considering different time-scales across the different Layback layers, we also incorporate an economic constraint in the allocation across different operators, which is enforced via the Lyapunov drift plus penalty: a method introduced in [Georgiadis *et al.*(2006),Neely(2006)], and extensively used in recent years for dynamic control. Numerical results suggest that the proposed approach can effectively minimize delays, by enabling a flexible resource redistribution of backhaul resources across different operators.s

4.3 Optimization Framework

Let us consider a general rooted tree network with $P+1$ layers and use $\mathcal{A}_p$ to denote the set of agents at the p-th layer, with $p = 0, 1, \ldots, P$. By convention $\mathcal{A}_0 = \{0\}$: we refer to this agent as *root* or *orchestrator*. The agents' goal is to maximize, at each

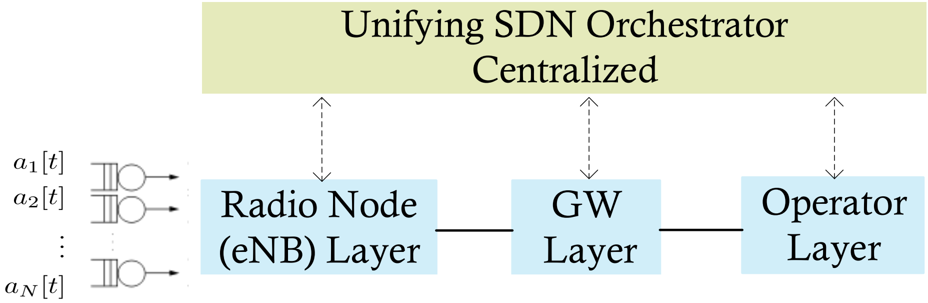


Figure 4.2.1: High level diagram of LayBack architecture: The centralized unifying SDN orchestrator orchestrates the operation of all LayBack layers.

instance t, a NUM at the P-th layer, e.g.

$$\underset{\boldsymbol{x}}{\text{maximize}} \quad \sum_{b \in \mathcal{A}_P} u_b(x_b; t) \tag{4.3.1a}$$

$$\text{subject to} \quad \mathbf{A}(t)\boldsymbol{x} \leq \mathbf{c}(t) \tag{4.3.1b}$$

We limit here to consider linear constraints in (4.3.1b): for non linear constraints, typical approaches are either a linearization of the constraint looking at the Hessian, or a relaxation of the problem by adding a penalty term in the objective that "reflects" the constraint. The formulation in (4.3.1) is typically complex to implement, since it requires information from all agents at the bottom layer P to be "centralized" to solve the optimization. This leads to use the vector $\boldsymbol{x}$ to indicate the decisions of all the agents in the network; the notation $\boldsymbol{x}_{\mathcal{A}}$, as previously, indicates the sub-vector with entries $i \in \mathcal{A}$. The set $\mathcal{A}_{p|a}$ is the set of elements $b \in \mathcal{A}_p$ that have the same father $a \in \mathcal{A}_{p-1}$. For the sake of solving (4.3.1), the decision variables of agents at intermediate layers are slack variables. As we will see however, these additional variables represent actual network decisions in the distributed and time-decomposed implementation of (4.3.1b). We then introduce the following recursive optimization problem by defining $\mathcal{U}^{\star}_{p,a}(x_a; t)$, for $p = 0, \ldots, P-1$ ($a \in \mathcal{A}_p$) as the optimal value of the subproblem

$$\underset{\boldsymbol{x}_{\mathcal{A}_{p+1|a}}}{\text{maximize}} \quad \sum_{b\in\mathcal{A}_{p+1|a}} \mathcal{U}^{\star}_{p+1,b}\left(x_b;t\right) \tag{4.3.2a}$$

$$\text{subject to} \quad \mathbf{A}_a(t)\boldsymbol{x}_{\mathcal{A}_{p+1|a}} \leq \mathbf{c}_a(x_a;t) \tag{4.3.2b}$$

As a starting point of the optimization, we have that x_0 is not a variable of the problem but a constant, e.g. at the *root* level we have $\mathcal{U}^{\star}_0(x_0,t) = \mathcal{U}^{\star}_0(t)$ and $\mathbf{c}_0(x_0,t) = \mathbf{c}_0$. Finally, $\mathcal{U}^{\star}_P\left(x_a;t\right)$ is the optimal value of the bottom layer utility maximization, i.e.:

$$\underset{\boldsymbol{x}_{\mathcal{A}_{P|a}}}{\text{maximize}} \quad \sum_{b\in\mathcal{A}_{P|a}} u_b(x_b;t) \tag{4.3.3a}$$

$$\text{subject to} \quad \mathbf{A}_a(t)\boldsymbol{x}_{\mathcal{A}_{P|a}} \leq \mathbf{c}_a(x_a;t) \tag{4.3.3b}$$

We now derive the sequence of updates of each variable to iteratively solve the problem we have just decomposed in different layers. The next subsection serves as a basis to tackle the problem at different timescales, imposed by the network infrastructure, which will be discussed in 4.3.2. It is however easier to derive them in the *ideal* static case first, given that the expressions in the dynamic case will have the same form, albeit having a different meaning.

4.3.1 Iterative Solution via Gradient Descent

The chain of optimization problems formulated in a recursive manner in (4.3.2)-(4.3.2) can be written as a unique optimization, by introducing the set of Lagrangian dual vectors $\{\boldsymbol{\lambda}_{p|a}\}$ for each constraint in (4.3.2b) and (4.3.3b). The notation $\mathbf{A}_{a,b}$ indicates the b-th column of the matrix $\mathbf{A}_a$. We define

$$\Phi_{2p+1}\left(x_a,\boldsymbol{\lambda}_{p|a}\right) \triangleq \boldsymbol{\lambda}^T_{p|a}\mathbf{c}_a(x_a) + \sum_{b\in\mathcal{A}_{p+1|a}} \max_{x_b} \ \Phi_{2p}(\boldsymbol{\lambda}_{p|a},x_b) \tag{4.3.4}$$

for $p = 0, 1, \dots, P-1, a \in \mathcal{A}_p$.

$$\Phi_{2p}(\boldsymbol{\lambda}_{p|a}, x_b) \triangleq \begin{cases} -\boldsymbol{\lambda}_{p|a}^T \mathbf{A}_{a,b} x_b + \min\limits_{\boldsymbol{\lambda}_{p+1|b}} \ \Phi_{2p+1}\left(x_b, \boldsymbol{\lambda}_{p+1|b}\right) & \text{for } p = 1, \dots, P-1 \\ -\boldsymbol{\lambda}_{p|a}^T \mathbf{A}_{a,b} x_b + u_b(x_b) & \text{for } p = P \end{cases}, \tag{4.3.5}$$

for $b \in \mathcal{A}_{p+1|a}$.

Note that we have used the same convention, above described, for the pseudo-variable x_0 and functions of it. By gradient descent from the functions just defined, we can derive the sequence of updates

$$\boldsymbol{\lambda}_{p|a}^{(k+1)} = \left[\boldsymbol{\lambda}_{p|a}^{(k)} - \alpha_{2p+1}^{(k)}\left(\mathbf{c}_a(x_a) - \sum_{b \in \mathcal{A}_{p+1|a}} \arg\max_{x_b} \ \Phi_{2p+2}\left(\boldsymbol{\lambda}_{p|a}^{(k)}, x_b\right)\right)\right]^+ \tag{4.3.6}$$

$$x_b^{(k+1)} = \left[x_b^{(k)} + \alpha_{2p}^{(k)}\left(\arg\min_{\boldsymbol{\lambda}_{p+1|b}} \Phi_{2p+3}(x_b^{(k)}, \boldsymbol{\lambda}_{p+1|b}) - \boldsymbol{\lambda}_{p|a}^T \mathbf{A}_{a,b}\right)\right]^+ \quad \text{for } b \in \mathcal{A}_{p+1|a} \tag{4.3.7}$$

Notice that, to ensure the convergence of the decomposition, the updates in (4.3.6)–(4.3.7) have to be read as follows: to reach the optimal $\boldsymbol{\lambda}_{p|a}$, the agent a needs to perform a sufficient number of iterations in (4.3.6). However, before computing one iteration of (4.3.6), all the agents, children of a in the network, should perform a sufficient number of iterations of (4.3.7) upon receiving the Lagrangian $\boldsymbol{\lambda}_{p|a}$, and so on. Unless a value can be computed in closed form in one shot, each update that includes the solution of an optimization problem (i.e., it has an $\arg\max$ / $\arg\min$ term in the update) requires a sufficient number of gradient descent updates at the lower level to approximate the solution of the maximization. Therefore, the indices k in (4.3.6)–(4.3.7) are *not* associated with the same time scale (i.e., they are not the same updates counter). If the computation at each layer and the communication delays among layers were all negligible, we would be in the *time-scale separation* regime that we mentioned in the discussion on prior works. However, this is not possible in a real system, since latencies play an important role and the framework we are about to explain explicitly takes these latencies into consideration. We use $\tau^{p}_{p+1}, p = 0, 1 \dots, P-1$ to indicate the Round Trip Time (RTT) between layers p and $p+1$. Let us start by considering the optimization at the bottom layer as the one that operates at the minimum latency, i.e., the time difference between the time indexes t and $t+1$ is the RTT τ^{P-1}_{P} (considered equal, for simplicity, for all agents between the two layers). For our application this layer will be the closest one to the user and to the information regarding traffic. To map all the time instants into integer values of t it is convenient to normalize all times with respect to τ^{P-1}_{P} (i.e. we set $\tau^{P-1}_{P} = 1$). The first aspect considered in our framework is that in actual network infrastructures one has constraints that prevent the redistribution of resources across different layers, i.e. the frequency at which $\boldsymbol{x}_{\mathcal{A}_{p|a}}$ can change is different for different p; typically layers with lower p, i..e closest to the root, can be changed at a slower frequency than layers closer to the bottom. Therefore, even if a genie could compute the optimal solution of the decomposed problem at each instant t, it might not be possible to implement the decision. Denoting with $\underline{T_p}$ the minimum refresh times for the decisions on the $(p+1)$-th layer, time t can be written according to a poly-phase decomposition as follows:

$$t = \sum_{p=0}^{P-1} m_p T_p, \; m_0 \in \mathbb{N}, 0 \leq m_p \leq \tilde{T}_p - 1 \text{ for } \; p = 1, \dots, P-1 \tag{4.3.8}$$

$$\tilde{T}_p \triangleq \frac{T_p}{\prod_{p'=p+1}^{P-1} T_{p'}} \tag{4.3.9}$$

where $T_p > \underline{T_p}$ and $\mod (T_p, \prod_{p'=p+1}^{P-1} T_{p'}) = 0, \; \forall p = 0, \dots, P-1$. In the next subsection, to comply with the refresh time limits, the greedy optimization, decoupled at any instant t, is mapped into the stochastic optimization we solve. Changing the objectives from deterministic values to expected values is necessary to capture the uncertainty of the impact of the decisions $\boldsymbol{x}$ on the utility obtainable at the bottom layer, which varies with time at a faster pace than the refresh times $T_p, 0 \leq p < P-2$.

4.3.2 Stochastic Optimization and Temporal Decomposition

Since the different layers cannot communicate instantaneously, the parameters of the utility $u_b(x_b;t),\ b \in \mathcal{A}_P$ might change dynamically underneath, while decision from the upper layers are not updated. Clearly, the objectives of the optimization have to be defined in such a way that they stay constant while the bottom layer changes stochastically from one state to the other. The proposed framework can be seen as a special case of *stochastic gradient descent* where the network dynamics impose the sequence of training samples' updates. In particular, the *root* agent operates its optimization at every time instant $t_{m_0} \triangleq m_0 T_0$ and in general the agent $a \in \mathcal{A}_p, p < P$ performs its optimization at every $t_{\mathbf{m}_0^p} \triangleq \sum_{p'=0}^{p} m_{p'} T_{p'}$. The recursive formulation is then modified to account for the different time scales and we obtain the following optimization, solved by the agent $a \in \mathcal{A}_p, p < P$

$$\underset{\boldsymbol{x}_{\mathcal{A}_{p+1|a}}}{\text{maximize}} \sum_{b \in \mathcal{A}_{p+1|a}} \frac{1}{\tilde{T}_{p+1}} \sum_{m_{p+1}=0}^{\tilde{T}_{p+1}-1} \mathbb{E}\left\{\mathcal{U}^{\star}_{p+1,b}\left(x_b; t_{\mathbf{m}_0^{p+1}}\right)\right\} \tag{4.3.10a}$$

$$\text{subject to} \quad \mathbf{A}_a\left(t_{\mathbf{m}_0^p}\right) \boldsymbol{x}_{\mathcal{A}_{p+1|a}} \leq \mathbf{c}_a\left(x_a; t_{\mathbf{m}_0^p}\right) \tag{4.3.10b}$$

for which the optimal value is defined as $\mathcal{U}^{\star}_{p,a}(x_a; t_{\mathbf{m}_0^p})$. The same considerations over the starting point and the bottom optimization at every T_{P-1} apply here as well. The updates derived in (4.3.6)–(4.3.7) will then be used to update the decisions $\boldsymbol{x}_{\mathcal{A}_{p|a}}$ at every T_p, for $p = 0, \ldots, P-1$, as if convergence to the solution of a static problem has been achieved in the time horizons of length T_p, respectively. Note, however, in practice the problem is not static, due to the underlying evolution of the utility function $u_b, b \in \mathcal{A}_P$ at every unit of time: we then proceed to discuss the nature and significance of the *static* approximation.

First, in light of the discussion on the updates in Sec. 4.3.1, we note that the number of iterations of each update needs to "fit into" the time intervals T_p that separate the decisions. Let us introduce K_{2p+1} as the number of iterations of each update in (4.3.6) for $\boldsymbol{\lambda}_{p|a}$ and K_{2p} for the number of iterations of each update in (4.3.7) for $x_b, b \in \mathcal{A}_{p|a}$. We then have the following relations

$$T_p \geq \max \left\{ \sum_{p'=p}^{P-1} \tau_{p'+1}^{p'} \prod_{p''=2p+1}^{2p'+1} K_{p''}, \underline{T_p} \right\} \text{ for } p = 0, 1 \ldots, P-1 \tag{4.3.11}$$

where, as anticipated, by convention we have $\tau_P^{P-1} = 1$ and, if the bottom layer optimization can be computed in one shot, then $K_{2P-1} = 1$. The inequalities in (4.3.11) indicate that, if we want to act fast, e.g., reduce T_p (possibly to the minimum refresh times) we need to perform fewer iterations. Vice versa, if we want to perform more iterations, we have to be willing to act slower in updating the decisions $\boldsymbol{x}_{\mathcal{A}_{p|a}}$. For the the particular case-study in the remainder of this chapter we will consider a fixed design for T_p, and $K_{p'}$, and explore the performance only numerically (see Section 4.4). Note that, the impact of the choice of the $K_{p'}$ has not been fully addressed in the literature, where these parameters are implicitly predetermined in the formulations studied. If we look at the static problem, as a "surrogate" for the dynamic problem (up to the next decision), increasing the number of iterations and delaying future decisions can guarantee a better accuracy for a static scenario; however, the ability of the algorithm to incorporate new dynamic information is compromised. The question just described is beyond the scope of this thesis. The key to distribute the updates at different time scales, as information changes underneath, is to view them as *stochastic gradient descent* iterations where the network dynamics impose the sequence of training samples' updates, i.e., samples of the gradient are used in lieu of expectations. As mentioned in Section 4.2 of this chapter, the work in [Lin *et al.*(2008)] gives a convergence result for a special case of the framework proposed. In particular, reformulating their result according to our notation, it has been proven that

Theorem. [Lin *et al.*(2008)] *For a network with $P+1=3$ layers, where the number of users per class follows a recurrent Markov Chain, and the utility to maximize at the bottom layer is a β-fairness function, i.e.:*

$$u_b(x_b) = \begin{cases} w_b \frac{x_b^{1-\beta}}{1-\beta}, & \textit{for } \beta > 0, \textit{and} \;\; \beta \neq 1 \\ w_b \log x_b, & \textit{for } \beta = 1 \end{cases} \tag{4.3.12}$$

then with $K_p = 1 \; \forall p = 0, 1, 2$ the optimal stability region can be achieved.

Even if the rationale of Lagrangians' updates for intermediate decisions is similar, optimality and convergence of the proposed approach when a different type of utility function is used, and when there is no Markovian assumption on the exogenous traffic, remains an open problem. In the example that follows, in the utility function (that corresponds to a max-projection scheduling) the rates are weighted by the different users (eNBs)' queues and they are therefore not statistically independent from the decisions, as the parameter w_b of the β-fairness function in (4.3.12). In the simulations however we consider Poisson arrivals with a non-stationary rate, which is a Markov process.

4.3.3 A Study Case for Minimum Delay

We consider a network with O distinct operators, indexed by $o \in \mathcal{A}_1$. Each operator manages a set of Smart-Gateways (GWs), indexed by $g \in \mathcal{A}_{2|o}$. In turn, each GW g manages a set of e-NodeBs (eNBs), indexed by $n \in \mathcal{A}_{3|g}$. The operators are coordinated by the SDN orchestrator (e.g. the *root* element in $\mathcal{A}_0$). The set of all eNBs is naturally $\mathcal{A}_3 = \bigcup_{o=1}^{O} \bigcup_{g \in \mathcal{A}_{2|o}} \mathcal{A}_{3|g}$ and similarly for the upper layers. The queues of each eNB $n \in \mathcal{A}_3$ are denoted by P_n and their dynamics are

$$Q_n[t+1] = [Q_n[t] - x_n[t]]^+ + a_n[t+1] \tag{4.3.13}$$

where $a_n[t]$ and $x_n[t]$ represent, respectively, the exogenous packets arrival process and service rate granted at eNB n, during the t-th slot. The work in [Thyagaturu *et al.*(2016)] introduced the concept of Smart Gateways and discussed the possibility of sharing bandwidth between operators to improve uplink throughput and efficiency. Let us start from the centralized optimization we wish to emulate. If the SDN orchestrator, having full control of the total service rate denoted by c_0, could allocate it directly the eNBs, the optimization:

$$\begin{aligned}
&\underset{\boldsymbol{x}_{\mathcal{A}_3}}{\text{minimize}} && \limsup_{\tau\to\infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} \sum_{n\in\mathcal{A}_3} \mathbb{E}\{Q_n[t]\} && (4.3.14a)\\
&\text{subject to} && \sum_{n\in\mathcal{A}_3} x_n[t] \le c_0 \ \forall t, && (4.3.14b)\\
& && 0 \le x_n[t] \le Q_n[t] \ \forall t, n \in \mathcal{A}_3. && (4.3.14c)
\end{aligned}$$

by minimizing the long term average queue, as a result of Little's theorem [Allen(1990)], would also minimize the end-to end delay in the network (4.3.14a). As discussed in Section 4.2, in the simplified scenario considered, a QMW policy is delay-optimal. We adapt such policy for our context, and write it as the greedy maximization, at each t, in Algorithm 4.1. Note that this corresponds to the formulation in (4.3.1) with a linear utility $u_n(x_n) = Q_n x_n$. The problem of uplink traffic management via SDN orchestrator falls naturally under the decomposition characteristics discussed in the previous section. The constraints for each layer correspond to a different rate allocation at each agent, e.g., at the operator level, at the GW level or at the eNB level. As a simplification for this case we will consider a scalar constraint for (4.3.2b) where $\mathbf{A}_a = \mathbf{1}^T$ and $\mathbf{c}(x_a; t) = x_a$, e.g. in the architecture we can grant any rate to

Algorithm 4.1: Ideal centralized scheduler (QMW)

```
At every $t = 0, 1, \ldots, \tau - 1$
    • receive $Q_n[t]$ from the operators
    • allocates the vector $\boldsymbol{x}_{\mathcal{A}_3}[t]$ by solving
$\max_{\boldsymbol{z}} \sum_{n \in \mathcal{A}_3} Q_n[t] z_n \text{ s.t. } \sum_{n \in \mathcal{A}_3} z_n \le c_0,\ 0 \le z_n \le Q_n[t]\ \forall n \in \mathcal{A}_3$    (4.3.15)
```

the "children" as long as the sum is below the rate assigned to the father. For this case, however, it is important however to remark that the allocation of $\boldsymbol{x}_{\mathcal{A}_1}$ needs to respect an "economic" constraint across the operators, that defines a contractual service obligation and prevents any operator from gaming the system (i.e., consistently acquiring more resources than what it paid for). This constraint is imposed on the long-run average of the decisions $\boldsymbol{x}_{\mathcal{A}_1}$, i.e.:

$$\limsup_{\tau \to \infty} \frac{1}{\tau} \sum_{t=0}^{\tau-1} x_o[t] \le c^{(o)} \quad \forall\ o \in \mathcal{A}_1 \tag{4.3.16}$$

where, for consistency of the problem, it is necessary to have $\sum_{o \in \mathcal{A}_1} c^{(o)} \le c_0$.

At the same time, by having an inequality constraint, we are not forced to assign resources to an operator that would be wasted if there is not sufficient uplink demand. We use the idea of *virtual queues*, following the Lyapunov drift-plus-penalty approach [Georgiadis *et al.*(2006)] to encode the constraint in (4.3.16), and modify (4.3.10) for $p = 0$ into:

$$\underset{\boldsymbol{x}_{\mathcal{A}_1}}{\text{maximize}} \quad \sum_{o \in \mathcal{A}_1} -\frac{\Theta_o x_o}{V} + \frac{1}{\tilde{T}_1} \sum_{m_1=0}^{\tilde{T}_1 - 1} \mathbb{E}\left\{ \mathcal{U}^{\star}_{1,o}\left(x_o; t_{\mathbf{m}_0^1}\right)\right\} \tag{4.3.17a}$$

$$\text{subject to} \quad \sum_{o \in \mathcal{A}_1} x_o \le c_0 \tag{4.3.17b}$$

After deciding $\boldsymbol{x}_{\mathcal{A}_1}[t_{m_0}]$, the virtual queues Θ_o's are updated as:

$$\Theta_o[t_{m_0+1}] = \left[\Theta_o[t_{m_0}] + (x_o[t_{m_0}] - c^{(o)}), 0\right]^+ . \tag{4.3.18}$$

The parameter V represents the "flexibility" of the constraint in (4.3.16), e.g., the higher V the more inclined we are to temporarily violate the constraint. The general formulation derived in the previous section can be used to derive the sequence of updates for $\lambda_0, \{\lambda_{1|o} : o \in \mathcal{A}_1\}, \{\lambda_{2|g} : g \in \mathcal{A}_2\}$ and the decisions $\{x_o : o \in \mathcal{A}_1\}, \{x_g : g \in \mathcal{A}_2\}, \{x_n : n \in \mathcal{A}_3\}$. Note that, in light of the addition of the Lyapunov penalty on the operators' redistribution, the update for $x_o, o \in \mathcal{A}_1$ will be

$$x_o^{(k+1)} = \left[x_o^{(k)} + \alpha_2^{(k)} \left(\arg\min_{\lambda_{1|o}} \Phi_3(x_o^{(k)}, \lambda_{1|o}) - \lambda_0 - \frac{\Theta_o}{V}\right)\right]^+ \tag{4.3.19}$$

The pseudocodes for the procedures at each layer (i.e. agent of the network) are reported in Algorithms 4.2-4.6 and Fig.4.3.1 illustrates the message exchange between layers and the time decomposition of the problem.

