Doctoral Thesis

Czech
Technical
University
in Prague

F3

Faculty of Electrical Engineering
Department of Radioelectronics

Distributed algorithms for Wireless Physical Layer Network Coding self-organisation in cloud communication networks

Ing. Tomáš Hynek

Supervisor: Prof. Ing. Jan Sýkora, CSc.
Ph.D.Programme: Electrical Engineering and Information Technology
Field of study: Radioelectronics
January 2017

Acknowledgements

I would like to thank you Mišu for everything ...

I would like to thank my family for never-ending support.

I would like to thank my supervisor Jan Sýkora for all valuable hints, comments and suggestions that made the research described in this thesis possible.

I would like to thank all my former colleagues from DiRaC research group for all disputes, discussions and solutions found. You made this research enjoyable. In alphabetical order:

- Jan Hejtmánek
- Miroslav Hekrdla
- Martin Mašek
- Pavel Procházka
- Lukáš Růžička
- Tomáš Uřičář

I would like to thank researchers from Toshiba Telecommunications Research Laboratory in Bristol, namely David Halls and Will Thompson, for splendid days of our collaboration.

Declaration

The presented doctoral thesis is based on my own research work carried out at Faculty of Electrical Engineering at Czech Technical University in Prague and in my close cooperation with Toshiba Telecommunications Research Laboratory in Bristol, United Kingdom. It has not been submitted in any form for another degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given in the bibliography.

Abstract

Communication networks of present-days are growing in complexity to follow a never-ending demand for fast, smooth, energy efficient and cheap connection. This demand is a driving force of new technologies. Two of them – cloud network concept and Wireless Physical Layer Network Coding (WPLNC) – are in the focus of this thesis. Both together represent a paradigm shift that can bring benefits to the networks of the future.

With WPLNC the communicating nodes are no more separated in orthogonal resources, such as time, frequency or code space. Instead, they are allowed to overlap directly at the level of electromagnetic waves. Overlapping signals are no more considered as nuisance, rather, they form new super-signals that are processed as one. With slightly more complicated signal processing some big advantages in throughput, reliability and/or efficiency can be achieved by this technique.

Cloud network concept introduces an intelligent network, that is able to react on changing conditions. The network is formed by self-aware relay nodes that cooperate to provide communication service to the terminals. The network is assumed distributed and decentralised to be able to react fast and adapt to local variations. WPLNC technique seems to be a favourable scheme for such a network.

In this thesis I focus on distributed algorithms for implementation of WPLNC in the context of cloud network. The algorithms equip the relay nodes with ability to self-adapt and self-organise their WPLNC processing, such that it is aligned optimally across the whole network. Particularly, as an example an algorithm that assigns WPLNC mappings to individual relays is provided. This algorithm is transformed into a form of suitable communication protocol. Some other versions of the algorithm are proposed that focus on more complex situations – when the relays do not perform fair and try to harm the network or when some form of behaviour is dependent on relay node state, such as battery level.

Whenever it is possible not only a mathematical solution of studied issues is provided but also software simulation is shown and more importantly a verification by hardware testing in real conditions is done and presented.

Keywords: Wireless physical layer network coding, Self-adapting algorithms, cloud networks

Abstrakt

Současné komunikační sítě narůstají ve své složitosti, tak aby pokryly neutuchající poptávku po rychlém, hladkém a levném připojení. Tento tlak je hybnou silou nových technologií. Na dvě z nich – Cloudovou strukturu sítě a Bezdrátové síťové kódování na fyzické vrstvě (WPLNC) – se tato dizertační práce zaměřuje. Obě společně představují významný posun v návrhu sítí a přináší řadu významných výhod sítím budoucím.

S technologií WPLNC již nejsou jednotlivé přenosy rozděleny ortogonálně do jednotlivých prostorů, jako je kmitočet, čas nebo kódový prostor. Namísto toho je jim umožněno, aby spolu přímo reagovaly na úrovni elektromagnetických vln. Tato interakce není chápána jako rušení, spíše je považována za nový signál a jako s takovým je s ním dále nakládáno. Za cenu náročnějšího zpracování získáváme značné výhody v propustnosti, spolehlivosti a/nebo účinnosti.

Cloudový přístup k sítím vytváří inteligentní strukturu, která je schopná samočinně a pružně reagovat na změny. Taková sít je tvořena relay uzly, které spolupracují na zajištění spolehlivého spojení mezi koncovými uživateli. Cloudová síť nemá centrální strukturu ani řízení, tak aby byla schopna přizpůsobovat se svižně na místní úrovni. Technika WPLNC je více než příhodná pro takovýto typ sítě.

V této dizertační práci se zaměřuji na distribuované algoritmy pro implementaci techniky WPLNC v prostředí cloudových sítí. Algoritmy vybavují relay uzly schopností přizpůsobit se a zorganizovat své zpracování signálů tak, aby bylo dosaženo optimálního výsledku v celé síti. Příkladem budiž jeden z algoritmů přidělující jednotlivým uzlům WPLNC mapování. Tento algoritmus je následně transformován do podoby příhodné pro komunikační protokol. Další jeho komplikovanější verze jsou dále diskutovány. Například situace, kdy relay uzle nehrají v duchu fair-play a snaží se poškodit síť, nebo situace, kdy je chování uzlu ovlivněno jeho stavem, například docházející baterií.

Kdykoli je to jen trochu možné, práce uvádí nejen matematické řešení problému, ale i jeho ověření pomocí počítačové simulace. Ještě důležitější je, že řada algoritmů je ověřena pomocí hardwarového testu ve skutečných podmínkách.

Klíčová slova: Bezdrátové síťové kódování na fyzické vrstvě, adaptující se algoritmy, cloudové sítě

Contents

Figures

Tables

Chapter 1

Introduction

1.1 Thesis outline

1.1.1 Aims and scope of the thesis

Throughout this thesis I would like to describe my achievements and research outcomes in a field of distributed self-adapting algorithms for WPLNC based networks with relays. My focus is mainly laid on algorithms for distributed WPLNC mapping allocation under various network set-ups, conditions, nodes' local knowledge, etc.. All studied problems are rigorously defined, mathematically described and solved. Whenever it was possible the solution was verified at least by a numerical simulation. In several cases there was also an option to convert the theoretical result into a practical form of a communication protocol. These protocols served as a basis for the most advanced WPLNC based self-adapting network demonstrator known by the times of writing of this thesis.

Although a care was take to make as realistic assumption about the network as was possible there were obviously some significant simplifications. Many times, orthogonally separated signals are used as a simple solution for signalisation, synchronisation or similar problems, proposed signal structures are tailored to meet the requirements of relatively simple networks, time slotting of transmission/reception activity of a node is assumed to be known, initial network synchronisation is assumed to be achieved by some other means and no error protection coding is used. All these simplifications are assumed mainly because they will otherwise hide the core topic of this thesis – distributed self-adaptation of the network – or they will made the analysis impractically complex, mathematically and notationally cumbersome. Nevertheless, all these known issues have a solution. Some of them are relatively easily solvable, some may be hard to solve in optimal way, although some suboptimal solutions may exist. All these identified issues lay far beyond the scope of this thesis. Whenever possible I at least indicate potential solutions and point the interested reader to the relevant sources.

All test-bed verification of proposed algorithms in this thesis was done by my close cooperation with Dr. David E. Halls from Toshiba Telecommunications Research Laboratory, Bristol, the United Kingdom.

My contribution is especially:

- Definition, analysis and solution of a distributed WPLNC mapping allocation problem under perfect information scenario.
- Conversion of this solution into a usable communication protocol.
- Extension of the proposed protocol to far more advanced scenario of more complex cloud network that is able to allocate WPLNC mappings even in the case of changing internal topology.
- Definition, analysis and solution of distributed WPLNC mapping allocation problem under imperfect information scenario - exemplified by two examples of a maliciously behaving relay and of a battery level dependent behaviour.
- Initial attempt to use a non-orthogonal signalisation for a network state estimation.

1.1.2 Contents

The rest of the thesis is organised as follows: The following sections of this chapter summarises my publications and projects I have participated and that are related to the content of this thesis. Chapter 2 briefly introduce and describes the most important area where many of application scenarios for the algorithms studied here comes from – the area of ad-hoc/mesh networks exemplified by (Industrial) Internet of Things ((I)IoT), although the proposed cloud structure goes even beyond nowadays network concepts. The next chapter – chapter 3 – introduces formally the basic concept of Wireless Physical Layer Network Coding (WPLNC) which is an enabling technique for cloud multi-node relay networks. In the chapter 4 an important algorithm for decentralised and distributed WPLNC mapping assignment is presented and thoroughly studied. The following chapter – chapter 5 – extends the application scenario of DLA to more complex and more realistic networks. Some other topics, such as incomplete and/or imperfect knowledge of the network state, are also analysed there. The final technical chapter – chapter 6 – presents an algorithm for network state identification based on the non-orthogonal signalisation, which is a potential way to even more decentralisation of the network. The thesis is concluded by the chapter 7 where the achievements are summarised and future research directions are highlighted.

Whenever it is feasible not only numerical simulations of proposed algorithms are shown but also a verification by true hardware (HW) is done and presented. A care is taken to achieve as much reality as possible to show applicability and usefulness of the underlying theory.

1.2 Publications

Lists of publications both related and unrelated to the thesis follow. The publications in both lists are divided into categories - publications in journals with impact factor, conference publications indexed by Web of Science (WoS), conference publications indexed by Scopus (i.e. not indexed by WoS, otherwise the publication will be shown in the previous category) and other conference publications. The lists and distribution among the categories reflects the actual state at the time of writing of this thesis. A share of authorship is noted for all publications in square brackets.

1.2.1 Publications related to the thesis

Publications in journals ranked by impact factor

- T. Hynek and J. Sykora, “Wireless physical layer network coding in potential presence of malicious relays - incomplete information game approach,” *Electronics Letters*, vol. 51, no. 16, pp. 1292–1294, 2015. [**TH 70%; JS 30%**]

Conference publications indexed by WoS

- T. Hynek and J. Sykora, “Non-cooperative broadcast game for distributed decision map selection of relay wireless network coding processing,” in *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th International Workshop on*, (Darmstadt, Germany), June 2013. [**TH 60%; JS 40%**]

Conference publications indexed by Scopus

- T. Hynek, D. Halls, and J. Sykora, “Practical implementation of cloud initialization procedure for wireless physical layer network coding clouds,” in *European Wireless, 2014. EW. 20th European Wireless Conference*, pp. 1–5, May 2014. [**TH 50%; DH 30%; JS 20%**]

- T. Uricar, T. Hynek, P. Prochazka, and J. Sykora, “Wireless-aware network coding: Solving a puzzle in acyclic multi-stage cloud networks,” in *The 10th International Symposium on Wireless Communication Systems*, (Ilmenau, Gemany), Aug. 2013. [**TU 25%; TH 25%; PP 25%; JS 25%**]

- T. Hynek and J. Sykora, “Initialization procedure of wireless network coding with hierarchical decode and forward strategy in random connectivity networks,” in *Multiple Access Communications* (B. Bellalta, A. Vinel, M. Jonsson, J. Barcelo, R. Maslennikov, P. Chatzimisios, and D. Malone, eds.), Lecture Notes in Computer Science, pp. 13–24, Springer Berlin Heidelberg, 2012. [**TH 65%; JS 35%**]

Other conference publications

- T. Hynek and J. Sykora, "Sequential incomplete information game in relay networks based on wireless physical layer network coding," in *Multiple Access Communications* (M. Jonsson, A. Vinel, B. Bellalta, and O. Tirkkonen, eds.), vol. 9305 of *Lecture Notes in Computer Science*, pp. 47–56, Springer International Publishing, 2015.**[TH 50%; JS 50%]**

- T. Hynek, D. Halls, and J. Sykora, "Hardware implementation of distributed learning algorithm for mapping selection for wireless physical layer network coding," in *Communication Workshop (ICCW), 2015 IEEE International Conference on*, pp. 2127–2132, June 2015. [**TH 40%; DH 20%; JS 40%**]

- T. Hynek, " Sequential inter-relay games in wireless physical layer network coding based networks," in *Proc. POSTER 2015 - 19th International Student Conference on Electrical Engineering*, (Prague, Czech Republic), May 2015. [**TH 100%**]

- T. Hynek, "Game-theoretical approach to self-adaptation of wireless cloud networks," in *Proc. POSTER 2014 - 18th International Student Conference on Electrical Engineering*, (Prague, Czech Republic), May 2014. **Best paper award**. [**TH 100%**]

- T. Hynek and J. Sykora, "Potential games for distributed coordination of wireless network coding based cloud," in *COST IC1004*, (Ilmenau, Germany), June 2013. **Invited for publication in COST Newsletter**. [**TH 60%; JS 40%**]

- T. Hynek, "Performance of hierarchical decode & forward strategy under channel estimation errors," in *Proc. POSTER 2013 - 17th International Student Conference on Electrical Engineering*, (Prague, Czech Republic), May 2013. **Best paper award**. [**TH 100%**]

- T. Hynek and J. Sykora, "Wireless network coding initialization procedure through multi-source automatic modulation classification in random connectivity networks," in *COST IC1004*, (Bristol, UK), Sept. 2012. [**TH 65%; JS 35%**]

- T. Hynek, "An algebraic approach to wireless network coding with hierarchical decode & forward strategy," in *Proc. POSTER 2012 - 16th International Student Conference on Electrical Engineering*, (Prague, Czech Republic), May 2012. [**TH 100%**]

1.2.2 Publication unrelated to the thesis

Conference publications indexed by WoS

- T. Hynek and J. Sykora, "Capacity regions of wireless network coding with hierarchical decode & forward strategy applied on impulse radio

UWB technology," in *2011 IEEE International Conference on Ultra-Wideband (ICUWB 2011)*, (Bologna, Italy), Sept. 2011. [**TH 60%; JS 40%**]

Conference publications indexed by Scopus

- T. Hynek and J. Sykora, "Hierarchical decode & forward strategy in IR-UWB communication systems," in *Proc. Future Network and Mobile-Summit 2011*, (Warsaw, Poland), June 2011. [**TH 50%; JS 50%**]

Other conference publications

- M. Hekrdla, T. Uricar, P. Prochazka, M. Masek, T. Hynek, and J. Sykora, "Cooperative communication in wireless relay networks," in *Workshop 2011 [CD-ROM]*, (Prague, Czech Republic), pp. 1–18, 2011. [**MH 18%; TU 18%; PP 18%; MM 18%; TH 18%; JS 10%**]

- T. Hynek, "Hierarchical MAC capacity in 2-WRC network based on IR-UWB technology," in *Proc. POSTER 2011 - 15th International Student Conference on Electrical Engineering*, (Prague, Czech Republic), May 2011. **Best paper award**. [**TH 100%**]

1.3 Grants

The content of this thesis is based on my participation in the following international as well as local research projects and grants.

- FP7-ICT-2011-8/ICT-2009.1.1: DIWINE Dense Cooperative Wireless Cloud Network, 2013-2015

- FP7 ICT/STREP (INFSO-ICT-248001): SAPHYRE Sharing Physical Resources Mechanisms and Implementations for Wireless Networks, 2010-2012

- FP7 ICT/STREP (INFSO-ICT-215669): EUWB Coexisting Short Range Radio by Advanced Ultra-Wide Band Radio Technology, 2010-2011

- Grant Agency of Czech Republic (GACR 102/09/1624): Mobile radio communication systems with distributed, cooperative and MIMO processing, 2009-2012

- Ministry of Education, Youth and Sports (LD12062): Wireless Network Coding and Processing in Cooperative and Distributed Multi-Terminal and Multi-Node Communications Systems, 2012-2015

- Ministry of Education, Youth and Sports (OC 188): Signal Processing and Air-Interface Technique for MIMO radio communication systems, 2007-2010

- EU COST IC1004: Cooperative Radio Communications for Green Smart Environments, 2011-2014

- EU COST 2100: Pervasive Mobile & Ambient Wireless Communications, 2006-2010
- Czech Technical University in Prague (SGS15/090/OHK3/1T/13): Software Defined Radio Test-bed for Advanced Physical Layer Algorithms, 2015
- Czech Technical University in Prague (SGS14/081/OHK3/1T/13): Wireless Cloud, 2014
- Czech Technical University in Prague (SGS13/083/OHK3/1T/13): Wireless Network Coding based Multi-node Dense Networks, 2013
- Czech Technical University in Prague (SGS12/076/OHK3/1T/13): Physical Layer for Dense Cooperative Communication Networks, 2013
- Czech Technical University in Prague (SGS10/287/OHK3/3T/13): Distributed, Cooperative and MIMO (Multiple-Input Multiple-Output) Physical Layer Processing in General Multi-Source Multi-Node Mobile Wireless Network, 2010-2012

Chapter 2

Current trends in wireless communications

Communications and especially wireless have undergone a big evolution from its origin in the second half of nineteenth century. Or it may be better to say that it has undergone several big revolutions. To name only a few of them – analogue to digital paradigm shift, packet oriented networks, advanced modulation schemes, advanced coding techniques, multiple antennae devices, software oriented radios and networks, etc. All these changes has one common goal – to make the communications better – in every possible meaning of that word – throughput, availability, reliability, efficiency, latency, etc. There is a never-ending demand for wireless connectivity nowadays [1]. A plenty of devices that we use everyday are connected to some communications network – personal computers, laptops, tables, mobile phones. And even more devices will be connected in near future.

This race for a better communications network recently give rise to the fifth generation of mobile networks (5G) [2] and/or (Industrial) Internet of Things ((I)IoT) [3]. In future almost everything, every single electronic device, is expected to be connected to Internet by some means. This enables various applications especially from areas of remote control and sensing. Machine to machine communication may even overgrow human to human traffic.

IoT concept together with 5G networks open new or radically change already existing areas. They give rise to a broad range of application scenarios including [4, 5]: smart wearable electronics that continuously monitors and reports health status [6] and/or sports achievements; cars equipped with plenty of sensors [7,8] that support vehicular to vehicular and also vehicular to infrastructure communications to increase traffic safety, support autonomous vehicles, reduce or avoid traffic jams; smart factories [9, 10], where IoT sensors enable faster, more reliable and safer production; smart homes [11–14] full of sensors and actuators to intelligently control lightning and heating, detecting fire, explosive gasses or water leakage, detecting and avoiding illegal intrusion etc.; smart buildings [15, 16] that are mostly commercial equivalents of smart homes; smart metering network and smart grids [17] for better allocation and distribution of electric energy, water and gas; smart cities [18] are connected smart homes, buildings with smart metering and intelligent transportation systems. There are also some exotic areas of IoT application including smart dust sensors [19] or bio-nano technology [20]. Future applications affect a huge range of industries as communications, utilities, living and leisure time, transportation, healthcare, consumer electronics, and many others [21]. IoT and 5G should not be limited to only a smart network for device to device communication. Human to human communication will also still take place and will gain from novel approaches.

Although both 5G and IoT concepts are already relatively well evolved, even with already proposed standards [4, 5], the opportunities offered by the networks of the future may need new enabling technologies.

In an area of computer networks a nowadays buzzword is a cloud (or more precisely a cloud computing). Google returns over 80 millions of results for this term. Moreover, every well-known cloud computing service, such as iCloud, Dropbox, OneDrive, Google Docs, Amazon AWS, etc., possesses several tens of millions search results each.

The cloud computing is based on providing a service (storage, computational power, applications, etc.) to system users in a pay-for-use basis. Each user can access to desired service from arbitrary device and from arbitrary place over the world at arbitrary time instant. Thus he or she can reduce his/her costs since the burden of computational capability, HW requirements, etc. is transferred to the cloud side. Each cloud service is implemented in client-server way when users are not interested in particular implementation, HW background, technology, etc. at the provider side. They only access via defined interface to the cloud and enjoy provided services. The comprehensive discussion of technology, implementation, pros and cons of this up-to-date phenomenon can be found in vast amount of references, e.g. [22–25].

A similar concept can play a role of an important enabler for the network of the future also for the communications. The concept of cloud communications was originally proposed by the FP7 project DIWINE [26]. Similarly to cloud computing a cloud communication network introduces a distant, self-contained network structure that is transparent from the point of view of the end user and that provides a reliable wireless connection among sources and destinations.

A cloud communication network consist of nodes. Their number can potentially be huge – tens or hundreds of them can be connected. These nodes will be called relays. Users of such a cloud networks are various sensors and actuators on one side and various fusion centres, recording or logging devices, terminals for human interaction etc. on the other side. Physical location of the users (all denoted as terminal nodes for simplicity) really does not matter. They can be outside as well as inside the cloud. Multiple terminal nodes can be collocated in one physical device. Also a relay can be collocated with a terminal node.

Relay nodes can be connected wirelessly or by wires. Wireless connection is especially useful for last few (usually one or two) hops from relays to terminals. This enables mobility of terminal nodes. The core network is most probably connected by wires, since it forms fixed infrastructure of the network. Of course even the core network can be wireless and mobile, there is no restriction.

The goal of the cloud network is to provide a reliable connection between the terminals. The cloud relay nodes can be seen as servers passing the information in various directions to various destinations (although packet oriented and routing based solution should not be anticipated). Similarly to the cloud computing the terminals ask the cloud for the service and are neither interested in the internal cloud structure nor the wants and wishes of the other terminals – they behave selfishly.

To illustrate a cloud network based smart building/city environment imagine a typical smart office building. Each room is equipped by several wireless sensors and actuators. There are temperature sensors and heating control sensors, lighting control, humidity sensors, fire and dangerous gases will be surely detected. It is not only a question of building maintenance, however, all employee devices including smart-phones, laptops and tablets are connected to some form of the network. Employees' RFID badges can be used for access management, there are closed-circuit TV cameras etc. All these devices form terminals for the cloud network. Because of necessary mobility they are mostly wireless. Some of them simple battery powered devices, some properly powered having bigger computational power. The cloud network begins with the set of relays frequently placed in each room, corridor or open-space. These relays form gateways to the cloud network. They handle the communication to and from terminals. Higher level of relays can be placed floor-wise. Several of these relays handle all room-level relays in each floor of the building. Floor-wise relays are fused by building level relays. Some of the communication stays in the building – such as internal communication and building maintenance, some goes out – to telecommunication providers, energy providers, municipality, fire and police departments, etc. Building-level relays are merged by street-level ones that itself are connected to quarter ones and so on. More and more network levels can be added to form a city of connected buildings. Somewhere at various levels there are also fusion centres that collect and process the information, issue commands to actuators that are transferred by the cloud network in the opposite direction, interact with humans, etc.

This is just an example of cloud network based application scenario from the area of smart IoT network. The concept of cloud communication network is of course not limited to this scenario.

Obviously, there are qualitative and quantitative differences that distinguish current networks and the networks of the future. A short list of them, especially within the focus of this thesis, although not necessarily in the order of their importance, is provided

- Internal intelligence of the network
- Ability to self-organisation
- Globally unknown, random and changing topology
- Decentralised and distributed control
- Massively interacting network

Mainly because of complexity of cloud networks formed by many communicating nodes it is almost impossible to control the network from a few central points. Communication overhead and traffic to and from these centres may be huge and may become a bottleneck. Also a decision that is made in a distant location may be done too late compared to one that is found out locally. Networks of the future thus will probably be highly decentralised without any central controlling authority at all. This concept already proved itself successful in the Internet network with multiple routing possibilities. Despite higher reaction times to changes outside it also provides higher reliability by more robustness against failures. Hand in hand with distributivity goes ability to self-adaptation [27] and internal intelligence of the cloud.

Especially the terminal nodes are expected to be moving, their connection to the cloud will change with time. A node that was once connected can disconnect itself in arbitrary time instant. New terminals can join the network freely. Also the cloud infrastructure can change its topology in time because of moving relay nodes or their disconnection because of failures, depleted batteries, service and maintenance etc. The cloud network thus must be able to react to changing connectivity among the nodes. Again, in large scale this issue is more flexibly solved with decentralised network structure. The relay nodes can react locally to fulfil the needs of communication terminals. All the practical network designs and signal processing algorithms will have to utilise only a local knowledge of the node's closest neighbourhood.

Because the cloud network is proposed to consist of many (tens, hundreds or even more) interacting nodes sharing of the same environment each transmission may be a subject of severe interference. Classical approach with orthogonal separation of the user may be very restrictive, significantly limiting the number of the nodes. Coexistence of many selfish terminals may be problematic. Some new approaches are necessary. In this thesis I focus on promising technique of Wireless Physical Layer Network Coding (WPLNC) that is introduced and deeply discussed in chapter 3.

There are many other issues related to communication in these scenarios. For example it may be hard to achieve time and frequency synchronisation in network-wide level in a distributed way [28, 29]. Because of traditional half-duplex limitation the network operation will have to be divided in time slots when some nodes transmit and others listen. To achieve this activity allocation in distributed way may also be hard. Typical IoT devices are simple sensors based on single micro-controller architectures. Such nodes are usually battery powered or dependent on energy harvesting [30, 31] although the whole concept is not limited to simple sensors as terminal devices. To extend battery life the transmission rate is usually quite low. Also the computational power of the node is usually not high. Some examples of typical power consumption for some IoT devices can be found in [31, 32]. For example some current IoT physical layer standards such as Blue-tooth Low Energy or IEEE 802.15.4 pay attention to energy efficiency. As a result, many traditional communication techniques, such as forward error correction, are disabled to avoid any additional energy spent at the transmitter. At the same time, to ensure data integrity, both standards employ Cyclic Redundancy Check (CRC) codes for error detection [33].

Because of all this unsolved or only partially solved issues I assume several simplification through out this thesis. Generally, frequency and time synchronisation is assumed to exist and to be perfect enough to enable wireless communication. Time slotting is given, that is, each node knows when to transmit and when to receive. Forward error correction is never applied, although simple coding techniques can be relatively easily added.

Chapter 3

Wireless Physical Layer Network Coding

3.1 Introduction

From a very beginning of an electrical transfer of an information the most common scenario was (and in many areas still is) a point-to-point communication link, either wireline or wireless. Each communicating pair of transmitter and receiver had dedicated slice of available resource - frequency, time, spreading (or hopping) code sequence, polarization, space, etc. This resource division serves mainly to suppress interferences caused by the other communicating pairs on the link of interest. And better (in terms of the orthogonality) it was done the better performance was achieved (measured in terms of capacity, etc.) since the interference was better suppressed. The interferences from the others were (and still are) the most limiting factor for reliability and speed of such communication.

As the telecommunications become an ordinary matter of an every day life much more complex networks arose, such as a continent spread telephone networks, under-oceanic intercontinental cable connection, radio and television broadcasting services, satellite communications, etc. Although the communication between the information source and its destination hops through multiple other network nodes (possibly all the way around the world) each hop is usually considered as a stand-alone point-to-point link. A role of intermediate network nodes is very simple. Such a node only decides whether the received information is intended to it and if not where to sent it further on. The information from the sources is routed by the network nodes through a web of the junctions to its destination(s). Such a communication network decomposes to a set of distinguishable data pipes through which the information flows towards its place of destination. Any interferences among those pipes (as well as from any other network) are harmful and have to be prevented by the signal processing.

The fact that the routing approach is a viable method how to deal with large and complex nowadays networks is clearly seen from our every day experience with usage of the Internet - the most complex network of the current world. Many of different routing protocols for the Internet as well as for the other networks were proposed. For exhaustive overview there is a plenty of references, e.g. [34–36]. But the question is whether the routing solution is the optimal one for complex multi-node networks or whether we could do better?

To summarise features of routing-based approach I point out two properties that epitomise it:

- Intermediate network nodes do not perform any other processing than receive, possibly store, route and forward.
- The whole network is assumed to be formed by a set of point-to-point links.

Those assumptions are quite limiting and in fact the performance of a network designed under those properties is strictly sub-optimal compared to possible achievements when the network is assumed in its entirety and complexity and intermediate nodes are allowed to perform a bit more complex signal processing.

3.2 Network coding - a way beyond routing

It was a seminal work of Ahlswede, Cai, Li and Yeung [37] at the beginning of this millennium that introduced an idea of Network Coding (NC). In the paper they propose a fundamental communication paradigm shift. Its impact on the theory of communications was figuratively speaking comparable to the impact of the Theory of Relativity on the Newtonian physics. Intermediate network nodes were now allowed to perform some combining operation on the incoming data streams rather than simple storing and forwarding as in the routing-only world. This small change had dramatical effect on the design of complex networks.

When the network nodes are allowed to combine incoming data it gains higher total throughput at reasonable price of higher demands on node capabilities. To illustrate this feature I borrowed from [37, 38] a classical illustrative example of NC - a wireline butterfly network, see Fig.3.1.

A source S produces information packets (denoted $x, y, z, \cdots$) that have to be multicast to both destinations D_1, D_2. Labels close to the links denote which packet passes through the given link, $-$ denotes the idle link in the given instant to avoid collisions and $\cdot, \cdot$ denotes that multiple, two in this case, resources are used for transmission. Left-hand side network shows a typical routing solution. It can be easily seen that the routing delivers three packets (x, y, z) by using two resources for transmissions, which is the throughput of one and half packet per resource. It is compared to the NC solution shown by the right-hand side network. Here the middle node is able to perform a signal processing function f upon the incoming packets. This function cannot be arbitrary, but it must allow all the destination to recover the remaining packet from it. For example a XOR function (denoted $\oplus$) can be used in this example since $x \oplus (x \oplus y) = y$ and $y \oplus (x \oplus y) = x$ can be performed in appropriate destination to recover the other packet. The NC solution now offers the delivery of two packets using a single resource, which is a significant

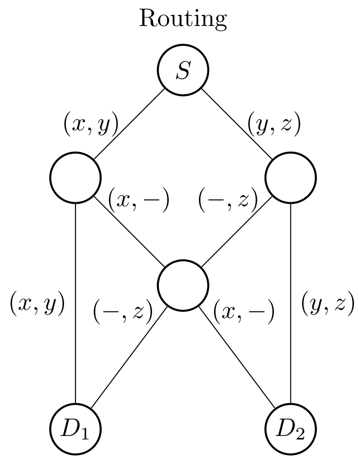

Figure 3.1: Butterfly network - routing vs. NC.

throughput gain over routing solution. Notice that dedicated wires (individual links) offer an ability to send different packet over different links (in different directions) simultaneously. In wireless networks this can be achieved only by orthogonal separation of transmissions such as frequency, time, code or spatially by terms of beam-forming.