Algorithm 4.2: Iterates for λ_0 (at the SDN orchestrator)

```
Input  : $\lambda_0^{(0)},\ k_1 = 0$
Output: $\lambda_0^{(K_1)}, \boldsymbol{x}_{\mathcal{A}_1}$
while $k_1 < K_1$ do
    Call Algorithm 4.3 with input $\lambda_0^{(k_1)}$ to all operators;
    Receive $\boldsymbol{x}_{\mathcal{A}_1}^{(K_2)}\Big(\lambda_0^{(k_1)}\Big)$ and update $\lambda_0^{(k_1+1)}$ via (4.3.6);
    $k_1 \leftarrow k_1 + 1$;
Decide $\boldsymbol{x}_{\mathcal{A}_1}[t_{m_0}]$ by projecting $\boldsymbol{x}^{(K_2)}\Big(\lambda_0^{(K_1-1)}\Big)$ onto the feasible set in
 (4.3.10b);
$m_0 \leftarrow m_0 + 1$;
Call Algorithm 4.2 with input $\lambda_0^{(0)} \leftarrow \lambda_0^{(K_1)}$;
```

Algorithm 4.3: Iterates for x_o (at the operator)

```
Input  : $\lambda_0,\ k_2 = 0$, ($x_o^{(0)}$ only if first call)
Output: $x_o^{(K_2)}$
while $k_2 < K_2$ do
    Call Algorithm 4.4 with input $x_o^{(k_2)}$;
    Receive $\lambda_{1|o}^{(K_3)}\left(x_o^{(k_2)}\right)$ and update $x_o^{(k_2+1)}$ via (4.3.19);
    $k_2 \leftarrow k_2 + 1$;
$x_o^{(0)} \leftarrow x_o^{(K_2)}$
```

Algorithm 4.4: Iterates for $\lambda_{1|o}$ (at the operator)

```
Input  : $x_o, k_3 = 0$, ($\lambda_{1|o}^{(0)}$ only if first call)
Output: $\lambda_{x_o}^{(K_2)}$
while $k_3 < K_3$ do
    Call Algorithm 4.5 with input $\lambda_{1|o}^{(k_3)}$;
    Receive $\boldsymbol{x}_{\mathcal{A}_{2|o}}^{(K_4)}\left(\lambda_{1|o}^{(k_3)}\right)$ and update $\lambda_{1|o}^{(k_3+1)}$ via (4.3.6);
    $k_3 \leftarrow k_3 + 1$;
Decide $\boldsymbol{x}_{\mathcal{A}_{2|o}}[t_{\mathbf{m}_0^1}]$ by projecting $\boldsymbol{x}_{\mathcal{A}_{2|o}}^{(K_4)}\left(\lambda_{1|o}^{(K_3-1)}\right)$ onto the feasible set in
  (4.3.10b);
$m_1 \leftarrow m_1 + 1$;
```

Algorithm 4.5: Iterates for x_g (at the GW $g \in \mathcal{A}_{2|o}$)

```
Input  : $\lambda_{1|o}, k_4 = 0$, ($x_g^{(0)}$ only if first call)
Output: $x_g^{(K_4)}$
while $k_4 < K_4$ do
    Call Algorithm 4.6 with input $x_g$ to solve (4.3.3);
    Receive $\lambda^{\star}_{2|g}(x_g^{(k_4)})$ and update $x_g^{(k_4+1)}$ via (4.3.7);
    $k_4 \leftarrow k_4 + 1$;
```

Algorithm 4.6: Solution of (4.3.3)

```
Input  : $x_g, \{Q_n : n \in \mathcal{A}_{3|g}\}$
Output: $\lambda^{\star}_{2|g}$
if $\sum_{n \in \mathcal{A}_{3|g}} Q_n \geq x_g$ then
    Find the permutation $\boldsymbol{\pi} = \{\pi_i : i = 1, \ldots, |\mathcal{A}_{3|g}|\}$ to
    sort the queues $\mathbf{Q}$ such that $i \geq j \Rightarrow Q_{\pi_i} \leq Q_{\pi_j}$;
    Find $i^* = \inf\{i : \sum_{j=1}^{i} Q_{\pi_j} \geq x_g\}$;
    $x_{\pi_j} = Q_{\pi_j}$ for $j < i^*$, $x_{\pi_{i^*}} = x_g - \sum_{j=1}^{i^*-1} Q_{\pi_j}$, $x_{\pi_j} = 0$ for $j > i^*$,
     $\lambda^{\star}_{2|g} = Q_{\pi_{i^*}}$;
else
    $x_n = Q_n \; \forall n \in \mathcal{A}_{3|g}, \lambda^{\star}_{2|g} = 0$;
Decide $\boldsymbol{x}_{\mathcal{A}_{3|g}}[t_{\mathbf{m}_0^2}]$ by solving the procedure in this Algorithm for $x_g = x_g[t_{\mathbf{m}_0^1}]$;
$m_2 \leftarrow m_2 + 1$;
```

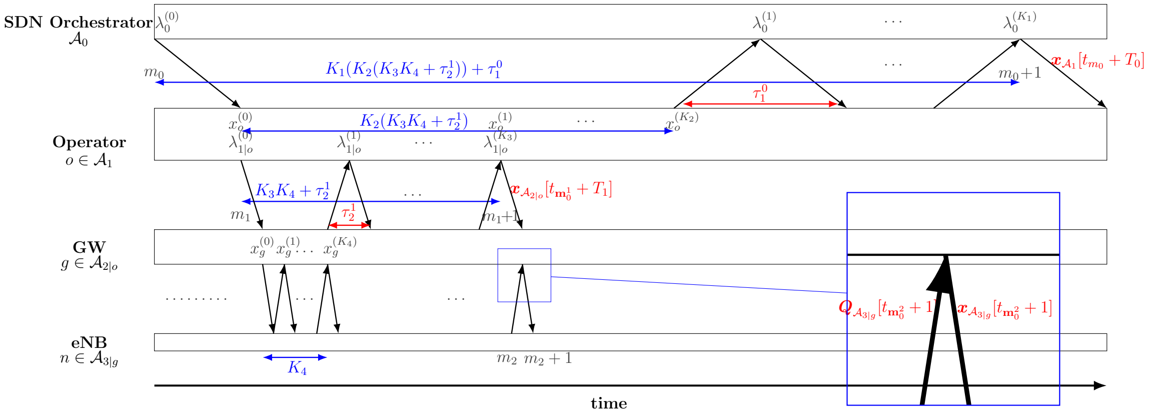
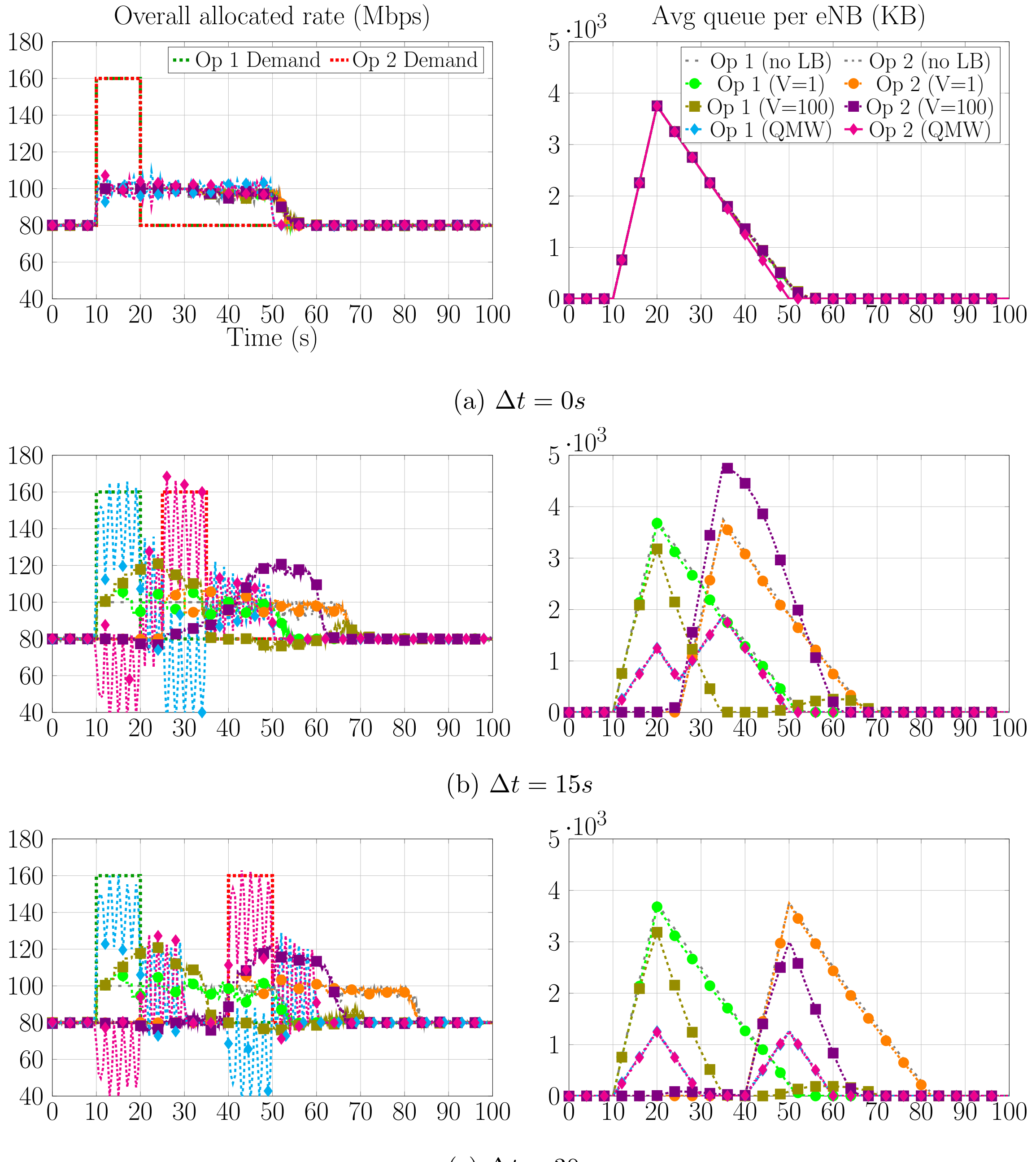

SDN Orchestrator $\mathcal{A}_0$
$\lambda_0^{(0)}$
$\lambda_0^{(1)}$
. . .
$\lambda_0^{(K_1)}$
m_0
$K_1(K_2(K_3K_4+\tau_2^1))+\tau_1^0$
m_0+1
$\boldsymbol{x}_{\mathcal{A}_1}[t_{m_0}+T_0]$
τ_1^0
Operator $o \in \mathcal{A}_1$
$x_o^{(0)}$
$K_2(K_3K_4+\tau_2^1)$
$x_o^{(1)}$
$x_o^{(K_2)}$
$\lambda_{1|o}^{(0)}$
$\lambda_{1|o}^{(1)}$
$\lambda_{1|o}^{(K_3)}$
$K_3K_4+\tau_2^1$
$\boldsymbol{x}_{\mathcal{A}_{2|o}}[t_{\mathbf{m}_0^1}+T_1]$
m_1
τ_2^1
m_1+1
GW $g \in \mathcal{A}_{2|o}$
$x_g^{(0)}$ $x_g^{(1)}$ . . . $x_g^{(K_4)}$
$\boldsymbol{Q}_{\mathcal{A}_{3|g}}[t_{\mathbf{m}_0^2}+1]$
$\boldsymbol{x}_{\mathcal{A}_{3|g}}[t_{\mathbf{m}_0^2}+1]$
eNB $n \in \mathcal{A}_{3|g}$
K_4
m_2 m_2+1
time

Figure 4.3.1: Illustration of the dynamics of the multi-timescale optimization framework within context of LayBack infrastructure: the optimal policy to minimize end-to-end delay is decoupled into multiple layers of sub-problems. The decisions' timescale decreases by moving down to the lower LayBack layers. The figure shows the dynamics between different layer, which are performed by the multiple entities in each Layback layer.

4.4 Simulation Results

In this section, we show the effectiveness of the proposed method in handling demand peaks across different operators by multiplexing resources dynamically. The bottleneck of the proposed approach is that, due to network latencies, high level decisions cannot be instantaneous and if one of the operators experiences a demand peak right after the other, the first of the event creates a response lag in addressing the subsequent events. In our experiments we test different values of the parameter V in (4.3.17). Our baselines are: 1) absence of the LayBack orchestrator, e.g. fixed allocation for $\boldsymbol{x}_{\mathcal{A}_1}$ (labeled "no LB" in the plots) and 2) a centralized optimal scheduler, e.g. the solution of Algorithm 4.1, with no latency and no long term constraints limiting operators (labeled "QMW" in the plots). The parameters in Fig. 4.3.1 are set to $K_1 = 1, K_2 = 5, K_3 = 1, K_4 = 10, \tau_1^0 = 100, \tau_2^1 = 10$, which correspond to 1s and 100ms for an RTT between GWs and eNBs of 10ms latency respectively. The Lagrangians λ_0 and $\lambda_{1|o}$ are initialized as standard uniform random variables. T_1 and T_0 are set to 20 and 200 respectively ($T_2 = 1$). For all the updates $\alpha = 0.4$. For numerical stability, the computation of $\lambda^{\star}_{2|g}$ uses the following queues' normalization $\frac{Q_n}{\sum_{n\in\mathcal{A}_{3|g}} Q_n}\frac{|\mathcal{A}_{3|g}|}{2}$, which does not alter the solution. The network has the following parameters: $O = 2, |\mathcal{A}_{2|o}| = 2\ \forall\ o, |\mathcal{A}_{3|g}| = 10\ \forall\ g \in \mathcal{A}_2, c^{(o)} = 100\text{Mbps}\ \forall\ o, c_0 = 200\text{Mbps}$. For our simulations we have considered a time interval of 100s. The aggregate rate demand for each operator is kept constant at 80Mbps, except for a peak of 10s duration of 160Mbps, for each operator. Operator 1 experiences the peak in demand rate at time $t = 10$s, whereas for Operator 2 the peak happens at time $t = (10 + \Delta t)$s. At all times, the traffic is homogeneous across the same operator's eNBs. For the selected time parameters and a packet size of 12.5 KBytes, the scenario just described corresponds to a process $a_n[t]$ in (4.3.13) as $Pois(0.4)$ in normal conditions and $Pois(0.8)$ when the demand peak occurs. In Fig. 4.4.1, we show three different simulations over time for different values of Δt: for $\Delta t = 0$ traffic is perfectly balanced, hence no redistribution across operators is enabled, for $\Delta t = 15$s the aforementioned overshadowing effect can be seen in the delay to which the system for $V = 100$ responds to the demand peak for operator 2. Finally for $\Delta t = 30$s, there is enough time for our decomposition to redistribute the resources and have both operators benefit from sharing.

The phenomenon just described is summarized in Fig. 4.4.2, where we plotted the time average queue size (over the whole simulation time) vs. the interval Δt that separates the two demand peaks. Notice how for small values of V (e.g $V = 1$) sharing is limited and performances are not significantly different from the absence of SDN orchestration. As V increases, we enable sharing, and when the demand peaks are sufficiently separated, we can guarantee smaller average queues for both operators, closing the gap with the optimal curves for the centralized solution. The shadowing effect described for small Δt is evident since Operator 1 has a smaller average queue size than in the centralized optimization, where it is not prioritized because there is



(a) $\Delta t = 0s$

(b) $\Delta t = 15s$

(c) $\Delta t = 30s$

Figure 4.4.1: Aggregate rate allocation for the two operators for different values of V and when no sharing across operators is enabled

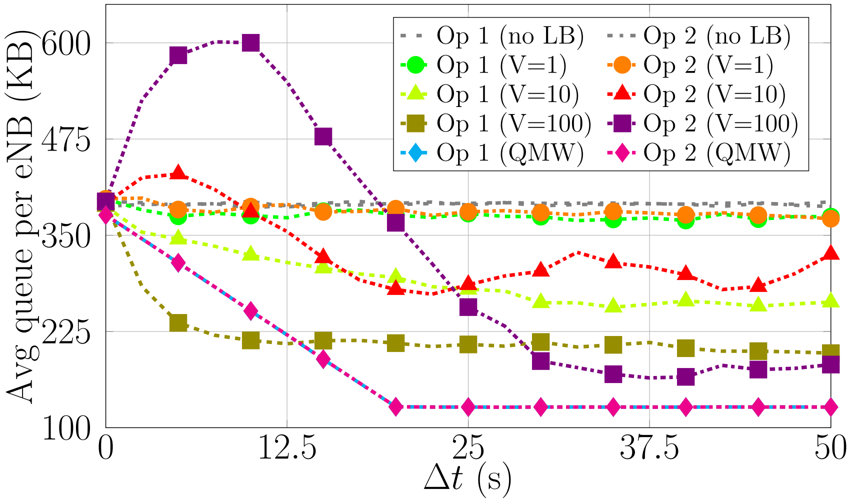


Figure 4.4.2: Average aggregate queue size at each operator for different time distances between demand peaks.

no lag in responding to events occurring later.

CHAPTER 5

CONCLUSIONS

In this thesis we have proposed three different problems for resource allocation that the new generation of Wireless Communications could benefit from. In the first part of the thesis we focused on the *time* resource, and proposed a cross-layer lightweight protocol (PulseSS) that, through signalling at the PHY layer, can achieve synchronization and scheduling in a locally connected network of sensors. The main benefits of the proposed protocol are resilience and robustness to network failures, attained via its decentralized nature, and lightness of the control overhead required: two features that make it highly desirable for industrial control and large area sensor deployments. The mathematical analysis of the updates' dynamics has provided important performance guarantees and design guidelines for the protocol implementation: a testbed evaluation and micro-controller implementation of the protocol has been conducted by my group and this has lead to two journal publications, two conferences and a filed provisional patent application. In the second part of the thesis, we moved our focus to the *frequency* resource: a newly defined analog front-end, named the Cognitive Utility Maximization Multiple Access (CUMMA) receiver, was proposed to enable sequential and adaptive Cognitive Spectrum Sensing. We started by introducing a POMDP formulation for a sequential single-band scanner: in doing so, we relaxed previous constraints in the literature and studied a more general case, enhancing the receiver's flexibility in the tests' design and decisions over the different sub-bands. We proposed a set of sub-optimal, numerically accessible, heuristics to approximate the optimal strategy, analytically derived. We then moved to formulate a set optimization problem to incorporate Compressed Sensing tests, e.g., tests that via a spectrum aliasing analog mixer, fold the spectrum prior to sampling and aim at providing informations over multiple sub-bands at the same time. We designed our measurements, following the spirit of prior works that have advocated for the use of sparse matrices to tackle the *noise-folding phenomenon*, caused by the spectrum aliasing. However, we show that the set optimization proposed and the feature of adaptivity in selecting the best combinations of tests can guarantee better performances, in the finite regime, than static methods as LASSO with dense matrices or belief propagation via message-passing and LDPC matrices. The performances have been measured via a utility function, which strikes the trade-off between exploration and exploitation. The utility function parameters have been derived considering outage rate and interference in a Cognitive Radio scenario, where the Secondary Users wants to avoid interfering with the Primary Users. For both scenarios considered, our strategies guaranteed a higher utility than prior schemes. Finally, we have proposed a new scheme for Decomposition of Infrastructure-based Dynamic Resource Allocation (DIDRA), to tackle the problem of *rate* allocation in uplink traffic management at the backhaul. Inspired by the extensive work on Network Utility Maximization (NUM) approaches to solve resource allocation problems, we have derived a unifying framework for redistribution of transmission resources across operators' core networks and gateways. Realistic latencies and time constraints have been included in our formulation to propose a distribution of computation tasks, governed by the Layback SDN architecture, proposed by Prof. Martin Reisslein's group here at Arizona State University, with whom we have collaborated. The proposed method combines the dynamic of Lagrangian updates (for gradient descent in primal-dual decomposition), with Lyapunov-drift-plus-penalty relaxation. The numerical results indicate that, while guaranteeing an economic fairness across the different operators, a flexible redistribution of resources can be enabled: if two operators experience demand peaks in uplink, separated in times by at least $15 - 20$ s, the method proposed can effectively guarantee a lower end-to-end delay for both, compared to a static redistribution of resources between the two. Convergence of the proposed approach, optimality-gap from the centralized implementation, and additional constraints at the different layers of the protocol stack are beyond the scope of this thesis.

REFERENCES

[Akyildiz *et al.*(2008)] Akyildiz, I., W.-Y. Lee, M. C. Vuran and S. Mohanty, "A survey on spectrum management in cognitive radio networks", Communications Magazine, IEEE **46**, 4, 40–48 (2008).

[Allen(1990)] Allen, A. O., "Statistics and queuing theory with computer science applications, vol. 2", (1990).

[Altman *et al.*(2012)] Altman, E., K. Avrachenkov and S. Ramanath, "Multiscale fairness and its application to resource allocation in wireless networks", Computer Communications **35**, 7, 820–828 (2012).

[Andrews *et al.*(2004)] Andrews, M., K. Kumaran, K. Ramanan, A. Stolyar, R. Vijayakumar and P. Whiting, "Scheduling in a queuing system with asynchronously varying service rates", Probability in the Engineering and Informational Sciences **18**, 2, 191–217 (2004).

[Ariananda and Leus(2012)] Ariananda, D. D. and G. Leus, "Compressive wideband power spectrum estimation", IEEE Transactions on signal processing **60**, 9, 4775–4789 (2012).

[Arias-Castro and Eldar(2011)] Arias-Castro, E. and Y. C. Eldar, "Noise folding in compressed sensing", IEEE Signal Processing Letters **18**, 8, 478–481 (2011).

[Arrow *et al.*(1949)] Arrow, K. J., D. Blackwell and M. A. Girshick, "Bayes and minimax solutions of sequential decision problems", Econometrica **17**, 3/4, pp. 213–244, URL `http://www.jstor.org/stable/1905525` (1949).

[Ashkiani and Scaglione(2012)] Ashkiani, S. and A. Scaglione, "Discrete dithered desynchronization", arXiv preprint arXiv:1210.2122 (2012).

[Atia and Saligrama(2012)] Atia, G. K. and V. Saligrama, "Boolean compressed sensing and noisy group testing", IEEE Transactions on Information Theory **58**, 3, 1880–1901 (2012).

[Aysal *et al.*(2008)] Aysal, T., M. Coates and M. Rabbat, "Distributed average consensus with dithered quantization", IEEE Transactions on Signal Processing **56**, 10, 4905–4918 (2008).

[Bagheri and Scaglione(2015)] Bagheri, S. and A. Scaglione, "The restless multi-armed bandit formulation of the cognitive compressive sensing problem", Signal Processing, IEEE Transactions on **63**, 5, 1183–1198 (2015).

[Banirazi *et al.*(2014)] Banirazi, R., E. Jonckheere and B. Krishnamachari, “Heat-diffusion: Pareto optimal dynamic routing for time-varying wireless networks”, in “INFOCOM, 2014 Proceedings IEEE”, pp. 325–333 (IEEE, 2014).

[Barbarossa and Celano(2005)] Barbarossa, S. and F. Celano, “Self-organizing sensor networks designed as a population of mutually coupled oscillators”, in “Signal Processing Advances in Wireless Communications, 2005 IEEE 6th Workshop on”, pp. 475–479 (2005).

[Baron *et al.*(2010)] Baron, D., S. Sarvotham and R. G. Baraniuk, “Bayesian compressive sensing via belief propagation”, IEEE Transactions on Signal Processing **58**, 1, 269–280 (2010).

[Baum and Veeravalli(1994)] Baum, C. W. and V. V. Veeravalli, “A sequential procedure for multihypothesis testing”, IEEE Transactions on Information Theory **40**, 6, 1994–2007 (1994).

[Bessler(1960)] Bessler, S., *Theory and Applications of the Sequential Design of Experiments, K-actions and Infinitely Many Experiments: Part I-Theory* (Department of Statistics, Stanford University., 1960), URL `https://books.google.com/books?id=M4OUAAAAIAAJ`.

[Biermann *et al.*(2012)] Biermann, T., L. Scalia, C. Choi, H. Karl and W. Kellerer, “CoMP clustering and backhaul limitations in cooperative cellular mobile access networks”, Pervasive and Mobile Computing **8**, 5, 662–681 (2012).

[Birmiwal *et al.*(2012)] Birmiwal, S., J. Nair, D. Manjunath and R. R. Mazumdar, “Delay minimization in multihop wireless networks: Static scheduling does it”, in “Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt), 2012 10th International Symposium on”, pp. 97–102 (IEEE, 2012).

[Braun *et al.*(2015)] Braun, G., S. Pokutta and Y. Xie, “Info-greedy sequential adaptive compressed sensing”, IEEE Journal of Selected Topics in Signal Processing **9**, 4, 601–611 (2015).

[Buck and Jology(1988)] Buck, A. J. and O. B. Jology, “Synchronous rhythmic flashing in fireflies ii”, Q. Rev. Biol pp. 265–289 (1988).

[Bui *et al.*(2009)] Bui, L. X., S. Sanghavi and R. Srikant, “Distributed link scheduling with constant overhead”, IEEE/ACM Transactions on Networking **17**, 5, 1467–1480 (2009).

[Campbell *et al.*(1999)] Campbell, S. R., D. L. Wang and C. Jayaprakash, “Synchrony and desynchrony in integrate-and-fire oscillators”, Neural Computation **11**, 7, 1595–1619, URL `http://dx.doi.org/10.1162/089976699300016160` (1999).

[Caromi *et al.*(2013)] Caromi, R., Y. Xin and L. Lai, “Fast multiband spectrum scanning for cognitive radio systems”, Communications, IEEE Transactions on **61**, 1, 63–75 (2013).

[Chan *et al.*(2011)] Chan, C. L., P. H. Che, S. Jaggi and V. Saligrama, "Non-adaptive probabilistic group testing with noisy measurements: Near-optimal bounds with efficient algorithms", in "Communication, Control, and Computing (Allerton), 2011 49th Annual Allerton Conference on", pp. 1832–1839 (IEEE, 2011).

[Chang and Liu(2009)] Chang, N. B. and M. Liu, "Optimal channel probing and transmission scheduling for opportunistic spectrum access", IEEE/ACM Transactions on Networking **17**, 6, 1805–1818 (2009).

[Chen *et al.*(2010)] Chen, L., J. Jin and Y. Gu, "A calibration system and perturbation analysis for the modulated wideband converter", in "Signal Processing (ICSP), 2010 IEEE 10th International Conference on", pp. 78–81 (IEEE, 2010).

[Chernoff(1959)] Chernoff, H., "Sequential design of experiments", Ann. Math. Statist. **30**, 3, 755–770, URL `http://dx.doi.org/10.1214/aoms/1177706205` (1959).

[Chiang(2008)] Chiang, M., "Stochastic network utility maximization", European Transactions on Telecommunications **22**, 1–22 (2008).

[Chiang *et al.*(2007)] Chiang, M., S. H. Low, R. Calderbank and J. C. Doyle, "Layering as optimization decomposition", Proc. IEEE **95**, 1, 255–312 (2007).

[Choudhury and Gibson(2007)] Choudhury, S. and J. Gibson, "Information transmission over fading channels", in "Global Telecommunications Conference, 2007. GLOBECOM '07. IEEE", pp. 3316–3321 (2007).

[Cohen and Eldar(2014)] Cohen, D. and Y. C. Eldar, "Sub-nyquist sampling for power spectrum sensing in cognitive radios: A unified approach", IEEE Transactions on Signal Processing **62**, 15, 3897–3910 (2014).

[Cohen and Zhao(2015a)] Cohen, K. and Q. Zhao, "Active hypothesis testing for anomaly detection", IEEE Transactions on Information Theory **61**, 3, 1432–1450 (2015a).

[Cohen and Zhao(2015b)] Cohen, K. and Q. Zhao, "Asymptotically optimal anomaly detection via sequential testing", IEEE Transactions on Signal Processing **63**, 11, 2929–2941 (2015b).

[Cohen *et al.*(2014)] Cohen, K., Q. Zhao and A. Swami, "Optimal index policies for anomaly localization in resource-constrained cyber systems", Signal Processing, IEEE Transactions on **62**, 16, 4224–4236 (2014).

[Cui and Yeh(2014)] Cui, Y. and E. M. Yeh, "Delay optimal control and its connection to the dynamic backpressure algorithm", in "Information Theory (ISIT), 2014 IEEE International Symposium on", pp. 451–455 (IEEE, 2014).

[Cui *et al.*(2016)] Cui, Y., E. M. Yeh and R. Liu, "Enhancing the delay performance of dynamic backpressure algorithms", IEEE/ACM Transactions on Networking **24**, 2, 954–967 (2016).

[Degesys and Nagpal(2008)] Degesys, J. and R. Nagpal, "Towards desynchronization of multi-hop topologies", in "Self-Adaptive and Self-Organizing Systems, 2008. SASO'08. Second IEEE International Conference on", pp. 129–138 (IEEE, 2008).

[Degesys *et al.*(2007)] Degesys, J., I. Rose, A. Patel and R. Nagpal, "Desync: Self-organizing desynchronization and tdma on wireless sensor networks", in "International Conference on Information Processing in Sensor Networks (IPSN)", (2007).

[Dimakis *et al.*(2010)] Dimakis, A., S. Kar, J. Moura, M. Rabbat and A. Scaglione, "Gossip algorithms for distributed signal processing", Proceedings of the IEEE **98**, 11, 1847–1864 (2010).

[Elson *et al.*(2002)] Elson, J., L. Girod and D. Estrin, "Fine-grained network time synchronization using reference broadcasts", SIGOPS Oper. Syst. Rev. **36**, SI, 147–163 (2002).

[Ernst *et al.*(1995)] Ernst, U., K. Pawelzik and T. Geisel, "Synchronization induced by temporal delays in pulse-coupled oscillators", Phys. Rev. Lett. **74**, 1570–1573, URL `http://link.aps.org/doi/10.1103/PhysRevLett.74.1570` (1995).

[F.Nunez *et al.*(2015)] F.Nunez, Y. Wang and F. J. D. III, "Global synchronization of pulse-coupled oscillators interacting on cycle graphs", Automatica **52**, 0, 202 – 209, URL `http://www.sciencedirect.com/science/article/pii/S0005109814004920` (2015).

[Foundation(2007)] Foundation, H. C., "International standard iec 62591", (2007).

[Frazier and Yu(2008)] Frazier, P. I. and A. J. Yu, "Sequential hypothesis testing under stochastic deadlines", in "Adv. Neural Inf. Process Syst", pp. 465–472 (2008).

[Frigui and Rhouma(2000)] Frigui, H. and M. Rhouma, "Clustering based on synchronization of pulse-coupled oscillators", in "Fuzzy Information Processing Society, 2000. NAFIPS. 19th International Conference of the North American", pp. 128–132 (2000).

[Fudge *et al.*(2008)] Fudge, G. L., R. E. Bland, M. A. Chivers, S. Ravindran, J. Haupt and P. Pace, "A nyquist folding analog-to-information receiver", in "Signals, Systems and Computers, 2008 42nd Asilomar Conference on", pp. 541–545 (IEEE, 2008).

[Gai *et al.*(2010)] Gai, Y., B. Krishnamachari and R. Jain, "Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation", in "New Frontiers in Dynamic Spectrum, 2010 IEEE Symposium on", pp. 1–9 (IEEE, 2010).

[Ganesan *et al.*(2008)] Ganesan, R., A. Rao and T. Das, “A multiscale bayesian SPRT approach for online process monitoring”, Semiconductor Manufacturing, IEEE Transactions on **21**, 3, 399–412 (2008).

[Gentz(2017)] Gentz, R., *Wireless Sensor Data Transport, Aggregation and Security*, Ph.D. thesis, Arizona State University (2017).

[Gentz *et al.*(2016)] Gentz, R., P. Hong, L. Ferrari and A. Scaglione, “Pulsess: A pulse-coupled synchronization and scheduling protocol for clustered wireless sensor networks”, IEEE Internet of Things Journal (2016).

[Gentz *et al.*(2015)] Gentz, R., A. Scaglione, L. Ferrari and Y. W. P. Hong, “Pulsess: A microcontroller implementation of pulse-coupled scheduling and synchronization protocol for cluster-based wireless sensor networks”, in “2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)”, pp. 536–541 (2015).

[Georgiadis *et al.*(2006)] Georgiadis, L., M. J. Neely and L. Tassiulas, “Resource allocation and cross-layer control in wireless networks”, Found. Trends Netw. **1**, 1, 1–144, URL `http://dx.doi.org/10.1561/1300000001` (2006).

[Gupta *et al.*(2009)] Gupta, A., X. Lin and R. Srikant, “Low-complexity distributed scheduling algorithms for wireless networks”, IEEE/ACM Transactions on Networking (TON) **17**, 6, 1846–1859 (2009).

[Hao *et al.*(2012)] Hao, J., F. Tosato and R. J. Piechocki, “Sequential compressive sensing in wireless sensor networks”, in “Vehicular Technology Conference (VTC Spring), 2012 IEEE 75th”, pp. 1–5 (IEEE, 2012).

[Herman and Tixeuil(2004)] Herman, T. and S. Tixeuil, “A distributed tdma slot assignment algorithm for wireless sensor networks”, Algorithmic Aspects of Wireless Sensor Networks pp. 45–58 (2004).

[Hong and Scaglione(2005)] Hong, Y.-W. and A. Scaglione, “A scalable synchronization protocol for large scale sensor networks and its applications”, Selected Areas in Communications, IEEE Journal on **23**, 5, 1085–1099 (2005).

[Huang and Bensaou(2001)] Huang, X. L. and B. Bensaou, “On max-min fairness and scheduling in wireless ad-hoc networks: analytical framework and implementation”, in “Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing”, pp. 221–231 (ACM, 2001).

[Israeli *et al.*(2014)] Israeli, E., S. Tsiper, D. Cohen, E. Shoshan, R. Hilgendorf, A. Reysenson and Y. C. Eldar, “Hardware calibration of the modulated wideband converter”, in “Global Communications Conference (GLOBECOM), 2014 IEEE”, pp. 948–953 (IEEE, 2014).

[Izhikevich(1999)] Izhikevich, E., “Weakly pulse-coupled oscillators, fm interactions, synchronization, and oscillatory associative memory”, Neural Networks, IEEE Transactions on **10**, 3, 508–526 (1999).

[Izhikevich(2007)] Izhikevich, E. M., *Dynamical systems in neuroscience : the geometry of excitability and bursting*, Computational neuroscience (MIT Press, Cambridge, Mass., London, 2007), URL `http://opac.inria.fr/record=b1125242`.

[Ji *et al.*(2013)] Ji, B., C. Joo and N. Shroff, "Throughput-optimal scheduling in multihop wireless networks without per-flow information", IEEE/ACM Transactions On Networking **21**, 2, 634–647 (2013).

[Jiang and Walrand(2011)] Jiang, L. and J. Walrand, "Approaching throughput-optimality in distributed csma scheduling algorithms with collisions", IEEE/ACM Transactions On Networking **19**, 3, 816–829 (2011).

[Johansson *et al.*(2006)] Johansson, B., P. Soldati and M. Johansson, "Mathematical decomposition techniques for distributed cross-layer optimization of data networks", IEEE Journal on Selected Areas in Communications **24**, 8, 1535–1547 (2006).

[Kar *et al.*(2008)] Kar, K., X. Luo and S. Sarkar, "Throughput-optimal scheduling in multichannel access point networks under infrequent channel measurements", IEEE Transactions on Wireless Communications **7**, 7 (2008).

[Kar *et al.*(2012)] Kar, K., S. Sarkar, A. Ghavami and X. Luo, "Delay guarantees for throughput-optimal wireless link scheduling", IEEE Transactions on Automatic Control **57**, 11, 2906–2911 (2012).

[Kaspi *et al.*(2015)] Kaspi, Y., O. Shayevitz and T. Javidi, "Searching for multiple targets with measurement dependent noise", in "Information Theory (ISIT), 2015 IEEE International Symposium on", pp. 969–973 (IEEE, 2015).

[Kelly(2001)] Kelly, F., "Mathematical modelling of the internet", in "Mathematics unlimited2001 and beyond", pp. 685–702 (Springer, 2001).

[Kelly *et al.*(1998)] Kelly, F. P., A. K. Maulloo and D. K. H. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability", The Journal of the Operational Research Society **49**, 3, 237–252 (1998).

[Kim *et al.*(2008)] Kim, A., F. Hekland, S. Petersen and P. Doyle, "When hart goes wireless: Understanding and implementing the wirelesshart standard", IEEE Emerging Technologies and Factory Automation Conference (2008).

[Kim and Shin(2008)] Kim, H. and K. Shin, "Efficient discovery of spectrum opportunities with mac-layer sensing in cognitive radio networks", Mobile Computing, IEEE Transactions on **7**, 5, 533–545 (2008).

[Kim *et al.*(2007)] Kim, S., R. Fonseca, P. Dutta, A. Tavakoli, D. CUller, P. Levis, S. Shenker and I. Stoica, "Flush: A reliable bulk transport protocol for multihop wireless networks", Proceeding of SenSys (2007).

[Krishnamurthy *et al.*(2016)] Krishnamurthy, V., A. Aprem and S. Bhatt, “Opportunistic advertisement scheduling in live social media: A multiple stopping time POMDP approach”, CoRR **abs/1611.00291**, URL `http://arxiv.org/abs/1611.00291` (2016).

[Kuramoto(1991)] Kuramoto, Y., “Collective synchronization of pulse-coupled oscillators and excitable units”, Physica D: Nonlinear Phenomena **50**, 1, 15 – 30, URL `http://www.sciencedirect.com/science/article/pii/016727899190075K` (1991).

[Lai *et al.*(2011)] Lai, L., H. Poor, Y. Xin and G. Georgiadis, “Quickest search over multiple sequences”, Information Theory, IEEE Transactions on **57**, 8, 5375–5386 (2011).

[Lee *et al.*(2006)] Lee, J.-W., M. Chiang and R. A. Calderbank, “Jointly optimal congestion and contention control based on network utility maximization”, IEEE Communications Letters **10**, 3, 216–218 (2006).

[Lee and Chen(2008)] Lee, L. and C. Chen, “Synchronizing sensor networks with pulse coupled and cluster based approaches”, Information Technology Journal **7**, 5, 737–745 (2008).

[Lennvall *et al.*(2008)] Lennvall, T., S. Svensson and F. Hekland, “A comparison of wirelesshart and zigbee for industrial applications”, IEEE Workshop on Factory Communication Systems (2008).

[Letaief and Zhang(2009)] Letaief, K. and W. Zhang, “Cooperative communications for cognitive radio networks”, Proceedings of the IEEE **97**, 5, 878–893 (2009).

[Lexa *et al.*(2011)] Lexa, M. A., M. E. Davies and J. S. Thompson, “Compressive and noncompressive power spectral density estimation from periodic nonuniform samples”, arXiv preprint arXiv:1110.2722 (2011).

[Li(2004)] Li, Q., “Global clock synchronization in sensor networks”, in “IEEE Transactions on Computers”, pp. 214–226 (2004).

[Liang *et al.*(2008)] Liang, Y.-C., Y. Zeng, E. Peh and A. T. Hoang, “Sensing-throughput tradeoff for cognitive radio networks”, Wireless Communications, IEEE Transactions on **7**, 4, 1326–1337 (2008).

[Lin *et al.*(2006)] Lin, X., N. B. Shroff and R. Srikant, “A tutorial on cross-layer optimization in wireless networks”, IEEE Journal on Selected areas in Communications **24**, 8, 1452–1463 (2006).

[Lin *et al.*(2008)] Lin, X., N. B. Shroff and R. Srikant, “On the connection-level stability of congestion-controlled communication networks”, IEEE Transactions on Information Theory **54**, 5, 2317–2338 (2008).

[Liu *et al.*(2016a)] Liu, H., H. Zhang, J. Cheng and V. C. Leung, “Energy efficient power allocation and backhaul design in heterogeneous small cell networks”, in “Proc. IEEE ICC”, pp. 1–5 (2016a).

[Liu *et al.*(2016b)] Liu, T., K. Wang, C. Ku and Y. Hsu, “QoS-aware resource management for multimedia traffic report systems over LTE-A”, Computer Networks **94**, 375–389 (2016b).

[Lu and Do(2008)] Lu, Y. M. and M. N. Do, “A theory for sampling signals from a union of subspaces”, IEEE transactions on signal processing **56**, 6, 2334–2345 (2008).

[Lucarelli and Wang(2004)] Lucarelli, D. and I.-J. Wang, “Decentralized synchronization protocols with nearest neighbor communication”, in “Sensys”, (2004).

[Maleki and Leus(2013)] Maleki, S. and G. Leus, “Censored truncated sequential spectrum sensing for cognitive radio networks”, IEEE Journal on Selected Areas in Communications **31**, 3, 364–378 (2013).

[Malioutov *et al.*(2010)] Malioutov, D. M., S. R. Sanghavi and A. S. Willsky, “Sequential compressed sensing”, IEEE Journal of Selected Topics in Signal Processing **4**, 2, 435–444 (2010).

[Maróti *et al.*(2004)] Maróti, M., B. Kusy, G. Simon and A. Lédeczi, “The flooding time synchronization protocol”, in “Proceedings of the 2Nd International Conference on Embedded Networked Sensor Systems”, SenSys ’04, pp. 39–49 (2004).

[Mathar and Mattfeldt(1996)] Mathar, R. and J. Mattfeldt, “Pulse-coupled decentral synchronization”, SIAM Journal on Applied Mathematics **56**, 4, 1094–1106, URL `http://dx.doi.org/10.1137/S0036139994278135` (1996).

[Michelusi and Mitra(2015)] Michelusi, N. and U. Mitra, “Cross-layer estimation and control for cognitive radio: Exploiting sparse network dynamics”, IEEE Transactions on Cognitive Communications and Networking **1**, 1, 128–145 (2015).

[Mirollo and Strogatz(1990)] Mirollo, R. and S. Strogatz, “Synchronization of pulse-coupled biological oscillators”, SIAM J. Appl. Math. **50**, 6, 1645–1662 (1990).

[Mishali and Eldar(2009)] Mishali, M. and Y. C. Eldar, “Blind multiband signal reconstruction: Compressed sensing for analog signals”, IEEE Transactions on Signal Processing **57**, 3, 993–1009 (2009).

[Mishali and Eldar(2010)] Mishali, M. and Y. C. Eldar, “From theory to practice: Sub-nyquist sampling of sparse wideband analog signals”, IEEE Journal of Selected Topics in Signal Processing **4**, 2, 375–391 (2010).

[Mishali *et al.*(2011)] Mishali, M., Y. C. Eldar and A. J. Elron, “Xampling: Signal acquisition and processing in union of subspaces”, IEEE Transactions on Signal Processing **59**, 10, 4719–4734 (2011).

[Mock *et al.*(2000)] Mock, M., R. Frings, E. Nett and S. Trikaliotis, “Continuous clock synchronization in wireless real-time applications”, in “Reliable Distributed Systems, 2000. SRDS-2000. Proceedings The 19th IEEE Symposium on”, pp. 125–132 (2000).

[Motter *et al.*(2005)] Motter, A. E., C. Zhou and J. Kurths, "Network synchronization, diffusion, and the paradox of heterogeneity", Phys. Rev. E **71**, 016116, URL `http://link.aps.org/doi/10.1103/PhysRevE.71.016116` (2005).