The NC is typically applied on upper layers of the communication system such as link or medium access layers. And typically operates on bits or symbols of the packets. Sometimes the NC principle is applied also to wireless networks, however, on upper layers of networks with established orthogonal wireless links. Which is significantly different from Wireless Physical Layer Network Coding (WPLNC) that I will introduce later.

The paper [37] shows that this NC solution is optimal in the sense of maximal throughput. The solution achieves a bound given by the Ford-Fulkerson Theorem of maximal flow through the network [39]. It also shows that the NC can achieve this bound in even much more complicated networks and that the routing solution is unable of it. Although this paper mainly deals with single source multi-cast problem (transmission from single source to all of multiple destinations) it also at least discuss the application of the NC in general multi-source networks.

To formalise the NC processing I assume a network to be an acyclic graphs that consists of vertices V formed by the network nodes and edges E which are links or wires in between the nodes.

A NC operation at some node $v \in V$ is generally denoted by

$$y^v = f_L^v(x^v), \tag{3.1}$$

where f_L^v is a local encoding function applied at node v, x and y represent inputs and output to/from this node, respectively. The local encoding function combines (all or a subset of) incoming packets to one output. It is also called a network coding function. A typical example of the network coding function is the XOR function.

At arbitrary node $v \in V$ (and especially at the final destination, since it its of the most interest) we can also express the global NC processing by

$$x^v = f_G^v(a, b, \cdots z), \tag{3.2}$$

where $a, b, \cdots z$ denotes packets originating at the source node(s), x^v is the input to the node v and f_G^v is a global network coding function. Note that f_G^v appertains to the node v and other nodes $u \in V$ may have generally different global encoding function f_G^u. The global network coding function describes how are the incoming packet created from the original source packets. Some of the incoming packets can already itself be network coded packets created by some nodes closer to original source nodes.

It is clear that while the local encoding functions represents the NC operation at one given node the global encoding function covers all the NC applied on the source packets on their way to the node v. When the node v is a destination the $f_G^v(\cdot)$ shows how are the packets coded by the network after passing through the it to their final destination. Since $f_G^v(\cdot)$ is a function the destination node v is able to decode the applied NC if and only if there is available an inverse function to $f_G^v(\cdot)$, which I denote $f_G^{v,-1}(\cdot)$. A process of decoding of the NC coded source packet is thus done by application of this inverse function to the received packets

$$(a, b, \cdots z) = f_G^{v,-1}(x_i^v). \tag{3.3}$$

3.3 Current network coding state-of-the-art

Since the publication of the seminal NC paper [37] in 2000 a plenty of publications has emerged, with both theoretical as well as practical results and implementations. Paper [40] restricts the network coding functions to be the linear ones. This provides very useful tool since the linear NC operations can be easily rewritten in form of the matrix equations, e.g. the local network coding function can be expressed as

$$y^v = \mathbb{X}_L^v x^v, \tag{3.4}$$

where x^v and y^v are appropriate representations of the node v's inputs and output, respectively, and $\mathbb{X}_L^v$ is a matrix equivalent of the local encoding function $f^v(\cdot)$. Note that elements of vectors and matrices are from some finite field $\mathbb{F}_p$ of order p, such as a binary field with elements 0,1.

This linear NC approach is formalised in terms of linear algebra in [41]. Invertibility of the global network coding functions (respectively of their matrix equivalents) can be verified through the matrix rank and the decoding functions (the inverse matrices to $\mathbb{X}_G$) can be found by the tools of the linear algebra.

Although the restriction to only linear encoding functions is still optimal for solving single source multi-cast problems, as was shown in [40]. The paper [42] showed that the linear NC is insufficient for solving general complex network of many sources. It presented several "diabolical" network topologies where the linear NC fails.

The NC originally proposed for acyclic network can be extended also to networks with cycles leading to convolutional network codes [43]. Principles of NC are also extended to provide generalised error protection in network-wide sense [44, 45].

The NC principle brings also other advantages not only the throughput gains. It thus propagates to many various areas of communication problems. Just to mention at least a few of them: communication reliability [46], information secrecy [47], distributed information storage [48], energy efficiency [49], data dissemination [50], etc. In fact Google returns over 400 000 results for the term NC and IEEE Xplore over 6000 conference and journal papers dealing with the NC in a plenty of research areas.

Albeit the fact that the previously mentioned paper are mostly theoretic ones there is also a huge number of practical implementations of the NC. The paper [51] defines a protocol named COPE, for application of the NC in wireless networks. Other popular application of the NC especially for mobile devices are for example [52, 53].

An excellent survey of history and achievements of the NC with over 300 relevant references is given in [54].

There is still a question how to assign the individual local network coding functions to the network nodes. A linear information flow algorithm is presented in [55] that is able to find the NC solution in polynomial time given the network topology. The second approach is based on a random selection of the local network coding functions rather than on their precise hand-crafting. Although it may seem strange that randomly selected functions can form invertible global network coding functions at the destinations the paper [56] shows that it is possible for the multicast network problem provided that the order of finite field is sufficiently high. Nevertheless, typically it is assumed that network coding functions are somehow assigned and somehow known to destinations such that they are able to invert the network code. Truly distributed and decentralised solution or even verified communication protocols are rare.

3.4 Wireless Physical Layer Network Coding

An idea of the Wireless Physical Layer Network Coding (WPLNC) was firstly proposed probably in a paper by Shengli Zhang *et al.* [57] in 2006, i.e. six years after the origin of the NC. Although several applications of the NC for wireless networks existed that days this seminal paper originally moved the NC principle directly towards the physical layer (PHY) of the wireless network as it was working with superposed electromagnetic waves.

Since the WPLNC is still emerging and evolving research area an unified theory as well as sufficiently generalised and stabilised terminology is still missing. Many research groups around the world use the different terms and shortcuts to describe the similar or closely related things. A term Physical

Layer Network Coding (PLNC) is mostly used in the research community for this principle [57] or a PNC as in [58, 59], also an Analog NC is closely related to this problematic [38, 60].

Throughout the whole thesis I will solely use the longest term Wireless Physical Layer Network Coding (WPLNC), since the PHY exist in wireline as well as in wireless scenarios and it is the fact of application of the NC directly at the wireless PHY that distinguishes it from the pure NC.

WPLNC also allows the intermediate network nodes to combine incoming transmissions into an output signal. The most important difference to NC is where (at which layer) the combination of the incoming transmissions occurs. In the case of the NC there are still distinguishable data lines leading to a node. Because of the nature of the wireless communication medium such lines do not exists but must be created. A common method is to separate individual wireless transmission into orthogonal subspaces of available resource – time slots, frequency bands, code space, polarisation, space, etc. Obviously the orthogonal separation of transmissions is also possible in wireline networks, however it is mainly used to increase the number of users simultaneously using the same wire. In the wireless environment the signals, when not separated into orthogonal subspaces, naturally superpose. Incoming electromagnetic waves are summed up (constructively or destructively) at a receiver antenna. The receiver observes one changing electromagnetic field and is unable to find out which source causes its changes.

This space of superposed electromagnetic waves is place where WPLNC is applied. A mixture of signals is created for free by the nature. Similarly to NC also WPLNC provide benefits over classical orthogonal routing as illustrated by Fig.3.2. A network that consists of three nodes - two of them are terminal nodes T_1 and T_2 which are collocated sources and opposite side destinations and wish to mutually exchange the information. Since there is no direct link available between the terminals the communication must proceed via the intermediate relay node R. To illustrate the benefits of the WPLNC I assume that the communication occurs in the same resource subspace and the orthogonal separation of different transmissions is used only when absolutely necessary. A separation into the time slots is assumed for this example, although it can be done in whatever other resource: different frequency bands, orthogonal spreading codes, etc. Different (orthogonal) pieces of resource are denoted by a different color.

A routing solution for this network is depicted in the upper part – it takes 4 time slots to deliver 2 packets (bits/symbols/etc.). The throughput is thus 1/2 packets per channel use. The NC approach is depicted in the middle part – the sources use separated time slot to avoid a collision when communicating with the relay, each individual packet are thus decodable separately and can be combined together by a network code at the output of the relay. The throughput increases to 2/3 packets per channel use since one common resource is used for deliver by the relay. Finally the lower part depicts the WPLNC solution. Notice mainly that communication from both sources to the relay occurs inside the same time slot. The combination of packets that

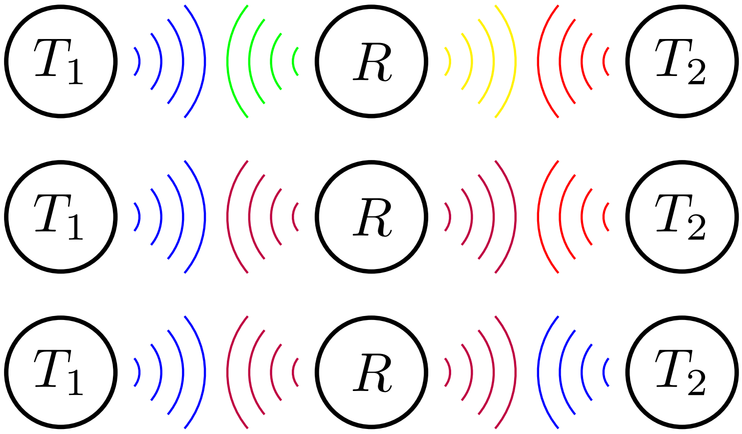

Figure 3.2: Comparison of techniques for wireless communications.

was artificially made by the relay in the case of the NC is done naturally and for free by the wireless environment. This inherent combination property is the source of the throughput gain since the WPLNC solution achieves a throughput of 1 packet per channel use.

Basically, WPLNC can be seen as NC applied on directly overlapped electromagnetic signals, where application means processing of the received signal by some form of function, either linear on non-linear. There are other important differences between WPLNC an NC caused by the nature of the channels. NC is mostly targeted on wireline communications where channels are quite stable, interference free and error-less. Those properties arise from the nature of copper wires or optical fibres. But none of them holds true for the wireless channel. There is almost nothing as unstable, changing, erroneous and full of various interferences as the wireless channel.

Wireless channel causes such phenomena as attenuation of the signal, multipath propagation, frequency selectivity, phase rotation, time delay, etc. Many of them can be fought with appropriate channel state estimation or by the design that is invariant to those phenomena. This may be quite hard in multi-node network and it is an area of intensive research nowadays [28,61–64]. When not coped properly those wireless channel effect can have significant influence on the performance of WPLNC as was shown for several particular examples of the WPLNC [65–68].

3.4.1 Relaying strategies

There are several various forms of signal processing performed by the relays that are covered by the term WPLNC. Relay processing is relay's input-output relation. The local network coding function is NC equivalent of this processing. Relay processing is traditionally called a relay strategy and there are basically three main families, depending on what level of decision is made by the relay about the incoming signals. Since WPLNC problematic is still evolving the following list may not be complete and exhaustive enough.

- Amplify and Forward (AF) family – when no decision is made at all, relay only multiplies the whole superposition and broadcasts it to the neighbouring nodes. This relaying strategy family is represented mainly by Amplify and Forward [69] and Analog NC [60] strategy. Those two terms almost totally coincide. Although this relaying strategy is a very simple one its main drawback is a propagation of an amplified noise through the network.

- Compress and Forward (CF) family – when only a partial (quantised or compressed) information is send to the neighbours. Relay is not making full and hard decision about the incoming signals. Methods from this family are usually named Compress and Forward [70] but other names such as Estimate or Quantize and Forward are often used [71, 72]. A common property of the CF methods is an application of general non-linear function to compress the information contained in the received superposition of the signals.

- Decode and Forward (DF) family – when some kind of hard decision is made by the relay. This is for sure the widest family of relay strategies. There are several specific methods how the decision is made, namely:

  - Joint Decode and Forward – which is closely related to the NC transfered to wireless [73,74]. The relay decodes all of the individual signals from the observed superposition separately and applies a combining function (a network code) upon those estimates. In fact the wireless environment has restrictive impacts on the performance of this strategy since JDF is trying to convert the situation back to dedicated channel per each source as with NC. The source transmission rates has to be set to enable the relay to reliably decode each of them, otherwise the network coding will be applied on erroneous data.

  - Hierarchical Decode and Forward – originally proposed in [75] and deeply elaborated in [76]. In the opposite to the JDF it does not distinguishes the individual source but processes directly superimposed codewords – so called hierarchical codewords. This property offers a capacity gains over the JDF especially in the high SNR regimes.

  - Denoise and Forward – scheme proposed in [74]. This scheme is very similar to HDF although in the subsequent works [58, 77] it is mainly focussing on symbol by symbol adaptive relay processing dealing with the wireless channel parametrization.

  - Compute and Forward – scheme proposed in [78] that directly processes the PHY superposition of the signals but utilising properties of lattices [79]. Currently this is one of the most elaborated relaying strategy for WPLNC. Although there were originally issues with complexity of this strategy (mainly because of dimensionality and decoding techniques) [80], currently there are some approaches that overcome it [81, 82].

There are many publications that provide an overview of the relaying strategies, some of them discuss the properties of the strategies including the evaluation of the capacity for various channels. I refer the interested reader to plenty of sources [38, 57, 59, 64, 69, 70, 73, 83]. An unified and exhaustive survey of WPLNC techniques similar to NC one is probably still missing.

Throughout this thesis I focus on DF relaying strategies only. With these strategies the relays perform some hard decision when creating their output signal. This thesis is heavily influenced by Hierarchical Decode and Forward and Denoise and Forward strategies, although all work that will be presented is directly or with only minor modifications applicable to any type of DF relaying. All algorithms for distributed self-organisation of the relay network operate with WPLNC encoding functions or mappings, which can be seen as abstract description of relay's combining function. It is important to highlight that the proposed algorithms can also be easily used with the Compute and Forward framework.

A DF strategy, assumed as the background technique, works as follows. Any step of DF relaying can be described on a network that consist of three elements – information sources, relays and information destinations. Sources produce information. From the point of view of particular relay node the sources can be either true source node producing new, original and independent information or some other relay whose output is already formed by WPLNC. Nevertheless it still can be considered as signal sources. Relays process incoming signals by DF WPLNC to produce their output signal. The recipients of relay output signals are either true destinations interested in true source data or the next set of relays. For simplicity I first focus on WPLNC description with the true sources and the true destinations.

Each source produces its information independently of any other network node. The information in a form of bit stream b_S is mapped to code-symbols/code-words c_S which are modulated to electromagnetic signal s_S by any means of modulation technique. Because of the wireless nature and since the WPLNC is applied at the physical layer the electromagnetic signals from the sources are allowed to overlap. They create a superposed signal $\sum_i s_{S_i}$ that is received and processed by the relay. Particular form of the superposed signal does not depend only on modulations used by the source but crucially on particular channels. A relay that receives the signal superposition applies a WPLNC function on it to create WPLNC encoded bits or symbols $\bar{c}_R = f\left(\sum_i s_{S_i}\right)$, where f is the WPLNC function or, as I will call it, the WPLNC mapping since it maps input superposition to the relay output. Relay output bits/symbols are modulated to relay's output electromagnetic signal s_R. One or multiple signals from relay(s) are received by destinations. Each of them recovers the sources data by application of inverse WPLNC functions f^{-1}. This processing is illustrated by Fig.3.3.

Particular implementations of DF relaying differ in a way how the superposition is dealt with. Particularly Hierarchical DF [76] does not decide about

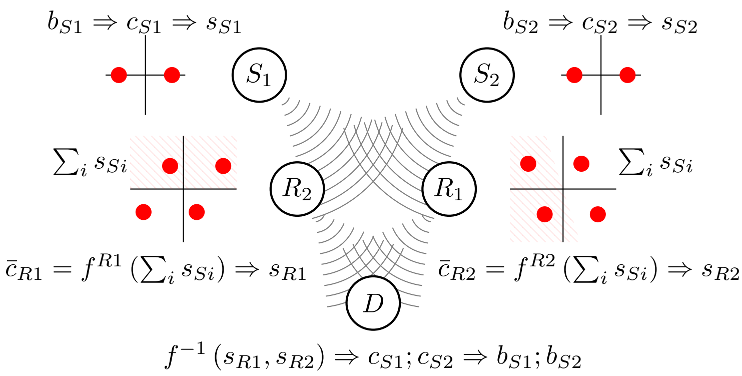

Figure 3.3: Illustration of WPLNC processing.

the individual source codewords (symbols, data, packets ...) at all. The only decision that is made by the relay is done at so called hierarchical level – an artificial "super-signal" that encapsulates all incoming signals. The decision at the relay for a two source example is given by

$$\bar{c}_{AB} = \arg\max_{c_{AB}} \mu(c_{AB}) = \arg\max_{c_{AB}} \mu\left(\bigcup_{c_A, c_B : f(c_A, c_B) = c_{AB}} \{c_A, c_B\}\right), \qquad (3.5)$$

where c represents packets/codewords/symbols/bits, subscript AB indicates hierarchical symbol jointly representing both sources A and B while subscripts A and B indicates single source, either A or B, μ is some decision-making metric. Notice that the decision is a hierarchical signal that maximises the metric and the decision is made upon all pairs of c_A, c_B that are mapped by WPLNC mapping f to hierarchical signal c_{AB}.

In comparison another relay strategy Joint DF [74] tries to estimate both individual sources from the received observation

$$[\hat{c}_A, \hat{c}_B] = \arg\max_{c_A, c_B} \mu(c_A, c_B), \qquad (3.6)$$

the relay output is made by applying a WPLNC mapping f a posteriori on individual estimates $\hat{c}_A, \hat{c}_B$, such that $\bar{c}_{AB} = f(\hat{c}_A, \hat{c}_B)$.

Both of these relaying strategies offer a throughput advantage over orthogonally separated communication. A communication rate $\mathcal{R}$ of a single source in orthogonally separated noisy channel is limited by the channel capacity given by Shannon's theorem [84]. Capacities of two orthogonally separated users are depicted on the left-hand side of Fig.3.4. When both share the same resource non-orthogonally a particular example od the maximum communication rate is given by capacity region [85] shown in the middle of Fig.3.4. Notice that the maximum rate of a user depends on the actual rate of the other. When both wish to communicate fast the overall sum rate is crucially limited. This capacity region also corresponds to the best available performance of JDF relaying since it tries to estimate both sources separately. On the other hand HDF offers a rectangular capacity region, shown on the right-hand side of

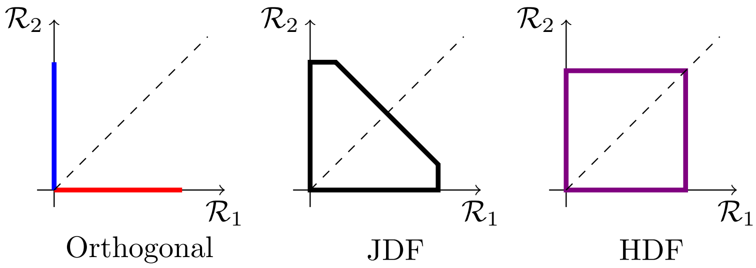

Figure 3.4: Capacity regions.

Fig.3.4. This region may [76] extend beyond the JDF one. And especially the sum rate can be highly increased. This is because HDF does not estimate incoming signals separately but rather jointly in the form of hierarchical signal. It is important to remind that there is still a common throughput advantage of any WPLNC relaying when the output signal is transmitted by the relay. Since it is a combined signal of the inputs it saves transmissions compared to orthogonal separation where each input signal must be send independently.

WPLNC not only combines incoming transmission into one outgoing but it also offer a chance to compress cardinality of the outgoing stream, similarly to NC. For simplicity assume WPLNC processing of source symbols from M sources each using integer-size alphabet of cardinality N. Usually, cardinalities that are some integer powers of two are used because of practical reasons. When no cardinality reduction is done the relay output cardinality is a product N^M. The corresponding WPLNC mapping is denoted as a full one. There is no reason to use higher cardinality. WPLNC mapping that produces symbols from alphabet of cardinality N, the same cardinality as used by the sources, is denoted as minimal one, abbreviated as MIN. Mapping whose symbols are from cardinalities between these bounds is denoted as extended one, abbreviated as EXT. WPLNC mapping whose output has the cardinality even lower than N is also possible, such a mapping is denoted as lossy.

Whenever a destination node (the one that wishes to decode WPLNC mapping and recover original source data) receives a WPLNC mapping with lower cardinality than that corresponding to full mapping it is unable to perform WPLNC mapping inversion. Since a part of information is missing due to compression connected with lossy, minimal or extended mappings. The missing information must be provided from some other source – typically from other relays in the network. If there is only one relay serving the information to that particular destination it must use full WPLNC mapping. However, in cloud networks (and generally in any mesh or sensor networks) connections to multiple relays are typical, such that multiple ways from sources to destinations almost surely exist. This multi-way property also allows utilisation of WPLNC throughput advantages. The destination to be able to recover original source data must receive a sufficient number of WPLNC encoded streams. Moreover, the WPLNC encoding must be such that there exists the inverse function to it. All mappings that together form invertible observation are called invertible WPLNC mapping. Ways to check if the mapping are invertible are described later when particular forms of WPLNC mapping descriptions are given.

A direct reception of source node by the destination node can be formally described also as a WPLNC processing – corresponding WPLNC mapping is identity.

There is a question what and how WPLNC mappings shall be assigned to individual relays within the network such that any destination is able to recover all original source data it is interested in. It can be relatively easy solved when there is a central authority with global knowledge of the network. However, this is not the case for distributed cloud networks. NC approaches with random selection of WPLNC mappings are possible, however, they do not avoid a non-invertible results. In this thesis I provide game-theory inspired algorithms that assign WPLNC mappings such that they are always invertible. Moreover, the mapping allocation is also optimised with regards to given utility function. The algorithm is presented in chapter 4 and its realistic demonstration is shown in chapter 5.

By compression of cardinality the relay not only improves throughput it also behaves energy efficiently. From point to point communications it is well known [86] that error performance (measured by BER) of at least linear modulations such as PSK, QAM, etc. is directly related to the minimal free distance. Thus having a transmission power constraint the lower cardinality modulations exhibit better error performance. Or vice-versa the same error performance is achieved with lower transmission power for lower cardinality modulations.

Mainly due to the simplicity of description and clear notation I will restrict myself to symbol-by-symbol WPLNC processing of uncoded data streams. Although this may seem highly unrealistic and limiting it in fact does not limit applicability of the proposed algorithms. In possible real deployment of cloud networks a plenty of battery powered or energy harvesting devices may be used which will not have enough computation capacity and planned battery life for some advanced coding technique as was already mentioned and as is already done in some IoT standards [33]. Nevertheless, simple algebraic codes such as Hamming are easily and directly applicable with any relaying strategy of WPLNC. A direct design of advanced codes that are good from both WPLNC and error protection point of view is provided in [76]. The authors show the possibility of serial concatenation of two codes - the first one serving for the error protection that can be arbitrary well known channel code such as Turbo or LDPC and the second one is suitable for WPLNC.

The WPLNC mapping selected by the relay should not only optimise throughput and energy efficiency. It must also perform well in wireless channel. To simplify the description assume that two sources communicates via a relay node. Each source node produces symbols q_i that are modulated into a signal $s_i = s(q_i)$. The superposed signal received at the relay

$$y = h_A s_A(q_A) + h_B s_B(q_B) + w, \tag{3.7}$$

where h_A, h_B are appropriate channel (for simplicity we assume frequency flat

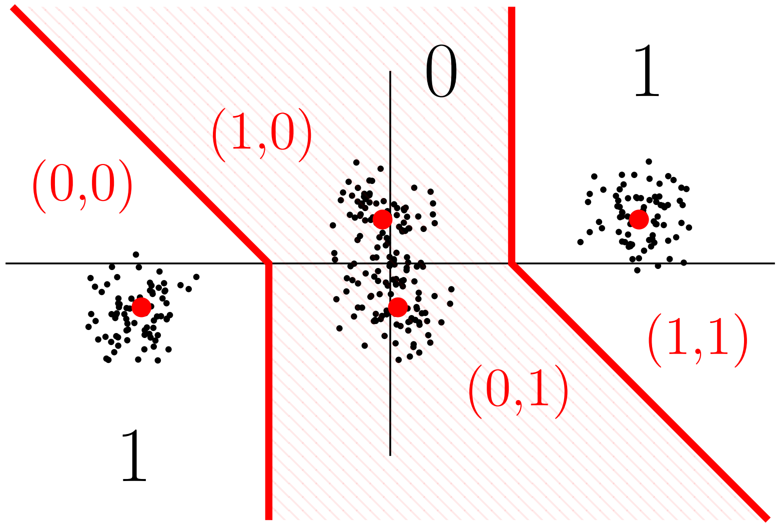

Figure 3.5: WPLNC in wireless channel.

block fading channel – described by one complex coefficient) and w is AWGN. Using signal space representation of individual symbols and having knowledge of the channels h_A, h_B and having metric μ defined the situation can be analysed using constellation diagrams and WPLNC can be transformed into decision region. Fig.3.5 shows superposition of two BPSK transmissions over random realisations of wireless channels. Particularly, $h_A = 1$ and $h_B = e^{j\frac{\pi}{9}}$. Noiseless superposed constellation is shown by red dots with appropriate pairs of source symbols (q_A, q_B) labelled nearby. For these particular channels, minimal distance decision metric and particular WPLNC mapping the decision regions are highlighted. WPLNC output values are shown by black number for each region. Notice that WPLNC mapping applied is the minimal one because two times one input bit is mapped into one output bit.

More importantly, notice that WPLNC applied for this particular situation has very favourable property. Neighbouring received signals that belong to either $(0, 1)$ or $(1, 0)$ are mapped into the same WPLNC output bit. It means that any erroneous decision between $(0, 1)$ and $(1, 0)$ is masked by WPLNC.

However, for different channel states this particular WPLNC mapping may be forced to distinguish points of superposed constellation that are close to each other. This may be hard especially for some particular channel states and low SNR conditions. It may lead to erroneous estimates of symbols. Similarly, when some other WPLNC mapping is applied to superposed constellation in Fig.3.5 it may distinguish between points corresponding to $(0, 1)$ or $(1, 0)$, which may again lead to erroneous estimates.

Any incorrect WPLNC symbol estimate may lead to symbol and bit errors at final destinations since this error propagates throughout the whole network and affects also the other source data contained in the erroneous symbol. Especially in the case of uncoded transmission or when only simple codes are used. This highlights the necessity to match WPLNC mapping used by the relays also to actual channel conditions. The situation may be different when advance coding techniques are employed, however, this is beyond the scope of the thesis.

This superposition behaviour is an important differentiator between WPLNC and NC. Error-free processing can be achieved by various approaches. An application of pre-rotation (or generally pre-distortion) of the signal at the transmitter side needs perfect knowledge of the channel (the own one as well as the channel of the other sources) at the transmitter side which is highly complicated especially in distributed networks.

An adaptive WPLNC mapping at relay node is mostly elaborated in [58,77]. Although no adaptation at the transmitters is necessary the errors can be erased only by the price of increased relay output cardinality. The scheme is also elaborated for simple networks only without any focus on invertibility issues. Moreover, this scheme is quite sensitive to the channel estimation errors as shown in [66]. Parametric invariant designs of modulation and codebooks are deeply elaborated in [68].

3.4.2 Representation of WPLNC mappings

Because this work mainly deals with proper design and assignment of the WPLNC mappings to each network node to establish reliable and high performance communication among the sources and the destination I will discuss first the ways to describe relay's input-output relationship. Several notation will be provided. Each of them offers different point of view on the problematic and can grant specific insight into the WPLNC processing.

All representation equivalently describe the same WPLNC processing. When channel characteristics are known and decision metric is selected they can be together transformed to constellation space decision regions for processing by particular DF relaying strategy such as Eq.(3.5) or Eq.(3.6).

Function

A description of relay processing as a function or a mapping f is a top-level point of view. The relay operation is described as a transformation from input signals/codewords/code-symbols/bits to output entity. It can be a many to one relation.

The j-th relay's output is given by

$$\bar{c}_j = f_L^j(c_a, c_b, c_c, \cdots), \tag{3.8}$$

where $\bar{c}_j$ is the output, $c_a, c_b, c_c, \cdots$ are inputs. With this description WPLNC operation is the same as that of NC. The space of the function's input and output can have various forms, e.g. bits from $\{0, 1\}$, code symbols from $\mathbb{N}$ or $\mathbb{F}_2^n$, codewords from $\mathbb{N}^L$ etc.

The most common example of the WPLNC function is the XOR function which is exemplified for two inputs to the relay R_j by

$$f_L^j : \mathbb{F} \times \mathbb{F} \to \mathbb{F} \qquad \bar{c}_j = f_L^j(c_k, c_l) = c_k \oplus c_l, \tag{3.9}$$

where $\mathbb{F}$ denotes the underlying field from which the signals are drawn which also defines how is the $\oplus$ operation applied on the elements from this field. Note that the output space dimension is the same as the input space - the XOR WPLNC mapping is the minimal one.

Similarly, the inversion done at the x-th destination can be expressed by appropriate inverse function or mapping

$$\hat{c}_a, \hat{c}_b, \hat{c}_c, \cdots = f_G^{-1,x}(c_j, c_k, c_l, \cdots), \tag{3.10}$$

where $c_j, c_k, c_l, \cdots$ are WPLNC encoded signals (created by Eq.3.8) from relays, $\hat{c}_a, \hat{c}_b, \hat{c}_c, \cdots$ are estimates of original sources and $f_G^{-1,x}$ is the mapping that inverts all WPLNC mappings used by previous relays.

Although the fact that this form of description is the most universal, it provides the minimal inside. Both linear as well as non-linear mapping can be applied. However an analysis of invertibility of WPLNC at the destinations is immensely hard. An encapsulation of WPLNC mappings – when a relays processes by WPLNC mapping incoming signals already formed by WPLNC encoded streams – can be expressed by

$$f_L^j(f_L^m(\cdot), f_L^n(\cdot)), \tag{3.11}$$

nevertheless complex network are not easy to be handled in this way.

Look-up table

Both linear as well as non-linear mappings can be efficiently implemented by look-up tables. Dimensions of the look-up table are given by the cardinalities of the incoming signals and may grow high to become an implementation obstacle. Two different WPLNC mappings differ in particular entries of the tables. The relay output cardinality and thus the kind of WPLNC mapping, is determined by the number of different table items.