[Mundhenk *et al.*(1997)] Mundhenk, M., J. Goldsmith and E. Allender, "The complexity of policy evaluation for finite-horizon partially-observable markov decision processes", in "Mathematical Foundations of Computer Science 1997, 22nd International Symposium, MFCS'97, Bratislava, Slovakia, August 25-29, 1997, Proceedings", pp. 129–138 (1997), URL `http://dx.doi.org/10.1007/BFb0029956`.

[Naghshvar and Javidi(2013a)] Naghshvar, M. and T. Javidi, "Active sequential hypothesis testing", Ann. Statist. **41**, 6, 2703–2738, URL `http://dx.doi.org/10.1214/13-AOS1144` (2013a).

[Naghshvar and Javidi(2013b)] Naghshvar, M. and T. Javidi, "Sequentiality and adaptivity gains in active hypothesis testing", IEEE Journal of Selected Topics in Signal Processing **7**, 5, 768–782 (2013b).

[Nakano and Saito(2002)] Nakano, H. and T. Saito, "Basic dynamics from a pulse-coupled network of autonomous integrate-and-fire chaotic circuits", Neural Networks, IEEE Transactions on **13**, 1, 92–100 (2002).

[Neely(2006)] Neely, M. J., "Energy optimal control for time-varying wireless networks", IEEE Trans. Inf. Theor. **52**, 7, 2915–2934, URL `http://dx.doi.org/10.1109/TIT.2006.876219` (2006).

[Neely(2013)] Neely, M. J., "Delay-based network utility maximization", IEEE/ACM Transactions on Networking **21**, 1, 41–54 (2013).

[Nitinawarat *et al.*(2013)] Nitinawarat, S., G. K. Atia and V. V. Veeravalli, "Controlled sensing for multihypothesis testing", IEEE Transactions on Automatic Control **58**, 10, 2451–2464 (2013).

[Niu *et al.*(2016)] Niu, B., Y. Zhou, H. Shah-Mansouri and V. W. S. Wong, "A dynamic resource sharing mechanism for cloud radio access networks", IEEE Transactions on Wireless Communications **15**, 12, 8325–8338 (2016).

[Nez *et al.*(2015)] Nez, F., Y. Wang and F. J. Doyle, "Synchronization of pulse-coupled oscillators on (strongly) connected graphs", IEEE Transactions on Automatic Control **60**, 6, 1710–1715 (2015).

[O'Keeffe *et al.*(2015)] O'Keeffe, K. P., P. L. Krapivsky and S. H. Strogatz, "Synchronization as Aggregation: Cluster Kinetics of Pulse-Coupled Oscillators", ArXiv e-prints (2015).

[Oksanen *et al.*(2010)] Oksanen, J., V. Koivunen, J. Lunden and A. Huttunen, "Diversity-based spectrum sensing policy for detecting primary signals over multiple frequency bands", in "Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on", pp. 3130–3133 (2010).

[Pagliari *et al.*(2010)] Pagliari, R., Y.-W. P. Hong and A. Scaglione, “Bio-inspired algorithms for decentralized round-robin and proportional fair scheduling”, IEEE Journal on Selected Areas in Communications, Special Issue on Bio-Inspired Networking **28**, 4 (2010).

[Pagliari and Scaglione(2011)] Pagliari, R. and A. Scaglione, “Scalable network synchronization with pulse-coupled oscillators”, IEEE Trans. Mobile Computing **10**, 3, 392–405 (2011).

[Palomar and Chiang(2006)] Palomar, D. P. and M. Chiang, “A tutorial on decomposition methods for network utility maximization”, IEEE Journal on Selected Areas in Communications **24**, 8, 1439–1451 (2006).

[Patel *et al.*(2007)] Patel, A., J. Degesys and R. Nagpal, “Desynchronization: The theory of self-organizing algorithms for round-robin scheduling”, in “IEEE Conference on Self-Adaptive and Self-Organizing Systems (SASO)”, (2007).

[Pei *et al.*(2011)] Pei, Y., Y. C. Liang, K. C. Teh and K. H. Li, “Energy-efficient design of sequential channel sensing in cognitive radio networks: Optimal sensing strategy, power allocation, and sensing order”, IEEE Journal on Selected Areas in Communications **29**, 8, 1648–1659 (2011).

[Peskin(1975)] Peskin, C. S., *Mathematical aspects of heart physiology* (Courant Institute of Mathematical Sciences, New York University, New York, NY, USA, 1975).

[Pham and Hwang(2017)] Pham, Q.-V. and W.-J. Hwang, “Network utility maximization-based congestion control over wireless networks: A survey and potential directives”, IEEE Communications Surveys & Tutorials **19**, 2, 1173–1200 (2017).

[Pham *et al.*(2015)] Pham, Q.-V., H.-L. To and W.-J. Hwang, “A multi-timescale cross-layer approach for wireless ad hoc networks”, Computer Networks **91**, 471–482 (2015).

[Pond and Li(1989)] Pond, L. C. and V. O. Li, “A distributed time-slot assignment protocol for mobile multi-hop broadcast packet radio networks”, in “Military Communications Conference, 1989. MILCOM’89. Conference Record. Bridging the Gap. Interoperability, Survivability, Security., 1989 IEEE”, pp. 70–74 (IEEE, 1989).

[Proskurnikov and Cao(2015)] Proskurnikov, A. V. and M. Cao, “Synchronization of pulse-coupled oscillators and clocks under minimal connectivity assumptions”, CoRR **abs/1510.02338**, URL `http://arxiv.org/abs/1510.02338` (2015).

[Quan *et al.*(2008)] Quan, Z., S. Cui and A. Sayed, “Optimal linear cooperation for spectrum sensing in cognitive radio networks”, Selected Topics in Signal Processing, IEEE Journal of **2**, 1, 28–40 (2008).

[Ramanathan(1997)] Ramanathan, S., "A unified framework and algorithm for (t/f/c)dma channel assignment in wireless networks", in "INFOCOM '97. Sixteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE", vol. 2, pp. 900–907 vol.2 (1997).

[Rappaport(2001)] Rappaport, T., *Wireless Communications: Principles and Practice* (Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001), 2nd edn.

[Rhee *et al.*(2006)] Rhee, I., A. Warrier, J. Min and L. Xu, "Drand: distributed randomized tdma scheduling for wireless ad-hoc networks", in "Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing", pp. 190–201 (ACM, 2006).

[Römer(2001)] Römer, K., "Time synchronization in ad hoc networks", in "Proceedings of the 2Nd ACM International Symposium on Mobile Ad Hoc Networking &Amp; Computing", MobiHoc '01, pp. 173–182 (2001).

[Rothkegel and Lehnertz(2014)] Rothkegel, A. and K. Lehnertz, "Synchronization in populations of sparsely connected pulse-coupled oscillators", EPL (Europhysics Letters) **105**, 3, 30003, URL `http://stacks.iop.org/0295-5075/105/i=3/a=30003` (2014).

[Sahu and Kar(2016)] Sahu, A. K. and S. Kar, "Distributed sequential detection for gaussian shift-in-mean hypothesis testing", IEEE Transactions on Signal Processing **64**, 1, 89–103 (2016).

[Samdanis *et al.*(2016)] Samdanis, K., R. Shrivastava, A. Prasad, D. Grace and X. Costa-Perez, "TD-LTE Virtual Cells: An SDN architecture for user-centric multi-eNB elastic resource management", Computer Commun. **83**, 1–15 (2016).

[Scarlett and Cevher(2016)] Scarlett, J. and V. Cevher, "Converse bounds for noisy group testing with arbitrary measurement matrices", in "Information Theory (ISIT), 2016 IEEE International Symposium on", pp. 2868–2872 (IEEE, 2016).

[Sharma and Murthy(2015)] Sharma, A. and C. R. Murthy, "On finding a subset of non-defective items from a large population using group tests: Recovery algorithms and bounds", in "Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on", pp. 4150–4154 (IEEE, 2015).

[Sichitiu and Veerarittiphan(2003)] Sichitiu, M. and C. Veerarittiphan, "Simple, accurate time synchronization for wireless sensor networks", in "Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE", vol. 2, pp. 1266–1273 vol.2 (2003).

[Siegmund(1985)] Siegmund, D., *Sequential analysis : tests and confidence intervals*, Springer series in statistics (Springer-Verlag, New York, 1985), URL `http://opac.inria.fr/record=b1092742`.

[Srikant(2004)] Srikant, R., "On the positive recurrence of a markov chain describing file arrivals and departures in a congestion-controlled network", in "IEEE Computer Communications Workshop", (2004).

[Su and Akyildiz(2005)] Su, W. and I. Akyildiz, “Time-diffusion synchronization protocol for wireless sensor networks”, Networking, IEEE/ACM Transactions on **13**, 2, 384–397 (2005).

[Taleb *et al.*(2015)] Taleb, T., Y. Hadjadj-Aoul and K. Samdanis, “Efficient solutions for enhancing data traffic management in 3GPP networks”, IEEE Systems Journal **9**, 2, 519–528 (2015).

[Tartakovsky *et al.*(2015)] Tartakovsky, A., I. Nikiforov and M. Basseville, *Sequential Analysis: Hypotheses Testing and Changepoint Detection*, vol. 136 of *Monographs on Statistics and Applied Probability* (CRC Press, 2015).

[Tassiulas and Ephremides(1992)] Tassiulas, L. and A. Ephremides, “Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks”, IEEE transactions on automatic control **37**, 12, 1936–1948 (1992).

[Tassiulas and Ephremides(1993)] Tassiulas, L. and A. Ephremides, “Dynamic server allocation to parallel queues with randomly varying connectivity”, IEEE Transactions on Information Theory **39**, 2, 466–478 (1993).

[Teng and Song(2017)] Teng, Y. and M. Song, “Cross-layer optimization and protocol analysis for cognitive ad hoc communications”, IEEE Access (2017).

[Thyagaturu *et al.*(2016)] Thyagaturu, A., Y. Dashti and M. Reisslein, “SDN based smart gateways (Sm-GWs) for multi-operator small cell network management”, IEEE Trans. Network and Service Management, in print (2016).

[Thyagaturu *et al.*(2017)] Thyagaturu, A., L. Ferrari, A. Scaglione and M. Reisslein, “Layback: Centralized management of distributed computing for wireless access services”, In Preparation (2017).

[Timme *et al.*(2004)] Timme, M., F. Wolf and T. Geisel, “Topological speed limits to network synchronization”, Phys. Rev. Lett. **92**, 074101, URL `http://link.aps.org/doi/10.1103/PhysRevLett.92.074101` (2004).

[Torikai and Saito(2004)] Torikai, H. and T. Saito, “Synchronization phenomena in pulse-coupled networks driven by spike-train inputs”, Neural Networks, IEEE Transactions on **15**, 2, 337–347 (2004).

[Tropp *et al.*(2010)] Tropp, J. A., J. N. Laska, M. F. Duarte, J. K. Romberg and R. G. Baraniuk, “Beyond nyquist: Efficient sampling of sparse bandlimited signals”, IEEE Transactions on Information Theory **56**, 1, 520–544 (2010).

[Tyrrell *et al.*(2008)] Tyrrell, A., G. Auer and C. Bettstetter, “On the accuracy of firefly synchronization with delays”, in “Applied Sciences on Biomedical and Communication Technologies, 2008. ISABEL ’08. First International Symposium on”, pp. 1–5 (2008).

[Unnikrishnan and Veeravalli(2010)] Unnikrishnan, J. and V. V. Veeravalli, "Algorithms for dynamic spectrum access with learning for cognitive radio", IEEE Transactions on Signal Processing **58**, 2, 750–760 (2010).

[Van Nguyen *et al.*(2013)] Van Nguyen, M., S. Lee and C. S. Hong, "Joint rate adaption, power control, and spectrum allocation in ofdma-based multi-hop crns", IEICE Transactions on Communications **96**, 1, 242–253 (2013).

[Veeravalli *et al.*(1993)] Veeravalli, V., T. Basar and H. Poor, "Decentralized sequential detection with a fusion center performing the sequential test", Information Theory, IEEE Transactions on **39**, 2, 433–442 (1993).

[Venkataramani and Bresler(2000)] Venkataramani, R. and Y. Bresler, "Perfect reconstruction formulas and bounds on aliasing error in sub-nyquist nonuniform sampling of multiband signals", IEEE Transactions on Information Theory **46**, 6, 2173–2183 (2000).

[Vreeswijk(1996)] Vreeswijk, C. v., "Partial synchronization in populations of pulse-coupled oscillators", Phys. Rev. E **54**, 5522–5537, URL `http://link.aps.org/doi/10.1103/PhysRevE.54.5522` (1996).

[Wald(1947)] Wald, A., *Sequential Analysis* (John Wiley and Sons, 1947), 1st edn., URL `http://books.google.com/books?id=oVYDHHzZtdIC&printsec=frontcover&dq=editions:oVYDHHzZtdIC&hl=en&ei=P5zFTYbWNdK1twe1sfCYBA&sa=X&oi=book_result&ct=book-thumbnail&resnum=1&ved=0CCwQ6wEwAA#v=onepage&q&f=false`.

[Wald and Wolfowitz(1948)] Wald, A. and J. Wolfowitz, "Optimum character of the sequential probability ratio test", Ann. Math. Statist. **19**, 3, 326–339, URL `http://dx.doi.org/10.1214/aoms/1177730197` (1948).

[Wang and Chen(2012)] Wang, K. and L. Chen, "On optimality of myopic policy for restless multi-armed bandit problem: An axiomatic approach", IEEE Transactions on Signal Processing **60**, 1, 300–309 (2012).

[Wang *et al.*(2012a)] Wang, S., Y. Wang, J. P. Coon and A. Doufexi, "Energy-efficient spectrum sensing and access for cognitive radio networks", IEEE Transactions on Vehicular Technology **61**, 2, 906–912 (2012a).

[Wang and Apsel(2007a)] Wang, X. and A. Apsel, "Pulse coupled oscillator synchronization for communications in uwb wireless transceivers", in "MWSCAS", (2007a).

[Wang and Apsel(2007b)] Wang, X. Y. and A. B. Apsel, "Pulse coupled oscillator synchronization for low power uwb wireless transceivers", in "2007 50th Midwest Symposium on Circuits and Systems", pp. 1524–1527 (2007b).

[Wang *et al.*(2011)] Wang, X. Y., R. K. Dokania and A. Apsel, "Pco-based synchronization for cognitive duty-cycled impulse radio sensor networks", IEEE Sensors Journal **11**, 3, 555–564 (2011).

[Wang *et al.*(2009)] Wang, X. Y., R. K. Dokania and A. B. Apsel, “Implementation of a global clocking scheme for ulp radio networks”, in “2009 IEEE International Symposium on Circuits and Systems”, pp. 912–915 (2009).

[Wang *et al.*(2012b)] Wang, Y., F. Nunez and F. J. Doyle, “Energy-efficient pulse-coupled synchronization strategy design for wireless sensor networks through reduced idle listening”, IEEE Transactions on Signal Processing **60**, 10, 5293–5306 (2012b).

[Wannamaker *et al.*(2000)] Wannamaker, R., S. Lipshitz, J. Vanderkooy and J. Wright, “A theory of nonsubtractive dither”, Signal Processing, IEEE Transactions on **48**, 2, 499–516 (2000).

[Werner-Allen *et al.*(2005)] Werner-Allen, G., G. Tewari, A. Patel, M. Welsh and R. Nagpal, “Firefly-inspired sensor network synchronicity with realistic radio effects”, in “Proceedings of the 3rd international conference on Embedded networked sensor systems”, pp. 142–153 (ACM, 2005).

[Woodroofe(1976)] Woodroofe, M., “Frequentist properties of bayesian sequential tests”, Biometrika **63**, 1, 101–110, URL `http://biomet.oxfordjournals.org/content/63/1/101.abstract` (1976).

[Wu *et al.*(2014)] Wu, X., J. L. Xu, M. Chen and J. Wang, “Work in progress paper: Energy-efficient sensing in cognitive radio networks”, in “Communications and Networking in China (CHINACOM), 2014 9th International Conference on”, pp. 564–567 (2014).

[Young(1996)] Young, C. D., “Usap: a unifying dynamic distributed multichannel tdma slot assignment protocol”, in “Military Communications Conference, 1996. MILCOM’96, Conference Proceedings, IEEE”, vol. 1, pp. 235–239 (IEEE, 1996).

[Yucek and Arslan(2009)] Yucek, T. and H. Arslan, “A survey of spectrum sensing algorithms for cognitive radio applications”, Communications Surveys Tutorials, IEEE **11**, 1, 116–130 (2009).

[Zeinalkhani and Banihashemi(2012)] Zeinalkhani, Z. and A. H. Banihashemi, “Iterative recovery algorithms for compressed sensing of wideband block sparse spectrums”, in “Communications (ICC), 2012 IEEE International Conference on”, pp. 1630–1634 (IEEE, 2012).

[Zeng *et al.*(2011)] Zeng, F., C. Li and Z. Tian, “Distributed compressive spectrum sensing in cooperative multihop cognitive networks”, IEEE Journal of Selected Topics in Signal Processing **5**, 1, 37–48 (2011).

[Zeng and Liang(2009)] Zeng, Y. and Y.-C. Liang, “Eigenvalue-based spectrum sensing algorithms for cognitive radio”, Communications, IEEE Transactions on **57**, 6, 1784–1793 (2009).

[Zhao *et al.*(2015)] Zhao, J., X. Wang and Q. Liu, “Cooperative sequential compressed spectrum sensing over wide spectrum band”, in “Sensing, Communication, and Networking (SECON), 2015 12th Annual IEEE International Conference on”, pp. 1–9 (IEEE, 2015).

[Zhao *et al.*(2008)] Zhao, Q., B. Krishnamachari and K. Liu, “On myopic sensing for multi-channel opportunistic access: structure, optimality, and performance”, IEEE Transactions on Wireless Communications **7**, 12 (2008).

[Zhao *et al.*(2007)] Zhao, Q., L. Tong, A. Swami and Y. Chen, “Decentralized cognitive mac for opportunistic spectrum access in ad hoc networks: A POMDP framework”, Selected Areas in Communications, IEEE Journal on **25**, 3, 589–600 (2007).

[Zhao and Ye(2010)] Zhao, Q. and J. Ye, “Quickest detection in multiple on-off processes”, IEEE Transactions on Signal Processing **58**, 12, 5994–6006 (2010).

[Zheng *et al.*(2013)] Zheng, L., C. W. Tan and it, “Cognitive radio network duality and algorithms for utility maximization”, IEEE Journal on Selected Areas in Communications **31**, 3, 500–513 (2013).

[Zheng *et al.*(2016)] Zheng, M., W. Liang, H. Yu and H. Sharif, “Utility-based opportunistic spectrum access for cognitive radio sensor networks: joint spectrum sensing and random access control”, IET Communications **10**, 9, 1044–1052 (2016).

APPENDIX A

PROOFS FOR CHAPTER 2

Proof of Proposition 2.4.2

We first show $\mathbf{\Delta}(t^*) = 0 \Rightarrow \forall t > t^* \;\; \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*)$ which indicates the synchronous state is a fixed point. The argument is straightforward, since all the firings events are received instantaneously by nodes that are also firing, i.e. their phase is equal to $0 \mod 1$ and it is immediate to see no node will change its phase, i.e. $\min\{(1+\alpha)0 \mod 1, 1\} = 0 \mod 1$ therefore no change in $\mathbf{\Delta}$ will occur from this point on. Then we prove $\forall t > t^* \;\; \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*) \Rightarrow \mathbf{\Delta}(t^*) = 0$ by contradiction. Let us assume $\Delta_{uv}(t^*) \neq 0$ for some uv. Without loss of generality we can assume $e_{uv} = 1$, in fact if $\Delta_{uv} \neq 0$ and $e_{uv} \neq 0$ then, by looking at the edges $k\ell$ over the path $\mathcal{P}_{uv}$ we must find a nonzero phase difference $\Delta_{k\ell} \neq 0$, $e_{k\ell} = 1$. Then, without loss of generality let us consider the firing of node $v$heard by node u. Node u will update its phase and Δ_{uv} will either increase or decrease, unless the phase of node u is 0 when node $v$is firing, which would contradict the hypothesis that $\Delta_{uv} \neq 0$. We can consider the isolated event "node j hears node v firing and updates its phase", since each other event that occurs simultaneously will not change the phase of node v(which is equal to $0 \mod 1$) and will only potentially move even forward node u, further increasing or decreasing Δ_{uv}.

Proof of Proposition 2.4.3

In order to prove the almost sure convergence of a $|\mathcal{V}| = 3$ nodes network to $\mathbf{\Delta} = \mathbf{0}$, let us label the center node as 1. It is easier to focus on the evolution of the variables

$$\mathbf{\Xi} \triangleq \{\Xi_{12}, \Xi_{13}\}, \quad \Xi_{uv} = (\Phi_v - \Phi_u) \mod 1 \tag{A.0.1}$$

and, since by definition (2.4.4) we have $\Delta_{uv} = \min\{\Xi_{uv}, \Xi_{ji}\}$, if we have convergence to $\mathbf{0}$ (synchronization) for $\mathbf{\Xi}$ we also have it for $\mathbf{\Delta}$. We focus on the line network with nodes $\{1, 2, 3\}$ and edges $\{(1, 2), (1, 3)\}$ since the case of a fully connected network is covered by [Mirollo and Strogatz(1990)]. Since the evolution of $\mathbf{\Xi}(t)$ occurs in jumps that are triggered by the firing events, we can define $\mathbf{\Xi}[k] \triangleq \mathbf{\Xi}(t_{f[k]})$, where $f[k]$ is the index of the node that is generating the k-th firing and $t_{f[k]}$ is the time for which the k-th firing occurs and focus on the evolution of $\mathbf{\Xi}[k], \mathbf{\Xi}[k+1], \ldots$.