The look-up table entries can have an arbitrary form, particularly useful are integers or any other elements of finite fields. Especially for illustration reasons I will use coloured look-up tables.

An example of the XOR WPLNC mapping look-up table representation for two incoming signals each with cardinality four is given in Fig.3.6. Notice that the dimensions of the look-up table are given by the input cardinalities and also notice the cardinality of the relay output which is the number of different colours inside the table. From this it can be again easily seen that the XOR WPLNC mapping is the minimal one.

The examples of the look-up tables with various cardinalities are depicted in Fig.3.7 for two incoming signals with cardinality four. The relay output cardinalities are from left to right 2,4,8 and 16, respectively, thus the mappings are lossy, minimal, extended and full.

A look-up table description of the WPLNC mapping is useful for analysis of the invertibility of the WPLNC at each destination. The invertibility condition converted to this notation is fulfilled if and only if there is a unique ordered n-tuple of look-up tables entries (one from each received WPLNC

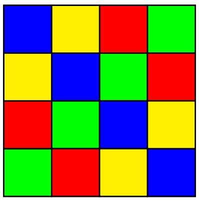

Figure 3.6: Look-up table representation of XOR WPLNC mapping.

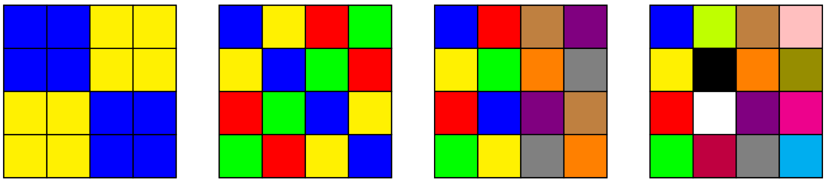

Figure 3.7: Look-up table representation of various WPLNC mappings.

encoded signal) for every possible input from the sources. To exemplify this assume a network from Fig.3.3 where the destination receives two WPLNC encoded inputs from two relays where each is a combination of two original sources. Assume that relay R_1 uses WPLNC mapping f_L^1 and R_2 uses mapping f_L^2. Look-up table representations of these mappings are shown in Fig.3.8 – f_L^1 on the left and f_L^2 on the right. I assume that each source produce an information of cardinality four. This gives two different four by four look-up tables. The destination is able to recover the original source data if and only if the WPLNC mapping are correctly chosen. Translated to the look-up table terminology it means that for every pair of column and row there is a unique pair of entries in look-up tables. It can be seen that example in Fig.3.8 is really the invertible case. Assume that source S_1 which defines column of the look-up table transmits a message with index three and the source S_2 transmits a message with index two. The destination thus receives two WPLNC encoded messages – green from R_1 and blue from R_2. Since green and blue (in this order) pair uniquely determines column and row indices the original source messages can be recovered correctly. Note also that each WPLNC mapping look-up table contains exactly four different colour entries, thus both the mappings are minimal.

WPLNC encapsulation is almost impossible even for simple networks with look-up table description.

Binary-field matrix

Only linear WPLNC mappings can be represented by a binary matrix $\mathbb{X}$. In fact this representability is an equivalent definition of mapping linearity. Since the symbols belong to the finite alphabet each symbol can be indexed, e.g. the symbols of the quaternary alphabet are from a set $\{1, 2, 3, 4\}$. The symbol's index can be easily converted into the form of a vector with the elements from $\mathbb{F}_2$, i.e. $\{0, 1\}$. The length of the resulting vector is given by

$$N = \lceil \log_2 |S| \rceil, \tag{3.12}$$

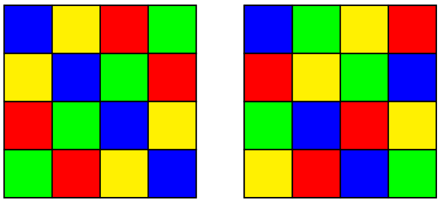

Figure 3.8: Look-up table invertibility example.

where $\lceil \cdot \rceil$ is the ceiling function and $|S|$ is the cardinality of the alphabet S.

A processing by a linear WPLNC mapping at the relay R_j that is described by the function f_L^j can be rewritten into the form of matrix multiplication. WPLNC encoding operation at the relay R_j is thus given by

$$\bar{c}^j = f_L^j(c_a, c_b, c_c, \cdots) \quad \Rightarrow \quad \bar{c}^j = \mathbb{X}_L^j \begin{bmatrix} c_a \\ c_b \\ c_c \\ \vdots \end{bmatrix}, \tag{3.13}$$

where $\mathbb{X}_L^j$ is the matrix form of the R_j's WPLNC and the right-hand side vector $[c_a \quad c_b \quad c_c \quad \cdots]^T$ is a concatenation of the binary representations of the incoming symbols $c_a, c_b, c_c, \cdots$.

The number of columns of the matrix $\mathbb{X}_L^j$ is given by the sum of the cardinalities of the inputs and the number of rows corresponds to the relay output cardinality. The elements of the matrix form are from the $\mathbb{F}_2$, i.e. $\mathbb{X}_L^j \in \mathbb{F}_2^{N \times M}$. The matrix representing the XOR WPLNC function for two sources with quaternary alphabets is

$$\mathbb{X}_L^j = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \tag{3.14}$$

Although the matrix form is usable only for linear mapping it has got a plenty of favourable features. Particularly, it is easy to check the invertibility of several WPLNC mapping matrices as well as it is easy to describe an encapsulation of WPLNC mapping when a relay combines several incoming WPLNC encoded signals by its own WPLNC mapping. The invertibility check is converted to the finite field matrix rank evaluation of concatenated matrices as illustrated by the following example. Assume a network from Fig.3.3 where the destination receives two WPLNC encoded inputs from two relays where each is a combination of two original sources. The relay R_1 receives the sources S_1 and S_2 and applies the WPLNC mapping $\mathbb{X}_L^1$. Similarly R_2 also receives the sources S_2 and S_3 but applies the mapping $\mathbb{X}_L^2$

Local encoding functions at the relays are

$$\bar{c}_1 = \mathbb{X}_L^1 \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} \qquad \bar{c}_2 = \mathbb{X}_L^2 \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}, \tag{3.15}$$

where $\bar{c}_j$ is the output from the relay R_j, c_i is the signal of the source S_i.

The global encoding function seen at the D is a concatenation of relays' matrices

$$\begin{bmatrix} \bar{c}_1 \\ \bar{c}_2 \end{bmatrix} = \mathbb{X}_G \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} \mathbb{X}_L^1 \\ \mathbb{X}_L^2 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}. \tag{3.16}$$

The global WPLNC mapping $\mathbb{X}_G$ is invertible if and only if there exist an inverse matrix to $\mathbb{X}_G$ which can be easily checked by computing of the finite field $\mathbb{F}_2$ rank of the $\mathbb{X}_G$ matrix. If the $\mathbb{X}_G$ is full rank (exactly has full column rank) its inverse (or pseudoinverse in the case of non-square matrices) exists. Decoding operation at the D is thus

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \mathbb{X}_G^{-1} \begin{bmatrix} \bar{c}_1 \\ \bar{c}_2 \end{bmatrix}. \tag{3.17}$$

The encapsulation of WPLNC mappings is handled in the following way. The relay R_j is using mapping defined by $\mathbb{X}_L^j$ and processes two WPLNC encoded signals encoded by mappings $\mathbb{X}_L^m$ and $\mathbb{X}_L^n$. The overall WPLNC processing is given by

$$\mathbb{X} = \mathbb{X}_L^j \begin{bmatrix} \mathbb{X}_L^m \\ \mathbb{X}_L^n \end{bmatrix}, \tag{3.18}$$

if the dimensions of the matrices match. In fact they must match otherwise the multiplication is not defined and WPLNC mapping will not be working properly. This is the reason why relays in complex layered networks must sometimes step back to worse mappings if there is a change in the previous layer as will be described in Chapter 5.

3.5 Notation

Because of the benefits of the binary matrix description of the WPLNC mappings they will be almost exclusively used throughout this thesis. This limits this thesis to linear WPLNC mappings only.

The matrices describing the WPLNC mappings will be denoted by $\mathcal{X}$, particularly a WPLNX matrix used by the node i is denoted $\mathcal{X}^i$. Since it is always obvious from the context whether the matrix is the local one applied by the node or the global one being observed by the destination the subscripts $_{L}$,$_{G}$ are omitted in the rest of the thesis.

As discussed in this chapter and in chapter 2 I will mostly focus on WPLNC performed at the symbol by symbol level of uncoded data. Thus in the rest of this thesis the inputs to the WPLNC mapping will be symbols q from some finite alphabet. Whenever it is not necessary to insist on symbols a symbol c is used to denote bit/symbol/codeword/.../signal that is processed by WPLNC.

Chapter 4

Distributed Learning Algorithm - basic scenarios

As was shown Wireless Physical Layer Network Coding (WPLNC) is a technique to improve the wireless networks of the future, gaining the advantage from innovative harnessing of interferences. WPLNC can properly work if and only if sufficient number of correctly selected WPLNC mappings are delivered to destinations. The mapping assignment can be relatively easily done in centralised and fixed networks, however, it becomes a crucial obstacle when the network is decentralised and with time varying topology. In any case the mappings must be matched to the actual network state. In fixed centralised network it can be done in advanced by network designer and driven from central coordination point. In distributed networks an algorithm that selects the mappings on demand is necessary. In this chapter I develop one such an algorithm, its fundamental properties are analysed and verified through both numerical and real-world implementation.

The work presented in this chapter is based on my published as well as unpublished research, especially [87, 88].

4.1 Problem statement

WPLNC is successfully applied by the cloud network if and only if it reliably delivers the source information to the destinations. This is true if for each destination D_k exists an inverse mapping $f^{-1,k}$ from received WLPNC encoded symbols q_{R_j} to original source data q_{S_i}:

$$f^{-1,k} : q_{R_m} \times q_{R_n} \times \cdots \rightarrow q_{S_e} \times q_{S_f} \times \cdots . \tag{4.1}$$

For the sake of simplicity I will not deal, at least for the beginning, with general situation when the relay inputs are not source symbols, but outputs of other relays, i.e. with WPLNC encapsulation. Which can obviously also be formed by WPLNC from other relays and/or symbols and so on *ad infinitum*. In the following I will assume layered topology of the cloud, shown for two layer case in Fig.4.1. In this case the layer of sources communicates only with the closest layer of relays, that communicates only with the next relay layer down the cloud towards the destination etc. Moreover for better illustration

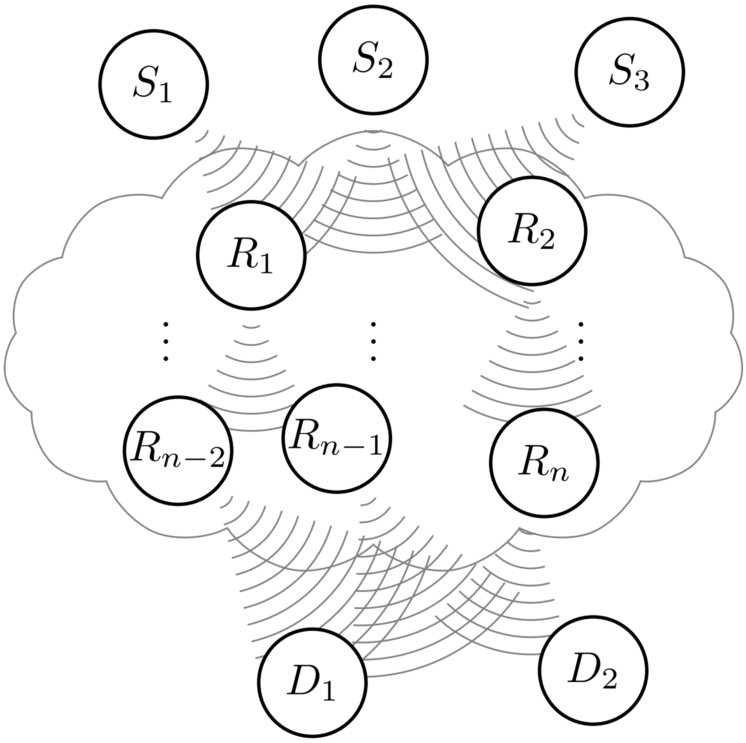

Figure 4.1: Layered cloud network.

I will mostly assume single layer cloud network only. Generalisation towards more complex networks will be shown later (see chapter 5). In this case the relay output is formed directly by WPLNC encoded source symbols. Each destination receives one or more such relay symbols and can construct (if possible) an inverse mapping to recover the original source data.

As was shown in the chapter 3, the existence of inverse mapping is equivalent to the state when the mappings applied by the relays are mutually invertible (also full rank for linear matrix description). There is a fundamental question how the WPLNC mappings shall be assigned to the individual network relay to enable invertibility.

Fairly simple solution exists for static network with known topology. Then individual WPLNC mappings can be tailored to exactly fit the situation. The solution can be even found in polynomial time [55], at least for single source multicast situation. In case of the distributed network we lack such an obvious solutions. There are different methods available, each having its advantages as well as disadvantages:

- Random Network Coding [56], a technique that assigns the WPLNC mappings to the relays randomly. Correct performance is guaranteed only by probability and the technique obviously works better in big networks. For mid size and smaller networks there is a significant probability of failure. Also randomly selected mappings does not guarantee optimal performance at all.

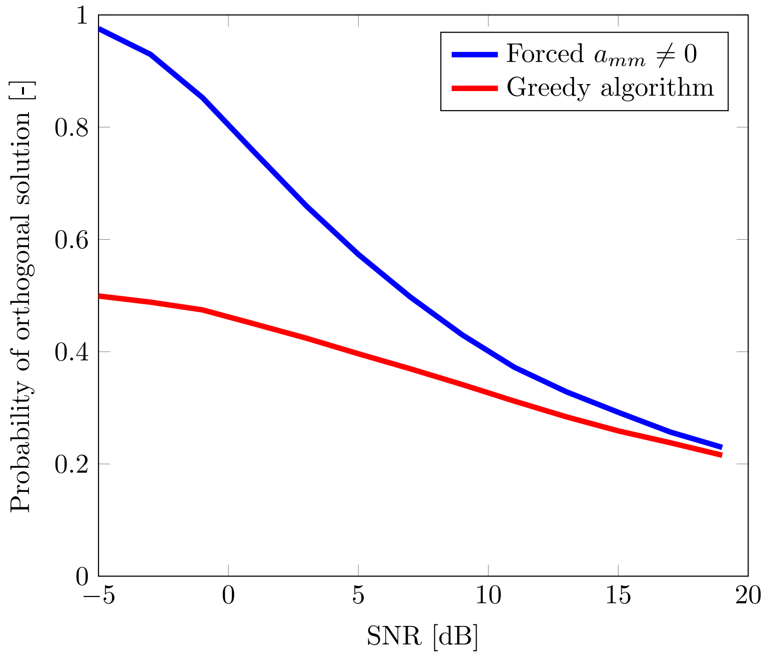

Figure 4.2: Probability of orthogonal solution for Compute and Forward technique.

- Greedy Algorithm is shown in [78, Section X, Fig.13, Eq(26)] that is developed for coefficient selection (which is similar concept to WPLNC mapping) for Compute and Forward technique. As reported therein it has quite high probability of failure since it does not deliver invertible observation to the destinations. Its improvement, that forces the m-th relay to always process at least the m-th source, on the other hand many times collapses into orthogonal solution, when the m-th source is the only source processed by the m-th relay. Fig.4.2 shows how often this happens.
- Authors in [89] provide an example of two source two relay network with WPLNC mapping allocation. Unfortunately it need full channel knowledge at all relays and thus is applicable only with high overhead or with the help of controlling authority.

In a distributed network the WPLNC assignment should arise from unsupervised decision made by each individual relay based on its local knowledge and observations. Seen as a general problem, there are multiple entities making decision that may affect the decisions of the others. A problem exactly fitting the area of game theory. Since every single relay R_j choosing its own WPLNC mapping f^j may cause some destination to be unable to recover the source data, because of violating the existence of the inverse mapping according to Eq(4.1), all mappings must be chosen with this condition in mind. In this chapter I will provide a game theoretic analysis of the presented issue that leads to viable protocol for WPLNC assignment across the network.

To my best knowledge there is no other similar solution published, presented or used. There are some game theoretic solutions such as [90–93], however, they mostly focus on other aspects to be solved by game theoretic tools as routing schemes, multiple access or resource allocation and more over if applied to relay networks then mostly to single source case only.

4.2 Network model

The network of interest consist of three separate layers, namely a layer of sources, relays and destinations. Particularly I will focus my attention to a layered network of two sources S_1, S_2, two relays R_R, R_C and two destination D_1, D_2 as illustrated in Fig.4.3.

This network topology is the simplest one that is able to show all game theoretic concepts I will present within this chapter. The network can be straightforwardly extended in width, more nodes can be added to any of the layers. All results that I am going to present are equally valid also for this extended network. On the other hand any such extension will make the presentation of the proposed concept unnecessarily complex, less illustrative and harder to follow, thus I restrict myself to the simple version. Extension in depth, i.e. adding more layers, is postponed to the next chapter of this thesis. Layered topology inherently assumes that the communication proceeds between the neighbouring layers. There is no communication going around or across the layers. The only exception from this rule is direct link from source(s) to destination(s) that will be commented later. Even more complex network, without layered structure, seems to be quite a challenging nowadays especially due to the possible existence of the loops. Such complex networks lie unfortunately beyond the scope of this thesis.

During the presentation I expect time and frequency synchronisation of all the nodes and known time slotting of transmission and reception activity. All these aspects are achievable by the means of distributed algorithms that again are far beyond the scope of this thesis. Both are at least particularly discussed in chapter 2.

A priori given and commonly known time slotting is somehow related to the layered topology of the network and is closely connected to the position of the node inside the network. Although there is now a priori knowledge about the node position expected, in real networks some knowledge will be available to the nodes. Some nodes will serve as sensors, thus forming the layer of the sources. Next, shown on the example of smart building/city in chapter 2, there will be a set of relays serving individual rooms, corridors, storeys, buildings, streets, quarters, ... Layered topologies may naturally arise in practical deployments of the cloud networks. Separated transmitting and receiving activity is enforced by half-duplex constraint which is typical for current wireless devices, although times may be changing [94].

Each source independently produces its data message to be delivered to destination(s). For simplicity I assume broadcast option, that is data delivery to all destinations within the network. Other cases are of course also possible.

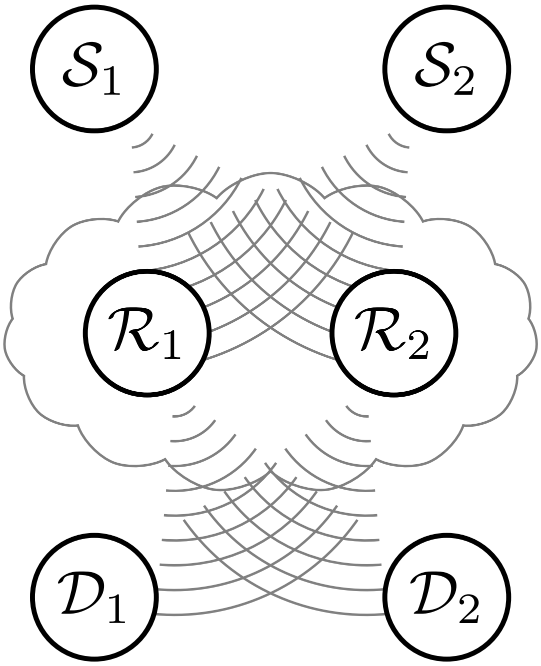

Figure 4.3: Two source, two relay, two destination cloud network.

Source message is composed of the bits b mapped to the symbols q. In my example the symbols will be from linear modulation families, such as #-PSK or #-QAM. Particularly, BPSK or QPSK will be used almost exclusively. The restriction to linear modulations is applied only to keep the discussed matter simple and not to obstruct it with tangled notation details. No error protection coding is assumed here, although again it is possible. Both traditional simple block or convolution as well as modern advanced Turbo and LDPC codes can be used. The latter have disadvantage of higher computational demands that can fairly soon deplete energy resources of small and simple battery powered devices. Also these modern coding techniques may be overkill for simple devices such as temperature sensors, where legacy or even no protection coding at all, just repeated transmission, may be more practical.

The processing at the i-th source side is given

$$\mathbf{b}_i = \{b_{i1}, b_{i2}, b_{i3}, \cdots\} \Rightarrow \{q_{i1}, q_{i2}, \cdots\} \Rightarrow s_i(q,t), \qquad (4.2)$$

where sequence of bits $\mathbf{b}_i$ is mapped symbols q_{in} (N bits to a symbol from 2^N-ary symbol alphabet) which are later converted to signal waveform $s_i(q,t)$ that is broadcast.

Transmission from the sources share, in agreement with WPLNC concept, the same wireless media with neither orthogonal separation nor interference reduction mechanism and thus naturally superpose upon each other. This superposition arrives to the first layer relays' antennae where it is received. The wireless channel causes attenuation, phase rotation, delay, frequency distortion (for frequency selective channels) etc. Also noise is added at the receiver input.

The noisy superpositions are processed by the relays. The superpositions will likely differ from relay to relay since also the channels among the relays and sources differ. Each relay processes the superposition by the WPLNC technique and produces its output, WPLNC encoded, message. Various types of WPLNC techniques can be used (shortly described and discussed in chapter 3). In this thesis I will exclusively focus on decode and forward type processing. Nevertheless the presented concept is generic enough to be directly (or with only minor modifications) applied to all other WPLNC processing types, which is important to highlight.

The relay output message is mapped to the relay output symbols, again without any error protection coding and using linear modulation families. The only encoding done is the WPLNC one which combines the incoming source messages into the relay output. The produced relay message is broadcast towards all destinations, again without any orthogonal separation and thus crating new superpositions on the way from relays to destinations.

Symbol by symbol processing at the j-th realy can be illustrated by

$$y_j(t) = h_{aj} s_a(q_a, t) + h_{bj} s_b(q_b, t) + \cdots + w_j(t) \Rightarrow q_j \Rightarrow s_j(q_j, t), \tag{4.3}$$

where superposed signals from sources $s_i(q_i, t)$ received at the relay as $y_j(t)$ are mapped by WPLNC to the relay symbol q_j from some alphabet (possibly completely different from the sources' alphabets, not only in terms of modulation family but also in terms of cardinality). Relay symbol is converted to signal $s_j(q_j, t)$ which is broadcast towards destinations. Multiplicative channel model h_{ij} is used solely for simplicity.

The destinations receive the relay transmissions. Then recover individual relay messages (which obviously may be tricky if there are too complex modulations and/or too many transmissions superposed in unfortunate way). From the recovered relay messages each destination tries to invert the applied WPLNC and obtain original source data. If it is possible then one communication round was successful and source data were delivered to intended destinations. If not, then the cloud network failed in its task, a lot of resources were spent in vain and the data shall be resend. Again, this highlights the importance of correctly assigned WPLNC mappings to the relays so as to make the destinations able to recover source data and thus avoid waste. The processing at the k-th destination consist of

$$y_{k,s}(t), y_{k,t}(t), \cdots \Rightarrow q_s, q_t, \cdots \Rightarrow q_a, q_b, \cdots, \tag{4.4}$$

where $y_{k,j}(t)$ is the signal from j-th relay. Multiple such signals are collected, relay symbols q_j are recovered and finally WPLNC is decoded which recovers original source symbols q_i and appropriate source bits b_i.

WLPNC performed in the network of interest (two source, two relays and at least one destination) will be illustrated by the following example where bit level description of symbol by symbol processing and matrix notation will be used for linear WPLNC mappings.

Example 4.1. Assume that the source S_1 sends a symbol q_1 consisting of bits b_{11}, b_{12}, i.e. $q_1 = (b_{11}, b_{12})^T$ from its quaternary alphabet, such as QPSK.

Similarly S_2 sends $q_2 = (b_{21}, b_{22})^T$. Then the output symbol q_{R_j} of the relay R_j $j \in \{R, C\}$, using an example of a minimal WPLNC mapping f^{R_j} shown by its matrix equivalent $\mathbb{X}^{R_j}$, is

$$\begin{bmatrix} b_{R_{j1}} \\ b_{R_{j2}} \end{bmatrix} = q_{R_j} = \mathbb{X}^{R_j} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \\ = \begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} b_{11} \\ b_{12} \\ b_{21} \\ b_{22} \end{bmatrix} = \begin{pmatrix} b_{12} \oplus b_{21} \oplus b_{22} \\ b_{11} \oplus b_{12} \oplus b_{21} \end{pmatrix}, \tag{4.5}$$

where $\oplus$ is bit-wise XOR operator. Notice that relay output cardinality equals the cardinality of each individual source – the WPLNC mapping used is a minimal mapping.

The existence of the inverse mapping of the pair of mappings f^{R_R}, f^{R_C} can easily be checked by finding the row rank (in the GF(2) sense) of the concatenated matrix that describes the WPLNC mappings. The row rank must be at least four in the case of two sources both using QPSK, i.e. two times two bits. It should be noted that all *full* mappings are invertible by itself. They are also invertible in a pair with an arbitrary WPLNC mapping. For illustration assume that two relays R_R and R_C use particular WPLNC mappings $\mathbb{X}^{R_R}$ and $\mathbb{X}^{R_C}$ to create their output symbols q_{R_R}, q_{R_C} respectively. The corresponding concatenated observation matrix $\mathbb{X}^k$ at the destination $\mathcal{D}_k$ is

$$\mathbb{X}^k = \left[\frac{\mathbb{X}^{R_R}}{\mathbb{X}^{R_C}}\right] = \left[\begin{array}{cccc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{array}\right], \tag{4.6}$$

where the individual relay matrices are are highlighted by horizontal line in the matrix concatenation.

It can be seen that in this case $\mathbb{X}^k$ is full rank (rank is four) in the GF(2) sense. Thus the pair of WPLNC mappings $\mathbb{X}^{R_R}$, $\mathbb{X}^{R_C}$ is invertible. The original source symbols q_1, q_2 can be recovered by multiplication with the (pseudo)inverse matrix $\mathbb{X}^{-1,k}$.

$$\begin{bmatrix} b_{11} \\ b_{12} \\ b_{21} \\ b_{22} \end{bmatrix} = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \mathbb{X}^{-1,k} \begin{bmatrix} q_{R_R} \\ q_{R_C} \end{bmatrix} = \\ = \left[\frac{\mathbb{X}^{R_R}}{\mathbb{X}^{R_C}}\right]^{-1} \begin{bmatrix} q_{R_R} \\ q_{R_C} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_{R_{R1}} \\ q_{R_{R2}} \\ q_{R_{C1}} \\ q_{R_{C2}} \end{bmatrix}. \tag{4.7}$$

4.3 Game theoretic analysis

4.3.1 Game definition

I seek a solution to the proposed problem of invertible WPLNC mapping assignment in a form of solution of game. The game is played among the relays that individually and independently choose their WPLNC mappings. The mapping selection must allow all destinations to recover original source data. It should reflect actual network conditions to make the mapping selection sensible also from the relays' local point of view. Both properties may be enforced by properly selected utility function that reflects both non-invertibility as well as actual network states.

More precisely the game of interest consist of the following entities:

- Set of players $\mathcal{P}$ - that is formed by the cloud network relays. For the sake of simplicity I assume that all the relay participate in one common game. This is inherent to two relay case that I will mostly study. In real cases there may be multiple sets of relays that do not compete each other for example because of big area separation among them, processing of different subsets of the sources, etc. In this case there will be multiple instances of the game.
- Set of actions $\mathcal{A}$ - that is formed by all the WPLNC mappings that are available to the relay node. Again, to simplify the case I will assume one common set that is available and known to all the relays. Practically, there is no obvious reason why the action sets should differ from relay to relay. In my case the action set will be formed by various linear WPLNC mappings. However, it is truly important to highlight that the analysis as well as the results hold true even for the action sets formed by non-linear mappings. The solution is general enough that it is also usable for any other WPLNC technique such as assignment of Compute and Forward coefficients, which are a different form of actions.
- Utility function $U(\cdot)$ - a real valued function from the set of actions of each relay which should reflect a goodness of WPLNC mapping to the cloud network as well as to the individual relay. It should penalise the situations when the final observation delivered to the destinations are not invertible. In this case the destination is unable to recover the source data and thus the transmissions and the resources were spent in vain. Similarly, it also should prefer the WPLNC mapping that improves the network performance, such mappings that multiply a throughput, lower the error rate at the destinations, save the most energy, etc. For simplicity the same utility function is assumed for all the relays. Although the different utility function for different relays should be sensible in some cases it complicates the initial analysis beyond necessity.

The game is assumed to be a complete information game, meaning that the rules of the game defined above are common knowledge of all the players.

The number and identity of players, action set and utility function is know in advance and to all players. Equivalently, a pay-off matrix of the game is known by the players. It is good to note that all the sets (particularly sets relays playing the game and of their actions) are finite, thus the game itself is finite.

An illustrative instance of game played by the relays forming the network shown in Fig.4.3 is specifically defined by:

- Player set $\mathcal{P} = \{R_R, R_C\}$ is formed by both relays that each selects its own WPLNC mapping. I also define an opponent set of player j denoted as $\mathcal{P}_{-j}$ where subscript $-j$ means all other players different from j. Particularly $\mathcal{P}_{-R} = \{R_C\}$ and vice versa.

- Set of actions $\mathcal{A}$ is formed by various WPLNC mappings a_i generally divided into cardinality categories and denoted as MIN1, MIN2, ..., EXT1, EXT2, ..., FULL1, FULL2, ... An example set is shown in a pay-off matrix of the game show as Tab.4.1. The set is finite although the number of the available actions is not specified since it is not critical. I naturally avoid less sensible actions sets such as an empty set or a set which does not posses at least one invertible pair. These sets do not lead to sensible outcomes and will barely emerge in reality. The action set is common to all the players. An action space of the game is formed by Cartesian product of the players actions sets, in this case $\mathcal{A}\times\mathcal{A}$ formed by pairs of WPLNC mappings $\{a_C, a_R\}$ played by relay R_C, R_R, respectively. Because the design should mainly avoid the case when non-invertible solution is provided to the destinations only pure strategies (i.e. one particular action from $\mathcal{A}$) are assumed. Mixed strategies (i.e. probability distributions over actions from the action set $\mathcal{A}$), although possible and obviously providing some additional NEs [95], cannot guarantee that non-invertible pair will never be selected, thus they are not analysed here. An interested reader shall find some results concerning mixed strategy NEs in my paper [96].

- Utility function $U(a_C, a_R)$ is a function from action space $\mathcal{A}\times\mathcal{A}$ to real numbers $\mathbb{R}$. Again this function is common to both players. I denote the action space, a space of all possible action pairs, as $\mathbb{A} = \mathcal{A}\times\mathcal{A}$. The action space is generally given by Cartesian product of all players' action sets. The utility function shall be designed such that it avoids non-invertible WPLNC mapping. Particularly, for the utility function assumed throughout this thesis, it will penalise all players with non-positive value $P \leq 0$ whenever any non-invertible WLPNC mapping is going to be used. Moreover, the utility function should also prefer WPLNC mappings that improve some of the performance metrics. Mostly used example will be an utility function that prefers MIN mappings over EXT and over FULL ones. Since this utility improves potential throughput, that grows as alphabet cardinalities go lower. Particularly, this utility function will award the relays by A for invertible MIN mapping, by B for invertible EXT mapping and finally by C for FULL mapping.

	MIN1	MIN2	MIN3	EXT1	EXT2	EXT3	FULL1	FULL2
MIN1	(P, P)	$(\mathbf{A}, \mathbf{A})$	$(\mathbf{A}, \mathbf{A})$	(A, B)	(A, B)	(A, B)	(A, C)	(A, C)
MIN2	$(\mathbf{A}, \mathbf{A})$	(P, P)	$(\mathbf{A}, \mathbf{A})$	(P, P)	(P, P)	(A, B)	(A, C)	(A, C)
MIN3	$(\mathbf{A}, \mathbf{A})$	$(\mathbf{A}, \mathbf{A})$	(P, P)	(A, B)	(A, B)	(A, B)	(A, C)	(A, C)
EXT1	(B, A)	(P, P)	(B, A)	(P, P)	(B, B)	(B, B)	(B, C)	(B, C)
EXT2	(B, A)	(P, P)	(B, A)	(B, B)	(P, P)	(B, B)	(B, C)	(B, C)
EXT3	(B, A)	(B, A)	(B, A)	(B, B)	(B, B)	(P, P)	(B, C)	(B, C)
FULL1	(C, A)	(C, A)	(C, A)	(C, B)	(C, B)	(C, B)	(C, C)	(C, C)
FULL2	(C, A)	(C, A)	(C, A)	(C, B)	(C, B)	(C, B)	(C, C)	(C, C)

Table 4.1: Pay-off matrix of WPLNC assignment game.

The individual utilities are ordered as follows:

$$A > B > C > 0 \geq P. \tag{4.8}$$

It can be easily seen that this utility does not reflect any channel state related parameters. It just awards the relays from cardinality point of view. In the following I will also mention the utility function that is related to the channel. Also a discussion why such utilities cannot be generally applicable will be provided.

Having all game elements defined a pay-off matrix of the game can be created. An instance of the pay-off matrix itself is shown in Tab.4.1. Action set of the player defines number of rows/columns of the matrix. Relay R_C selects its action a_C column-wise while R_R selects a_R as rows (this also explain the naming and subscripts of the nodes). Matrix entry on row a_R and column a_C has form $(U^R(a_R, a_C), U^C(a_R, a_C))$ where $U^R(.)$ is an award given to R_R when a pair of action a_R, a_C from $\mathbb{A}$ is played, similarly for $U^C(.)$ and player R_C. Non-invertible cases are those with reward equal to (P, P), where P is penalty to be paid by the nodes. Notice the both relays fail together and both pay penalty of $P \leq 0$. Also notice the fact that any FULL mapping is always invertible regardless the other WPLNC mapping, i.e. that there is no (P, P) in any of FULL mapping related row/column.

Since the proposed game is complete information game, this pay-off matrix is available to all the players and thus forms common knowledge. The complete set of the game rules is captured by this pay-off matrix. Notice, since that the matrix is diagonally symmetric in pay-off, the column and the row player can be mutually swapped without any effect, the proposed game belongs to symmetric game category.

4.3.2 Equilibrium analysis

If I avoid trivial cases such as single relay case, empty action sets, action sets that contain no pair of invertible mappings etc., it is rather easy to observe that the game possesses at least one pure strategy Nash equilibrium (NE).

In many cases there are even multiple of them. In Tab.4.1 all the NEs are highlighted by the bold font, all of them are formed by some pair of MIN WPLNC mappings for this particular case. The results of Nash [95] declares that there is always at least one NE if we allow mixed strategies. Since a mixed strategy is a probability distribution over the action set and thus it is not avoiding the non-invertible cases I focus only on the pure strategies (see [96] for mixed strategy discussion). In this case it is not hard to show that even some pure strategy NEs exist, for example by manually performing the best response moves until there is no point to move further. The formal proof, performed in different way, is provided in the next subsection.

4.3.3 Potential game

Having defined the game rules it is necessary to show if an existence of NE is just a mere coincidence in particular example or if it is a common property of this sort of games. Additionally it is good to show how one of the NEs can be achieved practically. First it is important to provide several definitions and propositions.

Definition 4.2 (Ordinal potential game). A non-cooperative game is called an *ordinal potential game* if there is a function $\Phi : \mathbb{A} \to \mathbb{R}$ such that

$$\operatorname{sign}\left[\Phi(x, a_{-i}) - \Phi(y, a_{-i})\right] = \operatorname{sign}\left[U^i(x, a_{-i}) - U^i(y, a_{-i})\right], \tag{4.9}$$

for all players i, for all actions $x, y \in \mathcal{A}_i$ of the player i and for all actions played by other players $a_{-i} \in \mathbb{A}_{-i}$, where sign is a signum function [97, 98].

Corollary 4.3. *Every finite ordinal potential game has at least one pure strategy Nash equilibrium [97].*

Corollary 4.4. *Assume a finite ordinal potential game. Then both the best reply dynamic and the random better reply dynamic will (almost surely) converge to a Nash Equilibrium in a finite number of steps [97, 99].*

Where dynamic means a process of action updates based on observed actions played by the others. Now the game will be played repeatedly and the players will be given a chance to adapt their actions based on the actions of the opponents. Best dynamic means that the best available action is always selected in the improvement step. Similarly, better dynamics tries to improve at least a bit. Differences between both and consequences of it will be discussed later. The dynamic update of used WPLNC will be a key to distributed implementation WPLNC assignment algorithm.

Theorem 4.5. *Game given by Tab.4.1 and all other similar games and instances are the ordinal potential game.*

Proof. To prove the theorem I show that an auxiliary function $\Phi(a_R, a_C) = \sum_{i \in \{R,C\}} U^i(a_R, a_C)$ fulfils Eq.(4.9) in all cases. Particularly, I will show all situations when (a_R, a_C) and (a'_R, a_C) allow or do not allow invertibility. Here a_R and a'_R are two different WPLNC mappings from $\mathcal{A}$ of relays R_R and $a_C \in \mathcal{A}$ is the action of relay R_C.

When (a_R, a_C) and (a'_R, a_C) both are invertible WPLNC mappings:

$$\begin{aligned}\text{sign}[U^R(a_R, a_C) + U^C(a_R, a_C) - U^R(a'_R, a_C) - U^C(a'_R, a_C)] = \\ = \text{sign}\left[U^R(a_R, a_C) - U^R(a'_R, a_C)\right],\end{aligned}$$

since $U^C(.)$ is independent of a_R when recoverable, i.e. $U^C(a_R, a_C) = U^C(a'_R, a_C)$, then Eq.(4.9) holds true.

When (a_R, a_C) is but (a'_R, a_C) is not invertible WPLNC mapping pair:

$$\begin{aligned}\text{sign}[U^R(a_R, a_C) + U^C(a_R, a_C) - 2P] = \\ = \text{sign}\left[U^R(a_R, a_C) - P\right],\end{aligned}$$

since award for invertible case is positive and $P \leq 0$ then Eq.(4.9) holds true.

When (a_R, a_C) is not but (s'_R, s_C) is invertible WPLNC mapping pair:

$$\begin{aligned}\text{sign}[2P - (U^R(a'_R, a_C) + U^C(a'_R, a_C))] = \\ = \text{sign}\left[P - U^R(a'_R, a_C)\right],\end{aligned}$$

since award for invertible case is positive and $P \leq 0$ then Eq.(4.9) holds true.

When both (a_R, a_C) and (a'_R, a_C) are not invertible WPLNC mapping pairs:

$$\text{sign}\,[2P - 2P] = \text{sign}\,[P - P]\,,$$

which is trivial and true for every P then Eq.(4.9) holds true.

Since I have listed all the cases that can occur and because of the symmetry of the game in players the existence of ordinal potential function is proved. This proof is general and can be applied to any instance of the proposed game even with multiple relays. □

There are two highly important consequences of the game being ordinal potential one

- First, at least one pure strategy NE always exists. Recall, that I have not assumed degenerated instances of the game (such as the action spaces that do contain no invertible WPLNC mapping pair), since they are of no practical interest.
- Second, there is a finite process, repeated game play, that achieves one (or some of multiple) equilibrium.

Both properties form basis for practical implementation, in a form of communication protocol, of WPLNC assigning algorithms.

It is obvious that ordinal potential property is only related to the form of the utility function assumed. The game has this property for the utility that prefers cardinality improvements and penalises the non-invertible outcomes. It can be shown [99] that if a game is constructed with random pay-offs, as in the case when the utility function is derived from random wireless channel, that the probability that it will be a potential game is zero. This fact limits the channel related utility functions to be the major one to dictate the rules of the game. Since without the potential property the game may not have pure strategy equilibria and/or no sensible process may exist to achieve the reasonable equilibrium outcome. Later in 4.6 I will describe channel related utility function that is use as additional criterion to assign WPLNC mappings.

4.4 Distributed learning algorithm

It is mainly the convergence of best/better dynamic process (see Corollary 4.4) that allows and makes favourable an implementation of WPLNC assignment algorithm into distributed cloud networks. This algorithm, from now on called a Distributed Learning Algorithm (DLA), is a wireless protocol created by direct implementation of best/better dynamics. Top-level pseudo-code for the basic network from Fig.4.3 without any particular implementation details is shown in Alg.1.

Algorithm 1 Distributed Learning Algorithm (DLA)

1: $\forall j$ R_j chooses a full WPLNC mapping optimising utility function $U(\cdot)$
2: $\forall j$ R_j adds information which WPLNC mapping was chosen to WPLNC encoded data
3: $\forall j$ R_j broadcasts towards the destinations
4: Signalise all used actions to $\forall j$ R_j
5: Select one relay R_{j^*} that will adapt its WPLNC
6: R_{j^*} chooses a new WPLNC that optimise $U(\cdot)$
7: Go back to line 2

The DLA starts from Full WPLNC mappings that are selected by each relay as a safe starting point. Safe, because any full WPLNC mapping is always invertible regardless other mappings, thus no advance knowledge of the other relays actions or even knowledge about their presence in the network is needed. Also, the reward obtained by the relay based on the proposed utility solely depends on mapping used by that relay not on the others if the final mapping combination is invertible. That is why all relays can independently initiate WPLNC communication with full mappings.

The selected mapping must be also indicated to the place where the original sources should be recovered. This is necessary for any form of WPLNC and even NC to construct an inverse mapping. Several forms of this signalisation are shown in [100] several others will be shown later in this thesis.

WPLNC encoded data together with WPLNC mapping information are broadcast by the relays to destinations where source data are recovered. This is possible even without any initial coordination between the relays because of the safe full mapping starting point.

The WPLNC mapping used by the other relay(s) will be made available to all other relays by special signalling channel, by overhearing other relays (both these methods assume dedicated and/or orthogonal channel(s) and will serve only as initial attempts) or will be signalised by the destination in backward communication round. Practical implementation of all these method will be shown later.

When the opponents actions are signalised back to all relays one relay should be selected to adapt its own mapping. Particularly, for example, column relay is being selected and knows the action (WPLNC mapping) previously used by the row player it can chose a new mapping improving its utility while invertibility is maintained. The improvement should follow one of better or best dynamics. The invertibility is guaranteed since the relay that changes the mapping perfectly knows (perfect signalling is assumed for simplicity) the state of the game and all non-invertible cases can be avoided due to utility function.

The important thing is that one and only one relay is allowed to adapt WPLNC mapping in given time instant. If a simultaneous move of multiple players is allowed it may result in non-invertible situation. This uniqueness of adapting relay can be enforced by various ways. To name a few (which will be also later shown during practical HW implementation) I mention fixed round robin, random selection with collision detection or random voting process.

When a new mapping is selected it is applied to newly incoming data and together with a newly created WPLNC encoded relay output delivered to the destination(s). The new WPLNC assignment is again signalised to all the relays and a new round of WPLNC adaptation can begin.

This process continues until equilibrium is achieved when it can stop, since there will be no more improvements from achieved equilibrium in static scenarios. In dynamic scenarios it may be beneficial to let DLA run even when equilibrium is achieved once. This helps with adaptation to potential change in network topology (such as (dis)connection of a node). Also if the utility function reflects the actual channel conditions it is necessary to let DLA continue to be able to modify WPLNC mapping according to the changing channel.

4.5 Hardware implementation and real-world testing

To verify functionality and properties of DLA a testbed, that closely resembles the toy network from Fig.4.3, is created. Software defined radio modules manufactured by Ettus Research [101] are used to perform radiofrequency and signal processing. For technical and physical layer related details see chapter 5 where a more developed test-bed is described. It is also valid for this initial DLA verification although with some simplifications and changes. Namely, all nodes are connected to common frequency reference – Ettus Octoclock clock distribution system [102] – such that there are no issues with carrier frequency synchronisation. Source nodes are time synchronised to sample time level such that simultaneous operation and transmission is achieved. Timing synchronisation of all remaining nodes is based on synchronisation pilot words. Time slotting is generally known. In a similar way also a channel

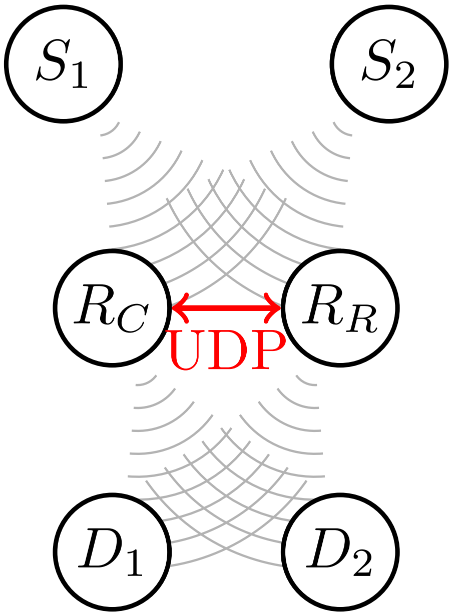

Figure 4.4: Two source, two relay, two destination network topology with UDP signalisation channel.

estimation is done. All this synchronisation is performed due to orthogonally separated signals and sequences. Initial attempt to use non-orthogonal signals instead please refer to chapter 6. In final and more advance test-bed all signalisation will be made closer to real system, if interested see chapter 5.

The network consist of a pair of sources, a pair of relays and a pair of destinations. Since this was the first and initial implementation of DLA there are two main differences compared to the theoretic concept: an independent and highly reliable User Datagram Protocol (UDP) [103] based channel is made between the relays to provide a signalisation and both relay to destination transmissions are for simplicity separated in time manner into two consecutive slots. Later these will be removed and replaced by more appropriate way. Due to the half-duplex limitation and time division assumption, one round of communication takes place over three time slots: Ss to Rs, R_R to Ds and R_C to Ds. A physical layer settings (OFDM parameters), time slotting, packet format and rules of underlying DLA game are known by all the network nodes. Whenever a transmission that does not fit into the slotting scheme is overheard then it is ignored, for example, the destinations ignore direct transmission of the sources. Similarly due to time separation of relay transmission there is a possibility to overhear an opponent relay. Again this transmission is ignored currently, although later it was used to replace UDP signalisation link. The true placement of the nodes in the laboratory is depicted in Fig.4.5.

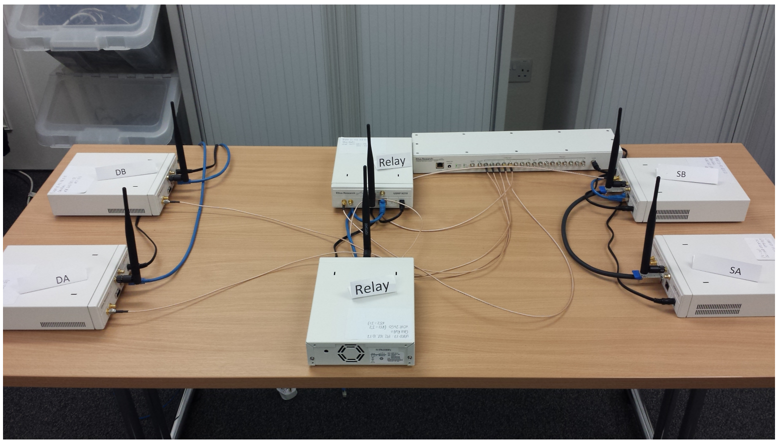

Figure 4.5: Node placement during laboratory experiment.

4.5.1 Implementation of DLA

The underlying game rules are as follows. The game is played between the relays R_R and R_C. The action set is common to both players. The actions in this set are ordered in the same way for both players and this ordering is used during signalisation to uniquely identify the WPLNC mapping that was used. Particular example of the action set is shown in Tab.4.2, where binary forms of WPLNC matrices $\mathbb{X}$ are given. The utility function reflects the relay output cardinality according to Eq(4.8). Thus it prefers the WPLNC mappings that minimise output cardinality and penalises non-invertible solutions. There is no channel related part in this utility function. The complete rules of the game are shown by pay-off table Tab.4.3. This pay-off table is the exact copy of the example shown in Tab.4.1 and is shown here for convenience. The same notation for pay-off rewarded is used and the Nash equilibria are highlighted by bold font. Notice that there are six of them, all formed by various MIN-MIN pairs of WLPNC mappings.

A particular modification of general DLA algorithm given in 1 is given in a pseudocode 2. Especially detailed implementation of signalisation and adapting relay selection is provided.

Again, both relays start from safe Full WPLNC mapping solution, particularly the relay R_R uses the mapping FULL1 from Tab.4.2 and R_C uses FULL2. Since there is no channel related part of the utility and since full mappings work independently of other mappings both relays should start with the same full WPLNC mapping. The actual reason for different starting points is merely to verify more options.

Each relay receives superposition of both source transmission, process the signalisation and synchronisation pilot signals (will be described in detail later). They also process data payload part with actual WPLNC mapping and create the output signal, that is send to destinations in appropriate time slot.

Together with output packet creation a coin is tossed to find out if a signalisation should be sent through UDP channel. A "win" or positive

WPLNC	$\mathbb{X}$	WPLNC	$\mathbb{X}$
MIN1	$\begin{matrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{matrix}$	EXT2	$\begin{matrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{matrix}$
MIN2	$\begin{matrix} 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{matrix}$	EXT3	$\begin{matrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 \end{matrix}$
MIN3	$\begin{matrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{matrix}$	FULL1	$\begin{matrix} 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{matrix}$
EXT1	$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{matrix}$	FULL2	$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$

Table 4.2: List of WPLNC mappings - UDP signalisation

Algorithm 2 Distributed Learning Algorithm - signalisation via UDP channel

```
∀j R_j chooses full WPLNC mapping
repeat
    for all j do
        R_j receives signal
        R_j processes pilots and payload
        R_j creates output signal and transmits it
        if Lottery tells to transmit UDP then
            Send out UDP packet
        else
            Do not send out UDP packet
        end if
        if Received valid UDP and does not send own then
            Update WPLNC mapping to improve utility u_j
        else
            Do not change WPLNC mapping
        end if
    end for
until Convergence achieved
```

outcome of coin toss is obtained with probability p, i.e. the signalisation data are sent over UDP with probability p. The form of signalisation data is the same as is used for signalisation pilot. When both relays output their

	MIN1	MIN2	MIN3	EXT1	EXT2	EXT3	FULL1	FULL2
MIN1	(P,P)	$(\mathbf{A},\mathbf{A})$	$(\mathbf{A},\mathbf{A})$	(A,B)	(A,B)	(A,B)	(A,C)	(A,C)
MIN2	$(\mathbf{A},\mathbf{A})$	(P,P)	$(\mathbf{A},\mathbf{A})$	(P,P)	(P,P)	(A,B)	(A,C)	(A,C)
MIN3	$(\mathbf{A},\mathbf{A})$	$(\mathbf{A},\mathbf{A})$	(P,P)	(A,B)	(A,B)	(A,B)	(A,C)	(A,C)
EXT1	(B,A)	(P,P)	(B,A)	(P,P)	(B,B)	(B,B)	(B,C)	(B,C)
EXT2	(B,A)	(P,P)	(B,A)	(B,B)	(P,P)	(B,B)	(B,C)	(B,C)
EXT3	(B,A)	(B,A)	(B,A)	(B,B)	(B,B)	(P,P)	(B,C)	(B,C)
FULL1	(C,A)	(C,A)	(C,A)	(C,B)	(C,B)	(C,B)	(C,C)	(C,C)
FULL2	(C,A)	(C,A)	(C,A)	(C,B)	(C,B)	(C,B)	(C,C)	(C,C)

Table 4.3: WPLNC mapping pairs and their payoffs - UDP signalisation

signals and toss their coins the outcome of adaptation step is resolved. The relay is selected for adaptation if and only if it has received valid signalisation data from the opponent and has not sent its own to the opponent in given communication round. This lottery mechanism enforces that only one relay will adapt in given time instant, which is necessary to avoid non-invertible solutions. If both relays output their signalisation data then a collision happened, that is easily detected by both and no one is allowed to adapt. Similarly, if none of the relays has send the signalisation then also adaptation cannot happen.

If the relay is selected for adaptation it updates its WPLNC mapping so as to improve utility in agreement with the game rules. Remind that adapting relay knows the opponent's WPLNC mapping through UDP signalling. This adaptation can follow both better or best dynamics. With better dynamics any mapping that improves utility at least a bit can be selected. While the best dynamics always selects the maximiser of the utility function. Obviously the best dynamics offers faster convergence speed, however it can get stuck in local optima. This can happen especially when the action sets are not the same for all the relays, which is not the case used here and also it is not a case of significant interest. Nevertheless, the better dynamics is almost exclusively used in this thesis, since it offers more detailed exploration of DLA and network behaviour, although the convergence is a bit slower.

A newly selected (and thus better) WPLNC mapping is used by that relay from that time instant for data processing as well as for signalisation. Then DLA continues until one of equilibria is achieved.

The DLA algorithm was implemented in MATLAB code which runs in the GNURadio flow graphs of the radio nodes representing sources, relays and destinations. The Matlab code is compiled as a shared library using Matlab Compiler Runtime. This can then be called within the C++ code contained within a GNURadio flow graph block.

Preamble	S_1 Idx	S_2 Idx	S_3 Idx	S_4 Idx	WPLNC Idx	Pad	CRC
0011	3 bits	3 bits	3 bits	3 bits	5 bits	7 bits	4 bits

Table 4.4: PiHRC Structure

4.5.2 WPLNC signalisation - hierarchical pilot

When WPLNC is employed the relay output should not only contain WLPNC encoded data but also some accompanying signalisation should be provided [100]. This signalisation should especially and importantly contain an identification of original sources and an identification of WPLNC processing done on the data part. In the sense of this thesis it especially means an identification of WPLNC mapping used to encode source data. There can be some additional information. Nevertheless this signalisation is necessary and mandatory for all WPLNC techniques.

This signalisation will be called an hierarchical pilot (PiHRC) throughout this thesis. It will be used in both ways – affixed to encoded data packet as well as used through dedicated signalisation channel if necessary.

The length of PiHRC of this particular implementation is fixed to 32 data bits with the format shown in Tab.4.4.

Fields S_i Idx serve to uniquely identify sources that are contained in the following data part of the packet. Field WPLNC Idx is basically an index to common list of available WPLNCs (available action set). Pad is an unused space to be possibly used in future and the PiHRC is protected by simple Cyclic Redundancy Check (CRC) encoding using polynomial $x^4+x^3+x^2+1$.

A packet that originates at source S_i node has only S_i Idx field filled and WPLNC Idx field contains a value that indicates no WPLNC, only pure source data. Due to the structure of the PiHRC, receiving nodes are able to recover the transmitted content. The relay combines the useful payload data using its WPLNC mapping and sends out the packet with updated PiHRC – S_i Idx fields filled by all sources combined by WPLNC and WPLNC Idx points to the used WPLNC mapping. The destination uses the PiHRC information to perform WPLNC inversion.

For output RF signal formation a newly created PiHRC is mapped to OFDM physical layer using orthogonal Constant Amplitude Zero AutoCorrelation (CAZAC) sequences [104]. The data are mapped to each of the 16 OFDM symbols, i.e. each pair of bits once mapped to a QPSK constellation, then are spread along the 48 data subcarriers in frequency using a CAZAC of length 47 with root 1, padded with a 0. This use of an odd sequence length provides beneficial cross-correlation properties. The CAZACs used for this spreading also uniquely identify the node transmitting the PiHRC by using a cyclic shift equal to the index of the node, i.e. 0 for S_A (stored as 000 in PiHRC) and 1 for S_B (stored as 001 in PiHRC).

As already mentioned, in future systems the dedicated signalisation through UDP channel will be replaced by feedback from destinations piggybacked to standard packets. This feedback will carry information about other WPLNC mappings received by each destination as well as a list of sources, for details see chapter 5.

4.5.3 Results

To verify the performance of DLA as well as presented implementation, a number of packet transmissions were carried out. We focused on the convergence time of the DLA, i.e. the number of adaptation steps before a (MIN,MIN) invertible pair is achieved, as a function of the probability of UDP inter-relay exchange p. A series of transmission rounds were executed for p in the range 0.1 to 0.9 for both 'better' and 'best' response dynamics and the number of adaptation trials until (MIN,MIN) was achieved was recorded. Fig.4.6 shows the average values ± 1 standard deviation for 150 independent runs for each value of p. It can be seen that the DLA converges reasonably fast for a wide range of p settings. Just 10 adaptation steps are sufficient on average for optimal p values when using 'better' response dynamics and only around 6 steps are required when using 'best' response dynamics.

The figure also shows the intuitive result that for extreme values of p ($\rightarrow 0$ or $\rightarrow 1$) it takes significantly more steps since at the limit where $p = 0$ and $p = 1$ there is no UDP communication or permanent UDP collision, respectively, so adaptation cannot take place. The average number of runs is approximately two times lower for $p \rightarrow 0$ than for $p \rightarrow 1$ since our algorithm stops the UDP transmission lottery at the relay and sets $p = 1$ when it achieves MIN mapping. From this time instant that relay is therefore constantly producing UDP packets. This thus improves the convergence for low values of p. For p close to 1 the probability of UDP collision increases, which slows the convergence rate, but the change is reasonably small since collision is quite probable even before p is set to one, e.g. it rises to $1 \cdot 0.9 = 0.9$ from $0.9 \cdot 0.9 = 0.81$ for $p = 0.9$.

The testbed results confirm that the DLA algorithm is fully functional in real-world conditions, at least with some simplifying assumptions on node synchronization and network topology. The algorithm finds optimal WPLNC mappings across the network quickly. From the results it can be seen that the optimal value of p is close to 0.3 for both dynamics but the algorithm performs well for a wide range of p values. Values of p between 0.2 and 0.7 require fewer than 10 adaptation rounds to achieve optimal result using 'better' response dynamics. The optimal response dynamic is 'best' since it converges faster than the 'better' one. However, 'best' dynamics has a higher probability of becoming stuck in a *local* optimum point. This is not the case in the example studied here since there are only global optima. However, local optima may exist, especially when the set of WPLNC mappings differs from relay to relay.

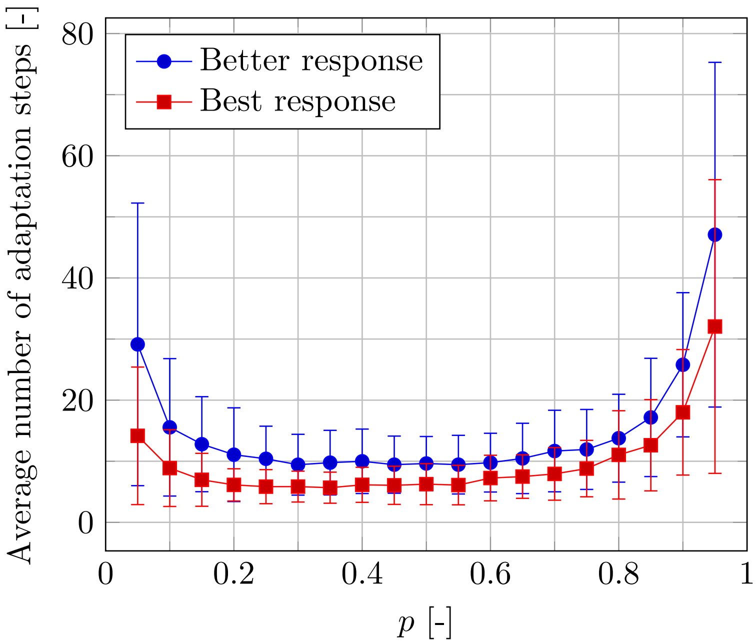

Figure 4.6: DLA average convergence time (in terms of number of adaptation steps) and its standard deviation.

4.6 Channel related utility function

Up to now the utility function used by the relays only accommodates their output cardinality. With a goal of its minimisation so as more efficient, in terms of throughput, energy consumption, etc., network operation is achieved. There was no particular attention given to actual channel conditions. However, it is known that particular channel states can literally destroy the WPLNC processing [68, 105] if an unsuitable mapping is used. In this section I would like to focus on channel related utilities. I will present two utilities, namely minimal distance and actual symbol error rate. Although there is no thorough analysis justifying selection of these utilities common sense should justify them as potential candidates with obvious link to actual channel states.

A constellation space representation of received superposition of transmission from two sources over two frequency flat channels is given by

$$y(t) = h_A s_A(q_A) + h_B s_B(q_B) + w \tag{4.10}$$

where $s_i(q_i)$ are constellation space representations of source symbols, $h_i = |h_i|e^{\varphi_i}$ are appropriate complex channels and w is a noise. This equation can be scaled by either h_A or h_B to make it a function of only one complex variable. Thus it becomes

$$\tilde{y} = h s_A(q_A) + s_B(q_B) + \tilde{w}, \tag{4.11}$$

where $h = \frac{h_a}{h_b}$ and all variables with tilde originated by division by h_b.