We highlight that for the case of 3 nodes (or any tree network for that matter) it is possible to consider each firing event separately, since the only case for which a node is affected by two simultaneous firing events is when node 1 can hear node 2 and node 3. To handle this case we first introduce

Lemma A.0.1. *Suppose $\Phi_2(t) = \Phi_3(t)$ and, equivalently, $\Xi_{12}(t) = \Xi_{13}(t)$. Then, $\Xi_{12}(t') = \Xi_{13}(t')$, for all $t' \geq t$, regardless of the sequence of firing events occurring after t.*

Proof. The statement follows from the fact that at the firing events of node 1, the two nodes update simultaneously and change their phase by the same amount, while when one of the two fires, the other one is also at the firing point and they do not affect each other. ■

Lemma A.0.1 also implies that when node 2 and 3 are synchronized, we have equivalently a fully connected two-node network whose convergence occurs for all initial values of the nodes' phases, except for a set of measure zero, as shown in [Mirollo and Strogatz(1990)]. Hence, the network converges almost surely to the fixed point $\mathbf{\Delta} = \mathbf{0}$ in this case. We can then proceed in our proof treating each firing event separately, and in particular consider two cases: the case where the firing order is maintained and the case where overtaking of the firing order may occur among nodes. We initially assume all nodes have different phases. Case 1: Suppose that the firing order does not change after some k_0 (the k_0-th firing event) and that the nodes are labeled in the order of their firing after this point, with node 1 being the node firing in the middle. That is, suppose that, for some k_0, $\Xi_{12}[k] < \Xi_{13}[k]$, for all $k \geq k_0$. If $f[k] = 1$ and no absorption occurs then, we have $\Xi_{13}[k+1] - \Xi_{12}[k+1] = (1+\alpha)(\Xi_{13}[k] - \Xi_{12}[k]) > \Xi_{13}[k] - \Xi_{12}[k]$, that is, the phase difference between nodes 3 and 2 increases. If $f[k] = 2$ or 3 and still no absorption occurs we can see that the phase difference between nodes 3 and 2 remains the same (i.e., $\Xi_{13}[k+1] - \Xi_{12}[k+1] = \Xi_{13}[k] - \Xi_{12}[k]$). This implies that, if no absorption occurred after each node has fired, the phase difference between nodes 3 and 2 must strictly increase, that is, $0 < \Xi_{13}[k] - \Xi_{12}[k] < \Xi_{13}[k+3] - \Xi_{12}[k+3]$. Since $\Xi_{13}[k'] - \Xi_{12}[k'] < 1$, for all k', an absorption must eventually occur in one of the cycles. If absorption occurs between nodes 2 and 3, then we are done since, by Lemma A.0.1, we have $\Xi_{12}[k] = \Xi_{13}[k]$ from that point on. If node 3 absorbs node 1, then the next one to fire is node 2 and that will trigger node 1 to overtake the position of node 3, which violates our assumption in Case 1. Finally, if node 1 absorbs only node 2 at some firing event k'' (that is, if $\Xi_{12}[k''] = 0$ and $\Xi_{13}[k''] > 0$), then the firing of node 3, which comes immediately after will cause either an absorption of both nodes 1 and 2 (which occurs if $\Xi_{13}[k''] \in [\frac{1}{1+\alpha}, 1)$) or an update of $\Xi_{12}[k''+1] = \alpha \Xi_{13}[k'']$, if $\Xi_{13}[k''] \in (0, \frac{1}{1+\alpha})$. In the former case, the nodes become synchronized and we are done; in the latter case, it must be true that $\Xi_{12}[k''+1] < \frac{\alpha}{1+\alpha}$ and, thus, node 1 will again absorb node 2 when it fires. Therefore, even though the phases of nodes 2 and 1 may temporarily deviate from each other, they will always become absorbed again once node 1 fires. In this case, we again have an equivalent two-node fully connected network (formed by node 3 and the combination of nodes 1 and 2) and, thus, the convergence to the fixed point $\mathbf{\Delta} = \mathbf{0}$ again follows from [Mirollo and Strogatz(1990)].

<u>Case 2:</u> Suppose that a node may overtake the position of another node in the firing order. This can occur only when the firing of node 2 triggers node 1 to increase its phase beyond the phase of node 3 (causing node 1 to fire again before node 3 fires). This is because, when node 3 fires (i.e., $f[k] = 3$), either node 1 (and, maybe also node 2) is absorbed (in which case the firing order is considered to be maintained) or no node is absorbed and the following update occurs: $\Xi_{13}[k+1] = (1+\alpha)\Xi_{13}[k] > \Xi_{12}[k] + \alpha\Xi_{13}[k] = \Xi_{12}[k+1]$, which also implies that the firing order remains the same; and, when node 1 fires, we again have $\Xi_{13}[k+1] \geq \Xi_{12}[k+1]$ and, thus, the firing order is again maintained. Now, suppose that the firing of node 2 causes node 1 to overtake the position of node 3 (that is, $\Xi_{13}[k] \geq \Xi_{12}[k]$ but $\Xi_{13}[k+1] < \Xi_{12}[k+1]$). This can occur only when $\Xi_{12}[k] \in (0, \frac{1}{1+\alpha})$; otherwise, the firing of node 2 would have absorbed both nodes 1 and 3 (in which case the convergence to the fixed point is immediately achieved). However, if overtaking occurs due to the firing of node 2, then this means that $\Xi_{13}[k+1] = \Xi_{13}[k] + \alpha\Xi_{12}[k] \mod 1 < \Xi_{12}[k+1] = (1+\alpha)\Xi_{12}[k]$. Since $\Xi_{13}[k] \geq \Xi_{12}[k]$, this is possible only when $\Xi_{13}[k] + \alpha\Xi_{12}[k] \geq 1$. Therefore, $\Xi_{13}[k+1] = \Xi_{13}[k] + \alpha\Xi_{12}[k] \mod 1 = \Xi_{13}[k] + \alpha\Xi_{12}[k] - 1 < \frac{\alpha}{1+\alpha}$ since $\Xi_{13}[k] < 1$ and $\Xi_{12}[k] \in (0, \frac{1}{1+\alpha})$. In this case, the firing of node 1, which comes immediately after, will again absorb node 3, causing the two nodes to be absorbed from that point on. Even though the firing of node 2 may trigger node 1 to temporarily overtake node 3, they will always become absorbed again once node 1 fires. Consequently, we again have an equivalent two-node fully connected network (formed by node 2 and

the combination of nodes 1 and 3); convergence to the fixed point $\mathbf{\Delta} = \mathbf{0}$ follows from [Mirollo and Strogatz(1990)]. In light of this extensive treatment, we can then consider the case where a node is added to a synchronized network. At the time this additional node fires, the nodes connected to it will jump in front of the others but, since no additional firing events will occur before the neighbors of the new node node fire, all the absorption with the other nodes will be restored at the end of the firing round and so we have again an equivalent two-node fully connected network (formed by the added node and the previous synchronized network) and convergence to synchronization will occur almost surely as previously discussed.

Proof of Proposition 2.4.4

We first prove $\forall t > t^*, \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*) \Rightarrow \mathbf{\Delta}(t^*) \in \mathcal{F}$ by contradiction, i.e. by showing that any point $\mathbf{\Delta}$ for which $\Delta_{uv} > \tau_{uv}$ for some uv s.t. $e_{uv} = 1$, cannot be a fixed point. We can initially assume, without loss of generality that $(\Phi_v - \Phi_u \mod 1) < (\Phi_u - \Phi_1 \mod 1)$ and look at what happens when node $v$fires. At the time node $v$fires, say $t_v > t^*$, since $\Delta_{uv}(t_v) > \tau_{uv}$ and $(\Phi_v - \Phi_u \mod 1) < (\Phi_u - \Phi_1 \mod 1)$ we have that $\frac{1}{2} < \Phi_u < 1 - \tau_{uv}$. Let us initially assume nothing happens (no firing events) before node j hears the firing of node $v$at time $t_v + \tau_{uv}$. The phase of node u at time $t_v + \tau_{uv}$ will be $\frac{1}{2} + \tau_{uv} < \Phi_u(t_v + \tau_{uv}) < 1$ and therefore outside of the refractory period, as long as $\rho < \frac{1}{2} + \min_{uv} \tau_{uv}$. This implies that node j will jump forward and the difference Δ_{uv} will decrease, which means it is not a fixed point. If some firing events happen before the update, i.e at time t where $t_v < t < t_v + \tau_{uv}$ then node $v$will not move since its phase will be $\Phi_v(t) < \tau_{uv}, \forall t_v < t < t_v + \tau_{uv}$ and therefore it is inside the refractory period $\rho > 2 \max_{uv} \tau_{uv}$. Node u instead, can either move even forward by hearing these additional events (and Δ_{uv} would further decrease, therefore keeping our argument by contradiction valid) or not move (and therefore there is no problem in neglecting such events), and this concludes our proof. The key aspect of the proof is that nodes can only jump forward after receiving a firing event. We then prove $\mathbf{\Delta}(t^*) \in \mathcal{F} \Rightarrow \forall t > t^*, \mathbf{\Delta}(t) = \mathbf{\Delta}(t^*)$ which indicates that all the points in $\mathcal{F}$ are fixed points. Let us consider a general node v and show it will note update its phase. In order to update its phase, node v needs to receive one or more firing events by its neighbours when its phase is outside the refractory period. Since $\Delta_{uv} \leq \tau_{uv}$ $\forall u, e_{uv} = 1$ there are then 2 cases.

Case 1: A set U of node v's neighbours fire at time t' when $0 \leq \Phi_i(t') \leq \min_{u \in U} \tau_{uv}$. All these firings will be heard by node $v$when $\min_{u \in U} \tau_{uv} \leq \Phi_v \leq \min_{u \in U} \tau_{uv} +$

$\max_{u \in U} \tau_{uv} \leq 2 \max_{uv} \tau_{uv} \leq \rho$, therefore no update will occur.

Case 2: A set U of node v's neighbours fire at time t' when $1 - \min_{u \in U} \tau_{uv} \leq \Phi_i(t') \leq 1$. All these firings will be heard when $(1 - \min_{u \in U} \tau_{uv} + \min_{u \in U} \tau_{uv}) \mod 1 \leq \Phi_v \leq \max_{u \in U} \tau_{uv} \rightarrow \ 0 \leq \Phi_v \leq \max_{u \in U} \tau_{uv} < \rho$, therefore no update will occur. If no node updates its phase, then $\mathbf{\Delta}(t)$ remains constant over time and this proves that all $\mathbf{\Delta} \in \mathcal{F}$ are fixed points.

Proof of Proposition 2.5.2

We know $\mathbf{M}^c$ is a stochastic matrix and then all of its eigenvalues are inside the unit circle except for one. After a few tedious but straightforward manipulation the $2n$-degree characteristic equation of $\mathbf{M}^c$ is ($n = |\mathcal{V}_c|$):

$$\begin{aligned} &\lambda^{2n} - \lambda^n - (\beta - 1)^2(\lambda^n - 1) + \beta^2\mu(2\lambda^n - \lambda^{n-1} - \lambda) \\ &- \beta\mu(\lambda^{2n-1} + \lambda^{n+1} - \lambda^{n-1} - \lambda) = 0. \end{aligned} \tag{A.0.2}$$

This specific form highlights that $\lambda = 1$ is solution and, also, that for $\mu = 0$ $\lambda^n = 1$. For $\mu > 0$ small, the $n-1$ roots of (A.0.2) inside the unit circle can be approximated as:

$$\lambda(k) = (1 - \varepsilon)e^{j(\frac{2\pi k}{n} - \vartheta)} \tag{A.0.3}$$

with $k = 1, 2, ..., n-1$ and here $j = \sqrt{-1}$ indicates the imaginary unit. We are interested in the second largest eigenvalue of the system, thus in the highest of these $n-1$ perturbed roots. For μ small enough we assume that $\varepsilon, \vartheta \ll 1$. Then it is possible to substitute (A.0.3) in (A.0.2) and consider:

$$\left((1 - \varepsilon)e^{-j\vartheta}\right)^n \approx (1 - n\varepsilon - nj\vartheta)$$

We are then able to solve (A.0.2) as a first order equation in $z = \varepsilon + j\vartheta$ and then consider the real part of the solution (ε) and the imaginary part (ϑ) to find the eigenvalues from (A.0.3). The solutions will have the following form:

$$z^*(k) = \frac{1}{n}\frac{\left(1 - \cos\frac{2\pi k}{n}\right)}{1 - \frac{1}{2}\exp\{-j\frac{2\pi k}{n}\} - \frac{1}{n}\sin\frac{2\pi k}{n} + \frac{1}{\beta\mu}(1 - \frac{\beta}{2} - \mu\cos\frac{2\pi k}{n})} \tag{A.0.4}$$

It is possible to show that the real part of $z^*(k)$ is concave down with respect to k, and thus from (A.0.3) the second largest eigenvalue is reached for $k = 1$ or $k = n-1$. A simple comparison leads to choose $k = n-1$ irrespective of the value for β and μ. At this point for n large enough, we can use the Taylor series for the trigonometric functions in (A.0.4) and approximate $|\lambda(n-1)| \approx 1 - \frac{2\beta\mu\pi^2}{n^3}$. Thus to find the second highest eigenvalue of $\mathbf{M}^c$ we take:

$$|\lambda_2^c| \approx \left(1 - \frac{2\beta\mu\pi^2}{|\mathcal{V}_c|^3}\right)^{|\mathcal{V}_c|} \approx 1 - \frac{2\beta\mu\pi^2}{|\mathcal{V}_c|^2} \qquad \text{(A.0.5)}$$

Proof of Theorem 2.5.6

One Shared Node

We will study the evolution of $\boldsymbol{\Upsilon}_1(t), \boldsymbol{\Upsilon}_2(t)$, as defined in (2.5.6). We will show that there is a unique fixed point irrespective of the initial configuration. If we have only one node shared between the two cliques we can define, without loss of generality, the two system vectors such that $\pi^1_{|\mathcal{V}_1|} = \pi^2_{|\mathcal{V}_2|} = v'$ with v' being the shared node. Except for the update of node v', the updates of the other nodes impact variables only in their clique, i.e. in only one of the system vectors. We can then consider for $c = 1, 2$ the following matrix

$$\tilde{\mathbf{M}}^c \triangleq \prod_{k=1}^{|\mathcal{V}_c|-1} \boldsymbol{M}_{\pi^c_k} = \begin{bmatrix} \tilde{\mathbf{M}}^c_{(2|\mathcal{V}_c|-1)\times(2|\mathcal{V}_c|-1)} & \mathbf{0} \\ 0 \ \cdots \ 0 & 1 \end{bmatrix} \tag{A.0.6}$$

Note that each $\boldsymbol{M}_{\pi^c_k}$ is a left stochastic matrix and so is the product $\tilde{\mathbf{M}}^c$. Because of the structure of $\tilde{\mathbf{M}}^c$, $\tilde{\mathbf{M}}^c_{(2|\mathcal{V}_c|-1)\times(2|\mathcal{V}_c|-1)}$ is also left stochastic and it is primitive because it contains all positive elements. Thus the Perron-Frobenius Theorem ensures that $\tilde{\mathbf{M}}^c_{(2|\mathcal{V}_c|-1)\times(2|\mathcal{V}_c|-1)}$ has exactly one eigenvalue equal to 1 and $2|\mathcal{V}_c| - 2$ eigenvalues inside the unit circle. Arguing as in [Pagliari *et al.*(2010)] we can show that the matrix $\tilde{\mathbf{M}}^c$, has the following two eigenvectors for the eigenvalue equal to one:

$$\tilde{\boldsymbol{\Upsilon}}^\star_{c,1} = \frac{\tilde{\gamma}^c}{\tilde{\boldsymbol{D}}^c}(\delta, D_{\pi^c_1}, \delta, D_{\pi^c_2}, \ldots, \delta, D_{\pi^c_{|\mathcal{V}_c|-1}}, \delta, 0)^T \tag{A.0.7}$$

$$\tilde{\boldsymbol{\Upsilon}}^\star_{c,2} = \quad (0, 0, 0, 0, \ldots, 0, 0, 0, 1)^T \tag{A.0.8}$$

with $\tilde{\boldsymbol{D}}^c = \sum_{k=1}^{|\mathcal{V}_c|-1} D_{\pi^c_k}$ and $\tilde{\gamma}^c = \dfrac{\tilde{\boldsymbol{D}}^c}{\tilde{\boldsymbol{D}}^c + |\mathcal{V}_c|\delta}$.

The complexity introduced by the presence of the shared node $v$is that the two nodes Pre(v, t), Suc(v, t) defined in (2.5.9)-(2.5.10) might change over time and belong

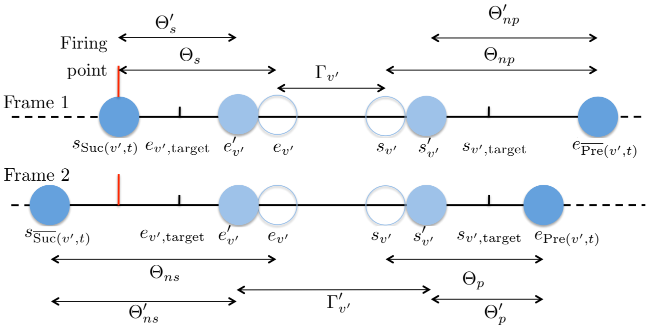


Figure A.0.1: Update of the shared nodes and representation of the variables $\Theta_s, \Gamma, \Theta_p, \Theta_{ns}, \Theta_{np}$.

to different cliques. We introduce the following two sets

$$\overline{\mathrm{Pre}}(v,t) \triangleq \bigcup_{c \in \mathcal{C}_v} \mathrm{pre}(v,c) \setminus \mathrm{Pre}(v,t) \tag{A.0.9}$$

$$\overline{\mathrm{Suc}}(v,t) \triangleq \bigcup_{c \in \mathcal{C}_v} \mathrm{suc}(v,c) \setminus \mathrm{Suc}(v,t) \tag{A.0.10}$$

In the case of two cliques, the sets $\overline{\mathrm{Pre}}(v',t), \overline{\mathrm{Suc}}(v',t)$ contain only one node and according to our notation, if for example, $\mathrm{Pre}(v',t) = \pi^1_{|\mathcal{V}_1|-1}$ then $\overline{\mathrm{Pre}}(v',t) = \{\pi^2_{|\mathcal{V}_2|-1}\}$ and so on. The update of node v' might cause change in 5 variables (see Fig.A.0.1) that for short notation we indicate with

$$\Theta_s(t) \triangleq \frac{e_{v'}(t) - s_{\mathrm{Suc}(v')}(t) \pmod{L}}{L} \tag{A.0.11}$$

$$\Gamma_{v'}(t) \triangleq \frac{s_{v'}(t) - e_{v'}(t) \pmod{L}}{L} \tag{A.0.12}$$

$$\Theta_p(t) \triangleq \frac{e_{\mathrm{Pre}(v')}(t) - s_{v'}(t) \pmod{L}}{L} \tag{A.0.13}$$

$$\Theta_{ns}(t) \triangleq \frac{e_{v'}(t) - s_{\overline{\mathrm{Suc}}(v')}(t) \pmod{L}}{L} \tag{A.0.14}$$

$$\Theta_{np}(t) \triangleq \frac{e_{\overline{\mathrm{Pre}}(v')}(t) - s_{v'}(t) \pmod{L}}{L} \tag{A.0.15}$$

Ignoring the dependence on time and using the apex to indicate the updated quantities we can write

$$\left[\Theta'_s, \Gamma', \Theta'_p, \Theta'_{ns}, \Theta'_{np}\right]^T = \hat{\boldsymbol{M}}_{v'} \left[\Theta_s, \Gamma, \Theta_p, \Theta_{ns}, \Theta_{np}\right]^T \tag{A.0.16}$$

where

$$\hat{\boldsymbol{M}}_{v'} = \begin{bmatrix} & & & 0 & 0 \\ & \mathbf{U}_{v'} & & 0 & 0 \\ & & & 0 & 0 \\ -\beta\frac{D+\delta}{D+2\delta} & \beta\frac{\delta}{D+2\delta} & \beta\frac{\delta}{D+2\delta} & 1 & 0 \\ \beta\frac{\delta}{D+2\delta} & \beta\frac{\delta}{D+2\delta} & -\beta\frac{D+\delta}{D+2\delta} & 0 & 1 \end{bmatrix} \tag{A.0.17}$$

where the first three rows are given by the submatrix $\mathbf{U}_{v'}$ defined in (2.5.20) and the last two rows are obtained from

$$\Theta'_{ns} = \Theta_{ns} - \Theta_s + \Theta'_s \tag{A.0.18}$$

$$\Theta'_{np} = \Theta_{np} - \Theta_p + \Theta'_p \tag{A.0.19}$$

From the structure of the matrix $\hat{\mathbf{M}}_{v'}$, we can conclude it has 3 eigenvalues equal to 1 and 2 strictly smaller than 1. In fact, the two right columns are two eigenvectors with eigenvalue 1 and the top-right block $\mathbf{U}_{v'}$ is positive stochastic, therefore gives a unique eigenvalue equal to 1 and others 2 smaller than 1. It is then possible to verify the third eigenvector associated with eigenvalue 1 has to satisfy the following constraint

$$\Theta^*_s = \Theta^*_p = \frac{\delta}{D_{v'}}\Gamma^*_{v'} \tag{A.0.20}$$

The fixed point for $\boldsymbol{\Upsilon}_1(t), \boldsymbol{\Upsilon}_2(t)$ needs to be such with respect to both the linear map $\boldsymbol{\Upsilon}'_c = \tilde{\mathbf{M}}_c\boldsymbol{\Upsilon}_c$ (i.e the updates of all the local nodes) and the update of the shared node just described. To be eigenvectors in (A.0.7)-(A.0.8) and meet the constraint $||\boldsymbol{\Upsilon}_c||_1 = 1$ we have

$$\boldsymbol{\Upsilon}^\star_c = (1-\lambda_c)\tilde{\boldsymbol{\Upsilon}}^\star_{c,1} + \lambda_c\tilde{\boldsymbol{\Upsilon}}^\star_{c,2} \quad \text{for} \quad c = 1, 2 \tag{A.0.21}$$

with $\lambda_c \in (0,1)$. However, since $\Gamma_{\pi^1_{|\mathcal{V}_1|}} = \Gamma_{\pi^2_{|\mathcal{V}_2|}} = \Gamma_v$ we have $\lambda_1 = \lambda_2 = \lambda$ and it is possible to verify there is only one value for λ that satisfies the constraint in (A.0.20), which correspond to the update of the shared node, according to the definition of $\Theta_s(t)$ and $\Theta_p(t)$ in (A.0.11)-(A.0.13). This value is given by $\lambda = \min_{c=1,2}\left\{\frac{\gamma^c}{\boldsymbol{D}^c}\right\} D_v$ which also tells us that both $\mathrm{Pre}(v', t)$ and $\mathrm{Suc}(v', t)$ at the steady state belong to the densest cluster, i.e. the cluster $c' = \arg\min_{c=1,2}\left\{\frac{\gamma^c}{\boldsymbol{D}^c}\right\}$.

Two or More Shared Nodes

If we have $|\mathcal{S}_{12}| > 1$ shared nodes among two cliques, we need to differentiate between the case where shared nodes occupy or not occupy consecutive portions of the frame. For the first case there is a straightforward extension from the previous argument. Let us define the two system vectors $(\boldsymbol{\Upsilon}_1(t), \boldsymbol{\Upsilon}_2(t))$ such that the variables associated to the shared nodes occupy the last positions in both vectors. Then for $c = 1, 2$ adjust the definition of $\tilde{\mathbf{M}}^c, \tilde{\boldsymbol{D}}^c, \tilde{\gamma}^c$ making the index k ranging from 1 to $|\mathcal{V}_c| - |\mathcal{S}_{12}|$ instead of $|\mathcal{V}_c| - 1$. With a similar argument as before, we conclude that the fixed points for the two system vectors, taking into account the update of local nodes, lie in the space spanned by:

$$\tilde{\boldsymbol{\Upsilon}}^\star_{c,1} = \frac{\tilde{\gamma}^c}{\tilde{\boldsymbol{D}}^c}(\delta, D_{\pi^c_1}, \ldots, \delta, D_{\pi^c_{|\mathcal{V}_c|-|\mathcal{S}_{12}|}}, \delta, 0, \ldots, 0)^T \tag{A.0.22}$$

$$\tilde{\boldsymbol{\Upsilon}}^\star_{c,u} = \mathbf{e}_{2(|\mathcal{V}_c|-|\mathcal{S}_{12}|)+u} \quad \text{for} \quad u = 2, \ldots, 2|\mathcal{S}_{12}| \tag{A.0.23}$$

If we now consider the consecutive updates of the shared nodes we find the following additional for the fixed point where the node Θ_s is defined with respect to the last (in order of firing) of the shared nodes and Θ_p is defined with respect to the first of the shared nodes:

$$\Theta^\star_s = \Theta^\star_{\pi^c_{|\mathcal{V}_c|-|\mathcal{S}_c|+2}} \tag{A.0.24}$$

$$\Theta^\star_p = \Theta^c_{\pi^c_{|\mathcal{V}_c|}} \tag{A.0.25}$$

$$\Gamma^\star_v = \frac{D_v}{\delta}\Theta^*_{\pi^c_{i+1}} \quad \text{for } v = |\mathcal{V}_c| - |\mathcal{S}_c| + 1, \ldots, |\mathcal{V}_c| \tag{A.0.26}$$

The fixed point for each of the system vectors ($c = 1, 2$) needs to have the following form

$$\boldsymbol{\Upsilon}_c^\star = \left(1 - \sum_{u=2}^{2|\mathcal{S}_{12}|} \lambda_{c,u}\right) \tilde{\boldsymbol{\Upsilon}}_{c,1}^\star + \sum_{u=2}^{2|\mathcal{S}_{12}|} \lambda_{c,u} \tilde{\boldsymbol{\Upsilon}}_{c,u}^\star \tag{A.0.27}$$

and once again we have $\lambda_{cu} = \lambda_u$ for $c = 1, 2$. By imposing the constraints in (A.0.24)-(A.0.25)-(A.0.26) we find there is a unique solution that respects the constraints and is consistent with the definition of the predecessor and successor nodes (relative to the firing order), i.e.

$$\lambda_u = \begin{cases} \min\limits_{c=1,2} \left\{ \dfrac{\gamma^c}{\boldsymbol{D}^c} \right\} \delta \\ \text{for} \quad u = (2m+1), m = 1, \ldots, |\mathcal{S}_{12}| - 1 \\ \min\limits_{c=1,2} \left\{ \dfrac{\gamma^c}{\boldsymbol{D}^c} \right\} D_{\pi^c_{|\mathcal{V}_c| - |\mathcal{S}_{12}| + m}} \\ \text{for} \quad u = 2m, m = 1, \ldots, |\mathcal{S}_{12}| \end{cases} \tag{A.0.28}$$

For the case 2), the notation is significantly more complicated but the proof can follow the same conceptual steps. Let us start by considering two shared nodes that are not consecutive in the firing order in at least one of the two cliques. In each clique c we will have two subsets of $\mathcal{L}_c$, namely $\mathcal{L}_{c,1}, \mathcal{L}_{c,2}$ such that the nodes in each of the two subsets are all consecutive in the firing order. Then we can define two different matrices

$$\tilde{\mathbf{M}}^{c,u} \triangleq \prod_{k:\ \pi_k^c \in \mathcal{L}_{c,u}} \boldsymbol{M}_{\pi_k^c} \tag{A.0.29}$$

for $u = 1, 2$. and also extend the definitions for $\tilde{\boldsymbol{D}}^{c,u}, \tilde{\gamma}^{c,u}$, accordingly for $u = 1, 2$. Reasoning as before we can see the fixed point $\boldsymbol{\Upsilon}_c^\star$ needs to mantain the proportionality between the nodes in $\mathcal{L}_{c,1}$ and the ones in $\mathcal{L}_{c,2}$ separately. Furthermore, the constraint in (A.0.20) continues to hold and applies to both shared nodes. Let us assume we are at the fixed point, and there is a certain “distance” (portion of the frame) between the two non-consecutive shared nodes, which is assigned in the two cliques to $\mathcal{L}_{c,1}$ and $\mathcal{L}_{c,2}$ respectively. If we indicate by v_1 and v_2 the two shared nodes (i.e., the counterclockwise order of nodes is $v_1, \mathcal{L}_{c,1}, v_2, \mathcal{L}_{c,2}$) we find that at the fixed point, i.e., $\forall t > t^*$:

$$\mathrm{Suc}(v_1, t), \mathrm{Pre}(v_2, t) \in \mathcal{L}_{1,1} \leftrightarrow \frac{\tilde{\gamma}^{1,1}}{\tilde{\boldsymbol{D}}^{1,1}} < \frac{\tilde{\gamma}^{2,1}}{\tilde{\boldsymbol{D}}^{2,1}} \tag{A.0.30}$$

$$\mathrm{Suc}(v_2, t), \mathrm{Pre}(v_1, t) \in \mathcal{L}_{1,2} \leftrightarrow \frac{\tilde{\gamma}^{1,2}}{\tilde{\boldsymbol{D}}^{1,2}} < \frac{\tilde{\gamma}^{2,2}}{\tilde{\boldsymbol{D}}^{2,2}} \tag{A.0.31}$$

In fact, if this is a fixed point, the distance between the two shared nodes is fixed and the other local nodes in each clique share that schedule proportionally, according to their demands. This implies the guard-spaces in each of the subframes are smaller if the overall demand of that subset of nodes is bigger, giving (A.0.30)-(A.0.31). But from this condition, we have a unique fixed point given by the condition in (A.0.20) that forces the two guard-spaces for any of the shared nodes to be equal and therefore we have enough constraints on the coefficients λ to write the two system vectors and find the schedule attainable. By introducing for $u = 1, 2$ the cluster $c_u^* = \arg\min_{c=1,2} \frac{\tilde{\gamma}^{c,u}}{\tilde{\boldsymbol{D}}^{c,u}}$ we have that the scheduling attainable by the unique fixed point is such that $(c, u = 1, 2)$

$$\Gamma_i = \begin{cases} \frac{D_v}{D_{v_1} + D_{v_2} + \frac{\tilde{\boldsymbol{D}}^{c_1^*,1}}{\tilde{\gamma}^{c_1^*,1}} + \frac{\tilde{\boldsymbol{D}}^{c_2^*,2}}{\tilde{\gamma}^{c_2^*,2}}}, & \text{for} \quad i = v_1, v_2 \\ \frac{\tilde{\gamma}^{c,u}}{\tilde{\boldsymbol{D}}^{c,u}} \frac{\tilde{\boldsymbol{D}}^{c_u^*,u}}{\tilde{\gamma}^{c_u^*,u}\left(D_{v_1} + D_{v_2} + \frac{\tilde{\boldsymbol{D}}^{c_1^*,1}}{\tilde{\gamma}^{c_1^*,1}} + \frac{\tilde{\boldsymbol{D}}^{c_2^*,2}}{\tilde{\gamma}^{c_2^*,2}}\right)} D_v & \text{for} \quad v \in \mathcal{L}_{c,u} \end{cases} \tag{A.0.32}$$

and the guardspaces before the first node in $\mathcal{L}_{c,u}$ $(c, u = 1, 2)$, after the last one and in between are all equal to

$$\delta_{\mathcal{L}_{c,u}} \triangleq \frac{\tilde{\gamma}^{c,u}}{\tilde{\boldsymbol{D}}^{c,u}} \frac{\tilde{\boldsymbol{D}}^{c_u^*,u}}{\tilde{\gamma}^{c_u^*,u}\left(D_{v_1} + D_{v_2} + \frac{\tilde{\boldsymbol{D}}^{c_1^*,1}}{\tilde{\gamma}^{c_1^*,1}} + \frac{\tilde{\boldsymbol{D}}^{c_2^*,2}}{\tilde{\gamma}^{c_2^*,2}}\right)} \delta \tag{A.0.33}$$

For the extension to an arbitrary number of shared nodes $|S_{12}|$ one should consider the shared nodes i_u in S_{12} and the subsets of local nodes $\mathcal{L}_{c,u}$ $(u = 1, 2, \ldots, |S_{12}|)$ and apply the same argument to each subset of consecutive nodes to find the unique fixed point.

Proof of Proposition 2.5.7

The proposition is proved by considering a generic subset of consecutive local nodes in any possible clique i.e., a generic $\mathcal{L}_{c,u}$ (see notation introduced in the previous proof, Appendix A). The matrix $\tilde{\mathbf{M}}^{c,u}$ will then have an eigenvector associated with eigenvalue 1 where the proportional fairness is enforced between the nodes in $\mathcal{L}_{c,u}$ and other several eigenvectors associated with eigenvalue 1 where the variables $\Gamma_i,\ v \in \mathcal{L}_{c,u}$ are equal to 0. Therefore, all the possible fixed points for the algorithm will respect the *partial proportional fairness criterion* as per Definition 2.5.3. To prove that for more than two cliques we can have in general a set of non-isolated fixed points let us consider the sample topology with 3 clusters ($c = 1, 2, 3$) and the following properties: $|\mathcal{V}_1| = |\mathcal{V}_3| = 4$, $|\mathcal{V}_2| = 3$, $|\mathcal{S}_{12}| = |\mathcal{S}_{23}| = 1$, $\mathcal{S}_{13} = \emptyset$, $\delta = 1, D_v = D = 4\ \ \forall v \in \mathcal{V}$. Then we have that the following configuration (the order of nodes in $\boldsymbol{\Upsilon}_2^\star$ is $\mathcal{S}_{12}, \mathcal{L}_2, \mathcal{S}_{23}$)

$$\boldsymbol{\Upsilon}_1^\star = \boldsymbol{\Upsilon}_3^\star = \left(\frac{1}{20}, \frac{1}{5}, \frac{1}{20}, \frac{1}{5}, \frac{1}{20}, \frac{1}{5}, \frac{1}{20}, \frac{1}{5}\right)^T$$
$$\boldsymbol{\Upsilon}_2^\star = \left(\theta, \frac{1}{5}, \frac{1}{10} - \frac{1}{6}\theta, \frac{2}{5} - \frac{2}{3}\theta, \frac{1}{10} - \frac{1}{6}\theta, \frac{1}{5}\right)^T$$

is a fixed point for any $\theta \in \left[\frac{1}{20}, \frac{3}{10}\right]$, since for all these values $\mathrm{Pre}(v,t)$ and $\mathrm{Suc}(v,t)$ for $v \in \mathcal{S}_{13} \cup \mathcal{S}_{23}$ continue to remain in $\mathcal{V}_1$ or $\mathcal{V}_3$ and the space left for the only node in $\mathcal{L}_2$ is proportionally distributed between the time schedule assigned to that node and the two guardspaces before and after the schedule.

Proof of Proposition 2.5.9

First, recalling the order of the partitions $\mathcal{A}_c$ introduced in (2.5.25), $\mathcal{A}_1$ is the partition with the highest demand. One can apply Theorem 2.5.6 to each pair of cliques $(\mathcal{V}_1, \mathcal{V}_c)$ with $1 < c \leq |\mathcal{C}|$ and obtain a possible assignment. In fact, if there are nodes shared among one of these pairs that belong also to other cliques, they have to belong to one of these two partitions by Assumption 2.5.8. In this case, we can see there is a unique fixed point by having the partitions $\mathcal{A}_1, \mathcal{A}_c$ assigning the unique schedule obtainable by Theorem 2.5.6 and then apply the argument to each pair of cliques $(\mathcal{V}_2, \mathcal{V}_c)$ with $2 < c \leq |\mathcal{C}|$. The only additional case we have to consider is that nodes that belong to the pair $(\mathcal{V}_2, \mathcal{V}_c)$ also belong to $\mathcal{V}_1$; but these nodes schedules have been already set by the pair $(\mathcal{A}_1, \mathcal{A}_c)$. For the generic pair case $(\mathcal{A}_2, \mathcal{A}_c)$ Theorem 2.5.6 applies directly if we consider the quantity $\mathscr{T}_2$ defined in (2.5.28) in subsection 2.5.3.1. The procedure is then iterated for every pair of cliques until every conflict has been considered, and this proves the Proposition.

Proof of Corollary 2.5.11

For a star network, the central node (say 1) can hear all the rest (i.e. $v = 2, \ldots, n+1$). There will be n cliques $\mathcal{V}_c = \{1, c+1\}$ with $c = 1, \ldots, n$. We simply have to enumerate nodes from 2 to $n+1$ in decreasing demand order to have $\mathcal{A}_1 = \{1, 2\}$ and $\mathcal{A}_c = \{c+1\}$ for $c = 2, \ldots, n$. Then Proposition 2.5.9 applies directly and there will be a unique fixed point that respects Definition 2.5.4 in light of Theorem 2.5.6. For a line network, Assumption 2.5.8 is not needed, since every node can belong to no more than two cliques, and then we can just apply Theorem 2.5.6 starting from the two cliques that contain the highest demand node, and then repeat a similar argument as in the proof of Proposition 2.5.9 until we reached the edges of the line.

APPENDIX B

PROOFS FOR CHAPTER 3

Proof of Lemma 3.3.2

We first notice that $\forall \boldsymbol{\omega}, \mathcal{A}$, $V(\boldsymbol{\omega}, \mathcal{A}, K{+}1) = 0$ implies $V_t^j(\boldsymbol{\omega}, \mathcal{A}, K) = -c$, $\forall\, j \in \mathcal{A}$ and therefore

$$V(\boldsymbol{\omega}, \mathcal{A}, K) = \max_{\mathcal{D}\subseteq\mathcal{A}}\{V_d(\boldsymbol{\omega}, \mathcal{D}) + \max_{j\in\mathcal{A}\backslash\mathcal{D}} V_t^j(\boldsymbol{\omega}, \mathcal{A}, K)\} = V_d(\boldsymbol{\omega}, \mathcal{A}) = \sum_{j\in\mathcal{A}} V_d^j(\omega_j) \quad \text{(B.0.1)}$$

which is convex in any $\omega_j, j \in \mathcal{A}$, since it is a positive sum of piece-wise linear functions (see (3.3.15)). We then prove the lemma by induction. We assume $V(\boldsymbol{\omega}, \mathcal{A}, k+1)$ is convex in ω_j and we show $V(\boldsymbol{\omega}, \mathcal{A}, k)$ is convex by showing all the possible functions $V_t^i(\boldsymbol{\omega}, \mathcal{A}, k)$ are convex in ω_j. We start by the case $j = i$. Without loss of generality, let us consider two $\boldsymbol{\omega}$ say $(\boldsymbol{\omega}^1, \boldsymbol{\omega}^2)$ which differ only in the j-th entry (i.e $\omega_\ell^1 = \omega_\ell^2, \forall \ell \neq i$). We want to prove that for $0 \le \lambda \le 1$ we have $\lambda V_t^i(\boldsymbol{\omega}^1, \mathcal{A}, k) + (1-\lambda) V_t^i(\boldsymbol{\omega}^2, \mathcal{A}, k) \ge V_t^i(\lambda\boldsymbol{\omega}^1 + (1-\lambda)\boldsymbol{\omega}^2, \mathcal{A}, k)$. From (3.3.13) we can write (we will use the short notation $\boldsymbol{\omega}^{\ell,o} = \Pi(\boldsymbol{\omega}^\ell, y, i)$ for $\ell = 1, 2$)

$$\begin{aligned}
&\lambda V_t^i(\boldsymbol{\omega}^1, \mathcal{A}, k) + (1-\lambda) V_t^i(\boldsymbol{\omega}^2, \mathcal{A}, k)\\
&= -c + \int \lambda V(\boldsymbol{\omega}^{1,o}, \mathcal{A}, k+1) f^i_{1-\omega_i^1}(y) + (1-\lambda) V(\boldsymbol{\omega}^{2,o}, \mathcal{A}, k+1) f^i_{1-\omega_i^2}(y) dy\\
&= -c + \int \left[\mu V(\boldsymbol{\omega}_i^{1,o}, \mathcal{A}, k{+}1) + (1-\mu) V(\boldsymbol{\omega}^{2,o}, \mathcal{A}, k{+}1)\right]\left[\lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y)\right] dy\\
&\overset{(a)}{\ge} -c + \int \left[V(\mu\boldsymbol{\omega}^{1,o} + (1-\mu)\boldsymbol{\omega}^{2,o}, \mathcal{A}, k+1)\right]\left[\lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y)\right] dy \quad \text{(B.0.2)}
\end{aligned}$$

where (a) follows from the assumption that $V(\boldsymbol{\omega}, \mathcal{A}, k+1)$ is convex in ω_i and $\mu = \frac{\lambda f^i_{1-\omega_i^1}(y)}{\lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y)}$. Now, if we define $\boldsymbol{\omega}^3 = \lambda\boldsymbol{\omega}^1 + (1-\lambda)\boldsymbol{\omega}^2$ we have

$$f^i_{1-\omega_i^3}(y) = \lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y) \quad \text{(B.0.3)}$$

since $f^i_{1-\omega_i}(y)$ is a linear affine function of ω_i and also

$$\Pi_i(\omega_i^3, y, i) = \frac{\omega_i^3 f_0^i(y)}{f^i_{1-\omega_i^3}(y)} = \frac{[\lambda\omega_i^1 + (1-\lambda)\omega_i^2] f_0^{(i)}(y)}{\lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y)}$$

$$= \frac{\lambda f^i_{1-\omega_i^1}(y)\dfrac{\omega_i^1 f_0^{(i)}(y)}{f^i_{1-\omega_i^1}(y)} + (1-\lambda) f^i_{1-\omega_i^2}(y)\dfrac{\omega_i^2 f_0^{(i)}(y)}{f^i_{1-\omega_i^2}(y)}}{\lambda f^i_{1-\omega_i^1}(y) + (1-\lambda) f^i_{1-\omega_i^2}(y)} = \mu\Pi_i(\omega_i^1, y, i) + (1-\mu)\Pi_i(\omega_i^2, y, i)$$

which implies

$$\boldsymbol{\Pi}(\boldsymbol{\omega}^3, y, i) = \mu\boldsymbol{\Pi}(\boldsymbol{\omega}^1, y, i) + (1-\mu)\boldsymbol{\Pi}(\boldsymbol{\omega}^2, y, i). \tag{B.0.4}$$

Therefore, by replacing (B.0.3),(B.0.4) in (B.0.2) we have

$$\lambda V_t^i(\boldsymbol{\omega}^1, \mathcal{A}, k) + (1-\lambda)V_t^i(\boldsymbol{\omega}^2, \mathcal{A}, k) \geq -c + \int V(\Pi(\boldsymbol{\omega}^3, y, i), \mathcal{A}, k+1) f^i_{1-\omega_i^3}(y)$$

$$= V_t^i(\boldsymbol{\omega}^3, \mathcal{A}, k) = V_t^i(\lambda\boldsymbol{\omega}^1 + (1-\lambda)\boldsymbol{\omega}^2, \mathcal{A}, k) \tag{B.0.5}$$

and this proves the convexity of $V_t^i(\boldsymbol{\omega}, \mathcal{A}, k)$ in ω_i. The convexity of $V_t^i(\boldsymbol{\omega}, \mathcal{A}, k)$ in ω_j, $j \neq i$ can be proved by considering two points $(\boldsymbol{\omega}^1, \boldsymbol{\omega}^2)$ with the same i-th coordinate (i.e $\omega_i^1 = \omega_i^2 = \omega_i$) and following similar steps as before, where this time we have

$$\boldsymbol{\Pi}(\boldsymbol{\omega}^3, y, i) = \lambda\boldsymbol{\Pi}(\boldsymbol{\omega}^1, y, i) + (1-\lambda)\boldsymbol{\Pi}(\boldsymbol{\omega}^2, y, i). \tag{B.0.6}$$

and we do not need to introduce μ to conclude our proof. To show the function $V(\boldsymbol{\omega}, \boldsymbol{\phi}_k, k)$ is convex in ω_j we rewrite the maximization over $\mathcal{D}$ in (3.3.11) as follows:

$$V(\boldsymbol{\omega}, \mathcal{A}_k, k) = \max\left\{ \max_{\{j\}\subseteq\mathcal{D}\subseteq\mathcal{A}_k} J(\mathcal{D}), \max_{\mathcal{D}\subseteq\mathcal{A}_k\setminus\{j\}} J(\mathcal{D}) \right\} \tag{B.0.7}$$

where the function $J(\mathcal{D})$ is defined in (3.3.18). Let us then call f_1 and f_2 the two maximizations inside (B.0.7) in the order they appear and let us omit the arguments for brevity. Now if we see f_1 and f_2 as functions of ω_j we have that both f_1 and f_2 are convex functions of ω_j. In fact ω_j in f_1 appears only as argument of $V_d^j(\omega_j)$ which is a piece-wise linear function of ω_j and therefore convex, whereas in f_2, ω_j is an argument of the second term of $J(\mathcal{D})$, which is the maximization over the index i of the functions $V_t^i(\boldsymbol{\omega}, \mathcal{A}_{k+1}, k)$ $(i, j \in \mathcal{A}_{k+1})$ that are all convex in ω_j. Therefore the maximum is convex and we can conclude f_2 is convex. The convexity of $V(\boldsymbol{\omega}, \mathcal{A}_k, k)$ follows from the fact it is the maximum of two convex functions.