For given channel states it is possible that some points of the superposed constellation y will be close to each other or will even overlap. This may result in symbol errors when the given WPLNC mapping requires the overlapped (or near-overlapped) points to be distinguished. The probability of overlaps increases obviously at lower SNR.

Assume that two received signals y, y' correspond to the transmission of two different pairs of source symbols q_A, q_B and q'_A, q'_B, i.e. $y = h_A s_A(q_A) + h_B s_B(q_B) + w$ and $y' = h_A s_A(q'_A) + h_B s_B(q'_B) + w'$. Particular channel conditions h_A, h_B may cause the two superposed signals y, y' to be precisely overlapped or close enough to be indistinguishable. If the relay uses a WPLNC mapping that maps the superposed signals y, y' to different relay output symbols and y is interchanged with y' (or vice versa) then a symbol error will occur. Moreover this error propagates to the final destination, due to the incorrect mapping of q_A, q_B to relay output, and will cause errors at the destination in the estimates of q_A, q_B.

Properly selected WPLNC mappings should take this effect into account, i.e. minimize the probability of symbol error by mapping collocated, or closely adjacent, received signals y and y' to the same relay output symbol. Thus if the signals *are* interchanged there is no difference caused to the output, and no error is caused. Fig.4.7 helps to exemplify the error behaviour of different mappings. The figure shows WPLNC decision regions based on two different mappings. The dots show the noisy superposition of two simultaneous BPSK transmissions, i.e. an example of a received superposed constellation at the relay, given by Eq(4.10). The background patterns indicate different relay output regions. At lower SNRs the noisy signals close to the middle become difficult to distinguish and may become interchanged with each other. With the WPLNC mapping on the left-hand side of Fig.4.7 this does not matter since both are mapped by WPLNC mapping to the same output (hatched background). However this constitutes an issue for the right-hand side mapping, since the points must not be interchanged otherwise there will be an error. It can be concluded that the mapping on the right is more susceptible to symbol error because some constellation points are easily mistaken by each other. In a non-static wireless channel, therefore, a fixed WPLNC mapping may not be optimal from the SER point of view. Using a dynamic mapping it is possible to choose the mapping that is most robust to symbol error, for given channel conditions.

As noted before, since any channel related utility function will be inherently a random variable it can be hardly used as a primary utility function since there is a high probability that the underlying game will lost its potential property. Any such utility can, however, serve as some additional criterion to select among WPLNC mappings preselected by DLA with cardinality utility to make them best fit actual channel conditions.

Whenever it was possible the algorithm was implemented and tested by existing testbed. Mostly only simple two source one relay sub-block was verified by this testing. Nevertheless, this simple topology is a building block of more complex networks.

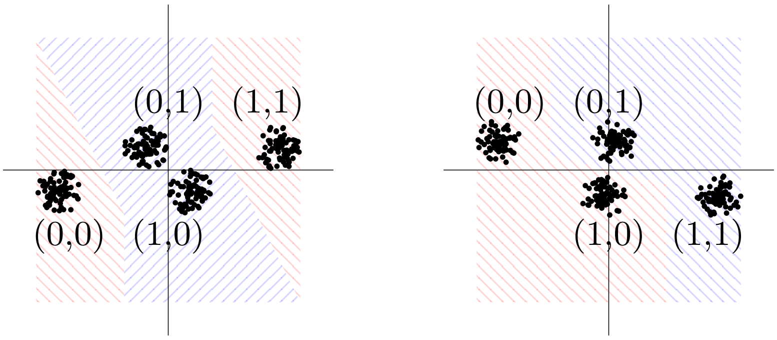

Figure 4.7: Illustration of channel impact on superposed constellation.

4.6.1 Minimum distance based utility

In point to point communication system a minimal distance of used modulation is one of the parameters that crucially determine performance [86]. Although it may not be also fully true for multi-node communications the minimal distance is a potential candidate for channel related utility. In the case of multi-node network the minimal distance d_{min} is defined as the shortest distance between two points of superposed constellation that belong to two different decisions regions. The decision regions are defined by a WPLNC mapping that is used and decision metric. A mutual distance among the constellation points within each of decision regions does not matter since all these point are mapped to the same output. Thus they can be freely interchanged with no effect on error performance, see Fig.4.7 for illustration. By this mechanism the minimal distance of superposed constellation is a function of the WPLNC mapping as well as actual state of the channels.

The minimal distance for channel model with one complex parameter is numerically evaluated for the case of two QPSK sources. Since there are 2^16-1 possible linear WPLNC mappings it is impossible to list all the results here. However to exemplify I show in Fig.4.8 squared minimal distance for some randomly selected minimal WPLNC mapping. For comparison Fig.4.9 shows the same for XOR WPLNC mapping. Although there is an appreciable difference in regions close to $h = \pm 1$ in the rest of the space both mappings provide similar or exactly the same minimal distance. This is quite a general conclusions, a lot of mappings, not only minimal ones, provide very similar minimal distances. Thus the minimal distance itself can be hardly used as additional utility since it does not effectively distinguish WPLNC mappings.

4.6.2 Symbol error rate based utility

To overcome issues of indecisiveness of minimal distance utility the next candidate is estimate of actual symbol error rate (SER) computed on-line on known transmitted data. Similarly to the minimal distance there is a negligible variation among full and extended mappings in SER performance, thus I focus only on minimal mapping selection optimisation.

From all available minimal mappings, $2^8 - 1$ in the case of two sources

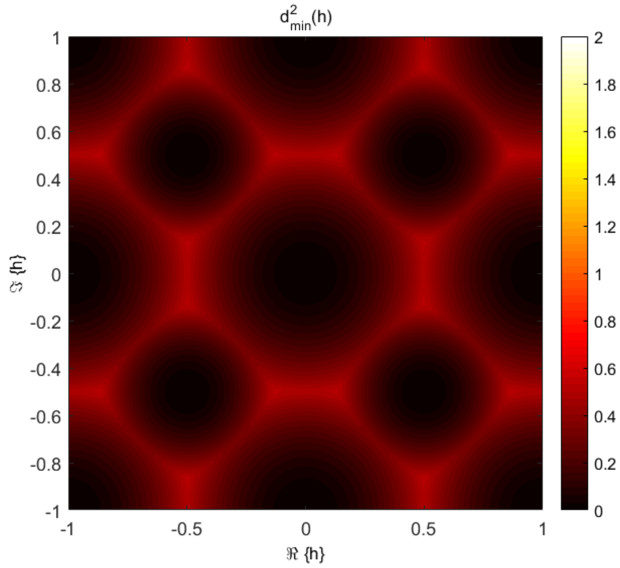

Figure 4.8: Squared minimum distance of some minimal WPLNC mapping - two QPSK sources.

using QPSK, we have preselected 16 mappings that evinced robust behaviour through thorough numerical simulations. Those selected had very low symbol error rates averaged over a large number of channel instances. This restricted set of mappings was used for relay adaptation. A care was taken to construction of the relays' action set, such that there still exist several optimal equilibria, even though the minimal mappings were preselected. The optimal mapping is chosen on a per packet basis by calculating the instantaneous SER for a subset of the data carriers in the first OFDM symbol of the data packet, for each of the 16 preselected mappings. A subset (every fourth carrier) is used to minimize the computational complexity. Only the first OFDM symbol is considered because the channel is assumed constant over the packet duration. Although there is an obvious computational burden connected with SER estimation it was minimised and actually is not to big to avoid run-time execution as was verified.

The proposed utility function for selection of WPLNCs has the form

$$\mathrm{WPLNC}_{\mathrm{new}} = \min(\widehat{\mathrm{SER}}), \tag{4.12}$$

where $\mathrm{WPLNC}_{\mathrm{new}}$ is the newly selected WPLNC mapping, $\widehat{\mathrm{SER}}$ is the estimated SER for the actual channel and WPLNC mapping and the minimum is over all WPLNC mappings that allow data recovery at the destination(s) which is obtained by execution DLA algorithm.

Both numerical and testbed verification were done. Results obtained from numerical simulations are based on transmission of 10^4 symbols over 10^4 channels with AWGN and random phase rotation. Care was taken also to ensure randomization of the channel for the USRP testbed results using suspended reflectors. This produces a relatively slow fading effect in an

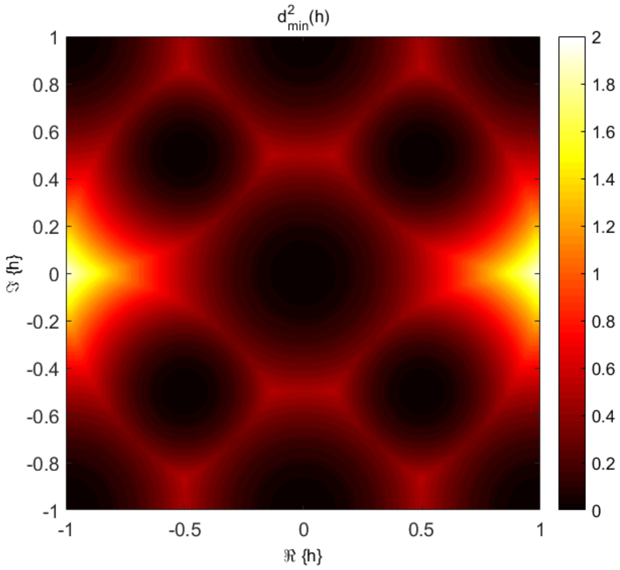

Figure 4.9: Squared minimum distance of XOR minimal WPLNC mapping - two QPSK sources.

otherwise static laboratory environment, thus giving comparable channels between simulation and the laboratory. For the optimization to work, the channel must remain approximately static only for the duration of each packet which is approximately 3 ms. The SER measurements obtained using the USRPs is based on 5000 packets collected. The reason for the smaller data set is due to processing delay in the relay since the SDRs are computer-hosted leading to slow baseband processing limiting the amount of data that can be collected in a timely fashion. SNR values above 30dB cannot readily be achieved using the hardware due to real-world effects, such as non-linearities in the power amplifier and phase noise.

Fig.4.10 shows both numerical simulation (solid lines) and USRP (dashed lines) SER as a function of SNR results for various mappings. 'Bit XOR' denotes one of the well known minimal WPLNC mappings. The other minimal, extended and full mappings are arbitrarily selected. Curves denoted 'opt' are based on the adaptive selection of WPLNC mappings matched to the actual channel conditions to minimize SER. It can be seen that the USRP results very closely match the simulation results, especially considering the simulation does not capture the effects of imperfect channel estimation, synchronization, phase noise and other non-linearities. Using fixed WPLNC mappings results in an SER of 10^{-2} at high SNR region. Although it may intuitively be expected, the use of fixed minimal mappings does not yield a big improvement in SER over fixed extended and full mappings due to very high error rate introduced by some channel conditions. Using the optimal minimal mapping selection, however, an SER below 10^{-5} can be achieved at approximately 20 dB SNR. Notice also that no channel coding to protect against the error is applied.

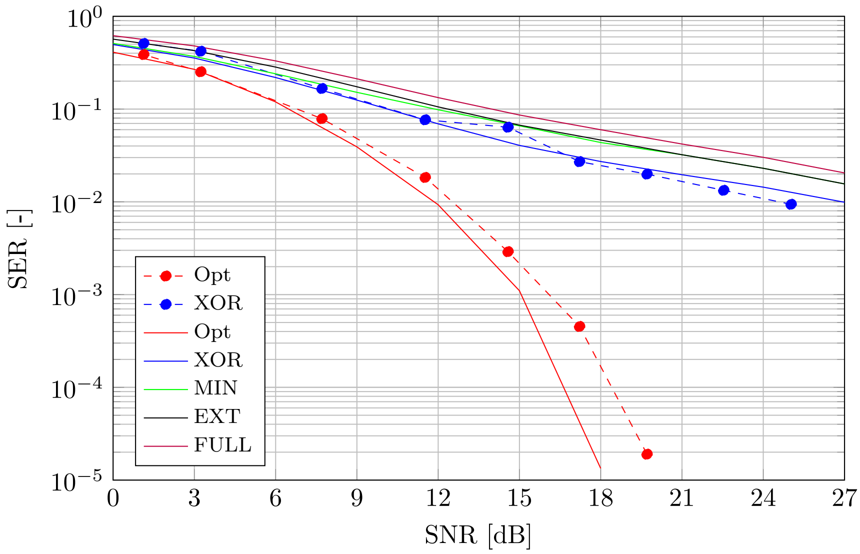

Figure 4.10: Results of SER utility verification.

4.6.3 WPLNC performance in unbalanced channel

Although it has not been truly highlighted up to now the signals from all sources have been assumed to be more or less equal in power. This naturally comes from symmetry of all the assumed networks and homogeneous environment leading to similar channel characteristics. There are thus natural questions what if the power received from individual sources is not well balanced, how well will WPLNC perform under this condition, what WPLNC mappings should be assigned to the relays when there are some comparatively weak sources, is it worth to try to recover such weak signals or is it better to ignore them?

It is an inherent property of all wireless systems that there is a minimum requirement on signal strength (usually expressed in terms of signal to noise ratio or similar measure) that must be achieved for smooth service. When this condition is not met than the connection cannot be established at all or is highly unstable with frequent drop-outs. WPLNC technique is not an exception from this rule. If the signal from the source is too weak to be recognised in the superimposed constellation then the relay should rather ignore it.

For example, the minimal distance (although not particularly used as utility and although minimal distance close to zero is more than common even if the signals are strong) comes to zero when one signal is weak. See Fig.4.9 for $h \to 0$. In this case it is really hard to reliably process the weak source and it may be highly favourable rather to ignore it.

There are WPLNC mappings that are designed for multiple sources (e.g. the corresponding dimensions of the WPLNC matrix fit the situation) but practically ignore some of the sources or transfer only the bits transmitted by weak sources that are most easily recognisable. All such "ignoring" WPLNC mapping contain some null part, for example in the case of linear WPLNC expressed by a matrix for two source case:

$$q_{R_j} = \mathbb{X}^{R_j} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} \mathbb{X}^{R_j,1} & 0 \end{bmatrix} \begin{bmatrix} b_{11} \\ b_{12} \\ b_{21} \\ b_{22} \end{bmatrix} = \mathbb{X}^{R_j,1} \begin{bmatrix} b_{11} \\ b_{12} \end{bmatrix}, \tag{4.13}$$

where 0 stands for zero matrix of appropriate size (2 by 2 in this case) and $\mathbb{X}^{R_j,1}$ is part of mapping $\mathbb{X}^{R_j}$ that relates only to the first source (again a 2 by 2 matrix in this case).

Such matrices that ignore some or parts of some sources can be added to the action sets to correctly cope with weak sources. They can still be used together with DLA without any modification. When this ignoring mapping is the minimal one it can also be easily and favourably used with additional SER optimising utility function.

The source that is ignored by some relays should be deliver to the destinations via other relay to which it has a stronger link. If no such relay exists then the source is effectively disconnected from the cloud network and could not expect that its transmissions will be received, processed and successfully delivered to destinations.

Numerical simulation was carried to evaluate error performance of WPLNC under unequal channel conditions. A two source one relay case was considered. Two WPLNC mappings were compared – XOR and modified XOR that carries over only the first source, with matrices given by

$$\mathbb{X}_{XOR} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \qquad \mathbb{X}_{S_A\text{-only}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{4.14}$$

Channel unbalance is express by channel ratio h and goes from 1 (both channels are equally strong) to 0 (one of the channels is totally absent). Signal to noise ratio of fixed channel is set to 20dB. By numerical simulation 10000 symbols were sent over a 1000 random realisations of complex frequency flat channel. Mean SER of relay decision is evaluated. The results are shown in Fig.4.11. When the channels are of comparable power the error performance of both mappings is also comparable. The SER of S_A-only mapping is slightly higher in this region, probably due to disturbances by S_B symbols under some channels. In weak signal conditions ($h < 0.5$) the SER of S_A-only mapping is unmeasurable with 10000 symbols sent, indicating very good performance matched to the actual channel conditions. On the other hand XOR mapping is still trying to process S_B symbols even though that channel is weak or completely missing. SER is very high as $h \to 0$ indicating that XOR mapping is not suitable.

To support the results somewhat bigger HW verification was done using already developed test-bed. The channel balance is now expressed by an SNR difference between two links, with S_A link SNR fixed to 21, 16 and 11 dB respectively. The SER optimisation (as described in sections above) was used with three different sets of available mappings. Opt denotes the

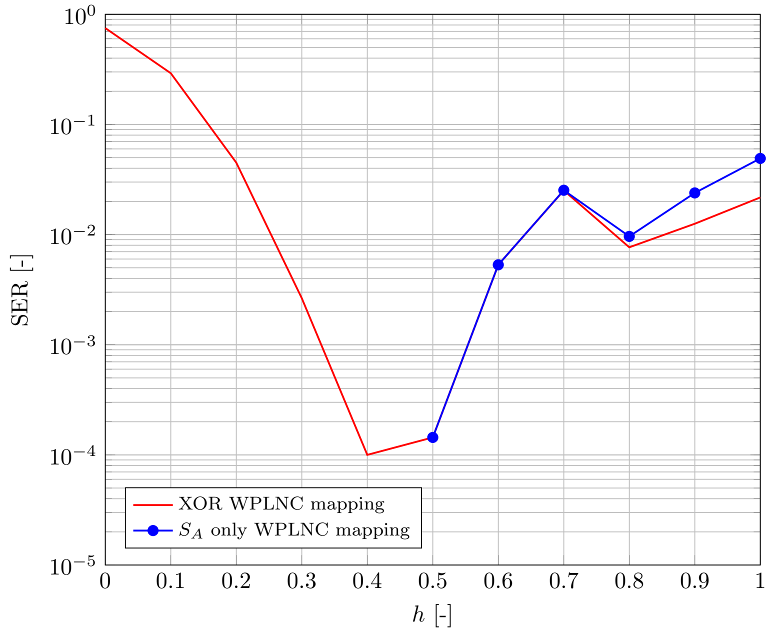

Figure 4.11: SER in unbalanced channel condition - numerical simulation.

original preselected set of well behaving mappings, Opt - SA is the same set but WPLNC mappings processing S_A only were removed from it and finally SA denotes single fixed arbitrary WPLNC mapping processing only S_A (i.e. part of that matrix corresponding to S_B is all zero, see 4.13). The results are shown in Fig.4.12. Considerably lower SER were achieved thanks to larger amount of symbols sent over randomly changing channel. The results are comparable to numerical simulations. Under balanced channel scenario ($\text{SNR}_B - \text{SNR}_A \to 0\text{dB}$) S_A-only mapping performs significantly worse then optimised mappings. On the other side of channel conditions the performance is similar, however S_A-only mapping (either constant or included in set for optimisation) performs a bit better. With this HW verification we did not observe superior performance of S_A-only mapping under weak channel conditions, which may be caused by some real-world or implementation related issue.

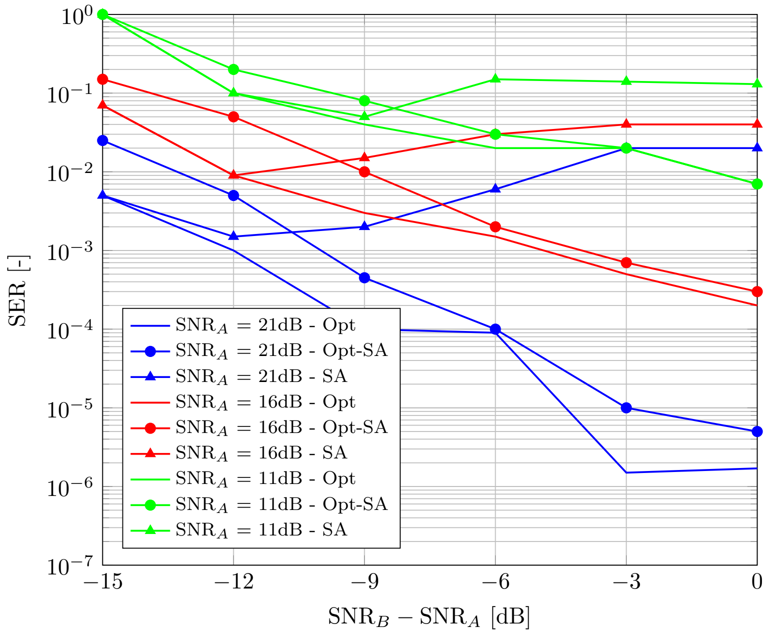

Figure 4.12: SER in unbalanced channel condition - HW verification.

Chapter 5

Distributed Learning Algorithm - advanced scenarios

The main reason of this chapter is to show applicability of Distributed Learning Algorithm (DLA) in more advanced and also more realistic scenarios, including larger scale environment testing and demonstration. Moreover, in two last sections original complete information game is relaxed even more to assume various types of relay nodes' behaviour. In two examples a malicious behaviour of the relay and a behavioural change enforced by node battery level is studied. These two modifications violate complete game rules knowledge assumption and add some uncertainty about the network. Game theoretic analysis of these relaxed assumptions could be used to proper design of the utility functions to cope correctly with potentially misbehaving relays. These two examples should show variability of possible real world issues and how they can be coped by proper design of the game rules.

The work presented in this chapter is mostly based on my work carried under DIWINE project. Presentation of incomplete information games stands on my publications [106, 107].

5.1 Larger scale realistic demonstrator

The main goal of this section is to present the latest evolution of DLA (see chapter 4) that is far more advanced version of originally proposed protocol. A care is given to test this protocol under realistic scenario making it as much developed cloud network demonstrator as was possible.

Some features of advanced DLA deployment to be highlighted are:

- Multiple layer of relays – One of the most important advances is an extension of the cloud network to two relay layers or tiers. The network consists of two sources, followed by two layers of relays, each containing two relays and then there are two destinations. Cloud network itself is thus formed by four relays separated by two into two layers.

- Bi-directionality – A distinction between the sources and the destinations is a bit artificial because the network is bidirectional now. Nevertheless the notation of sources and destinations is still used although also the destinations produce their independent information. Source nodes are the nodes that initiate the communication at the very beginning of the network operation. The destinations are the terminal nodes that are on the opposite side of the cloud network at that time instant. Also when different layers of relays are mentioned it is always counted in the direction from the actual origin of transmission. The first layer is always closer to the actual data transmitter being either sources or destinations. The second layer is the next one. Bi-directionality is favourably used to transfer necessary signalisation piggybacked to data packets. Some "true" data from sensors such as temperature and luminosity were sent by the sources.

- Variable network topology – The final demonstrator has a capability to adapt to changing cloud network topology. The relays were allowed to disconnect and reconnect from and to the cloud network. At least one relay should remain in each of the layer nevertheless all topologies in range from two source, two relays, two relays and two destinations to two sources, one relay, one relay, two destinations are allowed. Time instant of disconnection/reconnection is arbitrary without any necessary coordination or prior announcement. Typically only one packet is lost, the one that is being actually transferred by the cloud, in time instant of relay disconnection. DLA algorithm and thus WPLNC assignment process is restarted after topology change such that a new optimal allocation for a new network topology can be achieved.

- Real world deployment – In contrast to original DLA implementation (see chapter 4) which was deployed on a table with fairly static laboratory environment the advanced demonstrator was deployed in several rooms of an office area. There were common obstacles such as wooden doors, glass walls with aluminium window blinds, supporting concrete pillars and office furniture. The demonstrator was working correctly with people moving around and inside the cloud network making the channels time variable. Generally the links between the node were not obstacle-free line-of-sight-links.

- All in wireless – The next important feature is that all dedicated supporting links were removed. All necessary information is transferred by wireless link as a part of communication protocol. There is a dedicated synchronisation epoch prior WPLNC communication phase. During this epoch the synchronisation of nodes' local oscillators is done in a distributed manner. The algorithm is based on transmission of orthogonal Constant Amplitude Zero-AutoCorrelation (CAZAC) sequences. Its details are far beyond the scope of this thesis, however an interested reader can found them in [29, 108]. Throughout the WPLNC communication phase the synchronisation of frequency and time is achieved by processing dedicated orthogonalised pilot signals that are part of the packet. All necessary information for advanced DLA, such as voting to select adapting relay, WPLNC related information in form of hierarchical

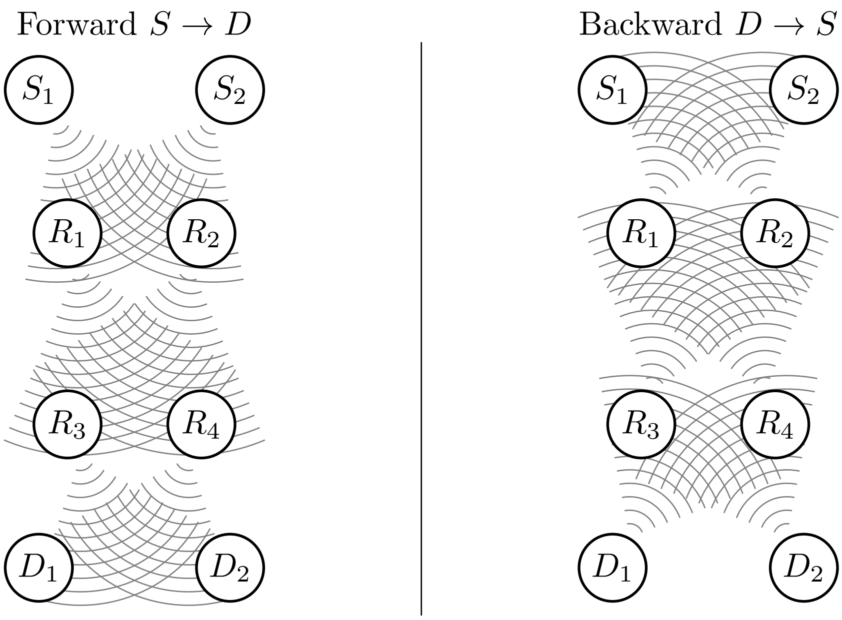

Figure 5.1: Network topology - advanced DLA.

pilot (PiHRC), is also transmitted as a part of data packet. No dedicated channels neither wired nor wireless are used for distributed WPLNC assignment.

There were several important changes made in original "basic" DLA to enable these features. They will be described in the following paragraphs.

5.1.1 Network model

The network consists of four terminal nodes, by annotation and bit artificially (just to have an ability to describe communication direction) divided into two sources and two destinations. The terminal nodes are wirelessly connected to WPLNC based cloud. The sources as well as destinations produce independently data to be transferred by the cloud network to all the nodes on the opposite side of the cloud. The cloud itself consists of four relays divided into two layers, each having two relays. The communication is bidirectional. Both directions and all communication links are shown in Fig.5.1.

Each node knows its position and role within the network. Moreover, a sequence of transmission and reception time slots is also known in advance. Practically, when the node is not transmitting it is receiving because there is no option for this demonstrator to make it idle. If an incorrect signal is overheard by the node it is ignored. For example relay nodes from the second layer ignore direct transmission from the sources if they are able to receive it. This significantly violates an assumption of decentralised and distributed network. However, time slotting and arrangement of transmission or reception phases in a distributed way is a challenging problem to be yet solved. As such it is far beyond the scope of this thesis.

5.1.2 Physical layer

The advanced DLA is verified and tested by a test-bed build from Ettus Research Universal Software Radio Peripherals (USRPs). Each network node is represented by computer-hosted Software Defined Radio (SDR). The USRPs connect to a host computer via a Gigabit Ethernet link, which the host-based software uses to receive and transmit the baseband IQ stream. The host controls the USRP using USRP Hardware Driver (UHD) commands, including setting parameters such as amplifier gain. The USRP then performs the necessary baseband processing then up/down-conversion and transmission/reception. The Ettus Research N210 USRP is used, which consists of a motherboard and a Radio Frequency (RF) daughterboard; more details can be found in [101]. In these experiments the XCVR2450 daughterboards are used, these are dual-band transceivers with 100 mW output at 2.4 - 2.5 GHz and 50 mW output at 4.9 - 5.85 GHz.

Prior to any WPLNC communication there is a synchronisation epoch during which the local oscillators of USRPs are made synchronous. There are specific signals used for this distributed synchronisation [108]. During the WPLNC communication phase the synchronisation is maintained by dedicated orthogonal signals that I will describe later.

The physical layer of the test bed is based on an OFDM which is implemented with $N_{FFT} = 64$ subcarriers. There are 48 data subcarriers, with a DC carrier, 11 guard carriers and 4 pilot carriers at -21, -7, 7 and 21. A cyclic prefix (CP) of $\frac{1}{4}$, i.e. 16 samples, is employed and a center frequency of $f_c = 2.4$GHz is used. The USRP N210 is limited (by the Gigabit Ethernet link) to support an RF bandwidth up to 25 MHz with 14-bit resolution, though a 1 MHz bandwidth, i.e. 1 MS/s sampling rate, is used here. Each data part of the packet contains 16 OFDM symbols which is a total of 768 bits assuming 48 data carriers per symbol and BPSK modulation. No channel coding is currently implemented in the system although some errors in some parts can be detected by CRC.

The WPLNC communication frame consists of synchronisation words (used to maintain synchronisation from synchronisation epoch) and data payload and is shown in Fig.5.2. Orthogonal synchronisation part is shown in blue/red based on the transmitter and data payload is shown in gray. It is important to note that data part of the payload is not orthogonalised it directly overlaps. Notice that the orthogonal separation of synchronisation maintaining signals assumes maximally two nodes communicating simultaneously. This is a strong limitation and future research of non-orthogonal ways of synchronisation/signalling is necessary. Some initial effort is given in chapter 6.

The OFDM symbol timing is maintained using Schmidl and Cox technique which is detailed in [109]. In brief, the symbol timing recovery relies on the use of a training symbol with two identical halves in the time domain, "Sync Word 1" in Fig.5.2, which will remain identical after passing through the channel, except that a phase difference is introduced between them by the carrier frequency offset. In [109] this is achieved by transmitting a pseudonoise

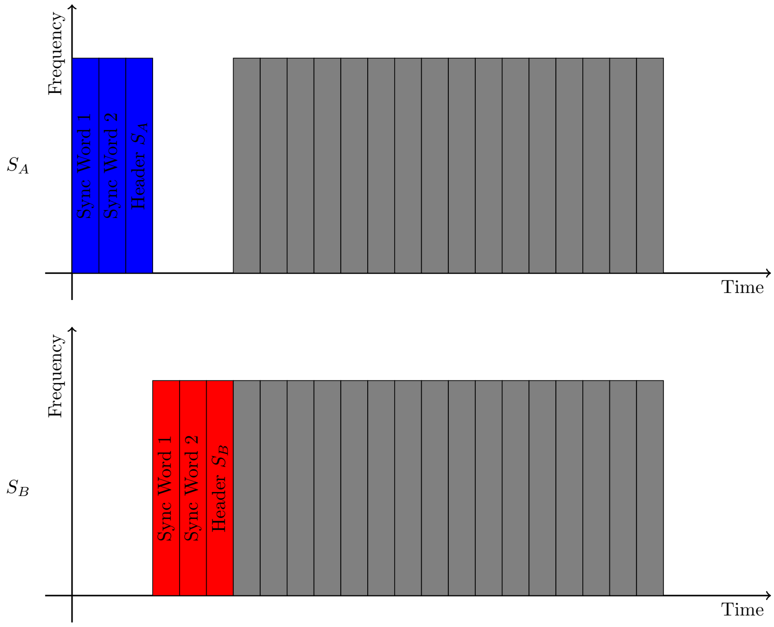

Figure 5.2: Synchronisation maintaining words and data payload frame structure.

(PN) sequence on the even carriers and zeros on the odd carriers. A second training symbol, "Sync Word 2" in Fig.5.2, contains a PN sequence on the odd frequencies to enable channel estimation and another PN sequence on the even frequencies to help determine the Carrier Frequency Offset (CFO). The PN sequence should be chosen to minimize the Peak to Average Power Ratio (PAPR). Any timing inaccuracy, i.e. sample shift, will result in a rotation in phase in the frequency domain. As long as the timing estimation error is less than the CP length, the received signal remains Inter-Symbol Interference (ISI) free [110]. Assuming this, the phase rotations cannot be distinguished from the phase delay caused by the channel, therefore they can be compensated through channel estimation and equalization [111].

During the WPLNC communication phase a carrier frequency offset among the nodes can arise from inaccuracy of the respective internal Temperature Compensated Crystal Oscillators (TCXO). Also in dynamic scenarios with moving nodes there can be additional frequency offset caused by Doppler effect. The CFO estimation is carried out as per [109], and is then applied to the received signal in the time domain:

$$\tilde{y}[n] = y[n]e^{-jn\tilde{\phi}}, \tag{5.1}$$

where $y[n]$ are the received samples in the time domain, and $\tilde{\phi}$ is the CFO estimate.

Classical orthogonal channel estimation technique is also performed. Unlike frequency and time synchronization, the channel estimation is performed in the frequency domain. It is performed after the Fast Fourier Transform (FFT) and the CP removal. At the beginning of each packet a wideband estimate can be obtained from the second synchronization symbol in the

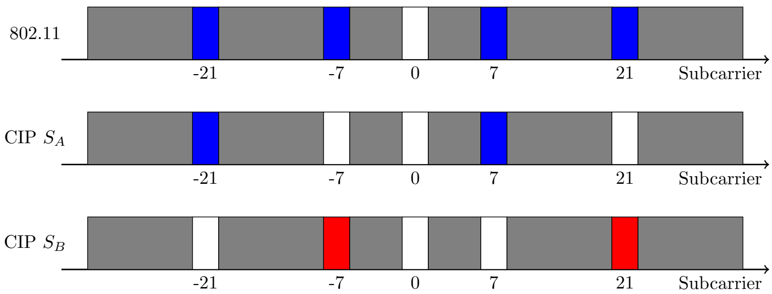

Figure 5.3: Channel estimation tracking pilot structure modified from IEEE 802.11.

preamble ("Sync Word 2" in Fig.5.2, abbreviated to "sync" in the following). The symbol transmitted from the first source in the frequency domain is $X_1^{sync}[k]$, where $k = 0, \cdots, N_{FFT} - 1$ and N_{FFT} is the number of FFT taps, i.e. 64 in this case. The synchronization symbol is a pseudo-random sequence as defined in [109] such that $X_1^{sync}[k] \in \{1, -1\}$ for all data subcarriers, and 0 elsewhere. The received samples of the first source Sync Word 2 are $y^{sync}[n]$, their frequency domain equivalent is $\tilde{Y}^{sync}[k] = \text{FFT}(y^{sync}[n])$ and thus channel estimation of $\hat{H}_1[k]$ is performed by:

$$\hat{H}_1[k] = \frac{\tilde{Y}^{sync}[k]}{X_1^{sync}[k]} \tag{5.2}$$

Depending on the coherence time of the channel and the number of symbols in a packet, the channel may have changed from the first OFDM symbol to later symbols in the packet. As a result the initial channel estimate from the preamble is no longer valid, so pilots are used to track the changes and update the channel estimate every symbol. The alteration to the 802.11 pilot design to allow for orthogonal pilots is shown in Fig.5.3, allocating two per transmitter. Subcarriers used for data are shown in gray, used pilots in blue/red and unused subcarriers in white. This means that it is possible to separately track the changes in $\hat{H}_1[k]$ and $\hat{H}_2[k]$ only.

If one considers a specific OFDM symbol, s, and allow $k' = -21$ and $k'' = 7$, i.e. the subcarriers occupied by the two pilots of the first source, then $\tilde{Y}^s[k']$ and $\tilde{Y}^s[k'']$ are the received signals in the frequency domain which contain only the signal related to the pilots of the first source due to the orthogonality of the pilots. Letting $P_1[k'] = P_1[k'']$ being pilot signal, the following are computed:

$$\Delta\hat{H}_1^s[k'] = \frac{\tilde{Y}^s[k']}{\hat{H}_1[k']P_1[k']} \tag{5.3}$$

$$\Delta\hat{H}_1^s[k''] = \frac{\tilde{Y}^s[k'']}{\hat{H}_1[k'']P_1[k'']} \tag{5.4}$$

Then linear interpolation is used to obtain $\Delta\hat{H}_1^s[k]$ where $k \neq k', k''$:

$$\Delta\hat{H}_1^s[k] = \Delta\hat{H}_1^s[k'] + \left(\frac{\Delta\hat{H}_1^s[k''] - \Delta\hat{H}_1^s[k']}{k'' - k'}\right)(k - k') \tag{5.5}$$

Finally, the channel estimate for the s-th symbol is obtained by modifying the initial estimate from the preamble:

$$\hat{H}_1^s[k] = \hat{H}_1[k]\Delta\hat{H}_1^s[k] \tag{5.6}$$

It is important to note that the use of these tracking pilots also resolves the issue of residual CFO estimation error, as can be seen from the following explanation. It is possible to model the output of the FFT of the receiver as [112]:

$$\tilde{Y}^s[k] = e^{-j2\pi s N_b \Delta f T_S} \sum_{i=1}^{2} X_i^s[k] H_i^s[k] + W[k], \tag{5.7}$$

where $X_i^s[k]$ is the modulated signal and $H_i^s[k]$ is the channel frequency response at the k-th carrier, of the s-th OFDM symbol from the i-th transmitter. The CFO estimation error (in Hertz) is given by Δf, N_b is the total number of samples per OFDM symbol, i.e. $N_{FFT} + N_{CP}$, T_s is the sampling period, and finally $W[k]$ represents the joint effect of Additive White Gaussian Noise (AWGN) and Inter-Carrier Interference (ICI) caused by CFO.

The phase term $e^{-j2\pi s N_b \Delta f T_S}$ represents the phase drift of subcarriers, ignoring the fixed phase offset caused by timing error which is consumed by the channel frequency response, which is dependent on s. This phase rotation is not dependent on frequency, though, and thus manifests itself as an additional phase term in the channel response and therefore is incorporated into the channel estimate.

After the synchronisation part of the packet there is a modified version of the hierarchical pilot (PiHRC). This part of the signal carries all WPLNC related information necessary to correctly process – combine at relays, decode at destinations – the data part. Since the communication is bidirectional and reverse direction is used to carry an information that enables DLA PiHRC has also two parts each dedicated to particular direction.

The structure of modified PiHRC is shown in Tab.5.1, where F/W and B/W is used to distinguish direction, F/W is used for current packet direction and B/W is used for the opposite side and previous packet. B/W part carries the feedback information and by reading it the node can learn what mapping is the opponent on the same layer using with the previous packet. The two three bit long fields S_i carry unique identification of terminal node ($S_1 = 000$, $S_2 = 001$, ...). Two special bit patterns are used for special occasions – $S_i = 101$ means unused field and $S_i = 111$ is used to indicate topology change and to restart DLA as will be described. Four bit long WPLNC fields carry indices to the common list of the available WPLNCs. There are also several patterns for specific purposes, particularly patter 1011 is used for empty field and patter 0001 is appropriate identity WPLNC mapping, meaning that only one source is contained in the following data packet. There are three fields for forward WPLNC mappings and two for backward, an explanation how this fields are used by DLA is given later. Vote field contains two bits for nomination used to allow adaptation according to DLA, again particular details are given later. The PiHRC pilot data are protected by

S_X	S_Y	F/W WPLNC1	F/W WPLNC2	F/W WPLNC3	B/W WPLNC1	B/W WPLNC2	Vote	CRC
3 bits	3 bits	4 bits	4 bits	4 bits	4 bits	4 bits	2 bits	4 bits

Table 5.1: Modified PiHRC structure

Cyclic Redundancy Check (CRC) code with polynomial $x^4 + x^3 + x^2 + 1$ resulting in four last bits.

Since the network topology as well as the structure of synchronisation maintaining signals assume maximally two concurrently transmitting nodes also the structure of PiHRC is designed in that way. It is important to note that no optimisation of PiHRC was done. It is designed open enough to meet the requirements of the demonstration test-bed. Better suited and more flexible structures should be found. The PiHRC length and its content capacity should be increased by various space-efficient data structures [113, 114]. This can also limit the overhead connected with WPLNC, which is nevertheless small for current test-bed and is in fact unavoidable as pointed in [100].

All 32 bits of PiHRC are mapped in pairs to QPSK symbols which are then used to modulate orthogonal CAZAC sequences. There is a dedicated CAZAC sequence to each node which uniquely identified it. The similar CAZAC sequence is used for distributed network-wide synchronisation.

After the PiHRC part of the packet there are finally useful data. This data part uses various single dimensional liner modulation over OFDM subcarriers. Terminal nodes use BPSK for all the time. The modulation used by the relays depends on their position in the network (particularly on the number of incoming bits) as well as on type of WPLNC mapping used - minimal, extended or full one. Particularly modulations from set of BPSK, QPSK, 8PSK and 16QAM are used.

5.1.3 Advanced Distributed Learning Algorithm

Each source produces data payload by reading actual state of sensors (temperature, luminosity,...). It constructs its PiHRC message – particularly, S_X field is filled by its unique number, S_Y is filled with empty sequence, F/W WPLNC1 is filled by preassigned number meaning that only one source is contained in the message, other WPLNC fields are filed with empty sequence, one bit from vote field is randomly set to one, the other is zeroed and finally protecting CRC field is computed.

Appropriate pilot signals are created and the signal is broadcast using OFDM physical layer forming the described frame structure. A BPSK modulation is used to modulate data part of packet to data OFDM subcarriers.

Superposition of source transmissions is received by each of the relays from the first layer. The relay processing is given by a pseudocode 3. The same processing is used also by the second layer of relays where the first layer plays a role of data sources, although there are some small modifications as will be explained next.

As described in previous section synchronisation and channel estimation is done (line 2) using pilot signal part of OFDM frame. When especially channel

Algorithm 3 Advanced DLA

```
1:  Receive signal
2:  Perform CFO, TO corrections and channel estimation
3:  Decode all PiHRCs
4:  if Error in CRC then
5:      Drop packet
6:  end if
7:  if Warning packet is detected then
8:      Stop operation, forward warning packet
9:  else
10:     if Change in incoming signals then
11:         Revert to full mapping corresponding to actual state
12:     else
13:         if Elected by voting then
14:             Update own WPLNC
15:         end if
16:     end if
17:     Crate new PiHRC - update source, WPLNC and voting
18:     Process superposed date by actual WPLNC mapping
19:     Crate output signal and broadcast
20: end if
```

is estimated all PiHRC data can be decoded. If there is an error, it is indicated by CRC (line 4) and erroneous packet is dropped. The most important part of algorithm starts at line 10. Here if any changed in number and/or shape of incoming WPLNCs is detected then the relay's actual WPLNC mapping is reverted to a full mapping that corresponds to the actual situation. A change or warning can be detected primarily on four occasions:

- At the very beginning of the WPLNC communication phase. As in classical DLA the relay starts from safe full WPLNC mappings which are initially set by this if-branch of the algorithm.
- When a node disconnects from the network or reconnects back. In this case the WPLNC has changed significantly when compared to the previous packet and thus invertibility condition can be violate. It is thus safe to return back to full mapping and restart the convergence process again to cope the change.
- As a special case of the previous item, when optimal allocation of WPLNC mappings is achieved in the layer closer to destinations the next layer is allowed to start mapping adaptation. Any change in this outer layer, however, will change the situation for the inner layer that should step back. This layered mechanism of advanced DLA will be described later.
- When the invertibility of the WPLNC mappings is not accidentally fulfilled, e.g. when a relay fails or disconnects from the cloud, then any

destination must send a warning packet to restart the network. The warning packet is identified by specific data filled in S_i part of PiHRC, this is identified by the relays that continue to broadcast warning packets until a valid packet is received and a safe full WPLNC mapping is automatically reused by this mechanism.

When there is no crucial change and all WPLNC mappings are valid as they where with the previous packet the relay may be given a chance to be elected by the previous layer of nodes to change its WPLNC mapping according to DLA (line 13. This is a implementation that allows only one change at particular time instant. The relay node is elected when all previous nodes set corresponding voting bit to 1. Otherwise it is not elected. When the relay is elected it uses information from B/W WPLNC fields in PiHRC to learn what are the other relays in the same layer doing and by following reply dynamics (see chapter 4) it can chose a new WPLNC mapping that improves its utility function and does not violate invertibility condition.

When a WPLNC mapping that is going to be used with the actual packet is selected then relay updates PiHRC data content and processes received superposition by the new WPLNC mapping. PiHRC data update consist mainly from updating source, WPLNC and voting fields. Particularly, the relays from the first layer fill all the source ids from all the sources that they receive into fields S_i. The relays from the second layer does not alter this fields and it is important to not that only ids of the original sources are transferred by PiHRC, relays are not reported.

The relays from the first layer also fill the id of WPLNC mapping that they are using for this particular data packet into the field F/W WPLNC1. The relays in the second layer fill fields F/W WPLNC1 and F/W WPLNC2 by the information they read from incoming PiHRC from the relays from the first layer and F/W WPLNC3 is WPLNC mapping actually used by the second layer relay. B/W WPLNC fields are always filled by the WPLNC ids received with the previous packet coming in the opposite direction. These are used for DLA operation to learn what the opponent is doing and to find out all invertible WPLNC mappings. Voting field is filled by random setting of exactly one of its bits to logical one, all other bits are zeroed. Sources vote immediately and forever, the relays in the second layer start to vote when they achieve their best WPLNC mapping. Voting is irrelevant for the first layer of relays since there are only sources upstream. Voting of the second layer can be temporarily stopped when there is a step back in their WPLNC mappings caused by the change in WPLNC mapping of the first layer.

This advanced DLA algorithm is performed in a layered manner. The layer closer to actual information destination starts first with WPLNC adaptation until the optimal allocation is achieved. By this the second layer starts to vote relays from the first layer that can adapt WPLNC mapping when elected. Since a change of WPLNC mapping in the first layer causes a change of incoming WPLNC mappings to the second layer, which may violate invertibility of overall WPLNC mapping, a step back to actual full mapping in the second layer is necessary.

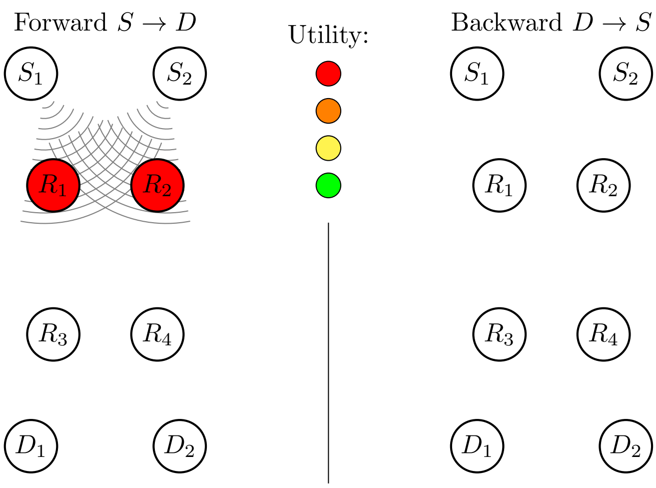

Figure 5.4: Advanced DLA - step 1.

The advanced DLA operation is illustrated step by step in the following figures 5.4 - 5.13. Some important steps of DLA algorithm are shown and explained. A semaphore-like color code is used to show utility improvements.

Fig.5.4 shows very first packet of WPLNC communication. The first layer of relays receive a superposition of two source signals and since it represents a change in WPLNC mapping according to line 10 all relays set their WPLNC mapping to full one. Particularly a mapping that maps two times one bit (two BPSKs from sources) to two bits (QPSK used by relays) is selected. Because of change any voting from the sources is ignored. The relays R_1, R_2 fill fields S_X, S_Y of their PiHRCs by ids of S_1, S_2, their newly used WPLNC mapping is put into F/W WPLNC1 field. All other fields of PiHRC are irrelevant.

Very similar situation is for the second layer of the relays, shown in Fig.5.5. Actual full mapping is set because of change. Now it is a mapping of two times two bits (two QPSKs from relays from the first layer) to four bits (16QAM used by the second layer relays). F/W WPLNC1 and F/W WPLNC2 contain WPLNC mappings used by R_1 and R_2 respectively and F/W WPLNC3 is filled by id of selected full mapping. All other fields are irrelevant now.

The first packet from sources is delivered to destinations on Fig.5.6. WPLNC applied by the cloud network is invertible since full mappings are used everywhere.

The first backward packet is transmitted by the destinations, see Fig.5.7. The WPLNC mapping of R_3 and R_4 in backward direction are set to full mapping because of change if-clause (line 10). However, since destination are already voting and are also providing by B/W WPLNC fields WPLNC mappings of R_3 and R_4 from previous S to D packet, thus R_3 and R_4 have all necessary information to start WPLNC adaptation by DLA. Let us assume that R_3 is elected by both destination and it changes its WPLNC mapping to be used with the next S to D packet to something better then the original full mapping.

The first backward packet is delivered to the sources in Fig. 5.8. Because

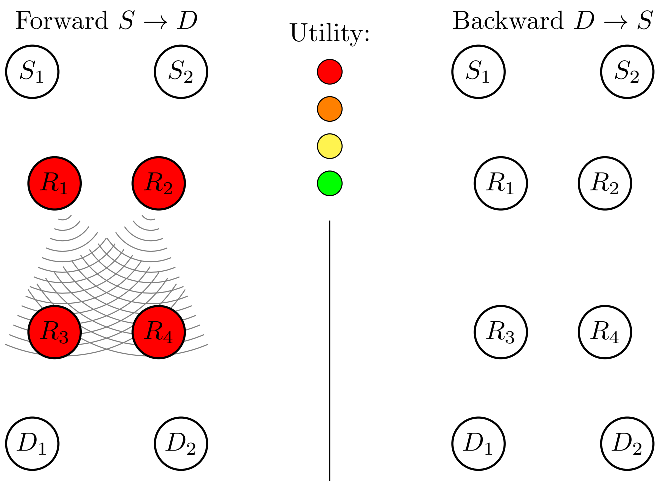

Figure 5.5: Advanced DLA - step 2.

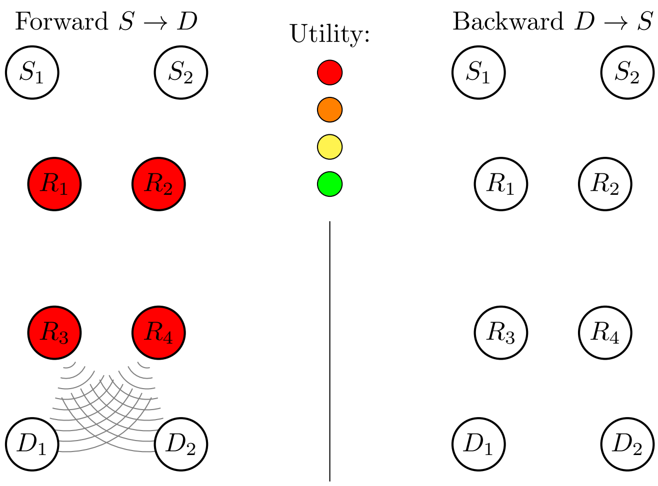

Figure 5.6: Advanced DLA - step 3.

of full mappings used it is successfully decoded. PiHRC signal of this packet carries information that will be used with the next packet in opposite direction to enable DLA for R_1 and R_2 in D to S direction.

The second packet is transmitted by the sources to destinations in Fig.5.9. Now voting and available information allows R_1 or R_2 to improve their utility by DLA. Assume that R_2 is elected and changes its mapping to something better than initial full mapping.

After several iterations (duration depends on voting process as well as selection of utility improving mappings) an optimal allocation of WPLNC mappings in outer layers is possible, see Fig.5.10. Particularly, all of the first layer relays (with the red utility level) still use full mapping (mapping two BPSKs to one QPSK) since there was no chance for them to adapt yet and all relays from the second layer (with yellow utility level) use corresponding minimal mappings (mapping two QPSKs to one QPSK). In this case all relays

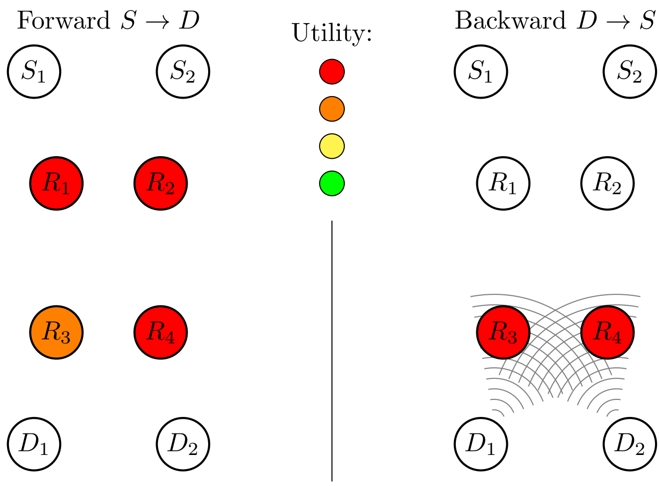

Figure 5.7: Advanced DLA - step 4.

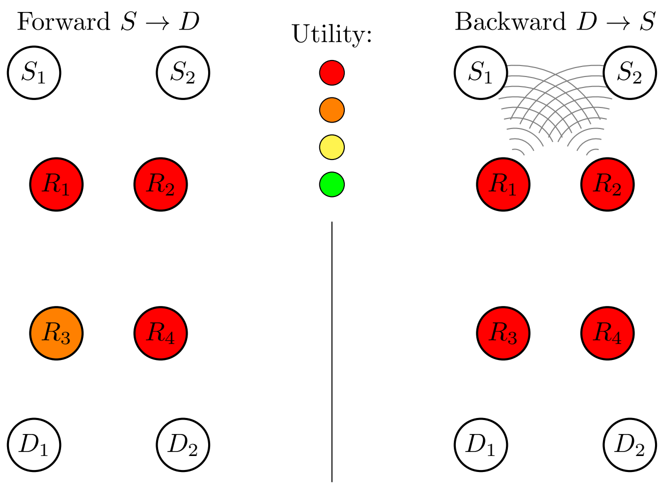

Figure 5.8: Advanced DLA - step 5.

from the second layer are voting the relays from the first layer to allow DLA also in this layer.

Situation in Fig.5.11 shows how WPLNC adaptation is allowed in the first layers. Backward packet carries information about WPLNC mappings used by R_1 and R_2 in previous forward communication together with votes. Assume that now R_1 is elected and improves its utility in the best way to minimal mapping (mapping two source BPSKs to one BPSK).

A change in WPLNC mappings used by the first layer however changes a situation for the second layer. With the previous packet the second layer relays were processing two QPSKs, now they have one BPSK and one QPSK. This may however violate invertibility if the overall WPLNC mapping or even the dimensions of underlying matrix representation may not allow their multiplication, recall WPLNC mapping encapsulation discussed for matrix description in chapter 3. A proper reaction of the second layer is to step back

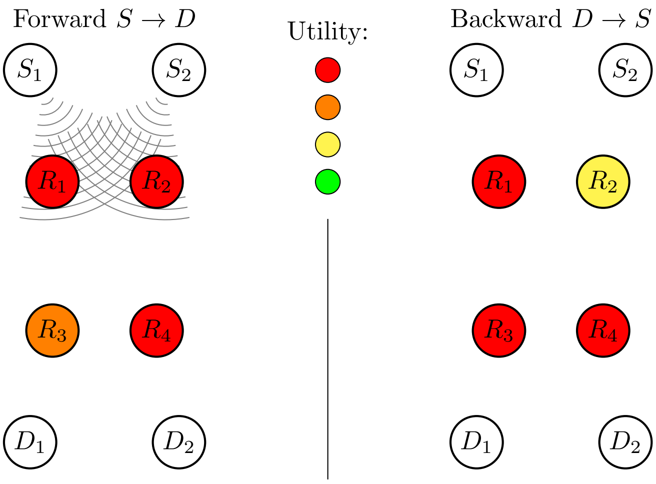

Figure 5.9: Advanced DLA - step 6.

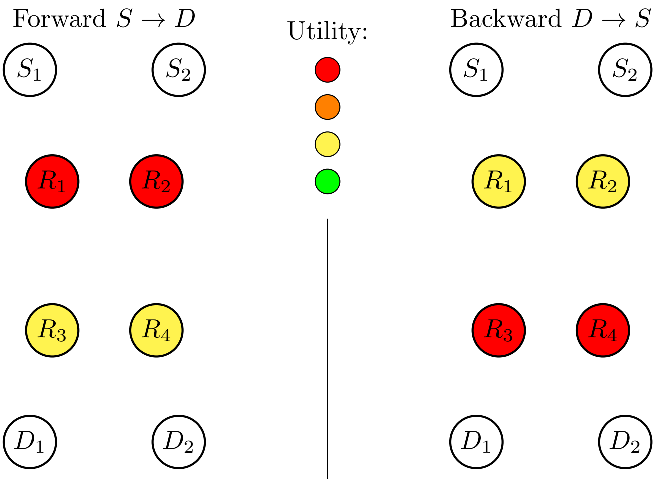

Figure 5.10: Advanced DLA - step 7.

to actual full mapping. Notice that full mapping in this case maps three bits (one BPSK and one QPSK symbol) to three bits (8PSK). Even though there is a step back the actual full WPLNC mapping has better utility than the full mapping used at the very beginning of the communication. Such step back is necessary and can be avoided if and only if also the first layer of the relays would have knowledge of WPLNC mappings used by the second layer. In this case DLA should work over the whole network. Obviously the overall overhead will be much bigger and coordination much harder. Possibly it may not be even doable in a distributed way without centralised control, especially in much bigger networks.

After several step backs (usually only a few of them is needed) the optimal allocation of WPLNC mappings is achieved as shown in Fig.5.13. The achievement is guaranteed by the properties of DLA (see chapter 4) which are not violated in this advanced implementation. With the optimal WPLNC

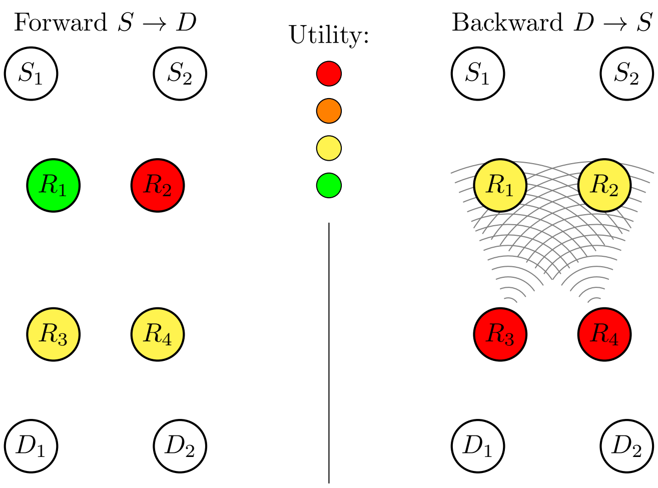

Figure 5.11: Advanced DLA - step 8.

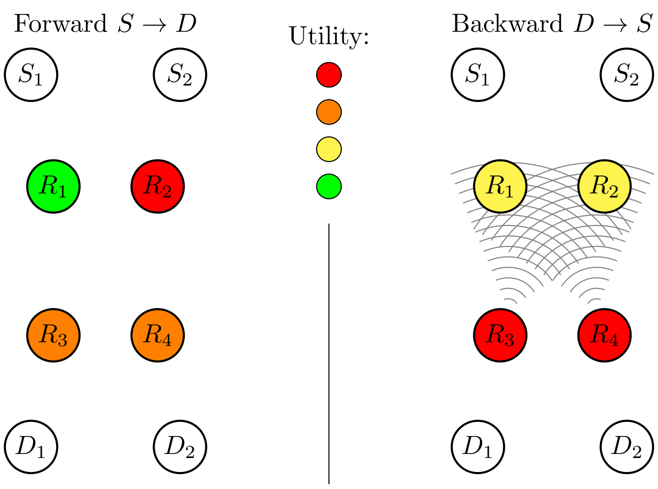

Figure 5.12: Advanced DLA - step 9.

mapping assignment minimal mappings are used across the whole network mapping two bits (two BPSKs) to one bit (one BPSK).

Prior to HW implementation and verification by office space demonstrator the advanced DLA algorithm was tested by numerical simulation. Obviously no RF signals were generated as well as no wireless medium issues such as channel estimation or synchronisation were resolved. It was mainly used to verify logic and sequence of steps of the proposed algorithm.