Proof of Lemma 3.3.4

Let us first introduce the following lemma

Lemma B.0.1. $\forall i \in \mathcal{N}, \omega_i \in [0,1], k = 1, \ldots, K-1$:

$$V_t^i(\omega_i, i, k) \geq V_t^i(\omega_i, i, k+1) + V_d^i(\omega_i) \tag{B.0.8}$$

Proof. We will prove once again by induction. First we show that (B.0.8) is true for $k = L-2$ (we use the short notation $\omega_i^y = \Pi_i(\omega_i, y, i)$ and $dF_i^y = f_{1-\omega_i}^i(y)dy$).

$$\begin{aligned} &V_t^i(\omega_i, i, K-1) + c = \int\limits_{\mathcal{Y}} V\left(\omega_i^y, i, K\right) dF_i^y = \int\limits_{\mathcal{Y}} V_d^i\left(\omega_i^y\right) dF_i^y \\ &\overset{(a)}{\geq} V_d^i\left(\int\limits_{\mathcal{Y}} \omega_i^y dF_i^y\right) \overset{(b)}{=} V_d^i(\omega_i) = V_t^i(\omega_i, i, K) + V_d^i(\omega_i) + c \end{aligned}$$

where (a) holds for the convexity of the function V_d^i and (b) holds for the martingale property of the prior belief update

$$\int_{\mathcal{Y}} \Pi_i(\omega_i, y, i) f_{1-\omega_i}^i(y) dy = \int_{\mathcal{Y}} \frac{\omega_i f_0^i(y)}{f_{1-\omega_i}^i(y)} f_{1-\omega_i}^i(y) dy = \omega_i.$$

Then we show that if (B.0.8) holds for k, then it holds for $k-1$.

$$\begin{aligned} &V_t^i(\omega_i, i, k-1) + c = \int\limits_{\mathcal{Y}} V\left(\omega_i^y, k\right) dF_i^y \\ &= \int\limits_{\mathcal{Y}} \max\left\{(K-k+1)V_d^i(\omega_i^y), V_t^i(\omega_i^y, i, k)\right\} dF_i^y \\ &\overset{(a)}{\geq} \int\limits_{\mathcal{Y}} \max\left\{(K-k+1)V_d^i(\omega_i^y), V_t^i(\omega_i^y, i, k+1) + V_d^i(\omega_i^y)\right\} dF_i^y \\ &= \int\limits_{\mathcal{Y}} \left[\max\left\{(K-k)V_d^i(\omega_i^y), V_t^i(\omega_i^y, i, k+1)\right\} + V_d^i(\omega_i^y)\right] dF_i^y \\ &= V_t^i(\omega_i, i, k) + c + \int\limits_{\mathcal{Y}} V_d^i(\omega_i^y) dF_i^y \end{aligned}$$

$$\overset{(b)}{\geq} V_t^i(\omega_i, i, k) + c + V_d^i\left(\int_{\mathcal{Y}} \omega_i^y dF_i^y\right) \overset{(c)}{=} V_t^i(\omega_i, i, k) + c + V_d^i(\omega_i)$$

where (a) holds for the induction hypotheses, (b) for the convexity of V_d^i and (c) for the martingale property of the belief update. ■

Now, to show the two thresholds respect (3.3.24)-(3.3.25) it is equivalent to prove the following statement

$$\forall \omega_i \in [0,1],\ (K-k+1)V_d^i(\omega_i) \geq V_t^i(\omega_i, i, k) \Rightarrow (K-k)V_d^i(\omega_i) \geq V_t^i(\omega_i, i, k+1)$$

and this can be proved since

$$(K-k)V_d^i(\omega_i) = (K-k+1)V_d^i(\omega_i) - V_d^i(\omega_i) \overset{(a)}{\geq} V_t^i(\omega_i, i, k) - V_d^i(\omega_i) \overset{(b)}{\geq} V_t^i(\omega_i, i, k+1)$$

where (a) holds by hypothesis and (b) follows from Lemma B.0.1, and this completes the proof.

Proof of Lemma 3.3.5

We start our proof by highlighting a property of the value function defined in (3.3.11) with the following lemma

Lemma B.0.2. $\forall i \in \mathcal{N}, \mathcal{A}' \in 2^{\mathcal{N}-i}, \omega_i \in [0,1], k = 1, \dots, K$

$$V(\boldsymbol{\omega}, \mathcal{A}', k) + V(\omega_i, i, k) \geq V(\boldsymbol{\omega}, \mathcal{A}' + i, k) \tag{B.0.9}$$

Proof. Let us consider the function $J(\mathcal{D})$ defined in (3.3.18) refers to the value function when $\mathcal{A}_k = \mathcal{A} = \mathcal{A}' + i$ and $J'(\mathcal{D})$ refers to the value function when $\mathcal{A}_k = \mathcal{A}'$. We will again use induction. Clearly (B.0.9) holds for $k = K$ where

$$\left[V_d(\boldsymbol{\omega}, \mathcal{A}') + V_d^i(\omega_i)\right] \geq V_d(\boldsymbol{\omega}, \mathcal{A})$$

Then, assuming (B.0.9) holds for $k+1$, we prove it holds for k. Let us write (B.0.9) as

$$\max_{\mathcal{D} \subseteq \mathcal{A}'} J'(\mathcal{D}) + V(\omega_i, i, k) \geq \max_{\mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D}) \tag{B.0.10}$$

and call $\tilde{\mathcal{D}} = \arg\max_{\mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D})$. There are three possible cases:

1. If $i \in \tilde{\mathcal{D}}$,

$$\max_{\mathcal{D} \subseteq \mathcal{A}'} J'(\mathcal{D}) + V(\omega_i, i, k) \geq J'(\tilde{\mathcal{D}} - i) + (K - k + 1) V_d^i(\omega_i) = J(\tilde{\mathcal{D}}) = \max_{\mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D})$$

2. If $i \notin \tilde{\mathcal{D}}$ and $i = \arg\max_{j \in (\mathcal{A}) \setminus \tilde{\mathcal{D}}} V_t^j(\boldsymbol{\omega}, \mathcal{A} \setminus \tilde{\mathcal{D}}, k)$

$$\begin{aligned}
&\max_{\mathcal{D} \subseteq \mathcal{A}'} J'(\mathcal{D}) + V(\omega_i, i, k) \overset{(a)}{\geq} (K - k + 1) V_d(\boldsymbol{\omega}, \tilde{\mathcal{D}}) + V(\boldsymbol{\omega}, \mathcal{A}' \setminus \tilde{\mathcal{D}}, k) + V_t^i(\omega_i, i, k) \\
&\geq (K - k + 1) V_d(\boldsymbol{\omega}, \tilde{\mathcal{D}}) + V(\boldsymbol{\omega}, \mathcal{A}' \setminus \tilde{\mathcal{D}}, k + 1) - c + \int_{\mathcal{Y}} V\left(\omega_i^y, i, k + 1\right) dF_i^y \\
&\overset{(b)}{\geq} (K - k + 1) V_d(\boldsymbol{\omega}, \tilde{\mathcal{D}}) - c + \int_{\mathcal{Y}} V\left(\boldsymbol{\Pi}(\boldsymbol{\omega}, y, i), \mathcal{A} \setminus \tilde{\mathcal{D}}, k + 1\right) dF_i^y
\end{aligned}$$

$$= (K-k+1)V_d(\boldsymbol{\omega}, \tilde{\mathcal{D}}) + V_t^i(\boldsymbol{\omega}, \mathcal{A} \setminus \tilde{\mathcal{D}}, k) = \max_{\mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D})$$

where (a) follows from the definition of our value function in (3.3.11) and (b) holds by the induction hypothesis.

3. If $i \notin \tilde{\mathcal{D}}$ and $i \neq j^* = \arg\max_{j \in \mathcal{A} \setminus \tilde{\mathcal{D}}} V_t^j(\boldsymbol{\omega}, \mathcal{A} \setminus \tilde{\mathcal{D}}, k)$ the proof is similar to the previous case.

■

Proving (3.3.26)-(3.3.27) is equivalent to prove:

$$\forall \omega_i \in [0,1],\ (K-k+1)V_d^i(\omega_i) \geq V_t^i(\omega_i, i, k) \Rightarrow \max_{\{i\} \subseteq \mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D}) \geq \max_{\mathcal{D} \subseteq \mathcal{A}'} J(\mathcal{D}).$$

To prove this, we call $\mathcal{D}^* = \arg\max_{\mathcal{D} \subseteq \mathcal{A}'} J'(\mathcal{D})$ and use the chain of inequalities

$$\max_{\{i\} \subseteq \mathcal{D} \subseteq \mathcal{A}} J(\mathcal{D}) \geq J(\mathcal{D}^* + i) = (K-k+1)V_d^i(\omega_i) + (K-k+1)V_d(\boldsymbol{\omega}, \mathcal{D}^*)$$
$$+ \max_{j \in \mathcal{A}' \setminus \mathcal{D}^*} V_t^j(\boldsymbol{\omega}, \mathcal{A}' \setminus \mathcal{D}^*, k) \overset{(a)}{=} V(\omega_i, i, k) + V(\boldsymbol{\omega}, \mathcal{A}', k) \overset{(b)}{\geq} V(\boldsymbol{\omega}, \mathcal{A}, k) \overset{(c)}{\geq} \max_{\mathcal{D} \subseteq \mathcal{A}'} J(\mathcal{D})$$

where (a) holds by hypotheses, since if $(K-k+1)V_d^i(\omega_i) \geq V_t^i(\omega_i, i, k)$, then $V(\omega_i, i, k) = (K-k+1)V_d^i(\omega_i)$ and definition of $J'(\mathcal{D})$, (b) follows by Lemma B.0.2 and (c) by definition of value function in (3.3.11), and this completes the proof.

Proof of Lemma 3.3.8

Since we are focusing on a specific resource i next, for brevity we drop the index i and define the following events:

$$A_k \triangleq \{\omega[k] \geq \nu_0[k]\},\ B_k \triangleq \{\omega[k] \leq \nu_1[k]\}, C_k \triangleq \overline{A_k \cup B_k},\ \mathcal{C}_k^\ell = \bigcap_{m=k}^{\ell} C_m$$

We then write the function $V_t(\omega, k)$ as follows (we use the short notation $P_0(A) = P(A|s=0)$ and same for $P_1(A)$):

$$V_t(\omega, k) = -c + \sum_{\ell=k}^{K-1} P\left(\mathcal{C}_{k+1}^\ell\right) \mathbb{E}\left[-u[\ell-k]c + (K-\ell)\right.$$
$$\left.[\omega[\ell] r P_0\left(A_{\ell+1}\right) - (1-\omega[\ell])\rho P_1\left(A_{\ell+1}\right)] \middle| \mathcal{C}_{k+1}^\ell\right] \tag{B.0.11}$$

To find bounds for $V_t(\omega, k)$ we will use:

1. an upper/lower bound for $P\left(\mathcal{C}_{k+1}^\ell\right)$

2. an upper/lower bound for $P_0\left(A_{\ell+1} \middle| \mathcal{C}_{k+1}^\ell\right)$

3. an upper/lower bound for $P_1\left(A_{\ell+1} \middle| \mathcal{C}_{k+1}^\ell\right)$

First we write:

$$P\left(\mathcal{C}_{k+1}^\ell\right) = P\left(C_\ell \middle| \mathcal{C}_{k+1}^{\ell-1}\right) P\left(\mathcal{C}_{k+1}^{\ell-1}\right) = \left[1 - P\left(A_\ell \middle| \mathcal{C}_{k+1}^{\ell-1}\right) - P\left(B_\ell \middle| \mathcal{C}_{k+1}^{\ell-1}\right)\right] P\left(\mathcal{C}_{k+1}^{\ell-1}\right) \tag{B.0.12}$$

We now find 2). 3) follows a similar approach and 1) will be otained using 2) and 3) (in our derivation with $P(\cdot|\varphi)$ we indicate we are conditioning on $\omega[\ell] = \varphi$).

$$P_0\left(A_{\ell+1} \middle| \mathcal{C}_{k+1}^\ell\right) = \int_{C_\ell} P_0\left(A_{\ell+1}|t\right) \frac{P_0\left(t \middle| \mathcal{C}_{k+1}^{\ell-1}\right)}{P_0\left(C_\ell \middle| \mathcal{C}_{k+1}^{\ell-1}\right)} dt$$
$$\overset{u.b.}{\leq} \max_{\nu_1[\ell]<t<\nu_0[\ell]} P_0\left(A_{\ell+1}|t\right) = P_0\left(A_{\ell+1} \middle| \nu_0[\ell]\right) \leq P_0\left(A_{\ell+1} \middle| \overline{\nu_0}[\ell]\right)$$

$$
\begin{aligned}
&= \bar{F}^1_{\omega[\ell+1]}\left(\nu_0[\ell+1]|\overline{\nu_0}[\ell],0\right) \le \bar{F}^1_{\omega[\ell+1]}\left(\underline{\nu_0}[\ell+1]|\overline{\nu_0}[\ell],0\right) \\
&\overset{l.b.}{\ge} \min_{\nu_1[\ell]<t<\nu_0[\ell]} P_0\left(A_{\ell+1}|t\right) = P_0\left(A_{\ell+1}|\nu_1[\ell]\right) \ge P_0\left(A_{\ell+1}|\underline{\nu_1}[\ell]\right) \\
&= \bar{F}^1_{\omega[\ell+1]}\left(\nu_0[\ell+1]|\underline{\nu_1}[\ell],0\right) \ge \bar{F}^1_{\omega[\ell+1]}\left(\overline{\nu_0}[\ell+1]|\underline{\nu_1}[\ell],0\right)
\end{aligned}
$$

where the upper and lower bound start by taking out of the integral the maximum and minimum value of
$P_0(A_{\ell+1}|\omega[\ell]=t)$ and notice that

$$\int\limits_{C_\ell} P_0\left(\omega[\ell]=t\middle|\mathcal{C}^{\ell-1}_{k+1}\right)dt = P_0\left(C_\ell\middle|\mathcal{C}^{\ell-1}_{k+1}\right) \tag{B.0.13}$$

Following similar steps for the event $B_{\ell+1}$ we can derive:

$$P_0\left(A_{\ell+1}\middle|\mathcal{C}^{\ell}_{k+1}\right) \overset{u.b.}{\le} \bar{F}^1_{\omega[\ell+1]}\left(\underline{\nu_0}[\ell+1]|\overline{\nu_0}[\ell],0\right) \tag{B.0.14}$$

$$P_0\left(A_{\ell+1}\middle|\mathcal{C}^{\ell}_{k+1}\right) \overset{l.b.}{\ge} \bar{F}^1_{\omega[\ell+1]}\left(\overline{\nu_0}[\ell+1]|\underline{\nu_1}[\ell],0\right) \tag{B.0.15}$$

$$P_0\left(B_{\ell+1}\middle|\mathcal{C}^{\ell}_{k+1}\right) \overset{u.b.}{\le} 1-\bar{F}^1_{\omega[\ell+1]}\left(\overline{\nu_1}[\ell+1]|\underline{\nu_1}[\ell],0\right) \tag{B.0.16}$$

$$P_0\left(B_{\ell+1}\middle|\mathcal{C}^{\ell}_{k+1}\right) \overset{l.b.}{\ge} 1-\bar{F}^1_{\omega[\ell+1]}\left(\underline{\nu_1}[\ell+1]|\overline{\nu_0}[\ell],0\right) \tag{B.0.17}$$

Exactly the same bounds can be found for the case $s=1$, replacing the index of P and the conditioned state on F from 0 to 1. For the case of $\ell=k$, which corresponds to the probabilities in the first term of the sum in (B.0.11), we can use the bounds in (B.0.14)-(B.0.15) by considering $\overline{\nu_0}[k]=\underline{\nu_1}[k]=\omega$. We can then use the recursion in (B.0.12) and the bounds previously found to derive:

$$P\left(\mathcal{C}^{\ell}_{k+1}\right) \overset{u.b.}{\le} \omega\prod_{m=k+1}^{\ell}\overline{\mu}[m|0]+(1-\omega)\prod_{m=k+1}^{\ell}\overline{\mu}[m|1] \tag{B.0.18}$$

$$P\left(\mathcal{C}^{\ell}_{k+1}\right) \overset{l.b.}{\ge} \omega\prod_{m=k+1}^{\ell}\underline{\mu}[m|0]+(1-\omega)\prod_{m=k+1}^{\ell}\underline{\mu}[m|1] \tag{B.0.19}$$

Once we upper and lower bound the probabilities for each term of the sum in (B.0.11), then the argument of the expectation is a monotonically increasing function of $\omega[\ell]$.

Therefore, since we are conditioning on the event $\mathcal{C}_{k+1}^{\ell}$, the expectation will be lower or upper-bounded by choosing $\omega[\ell] = \underline{\nu_1}[\ell]$ or $\omega[\ell] = \overline{\nu_0}[\ell]$ respectively. Gathering these results and considering the convexity of $V_t^i(\omega_i, i, k)$, we can write the upper and lower bound $\overline{V_t}(\omega, k), \underline{V_t}(\omega, k)$ as in (3.3.39)-(3.3.40) where we introduce the max to make sure our functions remain above c.

Proof of Lemma 3.3.9

Without loss of generality we can prove the lemma at $k = 1$ and for general k simply consider the time translation and the equivalent problem for $k' = 1, 2, \dots, K - k + 1$. For the remainder of the proof the index k is used as the time index for the test of resource i that starts at time 1. We indicate with $\Lambda_i^k \triangleq \sum_{\ell=1}^{k} \log\left(\frac{f_1^i(y[\ell])}{f_0^i(y[\ell])}\right)$ the log-likelihood of the samples collected for resource i up to time $k+1$. By considering the belief update in (3.3.8) and the optimal policy structure in Theorem 3.3.3, a final decision on resource i can be made as soon as

$$\Lambda_i^k \geq \Upsilon_1^i\left(\mathcal{A}_k, k\right) \;\vee\; \Lambda_i^k \leq \Upsilon_0^i\left(\mathcal{A}_k, k\right) \tag{B.0.20}$$

where $\Upsilon_{0,1}^i(\mathcal{A}_k, k) \triangleq \varsigma(\omega_i[1], \nu_{0,1}^i(\mathcal{A}_k, k))$, and we recall the resources in the set $\mathcal{A}_k$ are the resources for which a decision is still pending at time k. From (B.0.20) it follows that τ_i

$$\tau_i \triangleq \inf\left\{0 \leq k \leq K-1 \colon \Lambda_i^{k+1} \geq \Upsilon_1^i\left(\mathcal{A}_{k+1}, k+1\right) \vee \Lambda_i^{k+1} \leq \Upsilon_0^i\left(\mathcal{A}_{k+1}, k+1\right)\right\} \tag{B.0.21}$$

is a stopping time. From (3.3.28)-(3.3.29) with $k' = 1$ we can deduce
$\forall k = 1, \dots, K, \forall \mathcal{A}_k \subseteq \mathcal{N}$

$$\Upsilon_0^i(\mathcal{A}_k, k) \geq \varsigma(\omega_i[1], \overline{\nu_0^i}[1]) = \underline{\Upsilon_0^i}, \tag{B.0.22}$$

$$\Upsilon_1^i(\mathcal{A}_k, k) \leq \varsigma(\omega_i[1], \underline{\nu_1^i[1]}) = \overline{\Upsilon_1^i}. \tag{B.0.23}$$

Therefore

$$\overline{\tau}_i \triangleq \inf\left\{0 \leq k \leq K-1 : \Lambda_i^{k+1} \geq \overline{\Upsilon_1^i} \vee \Lambda_i^{k+1} \leq \underline{\Upsilon_0^i}\right\} \tag{B.0.24}$$

is a stopping time always greater than τ_i, i.e. $P(\overline{\tau}_i > \tau_i) = 1$. We then derive a bound on the expected value of $\overline{\tau}_i$ and this will also hold for τ_i. We will use a similar technique as the one used for the thresholds approximation (see Proof of Lemma 3.3.8 in Appendix B) to write

$$\mathbb{E}_1\left[\Lambda_i^{\overline{\tau}_i}\right] = E_1\left[\Lambda_i^{\overline{\tau}_i-1} + \Lambda_i^1\right] \leq \overline{\Upsilon_1^i} + \mathbb{E}_1[\Lambda_i^1|\Lambda_i^1 > 0] \tag{B.0.25}$$

$$\mathbb{E}_0\left[\Lambda_i^{\overline{\tau}_i}\right] = E_0\left[\Lambda_i^{\overline{\tau}_i-1} + \Lambda_i^1\right] \geq \underline{\Upsilon_0^i} + \mathbb{E}_0[\Lambda_i^1|\Lambda_i^1 < 0] \tag{B.0.26}$$

where the bounds follow by $\overline{\tau_i}$ being the stopping time, therefore we know $\underline{\Upsilon_0^i} < \Lambda_i^{\overline{\tau_i}-1} < \overline{\Upsilon_1^i}$ and $\Lambda_i^{\overline{\tau_i}} > \overline{\Upsilon_1^i}$ or $\Lambda_i^{\overline{\tau_i}} < \underline{\Upsilon_0^i}$. By Wald's identity we have

$$\mathbb{E}\left[\Lambda_i^{\overline{\tau_i}}\right] = \mathbb{E}[\overline{\tau_i}]\mathbb{E}\left[\Lambda_i^1\right] \tag{B.0.27}$$

and since $\mathbb{E}_0\left[\Lambda_i^1\right] = -D(f_0^i||f_1^i)$, $\mathbb{E}_1\left[\Lambda_i^1\right] = D(f_1^i||f_0^i)$ we can write

$$\mathbb{E}_0\left[\overline{\tau_i}\right] \leq \frac{-\underline{\Upsilon_0^i} - \mathbb{E}_0[\Lambda_i^1|\Lambda_i^1 < 0]}{D(f_0^i||f_1^i)} \tag{B.0.28}$$

$$\mathbb{E}_1\left[\overline{\tau_i}\right] \leq \frac{\overline{\Upsilon_1^i} + \mathbb{E}_1[\Lambda_i^1|\Lambda_i^1 > 0]}{D(f_1^i||f_0^i)} \tag{B.0.29}$$

and combining (B.0.28)-(B.0.29) (conditioned on the status of the resource) and evaluating the two bounds in (B.0.22)-(B.0.23) at any time $k \neq 0$ we can obtain the bound in (3.3.41).