Typical progress of advanced DLA run is shown in Fig.5.14 where number of relays' output bits is recorded for communication in direction from sources to destinations. Relays R_1 and R_2 start with full WPLNC mapping (corresponds to two output bits of QPSK symbol) while R_3 and R_4 use also full WPLNC mapping, however, sending out four bits by 16QAM. WPLNC mapping adaptation starts between R_3 and R_4 and around the 15th packet the optimal (given that the first layer has not yet started with optimisation) allocation is

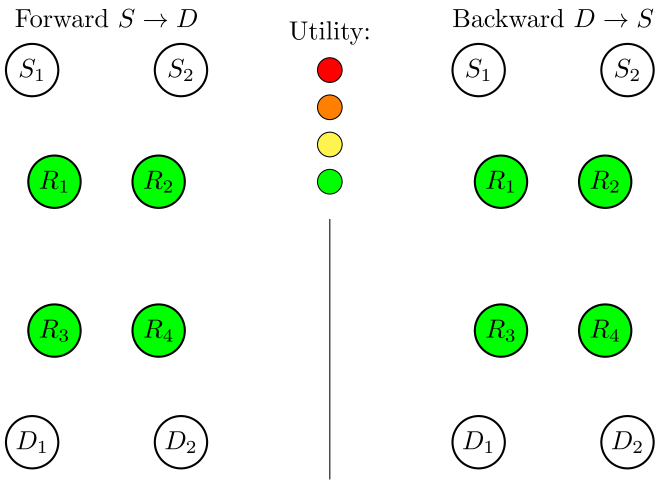

Figure 5.13: Advanced DLA - step 10.

achieved. With packet 21 relay R_2 is elected to made the first improvement also in the first layer. This change is reflected by step back of relays R_3 and R_4 to use actual full mapping. This step back situation is repeated around the 33th packet when also R_1 improves its utility. The global optimal allocation is achieved before the 50th packet is sent. Notice that the total number of transmitted bits changed from twelve to four. There are obvious relatively long periods of no activity. During these stages no relay is elected by voting mechanism. It is an unavoidable phenomenon of distributed implementation and of a need of only one change of WPLNC mapping at each time.

Thanks to warning packets the advanced DLA algorithm can successfully survive disconnection or failure of any relay node. Because of given and fixed time slotting multiple relays can disconnect from the network until there is at least one relay in each layer. The warning packet is transmitted by any destination that is unable to decode the WPLNC packet because the WPLNC mapping is not invertible. Because of the properties of DLA this can happen only if some part of the information is missing, for example if a relay is lost during the transmission of the packet. The warning is raised by specific sequence send in S_i field of the PiHRC message. Whenever the warning packet is received by the relay (see line 7) it causes that WPLNC operation is stopped, all actually used WPLNC mappings are discarded and the warning packet is forwarded to the next parts of the network. When there is again a valid packet send from the terminal nodes the operation of advanced DLA is restored and normal WPLNC operation mode is used by change detection mechanism (see line 10) and an appropriate full mapping is used as a new starting point.

An example of advance DLA algorithm under failure of one relay is show in Fig.5.15. Again the number of relays' output bits is shown. Notice that with the 14th R_3 is disconnected from the network. This (probably) leads to a loss of this packet. The warning packet is send across the network and all WPLNC mappings are returned to appropriate full mappings with the

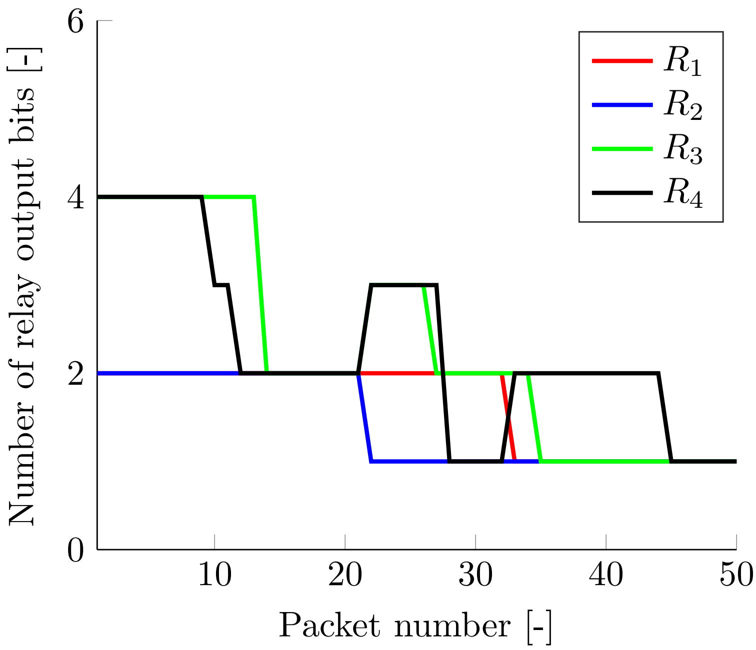

Figure 5.14: Advanced DLA - simulation.

15th packet. Advanced DLA is restarted from this new starting point and optimal allocation is achieved around the 23rd packet. Also notice that now R_4 could not get down to BPSK since it must use a full mapping for two incoming BPSK transmission from R_1 and R_2 because it is a sole relay within the second layer. Nevertheless this is still the best and the optimal allocation of WPLNC mappings for this network.

5.1.4 Real-world demonstration

The demonstration of advanced DLA system was hold in March 2016 at the Toshiba Telecommunications Laboratory premises in Bristol, the United Kingdom, during the final review meeting of DIWINE project. The main goal was to validate advance DLA design in realistic scenario of office building.

The four relay node cloud plus four terminal node network was established using USRPs. Its compact placement is shown in Fig.5.16. 3D printed houses, see Fig.5.17, were used to house sources of useful information when their internal temperature and luminosity were measured by sensors and transferred by USRPs to the opposite side of the cloud. This is a very simple model of smart-metering IoT network.

For demonstration the network nodes were scattered across an office area consisting of several compartments. Approximate node positions are illustrated in Fig.5.18 where obstacles such as walls, doors, pillars and furniture are also depicted.

Snapshots of GUI of running demonstration are shown in Figs.5.19 and 5.20 where forward (S to D) and backward (D to S) packets are shown,

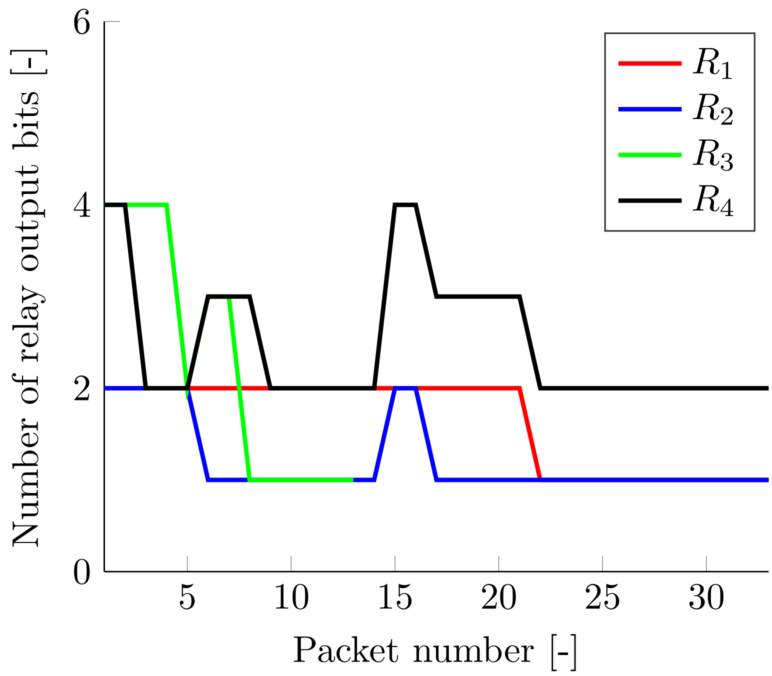

Figure 5.15: Advanced DLA - simulation - relay disconnected.

respectively. Relay output constellations used for the transmission at that particular time instant are depicted close to relay nodes. The user can observe them changing during the run of the demonstration through GUI. Sensor data from the sources are shown in the upper right corner and the number of actually successfully transmitted packets in the lower right. The relays can be switched off/on simply by clicking on the node in the GUI.

All proposed abilities of advanced DLA demonstrator, such as optimal WPLNC allocation, adaptability to changing network topology and ability to recover from node failures were successfully verified in highly realistic conditions of office space environment. No exact measures of convergence speed, throughput, reliability, etc. were taken although the WPLNC cloud network was compared to software emulation of classical network with orthogonally separated transmission as well as to a WPLNC network with fixed WPLNC mapping allocation.

When compared to classical system the demonstrator showed significant throughput increase – two time slots are needed in the classical case for transmissions between any two layers compared to single slot for WPLNC. A direct comparison between real-time running WPLNC network with the advanced DLA (upper part) and software emulation of classical orthogonal network (lower part) is shown in Fig.5.21. Notice especially the number of successfully transmitted and received packets in the lower right corners. Also gaps in the measurement data are visible in the upper right corner of orthogonal network emulation since the packet were not delivered in time compared to WPLNC network.

Figure 5.16: Compact demonstrator network.

The fixed WPLNC mapping allocation could not resolve topology changes caused by relay disconnection. Under this condition it completely failed and communication was impossible.

5.1.5 Summary

A highly advanced demonstration of self-adaptation algorithm for relatively complex cloud network is provided. The demonstrator consists of four outer terminal nodes and four internal relay nodes. Distributed Learning Algorithm is modified to fit this situation and equip the cloud with abilities to allocate the optimal WPLNC mappings in a distributed way. These modification allows reliable bi-directional communication provided to the terminal nodes. The cloud itself is able to adapt to changes in its internal structure.

As much realistic deployment as was possible was done. The network was deployed in an office area such that the communication faced typical application environment. It smoothly worked over typical obstacles (doors, walls, furniture, etc.) with people moving inside the environment.

Also a grate care was given to make the physical layer realistic. All necessary signals were transmitted wirelessly without any cheating. There were of course several simplifications, mainly time slotting of transmission/reception phases is given and globally known and followed, also orthogonal signals were used to maintain time and frequency synchronisation and channel estimation.

Despite these limitation the presented demonstrator is, to my best knowledge, far more advanced demonstrator of WPLNC principle in self-adapting networks. There are only a few of other demonstrators of WPLNC [115–120] each mostly assuming only one relay network with two terminal nodes. In this case a fixed XOR WPLNC mapping is commonly used and thus there is no distributed WPLNC mapping allocation. It is highly important to note that advanced DLA algorithm will work correctly also in this simple scenario.

Figure 5.17: 3D printed houses used for smart-metering demonstration.

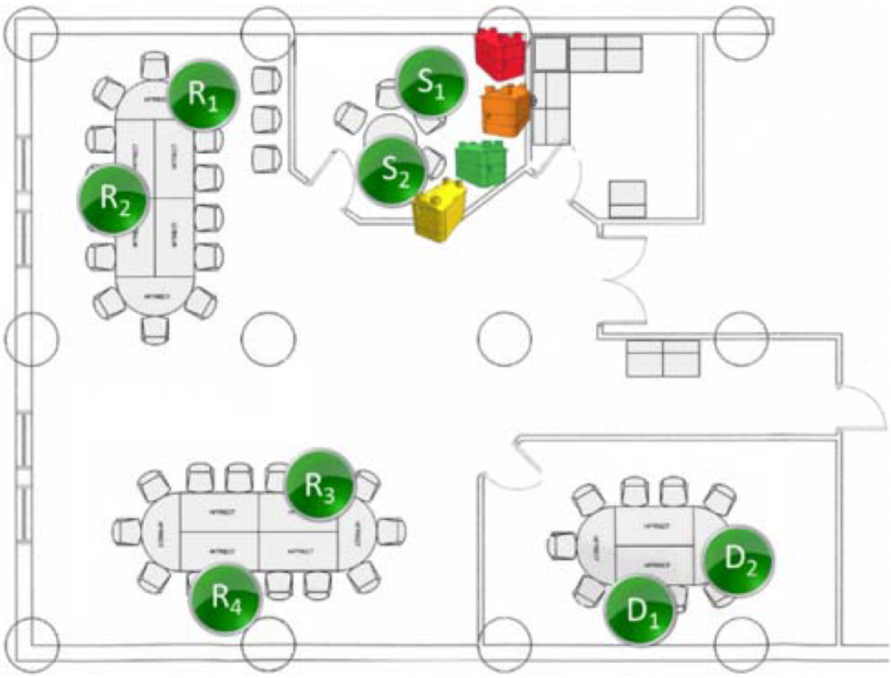

Figure 5.18: Illustration of demonstrator deployment in the office area.

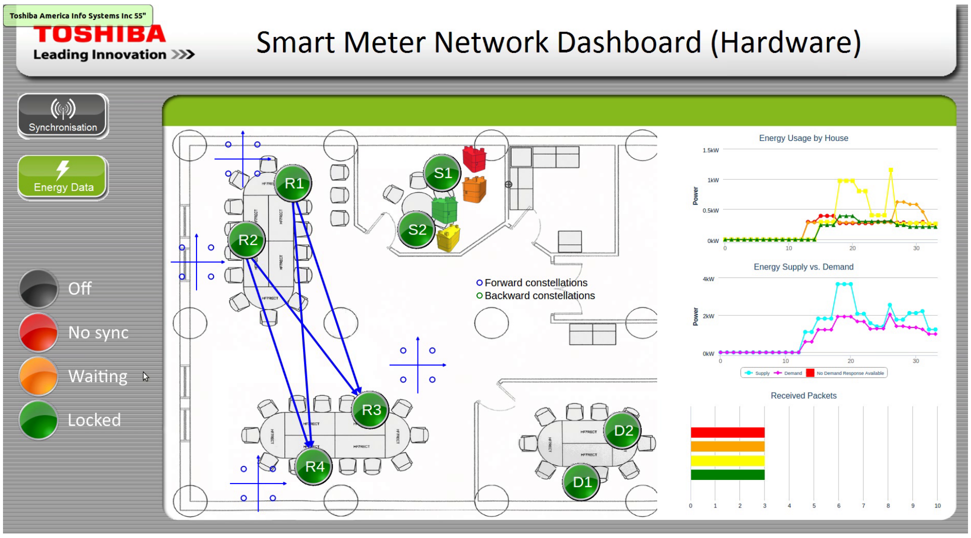

Figure 5.19: Snapshot of GUI of forward packet transmission.

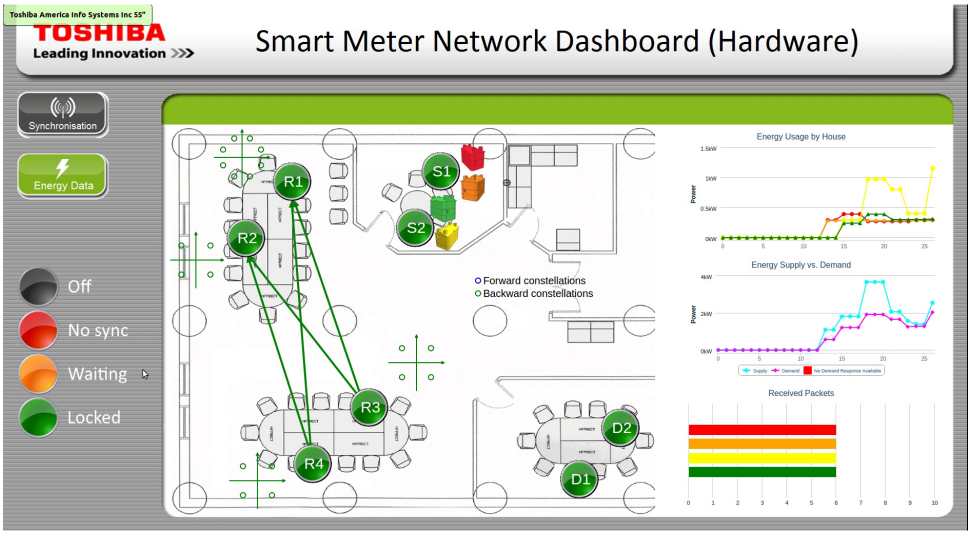

Figure 5.20: Snapshot of GUI of backward packet transmission.

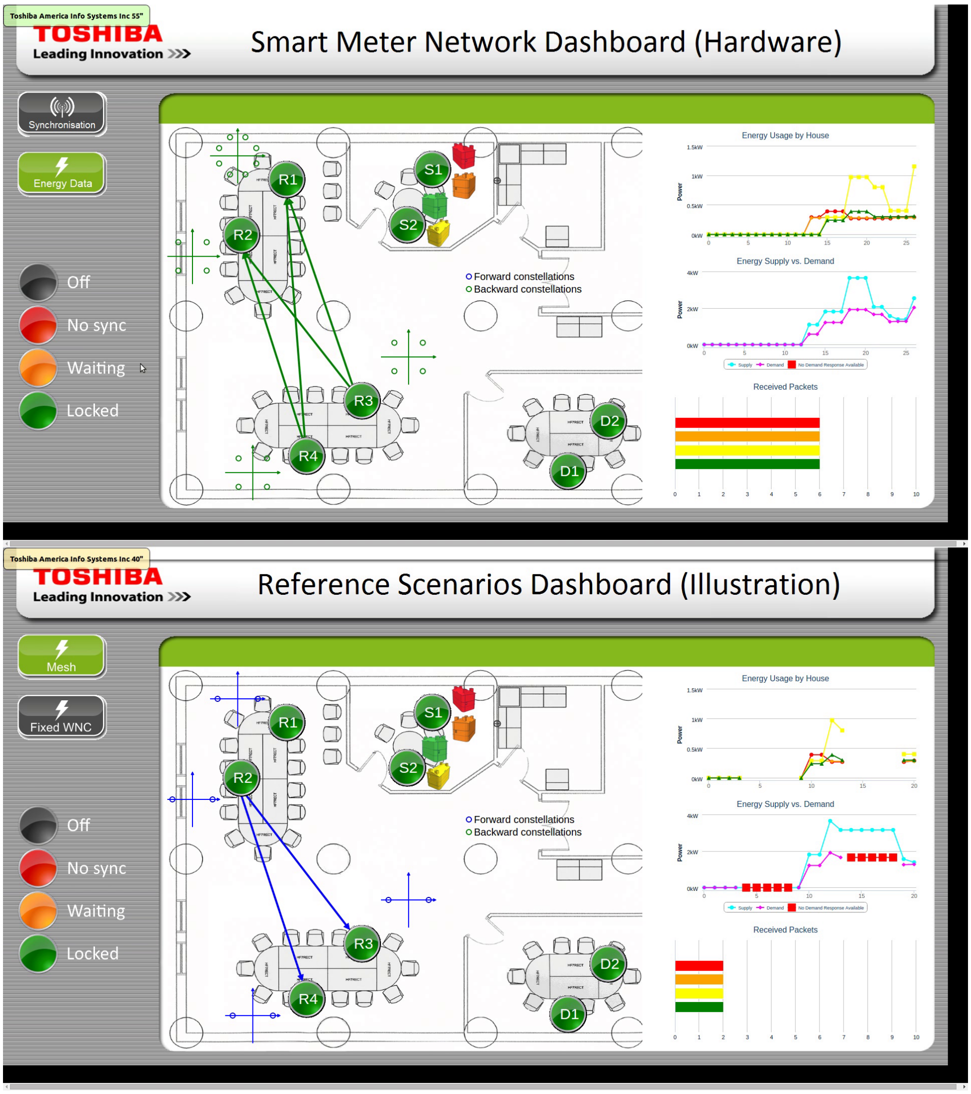

Figure 5.21: Snapshot of full GUI with comparison to classical system.

5.2 Malicious relay - incomplete information game

A creation of invertible WPLNC mapping is a collaborative task of multiple relays. The most of current research work optimistically assumes that all the relays altruistically cooperate on their task of source to destination communication. As was shown in the chapter 4 in distributed networks it is relatively easy to create a non-invertible mapping even in the case of relays that behave friendly. A maliciously behaving relay that is present inside the network can intentionally cause significant harm to the network simply by taking inappropriate WPLNC mapping. It can of course also transmit completely nonsense data to cause a disorder, but it is not studied here.

The question is if there is any protecting mechanism that can friendly relays use to minimise impact of the malicious relays. As usual I am interested in distributed decision making taken by the relays and thus the problem is solved in terms of game theory, by proper design and analysis of rules of a game.

However, a type - either friendly or malicious - of relay behaviour is only its private information unknown to the other relays nor the destinations. By selecting its own WPLNC mapping the malicious node attacks the friendly behaving relays as well as the destinations. All the relays - both friendly and malicious - are rewarded/penalised according to how well their perform their appropriate tasks.

I describe the scenario as a static (single shot) incomplete information game among the relays. Incompleteness of the game comes from the existence of the private information. The game rules are thus not globally known. I mostly focus on conditions for existence of particular equilibrium points given the probability of malicious node existence and the valuation of the utility functions.

The presence of maliciously behaving node among the fair players is widely analysed in the area of sensor networks. Game theoretical approach to coexistence in point to point scenario is presented in [121]. Relay networks (with single or multiple relay) with malicious nodes are in [122, 123] however both are single source cases. To my best knowledge there is no work yet analysing the game theoretical aspects of maliciously behaving relay in multi-source multi-relay scenario.

The work presented here is mostly based on my work [106].

5.2.1 Network model

A network of interest consists of two independent sources S_1, S_2, two relays R_R, R_C, where subscript R stands for row and C for column as usual, and a destinations D_k, see Fig.5.22.

The important aspect is that the relay R_C is always "friendly" - its aim is always to support source to destination communication and it is rewarded for this task. However the relay R_R is not always friendly, with some probability it can behave maliciously. When this relay is malicious its aim is to cause

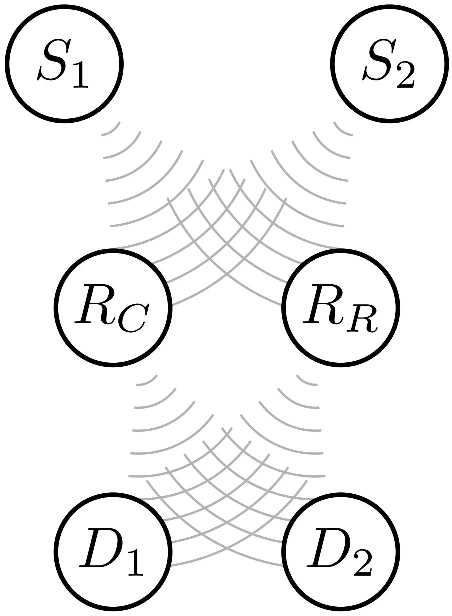

Figure 5.22: Network topology - malicious relay game.

harm to both R_C and the destinations by using such a WPLNC mapping that prevents Ds from recovery of the source data. The malicious relay is also rewarded for the harm caused. The relay R_C does not know the type of the R_R while R_R itself knows it.

5.2.2 Incomplete information game

To analyse and study the problem of distributed selection of WPLNC mapping by the relays in scenarios with potential presence of malicious relays I describe the situation as an incomplete information non-cooperative game of two players R_R, R_C. The game is the incomplete information game since the relays do not know the type (friendly/malicious behaviour) of the other relay. Thus the players do not know all the information relevant for their decision, a part of the information is private, revealed only to one of the nodes. This is a big contrast to fully known game rules studied in chapter 4.

A standard method to cope with an *incomplete* information game is to transform it to an *imperfect* information game, where the history of the turns is not fully known. This is done by Harsanyi transform [124]. Practically, a new player that plays only once and always at the beginning of the game is introduced. This new player is usually called a Nature and assigns types to the rest of the players. Particularly, it this case the Nature assigns a type t_R (Friendly/Malicious) to relay R_R at the beginning of the game. This transforms the game to a shape that is solvable by standard game theoretic tools. All the nodes' uncertainties about the game (behaviour type of the other relay in our case) are transformed into the pay-off uncertainties. The Nature's turn is a lottery but its prior probability is a common knowledge available to all players. After the type of each relay is revealed by the nature, the relays simultaneously choose their action. Based on the actions played and also on the players' types the pay-off is received by the players.

More formally the game is defined as follows.

- The set of players consists of R_R and R_C. There is also an artificial player – the Nature – that assigns type to R_R.
- The set of players' types is $t_R \in \{\mathrm{F}, \mathrm{M}\}$ and $t_C \in \{\mathrm{F}\}$, where F stands for Friendly and M for Malicious. The type of R_R in particular realisation of the game G depends on the Nature's turn - friendly version is chosen with $\Pr\{t_R = \mathrm{F}\} = p$ and malicious with $\Pr\{t_R = \mathrm{M}\} = 1 - p$. Notice that for the sake of simplicity we assume that only R_R has two possible incarnations, R_C is always friendly, i.e. $\Pr\{t_C = \mathrm{F}\} = 1$.
- The set of actions of each player consists of several various WPLNC mappings f_i. To describe the problem generally I do not exactly specify the mappings f_i now. Although the players' action sets can be much wider, the minimal action subset is as follows: two different minimal mappings $\mathrm{MIN}_{1,2}$, two different extended ones $\mathrm{EXT}_{1,2}$ and one full mapping. This set is capable of describing all the situations that can occur in specific instances of the game - such as two minimal mappings that form an invertible pair (i.e. their inverse f_j^{-1} exists), a non-invertible pair of the minimal and the extended mapping (i.e. their inverse f_j^{-1} does not exist), etc. For the sake of simplicity I assume the minimal action set to be common to both players.
- In complete information games the terms action and strategy more or less coincide. The situation for incomplete information games is quite different. A strategy is a function that for every player's type assigns an action from the set of actions. Thus the strategy for player R_R, with two possible types, is a doublet $s_R = (a_R(t_R = \mathrm{F}), a_R(t_R = \mathrm{M}))$, e.g. $s_R = (\mathrm{MIN}_1, \mathrm{MIN}_2)$ means that the relay R_R uses MIN_1 when it is the friendly relay while its malicious version uses MIN_2. Since there is only one type of R_C its strategy coincides with its action, e.g. $s_C = \mathrm{EXT}_2$. I will use notation $s_i(t_i)$ to denote particular action used by player i of type t_i.
- The players have beliefs about the opponent types, such as $p_R(t_C|t_R)$. In the analysed game it is assumed that player's belief is independent of the player's own type, e.g. $p_R(\mathrm{F}|\mathrm{F}) = p_R(\mathrm{F}|\mathrm{M}) = 1$.
- Finally there is an utility function π_i of each player that assigns a reward to the player i. In contrast to complete information game the pay-off function of the player i depends not only on the actions played by all the players but also on their types, i.e. $\pi_i(a_R, a_C; t_R, t_C)$ where a_R, a_C are actions of R_R, R_C and t_R, t_C are their appropriate types. For simplicity I assume the utility function does not depend on the actual channel state. Rather it is a function of cardinality and also types of players.

F vs. F	MIN_1	MIN_2	EXT_1	EXT_2	FULL
MIN_1	P, P	$\mathbf{A}, \mathbf{A}$	A, B	P, P	A, C
MIN_2	$\mathbf{A}, \mathbf{A}$	P, P	P, P	A, B	A, C
EXT_1	B, A	P, P	B, B	P, P	B, C
EXT_2	P, P	B, A	P, P	B, B	B, C
FULL	C, A	C, A	C, B	C, B	C, C

Table 5.2: Pay-off matrix Friendly vs. Friendly game.

M vs. F	MIN_1	MIN_2	EXT_1	EXT_2	FULL
MIN_1	W, P	P, A	P, B	W, P	P, C
MIN_2	P, A	W, P	W, P	P, B	P, C
EXT_1	P, A	W, P	P, B	W, P	P, C
EXT_2	W, P	P, A	W, P	P, B	P, C
FULL	P, A	P, A	P, B	P, B	P, C

Table 5.3: Pay-off matrix of Malicious vs. Friendly game.

Since there are two types of the player R_R two pay-off matrices can be constructed to describe two game incarnations. The first one, when both relays are friendly, is shown in Tab.5.2. The second one, when the R_R is malicious is in Tab.5.3. In both matrices the column is determined by the action played by R_C and the row by the action of R_R. The entries of the matrices have a form $\pi_R(a_R, a_C; t_R, t_C), \pi_C(a_R, a_C; t_R, t_C)$, the first number is a reward of R_R of type t_R when a pair of actions a_R, a_C is played and type of R_C is t_C. The second number gives a reward of R_C of type t_C. Notice that those two games are complete information games, it is the Nature who selects which one of those games will be played by selecting the players' types.

For friendly players, see Tab.5.2, a utility function is assumed, such that the pay-off is determined only by the cardinality of the mapping (MIN, EXT, FULL) and their invertibility. The friendly relay (R_C always, R_R if its type t_R is Friendly) is rewarded by A for using minimal mapping, B for the extended one and C for the full mapping. However when the friendly relays select a non-invertible pair of mappings, that do not allow the destinations to recover the source data, both relays have to pay a non-positive penalty P as a punishment. This situation is expressed in Tab.5.2 by (P, P) entries. In this case both relays play the very same game (one of its instances) as was described and analysed in the chapter 4. The pay-off preference for friendly relays is given by

$$A > B > C > 0 \geq P, \tag{5.8}$$

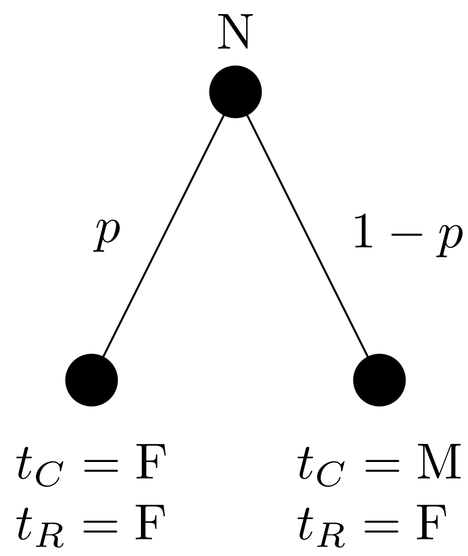

Figure 5.23: Order of turns and rules of the malicious relay game.