Motivation for the index in our decision algorithm

The reason to approximate the optimal selection rule with the index $\frac{\omega_i[k]r_i}{\mathbb{E}[\tau_i-k|k]}$ finds its motivation in the asymptotic utility growth. Let us consider L large enough such that we can neglect the probability of taking a wrong decision over a certain resource, and we can assume the sensing time for each resource is not affected by having spent time sensing other resources before. We also further limit our strategy to sequentially sense each resource until a decision is made and then switch to a different resource. Then there is an optimal sorting for this strategy which maximizes the expected utility. To find such sorting, we use an interchange argument. Let us consider a pair of arbitrary resources, say 1 and 2, and show the sorting $1, 2$ at time 1 with K instants available is optimal if (we use the short notation $\mathbb{E}[\tau_i]$ for $\mathbb{E}[\tau_i - k|k]$ when $k = 0$):

$$\begin{aligned}
&-c\mathbb{E}[\tau_1] + (K - \mathbb{E}_0[\tau_1])\omega_1 r_1 - c\mathbb{E}[\tau_2 - \tau_1|\tau_1] + (K - \mathbb{E}[\tau_1] - \mathbb{E}_0[\tau_2 - \tau_1|\tau_1])\omega_2 r_2 > \\
&-c\mathbb{E}[\tau_2] + (K - \mathbb{E}_0[\tau_2])\omega_2 r_2 - c\mathbb{E}[\tau_1 - \tau_2|\tau_2] + (K - \mathbb{E}[\tau_2] - \mathbb{E}_0[\tau_1 - \tau_2|\tau_2])\omega_1 r_1 \quad \overset{(a)}{\Rightarrow} \\
&-c\mathbb{E}[\tau_1] + (K - \mathbb{E}_0[\tau_1])\omega_1 r_1 - c\mathbb{E}[\tau_2] + (K - \mathbb{E}[\tau_1] - \mathbb{E}_0[\tau_2])\omega_2 r_2 > \\
&-c\mathbb{E}[\tau_2] + (K - \mathbb{E}_0[\tau_2])\omega_2 r_2 - c\mathbb{E}[\tau_1] + (K - \mathbb{E}[\tau_2] - \mathbb{E}_0[\tau_1])\omega_1 r_1 \\
&\Leftrightarrow -\mathbb{E}[\tau_1]\omega_2 r_2 > -\mathbb{E}[\tau_2]\omega_1 r_1 \quad \Leftrightarrow \quad \frac{\omega_1 r_1}{\mathbb{E}[\tau_1]} > \frac{\omega_2 r_2}{\mathbb{E}[\tau_2]}
\end{aligned}$$

where (a) follows from the assumption the sensing time for the second resource is not affected by the time spent in sensing the previous one, which is a reasonable assumption for L large enough. However, due to the time variant threshold and the finite horizon scenario, the expected sensing time strongly depends on the time k the test starts, and is therefore affected by the sorting. Nevertheless, this index represents a good approximation that takes into account the identifiability of the resource and simulation results will corroborate our intuition.

Derivation of (3.5.15)

The instantaneous SNR at the primary receiver in presence of interference from the secondary transmitter can be expressed as $\gamma_{i,\bar{\xi}_i}^P = \frac{\bar{\gamma}_i^P \left|h^P\right|^2}{1+\bar{\xi}_i \left|h^S\right|^2}$, where h^P, h^S are the two complex channel gains and for Rayleigh fading their absolute value squared is exponential with unitary mean. The primary transmitter is transmitting at constant rate

$$C_{i,out}^P = W_0 \log_2 \left(1 + \gamma_{i,min}^P\right) \tag{B.0.30}$$

according with a certain threshold $\gamma_{i,min}^P$ that corresponds to its designed outage probability $P_{i,out}^P$ such that

$$P_{i,out}^P = P \left(\gamma_i^P < \gamma_{i,min}^P\right\} . \tag{B.0.31}$$

which gives the inverse relationship (assuming Rayleigh fading) [Choudhury and Gibson(2007)]

$$\gamma_{i,min}^P = -\bar{\gamma}_i^P \ln \left(1 - P_{i,out}^P\right) \tag{B.0.32}$$

Thus, in presence of interference, the probability of the primary receiver to not be able to decode succesfully the symbols transmitted by the primary receiver becomes

$$P_{i,\bar{\xi}_i}^P = P \left(\bar{\gamma}_i^P \left|h^P\right|^2 - \bar{\xi}_i \gamma_{i,min}^P \left|h^S\right|^2 < \gamma_{i,min}^P\right) \tag{B.0.33}$$

where $\gamma_{i,min}^P = -\bar{\gamma}_i^P \ln \left(1 - P_{i,out}^P\right)$. By assuming independence between $\left|h^P\right|^2$ and $\left|h^S\right|^2$ this probability can be computed by defining a random variable $Y = X_1 - X_2$, where $X_1 \sim Exp(\alpha_1)$, $X_2 \sim Exp(\alpha_2)$ and deriving the cdf $P\left(Y \leq y\right)$ for $y > 0$. The result gives

$$P\left(Y \leq y\right) = 1 - \frac{\alpha_1}{\alpha_1 + \alpha_2} e^{-\frac{y}{\alpha_1}} \qquad y > 0.$$

Therefore, we can substitute this CDF in (B.0.33) to find

$$P_{i,\bar{\xi}_i}^P = \frac{P_{i,out}^P - \bar{\xi}_i \ln \left(1 - P_{i,out}^P\right)}{1 - \bar{\xi}_i \ln \left(1 - P_{i,out}^P\right)} \tag{B.0.34}$$

and the effective rate for the primary receiver will be the one expressed by (3.5.15).

Derivation of $\bar{F}^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, s)$ in (3.5.18)-(3.5.19)

We focus on the derivation of (3.5.18) for $s = 0$. The proof for $s = 1$ in (3.5.19) follows a similar argument. From the update belief rule in (3.3.8) we know

$$\omega_i[k+\ell] = \frac{1}{1 + \frac{1-\omega_i[k]}{\omega_i[k]} \prod_{w=0}^{\ell-1} \frac{f_1^i(o[k+w])}{f_0^i(o[k+w])}} \tag{B.0.35}$$

therefore we can rewrite $F^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, 0)$ as follows

$$F^{\ell}_{\omega_i[k+\ell]}(\omega|\varphi, 0) = P\left(\prod_{w=0}^{\ell-1} \frac{f_1^i(o[k+w])}{f_0^i(o[k+w])} < \frac{\varphi(1-\omega)}{(1-\varphi)\omega} \middle| s_i = 0\right) \tag{B.0.36}$$

But for our choice of distribution $f_j^i(y), \quad j = 0, 1$ in (3.5.13) we can write

$$\prod_{w=0}^{\ell-1} \frac{f_1^i(o[k+w])}{f_0^i(o[k+w])} = \left(\frac{\theta_0^i}{\theta_1^i}\right)^{\ell} \exp\left\{\left(1 - \frac{\theta_0^i}{\theta_1^i}\right) \sum_{w=0}^{\ell-1} \frac{o[k+w]}{\theta_0^i}\right\} \tag{B.0.37}$$

and if we condition on $s_i = 0$ then $\frac{y[k]}{\theta_0^i} \overset{i.i.d.}{\sim} Exp(1)$ which implies $\sum_{k=1}^{\ell} \frac{y[k]}{\theta_0^i} \sim \Gamma(\ell)$ where $\Gamma(\ell)$ is the sum of ℓ exponential random variables with unitary mean. Equation (3.5.18) is then obtained by substituting (3.5.14)-(3.5.17)-(B.0.37) in (B.0.36) and doing simple algebraic steps, given that $\frac{\theta_1^i}{\theta_0^i} = 1 + \bar{\zeta}_i$. For $k = 1$, it will never be more convenient to make a test since there would be no time instants to achieve the utility. This is equivalent to set $\nu_0^i[1] = \nu_1^i[1]$ and from simple geometric consideration, it is possible to see that setting $\nu_0^i[1] = \nu_1^i[1] = \frac{\rho_i}{\rho_i + r_i}$ will be consistent with the Lemma, since $\forall k \ \nu_0^i[k] \geq \frac{\rho_i}{\rho_i + r_i}$ and $\nu_1^i[k] \leq \frac{\rho_i}{\rho_i + r_i}$.

Proof of Lemma 3.4.3

To prove the submodularity of $U^{DI}(\mathcal{A})$ we show that to prove the property

$$f(\mathcal{A}+a)+f(\mathcal{A}+b)\geq f(\mathcal{A}+a+b)+f(\mathcal{A}). \tag{B.0.38}$$

is equivalent to prove

$$U_a(\omega_a)+U_b(\omega_b)\geq 0 \tag{B.0.39}$$

which is true by assumption on the function $U_i(\omega_i)$: since $U_i(\omega_i)=0$ for $\alpha_i=1, \beta_i=0$, we have $U_i(\omega_i)\geq 0, \quad \forall i\in\mathcal{N}$ for the optimized α_i^*, β_i^*.

Proof of Lemma 3.4.4

The function is the sum of two terms, to prove the first one is sub-modular one can follow the same steps in Appendix B. For the second term, it is enough to show that, for any i, $-\Upsilon(\deg_{\mathcal{E}}(i))$ is a sub-modular function of $\mathcal{E}$. The function is clearly sub-modular since is a concave function of the nodal degree, and from this we can conclude the second term is a positive sum of sub-modular functions, hence sub-modular. To prove the equivalence of the two optimizations in (3.4.26)-(3.4.28), we first note that for any $\mathcal{E}$ that satisfies the constraints in (3.4.26), the second term of the objective in (3.4.28) is equal to 0 and the two objectives are equal. It follows that we simply need to verify that no set of edges that violates the constraint on the nodal degree would be the optimal solution for (3.4.28). To show this, we note that any unfeasible set of edges (according to (3.4.26)) can be made feasible by removing some edges. For M large enough, i.e. $M > K \max_{ij} u_{ij}$ it is relatively straightforward to verify that such remotion of edges would improve the objective, preventing an infeasible solution for (3.4.26) to be optimal for (3.4.28), and this concludes the proof.

Proof of Lemma 3.4.5

We want to prove

$$U^{GT}\left(\mathscr{C}^G\right) \geq \alpha U^{GT}(\mathscr{C}^{OPT}) \tag{B.0.40}$$

where $\alpha = \frac{1}{\min\{L_{\text{eff}}, \frac{K}{2}\}} \frac{K-1}{K-\min\{L_{\text{eff}}, \frac{K}{2}\}}$ and $L_{\text{eff}} \leq |\boldsymbol{\phi}_k|$ is the largest test size returned by the greedy algorithm. We also rewrite

$$U^{GT}\left(\mathscr{C}^G\right) = (K - |\mathscr{C}^G|)U_{\mathscr{C}^G}$$
$$U^{GT}(\mathscr{C}^{OPT}) = (K - |\mathscr{C}^{OPT}|)U_{\mathscr{C}^{OPT}}$$

To prove the claim we look at the graph obtained by the union of the cycles in the optimal and the greedy solution. Since in each of the solution, no node can be in two cycles it follows that in the obtained graph, no node can be in more than two cycles. Let us start by assuming there is a cycle $\mathcal{C}$ with associated utility $u_{\mathcal{C}}$ in the optimal solution that does not share any node with the greedy solution. This means

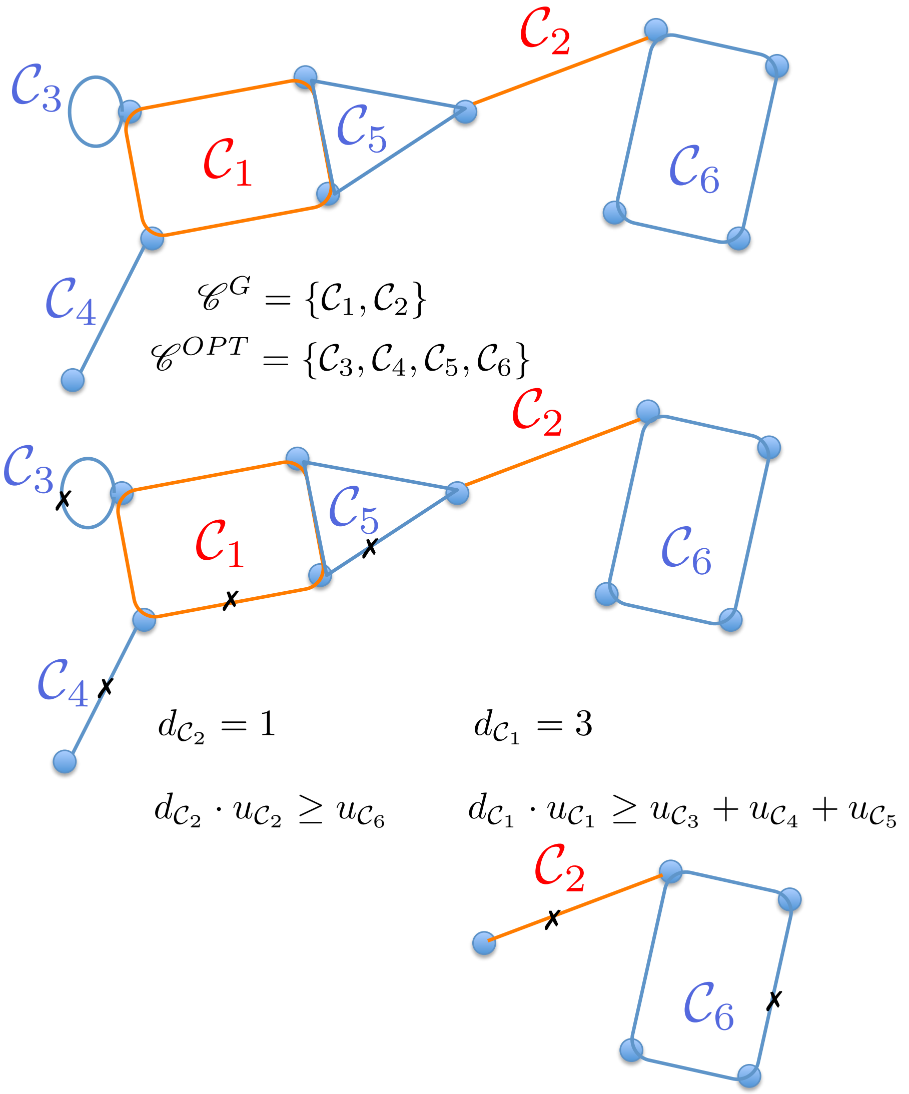
$$\frac{U_{\mathscr{C}^{OPT}} - u_{\mathcal{C}}}{K - |\mathscr{C}^{OPT}|} \leq u_{\mathcal{C}} \leq \frac{U_{\mathscr{C}^G}}{K - |\mathscr{C}^G| - 1} \tag{B.0.41}$$

$$\rightarrow U_{\mathscr{C}^{OPT}} \leq (K - |\mathscr{C}^{OPT}| + 1)u_{\mathcal{C}} \leq U_{\mathscr{C}^G}\frac{K - |\mathscr{C}^{OPT}|}{K - |\mathscr{C}^G|} \tag{B.0.42}$$

where (B.0.41) follows from the fact that adding $\mathcal{C}$ to $\mathscr{C}^{OPT} \setminus \{\mathcal{C}\}$ improves the objective but would no improve the objective for the greedy solution. From (B.0.42) we could then conclude $|\mathscr{C}^G| > |\mathscr{C}^{OPT}|$, since for $|\mathscr{C}^G| \leq |\mathscr{C}^{OPT}|$ we would find from (B.0.42) that $U^{GT}\left(\mathscr{C}^G\right) \geq U^{GT}(\mathscr{C}^{OPT})$. This means we can replace a cycle in $\mathscr{C}^G$ with this isolated cycle, to form a set of cycle $\tilde{\mathscr{C}}^G$ whose objective is lower than $\mathscr{C}^G$ by greedy search, and that in light of (B.0.42) we can iterate this process by always picking the cycle to be replaced in such a way that all the cycles in the optimal solution share at least one node with the set of cycles in $\tilde{\mathscr{C}}^G$. Now we have that all the cycles in the optimal solution share at least one node with a cycle in $\tilde{\mathscr{C}}^G$. If instead one has that no cycle $\mathcal{C}$ in the optimal solution is isolated and that there are isolated cycles in the greedy solution, then the set $\tilde{\mathscr{C}}^G$ is formed by removing these cycles from $\mathscr{C}^G$, lowering the objective (by submodularity and greedy search) and one would again obtain that all the cycles in the optimal solution share at least one node with the set of cycles in $\tilde{\mathscr{C}}^G$. We now iteratively remove cycles from $\tilde{\mathscr{C}}^G$ and $\mathscr{C}^{OPT}$, while bounding the loss in performance and therefore obtain the factor approximation we want to prove. We can remove cycles from $\tilde{\mathscr{C}}^G$ in decreasing order of utility and since we know that for each cycle $\mathcal{C}'$ of length L in $\tilde{\mathscr{C}}^G$ there are at most L different cycles in the optimal solution that share a node with $\mathcal{C}'$, by greedy search we have that $L \cdot u_{\mathcal{C}'}$ is greater than the utility given by the cycles in the optimal solution that are adjacent to cycle $\mathcal{C}'$. Let us then define $\hat{\mathscr{C}}^G$ as the minimal subset of $\tilde{\mathscr{C}}^G$ that can cause the removal of all the cycles in the optimal solution when iterating the procedure just described, i.e. the set containing the first $|\hat{\mathscr{C}}^G|$ in decreasing order of utility contained in $\tilde{\mathscr{C}}^G$. Again by sub-modularity and greedy search one can easily find that the objective for $\hat{\mathscr{C}}^G$ is lower than $\tilde{\mathscr{C}}^G$, since if the objective could not be improved by removing a cycle from $\mathscr{C}^G$, then it also cannot improve the objective for $\tilde{\mathscr{C}}^G$ which has utility strictly greater than $\mathscr{C}^G$. At this point we can prove (B.0.40) for $\hat{\mathscr{C}}^G$and this will prove it for $\mathscr{C}^G$. We then use $d_{\mathcal{C}}$ to define the number of cycles in $\mathscr{C}^{OPT}$ that can be removed by removing the cycle $\mathcal{C}$ in $\hat{\mathscr{C}}^G$ and $d \triangleq \max\limits_{\mathcal{C} \in \hat{\mathscr{C}}^G} d_{\mathcal{C}}$. By then iterating our procedure described above we end up having

$$
\begin{aligned}
&(K - |\hat{\mathscr{C}}^G|)U_{\hat{\mathscr{C}}^G} = \frac{1}{d}\frac{K - |\hat{\mathscr{C}}^G|}{K - |\mathscr{C}^{OPT}|}(K - |\mathscr{C}^{OPT}|)d \cdot U_{\hat{\mathscr{C}}^G} \\
&\geq \frac{1}{d}\frac{K - |\hat{\mathscr{C}}^G|}{K - |\hat{\mathscr{C}}^G| - d + 1}(K - |\mathscr{C}^{OPT}|)d \cdot U_{\hat{\mathscr{C}}^G} \\
&\geq \frac{1}{d}\frac{K - |\hat{\mathscr{C}}^G|}{K - |\hat{\mathscr{C}}^G| - d + 1}(K - |\mathscr{C}^{OPT}|)U_{\mathscr{C}^{OPT}} \\
&\geq \frac{1}{\min\{L_{\text{eff}}, \frac{K}{2}\}}\frac{K - 1}{K - \min\{L_{\text{eff}}, \frac{K}{2}\}}(K - |\mathscr{C}^{OPT}|)U_{\mathscr{C}^{OPT}}
\end{aligned}
$$

and since $(K-|\mathscr{C}^G|)U_{\mathscr{C}^G} \geq (K-|\hat{\mathscr{C}}^G|)U_{\hat{\mathscr{C}}^G}$ this concludes the proof. We have used the fact that $d \leq L_{\text{eff}}$ and that the function $d(K-d)$ has its maximum in $d = \frac{K}{2}$. Fig.B.0.1 shows an example of the iterative procedure to obtain the bound just derived.



$$\mathscr{C}^G = \{\mathcal{C}_1, \mathcal{C}_2\}$$

$$\mathscr{C}^{OPT} = \{\mathcal{C}_3, \mathcal{C}_4, \mathcal{C}_5, \mathcal{C}_6\}$$

$$d_{\mathcal{C}_2} = 1 \qquad d_{\mathcal{C}_1} = 3$$

$$d_{\mathcal{C}_2} \cdot u_{\mathcal{C}_2} \geq u_{\mathcal{C}_6} \qquad d_{\mathcal{C}_1} \cdot u_{\mathcal{C}_1} \geq u_{\mathcal{C}_3} + u_{\mathcal{C}_4} + u_{\mathcal{C}_5}$$

$$\max\{d_{\mathcal{C}_1}, d_{\mathcal{C}_2}\}(u_{\mathcal{C}_1} + u_{\mathcal{C}_2}) \geq (u_{\mathcal{C}_3} + u_{\mathcal{C}_4} + u_{\mathcal{C}_5} + u_{\mathcal{C}_6})$$

$$|\mathscr{C}^{OPT}| = |\mathscr{C}^G| + (d_{\mathcal{C}_1} - 1) + (d_{\mathcal{C}_2} - 1) \geq |\mathscr{C}^G| \max\{d_{\mathcal{C}_1}, d_{\mathcal{C}_2}\} - 1$$

$$\begin{aligned}(K - |\mathscr{C}^G|) \sum_{\mathcal{C} \in \mathscr{C}^G} u_{\mathcal{C}} &\geq \frac{1}{\max\{d_{\mathcal{C}_1}, d_{\mathcal{C}_2}\}} \frac{K - |\mathscr{C}^G|}{K - |\mathscr{C}^G| - \max\{d_{\mathcal{C}_1}, d_{\mathcal{C}_2}\} + 1} (K - |\mathscr{C}^{OPT}|) U_{\mathscr{C}^{OPT}} \\ &\geq \frac{1}{4} \frac{K-1}{K-4} (K - |\mathscr{C}^{OPT}|) U_{\mathscr{C}^{OPT}}\end{aligned}$$

Figure B.0.1: Representation of the iterative procedure to obtain the factor approximation of the greedy algorithm