The malicious relay (R_C never, R_R if its type t_R is Malicious) is rewarded for the caused disorder, see Tab.5.3. It obtains a pay-off W whenever it selects a WPLNC mapping that forms a non-invertible pair with the friendly relay. Under this condition the destination is unable to decode the source data and thus the friendly relay is transmitting and spending its resources in vain. When the malicious relay node selects a mapping that forms invertible pair with the friendly node it has failed in its task to cause harm to the network. In this case the malicious relay is in fact supporting the friendly relay and thus spends in vain its resources and it is penalised by P. The pay-off preference of the malicious relays is given by

$$W > 0 \geq P, \tag{5.9}$$

thus the malicious relay prefers to create the non-invertible pair. It can be easily seen that the full mapping is never optimal for such a relay since it is always invertible.

The incarnation of the is given by the Nature's turn. Fig.5.23 shows a game tree of this game. It is important to note that the game is played only once and both players turn simultaneously when the types are revealed. Sequential play is studied in the next section in a different game example. Repeated games lay beyond the scope of this thesis, however.

5.2.3 Equilibrium analysis

In static (single shot and simultaneous turn) incomplete information games the solution concept of Nash equilibria is usually extended to so called Bayesian Nash equilibria (BNE) [124]. Basically, it is an equilibrium that maximises an expected utility function given the beliefs about the player types.

A strategy profile $s^* = (s^*_R, s^*_C)$ is a pure strategy Bayesian Nash equilibrium if for each player i and each of its type t_i holds

$$s^*_i(t_i) = \arg\max_{a_i} \sum_{t_{-i}} p_i(t_{-i}|t_i)\pi_i(a_i, s^*_{-i}(t_{-i}); t_i, t_{-i}), \tag{5.10}$$

where subscript $_{-i}$ denotes the player other than i and $p_i(t_{-i}|t_i)$ is belief that player i of type t_i has about the types of the others [125]. BNE strategy thus maximises the expected pay-off of the player given the player's type and belief about the opponent. Similarly to the Nash equilibria (NE) in complete information games the BNE represents the point where no player has any incentive to alter its strategy. In contrast to NE the BNE depends not only on the strategies but also on players' types and their beliefs.

It can be easily shown, by proofs and reasoning from the chapter 4, that friendly vs. friendly incarnation of the game has multiple (two for assumed action set of the players) pure strategy NE, highlighted by bold font in Tab.5.2. Both equilibria use MIN WPLNC mappings and give optimal (in Pareto sense) outcomes to both players. In contrast it can be shown that malicious vs. friendly game incarnation has no pure strategy equilibria. Notice that for both versions of game some mixed strategies also exist, although I do not analyse them.

Since friendly version of the game has NE that provides optimistic results from global network point of view – the information from the sources is successfully delivered to the destination while the relays' utilities are maximised – I focus on those particular equilibria and test if they are also BNE of the incomplete information game where the maliciously behaving relay is possible.

Particularly I search for some conditions under which the strategies containing some MIN WPLNC mappings are BNE.

Proposition 5.1. *A pair of strategies $s_R = (MIN_2, MIN_1)$, $s_C = MIN_1$ is the BNE of the presented game under some constrains on the probability p of R_R being friendly and valuation of the utility functions $\pi_i(\cdot)$.*

Proof. To prove the proposition I have to show that aforementioned s_R and s_C are mutual best responses and find out the conditions for them to be the BNE.

According to Eq.(5.10) the best response $s^{\mathrm{BR}}_C(t_C)$ of R_C of type t_C on $s_R = (\mathrm{MIN}_2, \mathrm{MIN}_1)$ is

$$s^{\mathrm{BR}}_C(t_C) = \arg\max_{a_C} \sum_{t_R} p_C(t_R|t_C)\pi_C(s_R(t_R), a_C; t_R, t_C) \tag{5.11}$$

$$= \arg\max_{a_C} \{p\pi_C(\mathrm{MIN}_2, a_C; \mathrm{F}, \mathrm{F})+ \tag{5.12}$$

$$+(1-p)\pi_C(\mathrm{MIN}_1, a_C; \mathrm{M}, \mathrm{F})\} \tag{5.13}$$

This gives the following expected pay-off for various actions a_C of R_C:

$$a_C = \mathrm{MIN}_1: \quad pA + P - pP \tag{5.14}$$
$$a_C = \mathrm{MIN}_2: \quad pP + A - pA \tag{5.15}$$
$$a_C = \mathrm{EXT}_1: \quad pP + B - pB \tag{5.16}$$
$$a_C = \mathrm{EXT}_2: \quad pB + P - pP \tag{5.17}$$
$$a_C = \mathrm{FULL}: \quad C \tag{5.18}$$

For MIN_1 to be the best response to $s_R(t_R)$ the expected pay-off in Eq.(5.14) must be higher than expected pay-off from any other action a_C in Eq.(5.15 - 5.18). It can be easily shown that this is true for

$$p > \frac{1}{2} \quad \text{and} \quad p > \frac{C-P}{A-P}. \tag{5.19}$$

Conversely, $s_R^{\mathrm{BR}}(t_R)$ of R_R of type t_R on $s_C = \mathrm{MIN}_1$ is

$$s_R^{\mathrm{BR}}(t_R) = \arg\max_{a_R} \sum_{t_C} p_R(t_C|t_R)\pi_R(a_R, s_C(t_C); t_R, t_C), \tag{5.20}$$

which is for friendly type of R_R

$$s_R^{\mathrm{BR}}(F) = \arg\max_{a_R} \pi_R(a_R, \mathrm{MIN}_1; \mathrm{F}, \mathrm{F}) = \mathrm{MIN}_2, \tag{5.21}$$

since the preference relation $A > B > C > 0 \geq P$.

Similarly, for the malicious type of R_R it is

$$s_R^{\mathrm{BR}}(M) = \arg\max_{a_R} \pi_R(a_R, \mathrm{MIN}_1; \mathrm{M}, \mathrm{F}) = \mathrm{MIN}_1 \text{ or } \mathrm{EXT}_2, \tag{5.22}$$

since the preference relation $W > 0 \geq P$.

Thus I have shown that $s_R = (\mathrm{MIN}_2, \mathrm{MIN}_1)$, $s_C = \mathrm{MIN}_1$ are mutual best responses and thus form BNE of the presented incomplete information game. □

By similar reasoning and the game symmetry the following strategy pair can be also shown to be BNE under the same conditions

$$s_R = (\mathrm{MIN}_1, \mathrm{MIN}_2) \qquad s_C = \mathrm{MIN}_2 \tag{5.23}$$

Under slightly varied conditions (details omitted) also

$$s_R = (\mathrm{MIN}_1, \mathrm{EXT}_1) \qquad s_C = \mathrm{MIN}_2 \tag{5.24}$$
$$s_R = (\mathrm{MIN}_2, \mathrm{EXT}_2) \qquad s_C = \mathrm{MIN}_1, \tag{5.25}$$

are the pure BNE. It is also possible that some other pure BNE, including both MIN and EXT WPLNC mappings, exist for completely different utility valuations. Also from the well-known game theoretic results a mixed strategy BNE is guaranteed to exist for any incomplete information game. Those results are not analysed here.

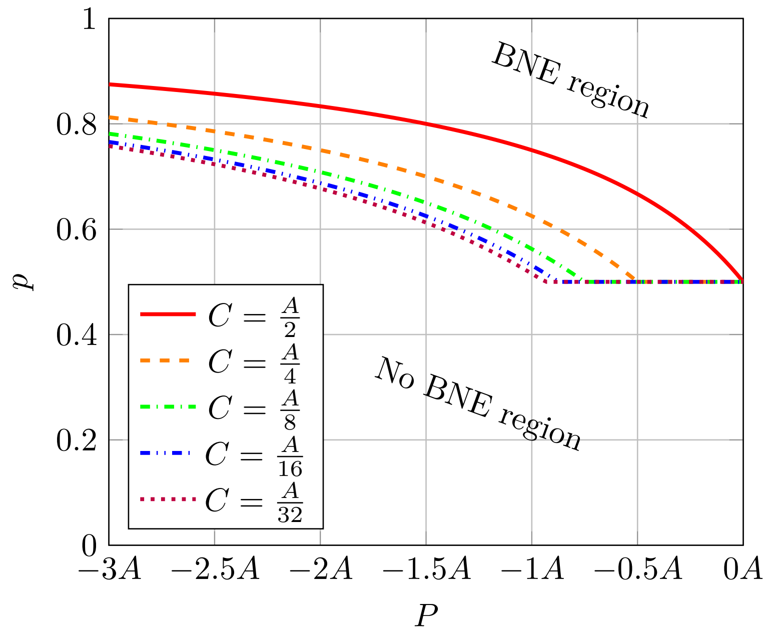

Figure 5.24: Region of Bayesian Nash equilibrium existence.

Fig.5.24 shows the region, defined by parameters in Eq.(5.19), where the strategy pair from Proposition 1 is the BNE. The utility function valuations are expressed relatively to the value of A. The figure shows that given the probability p the lower the difference between the MIN and FULL mapping pay-off A and C the lower penalty P must be paid by the nodes to have that particular BNE strategy. When the nodes are over-penalised and/or malicious relays are highly probable then there exits no studied strategy pair that is BNE.

5.2.4 Summary

The single turn incomplete information game, where the type of one relay could be friendly or malicious and is unknown to the other relays, is analysed. I provide proof and conditions under which some Bayesian Nash equilibria exist. These conditions are useful to design the utility function given the probability of presence of the malicious version of the relays. When the game does not possess this equilibria it can hardly be played rationally. On the other hand if the equilibrium exists ans is achieved then it maximises utility functions given beliefs about the other players. It does not mean that malicious node is eliminated by the friendly relays from the network, it still causes harm. But this harm is somewhat optimised given the probability p of being malicious.

To be able to eliminate the impact of malicious relay the game should be played repeatedly among the same players and some additional mechanism must be employed. The situation can be even more complicated with intelligent malicious relay that is able to change its type deliberately to hide its maliciousness. This, however, lay far beyond the scope of the thesis, but it is a perfect goal for future research.

Due to the time constraints there was not a chance to implement the studied game into the existing test-bed. That is also why a practically implementable protocol (presumably an extension of DLA) that forces the relay nodes to select reasonable WPLNC mappings in the presence of malicious relays is missing.

5.3 Battery state - sequential incomplete information game

This section focuses on a different network where the rules of underlying game are not fully and globally known. Similarly to the previous example there are also different variants of on relay. Particularly, I allow one of the relays to change its behaviour according to its battery level. When the battery is fully charged the relay behaves as usual, however, as available energy gets lower the relay starts to behave selfishly. Under low battery state it is not willing to us less energy efficient full WPLNC mappings but it rather selfishly grabs its last chance to deliver data before the battery is empty and thus prefers minimal mappings. This behaviour may be unexpected by the other relays since they do not directly know battery level of opponents.

However, now the game is played in a sequence of turns. The relays takes their actions in sequence such that the later playing relays are able to deduce a behaviour of the previous ones through observed actions. The relay can altruistically help to the opponent suffering from low battery level by reducing its own energy efficiency. Particularly, in my example a relay with varying battery plays first. Practically, this sequential behaviour can arise from relays' position in the cloud. The helping relay is more downstream in the cloud towards the destinations, although it is still able to receive the original sources. In a context of IoT, these "closer-to-destinations" relays would be probably part of backbone network and thus more fixed and better powered compared to outer relays and sensors, which at least a bit justifies the studied scenario.

This high/low battery example is just an illustration of much wider area of sequential incomplete information games and can be extended to a variety of similar problems. The work presented here is based on my publication [107]. Unfortunately there was also no chance for hardware verification of this work.

5.3.1 Network model

A network of interest as usual consists of two independent sources S_1, S_2, two relays R_R, R_C and destinations D_k, see Fig.5.25. The networks with more nodes are obviously possible but their analysis is impractical, however, the results shown here can be straightforwardly extended.

The important aspect is that I now assume that the relay R_C can suffer from low battery situation with probability p. When it is depleting its resources its main intent is to deliver as much as possible information to the destination

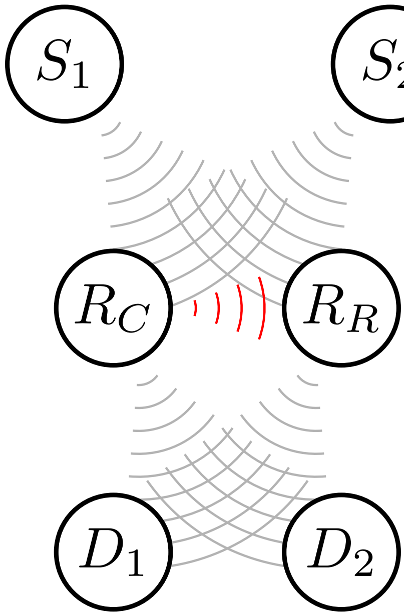

Figure 5.25: Network topology - battery level game.

in the most energy saving way. The other relay R_R has available observation of R_C's actions (shown in red in Fig.5.25) and can deduce its battery state and consequently support it. Since the R_C knows about its battery state I will also call it an informed relay (player). The uninformed one – R_R – has only a chance to learn about R_C state from the observed signal.

The network is still based on WPLNC technique such that the transmissions of the source nodes are not orthogonally separated i.e. relay R_i, $i \in \{R, C\}$ receives superposition of both sources. Moreover the uninformed relay R_R also receives a signal from the informed one – R_C – separately.

Now, I seek for an algorithm similar to the DLA (see chapter 4) that assigns invertible WPLNC mapping to both (generally all) relays even under the situation of battery level dependent behaviour of relay R_C and limited knowledge of R_R.

5.3.2 Sequential incomplete information game

This game is a sequential incomplete information game. When compared to the previous example of incomplete information game of malicious relay the significant difference is in a sequence of turns. By Harsanyi transform [124] – i.e. by introducing a fictitious Nature player that first assigns types to the others – the game is transformed to sequential imperfect information game. After observing its own type the informed relay takes its turn by choosing a WPLNC mapping. This mapping choice is observed by the uninformed relay that can use it to deduce the type of the informed one and to select its own mapping. Since one particular mapping can be chosen by both types of the informed relay the uninformed one forms so called information sets – a state of the game that can be achieved by possibly multiple ways but its details are undistinguishable by the uninformed relay, only belief can be formed. For example a minimal mapping can be used by both types of the informed player, than after observing this minimal mapping the uninformed relays can have some belief that it was used by the low battery type and another belief that it was used by the high battery type.

The game of interest is particularly defined as follows:

- A set of players is R_C and R_R. There is also the Nature that plays first and assigns the types to R_C.

- Relay node R_C has two possible incarnations – types – it has either high or low battery level. Type of the relay R_C is $t_C \in \{H, L\}$, where H denotes high battery level and L low level. Relay R_R has only one type, it always has high battery. The type t_C is a private information of R_C and is unknown to R_R. A battery level is selected by Nature prior to the beginning of the relay game. Low battery level is selected with probability p, this a priori probability is available to R_R too, but the actual battery level of R_C is not.

- Each relay has a set of actions, both sets contain various WPLNC mappings. To make the situation a bit easier I reduce the action set of R_R to two different minimal mappings MIN1 and MIN2 and a full mapping FULL. The action set of R_C depends on its type, high battery level type has the same action set as R_R since they are in fact the same. But low battery type uses minimal mappings only, i.e. its action set contains MIN1 and MIN2, this is because the minimal WPLNC mapping delivers data in the most energy saving way. For the sake of simplicity I do not assume extended mappings in this analysis, nevertheless they can be easily added.

- The relays are rewarded for the WPLNC mappings used. The utility function is related to energy efficiency not to the actual channel state, although some channel related assumptions are also made. The utility is as follows: at high battery level each rely is more or less selfish, it tries to use the best (from its own point of view) possible mapping. Such a relay is awarded by $2A$ for using MIN1 mapping, by A for MIN2 and by B for full mapping, where $A > B > 0$. We assume without loss of generality that there are some minimal mappings that perform better (in terms of SER, etc. see chapter 4) in given channel conditions, that is why the rewards for various minimal mappings differ. We also assume that a minimal mapping is always better than a full mapping (from energy savings point of view). At low battery mode the relay R_C tries to deliver as much as possible information before its battery is depleted (ignoring any other performance measure, only energy efficiency matters). It is rewarded by $4A$ for any minimal mappings it uses, because of important energy savings. The relay R_R, when facing low battery node, tries to help it and is rewarded by $4A$ for usage of full mapping since it helps most to R_C and by A for any invertible minimal mapping. Notice that

	MIN1	MIN2	FULL
MIN1	P,P	2A,A	2A,B
MIN2	A,2A	P,P	A,B
FULL	B,2A	B,A	B,B

Table 5.4: Payoff matrix for high battery vs. high battery game.

	MIN1	MIN2
MIN1	2P,2P	2A,4A
MIN2	A,4A	2P,2P
FULL	4A,4A	4A,4A

Table 5.5: Payoff matrix for high battery vs. low battery game.

usage of full mapping by R_R is enough to deliver the source data to the destinations regardless the mapping used by R_C. It is a natural question why should R_C transmit at all under this condition. Nevertheless the low battery state must be still somehow signalised to the rest of the network. More importantly, additional transmission from R_C can be favourable also from the point of view of reliability and diversity. These two aspects are, however, beyond the scope of the thesis. This example is thus a bit "academic" mainly because of its simplicity although it illustrates important concepts that can be topics for future research.

Whenever both relays select a pair of mappings that is non-invertible then they waste their resources in vain and are penalised by $P \leq 0$ since they do not fulfil their task to enable S to D communication. Because wasting of resources is critical especially in low battery mode the relays are penalised by $2P$ when R_C is in low battery state. The payoff matrices of both game incarnations are given in Tabs. 5.4 and 5.5.

- The uniformed player (R_R) can form beliefs about the informed one based on observed actions and a priori type probability, e.g. after observing MIN1 mapping the R_R can create a belief $\mu(H|MIN1)$ of R_C being type H. The belief system is important when evaluating expected utilities and when searching for the game equilibria. Because R_C does not observe any action of R_R and R_R has only one type there is no reason for R_C to form its belief about R_R.

The game sequence and rules are depicted in Fig.5.26. The dashed regions show information sets, e.g. the relay R_R is unsure about the type of R_C when it observes MIN1 mapping, it has only belief about the type, so both paths leading to MIN1 dashed region belong to the same information set. It can be also seen that FULL mapping is a singleton information set since it automatically means that $t_C = H$, belief for this singleton information set

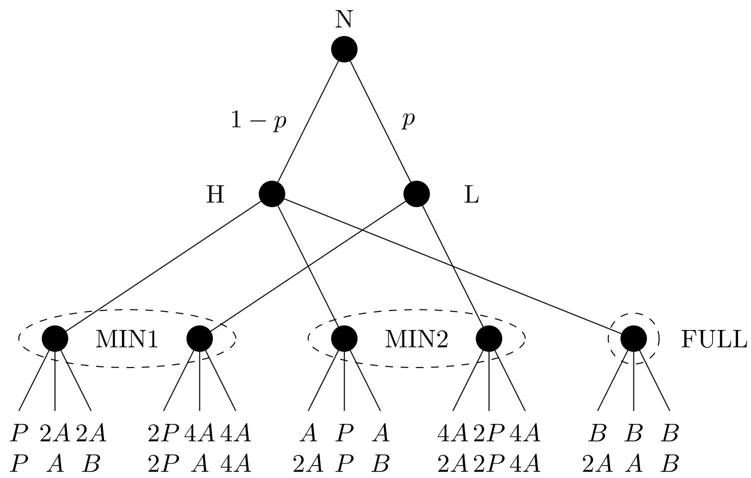

Figure 5.26: Order of turns and rules of the battery game.

is thus simple. Edges coming out of the information sets are R_R's actions and are from left to right MIN1, MIN2 and FULL. Rewards are shown at the leaves of the game tree, R_C's upper and R_R's lower. Compare this game tree with the one shown in Fig.5.23, where the game is simultaneous turn complete information one after types are assigned.

5.3.3 Equilibrium analysis

A widely accepted solution for sequential incomplete information games is a Perfect Bayesian Equilibrium (PBE) [125]. A PBE is a strategy profile $s^* = (a_C^*, a_R^*)$ and a belief system μ^* with two additional properties. The strategies are sequentially rational given the belief system and the belief system is consistent given the strategy profile. Sequential rationality means that the action taken in given information set maximises expected pay-off given the belief and actions of the other players. Consistent belief systems means that the beliefs at least in the information sets on-the-equilibrium-path are given by Bayes' rule. The information set is a point in the game tree connected with particular observation the player has. In the game tree shown in Fig.5.26 there are three information sets, for example after observing MIN1 action from R_C by R_R there are two possible ways (histories) how the game should come there – R_C being either H or L. Thus observation of MIN1 forms one information set for R_R. Within the information set R_R is not sure what is the exact history of the game, which way was actually taken to reach this information set. It can only have some probabilistic belief about the histories. Some of the information sets are said to be off-the-equilibrium-path, these can be never reached given the particular strategy. Thus the beliefs for off-the-equilibrium-paths cannot follow the Bayes' rule and can be set arbitrarily, also they are removed from belief consistency condition. The concept of PBE is applied to avoid some implausible equilibria that can arise when sequential rationality and belief consistency are omitted.

Again, since this is an incomplete information game there is a different meaning of a strategy and an action. The strategy assigns action to each particular type of the player. Thus since the relay R_C has two possible types its strategy is a doublet assigning an action to both types, e.g. $a_C = (a_C(H), a_C(L)) = (\text{MIN1}, \text{MIN2})$ means that R_C of type $t_C = H$ uses MIN1 while type $t_C = L$ plays MIN2. Since R_R has only one type its strategy is a singleton action a_R such as $a_R = \text{MIN1}$. When the actions played by different types are the same the strategy is called a pooling strategy, otherwise, when different types do different actions, it is called a separating strategy.

In the proposed game there four separating:

$$a_C = (a_C(H), a_C(L)) = (\text{MIN1}, \text{MIN2}) \tag{5.26}$$
$$a_C = (a_C(H), a_C(L)) = (\text{MIN2}, \text{MIN1}) \tag{5.27}$$
$$a_C = (a_C(H), a_C(L)) = (\text{MIN1}, \text{FULL}) \tag{5.28}$$
$$a_C = (a_C(H), a_C(L)) = (\text{MIN2}, \text{FULL}), \tag{5.29}$$

and two pooling strategies of R_C:

$$a_C = (a_C(H), a_C(L)) = (\text{MIN1}, \text{MIN1}) \tag{5.30}$$
$$a_C = (a_C(H), a_C(L)) = (\text{MIN2}, \text{MIN2}) \tag{5.31}$$

Proposition 5.2. *The only PBEs of the proposed game are connected with the pooling strategy $a_C^* = (a_C^*(H), a_C^*(L)) = (MIN1, MIN1)$. Otherwise there is no PBE.*

Proof. To prove the Proposition 5.2 I have to show that there is a consistent belief system and sequentially rational strategies for both players. First of all define belief system: on the equilibrium path R_R's belief about R_C having high battery level after observing MIN1 is $\mu(H|\text{MIN1}) = 1 - p$ which follows from Bayes' rule. Off the equilibrium path beliefs can be arbitrary, such as $\mu(H|\text{MIN2}) \in [0, 1]$. Since the off-the-equilibrium-path information set of FULL mapping is singleton the belief $\mu(H|\text{FULL})$ is simply 1 – FULL cannot be played by L type R_C. Thus the belief system is consistent (it is Bayesian at least on the equilibrium path and arbitrary elsewhere) given a_C^*.

Having the consistent belief system I show the sequential rationality of the strategy. I first seek the best responses of R_R on observed signals from R_C having R_R's beliefs set above. The expected pay-off of R_R obtained in response to $a_C = \text{MIN1}$ for all three actions of R_R is:

$$\text{MIN1:} \quad (1-p)P + p2P = P - pP \tag{5.32}$$
$$\text{MIN2:} \quad (1-p)A + pA = A \tag{5.33}$$
$$\text{FULL:} \quad (1-p)B + p4A = B - pB + p4A \tag{5.34}$$

Thus it can be shown that R_R's best response on action $a_C = \text{MIN1}$ is $a_R = \text{MIN2}$ if $p < \frac{A-B}{4A-B}$ or $a_R = \text{FULL}$ elsewhere, of course $p \in [0, 1]$.

The R_R's best response on action $a_C = \text{MIN2}$ is undefined since the belief $\mu(H|\text{MIN2})$ is not defined because this information set is off the equilibrium path. Later I will refer to this unspecified best response as $a_R(\text{MIN2})$.

The R_R's best response on action a_C = FULL is a_R = MIN1 simply because of utility function ordering $2A > A > B$, see Fig.5.26.

Knowing the best responses of R_R, the strategy of R_C and belief system the assumed strategy is PBE if and only if R_C has no incentive to change its strategy, i.e. its expected pay-off should be the highest among all possible alternative strategies. First, start with the situation when the best response on a_C = MIN1 is a_R = MIN2. Then the expected pay-off of R_C must be:

$$\pi_C(H, \text{MIN1}, \text{MIN2}) \geq \pi_C(H, \text{FULL}, \text{MIN1}) \tag{5.35}$$
$$\pi_C(H, \text{MIN1}, \text{MIN2}) \geq \pi_C(H, \text{MIN2}, a_R(\text{MIN2})) \tag{5.36}$$
$$\pi_C(L, \text{MIN1}, \text{MIN2}) \geq \pi_C(L, \text{MIN2}, a_R(\text{MIN2})), \tag{5.37}$$

where $\pi_C(t_C, a_C, a_R)$ is the pay-off of R_C of type t_C playing action a_C while R_R plays a_R. All these equations are true since (5.35) is $2A \geq B$, (5.36) is $2A \geq A$ since A is the maximal possible pay-off of R_C given that strategy and (5.37) is $4A \geq 4A$ since $4A$ is the maximal possible pay-off of R_C given that strategy.

Secondly, assume that the best response on a_C = MIN1 is a_R = FULL. Then the expected pay-off of R_C must be:

$$\pi_C(H, \text{MIN1}, \text{FULL}) \geq \pi_C(H, \text{FULL}, \text{MIN1}) \tag{5.38}$$
$$\pi_C(H, \text{MIN1}, \text{FULL}) \geq \pi_C(H, \text{MIN2}, a_R(\text{MIN2})) \tag{5.39}$$
$$\pi_C(L, \text{MIN1}, \text{FULL}) \geq \pi_C(L, \text{MIN2}, a_R(\text{MIN2})). \tag{5.40}$$

All equations are true since (5.38) is $2A \geq C$, (5.39) is $2A \geq A$ since A is the maximal possible pay-off of R_C given that strategy and (5.40) is $4A \geq 4A$ since $4A$ is the maximal possible pay-off of R_C given that strategy.

This holds true for arbitrary belief $\mu(H|\text{MIN2}) \in [0, 1]$. I thus have shown that there are two PBEs based on mutual relation between A, B and p – namely

$$s_1^* = (a_C^*(H), a_C^*(L), a_R^*) = (\text{MIN1}, \text{MIN1}, \text{MIN2}), \tag{5.41}$$
$$s_2^* = (a_C^*(H), a_C^*(L), a_R^*) = (\text{MIN1}, \text{MIN1}, \text{FULL}), \tag{5.42}$$

with belief system $\mu^*(H|\text{MIN1}) = 1-p$, $\mu^*(H|\text{MIN2}) \in [0, 1]$ and $\mu^*(H|\text{FULL}) = 1$. s_1^* is PBE if $p < \frac{A-B}{4A-B}$ otherwise it is s_2^*.

By similar reasoning it can be easily shown that aforementioned pooling strategy of R_C is the only one PBE of the proposed game. □

Fig.5.27 shows a surface that divides regions of existence of both PBE. Below this surface s_1^* is the PBE of the game, above it is s_2^*. The surface is given by probability of low battery state $p = \frac{A-B}{4A-B}$ as a function of rewards A, B. It can be concluded that the higher the probability of depleted battery and the higher the reward B the relay R_R will prefer action FULL since it is rewarded by higher pay-off by supporting R_C.

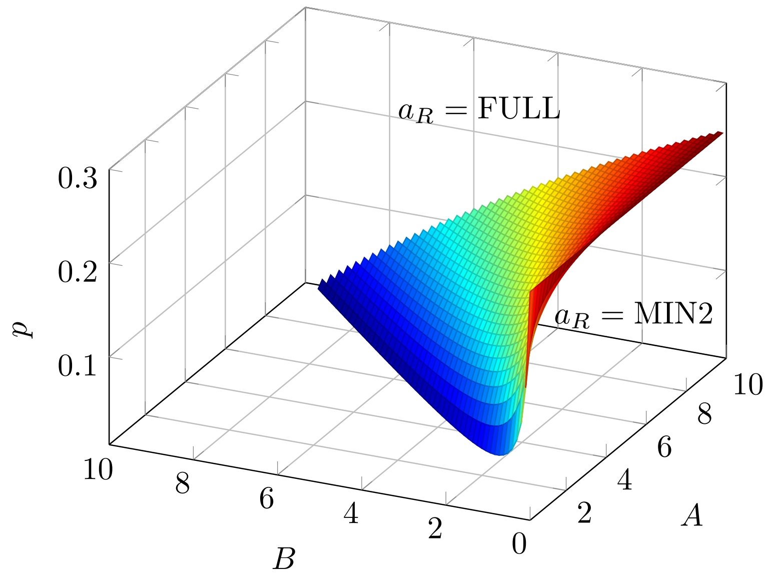

Figure 5.27: Regions of existence of game equilibria - incomplete information game.

5.3.4 Summary

The sequential incomplete information game where the first playing relay may have two types dependent on its battery level is analysed. The battery level affects the actions taken by that relay which consequently determines the performance of the whole network. The other relay has a chance to observe the battery state only through observing the actions taken. This relay can potentially modify its operation to improve the performance and to support the node with low battery.

I provide proof and conditions under which some perfect Bayesian equilibria exist. Similarly to the example with the malicious relay this analysis should serve as a tool for utility function design such that some feasible equilibrium state of the network is achieved.

Due to the time constraints there was not a chance to implement the studied game into the existing test-bed. That is also why a practically implementable protocol (presumably as an extension of DLA) is missing.

Chapter 6

Non-orthogonal signalisation in cloud networks - Cloud Initialisation Procedure

It was shown in the previous chapters and for example in [100] that any form of Wireless Physical Layer Network Coding (WPLNC), and also even any form of Network Coding, needs some accompanying signalling. Minimally, an applied WPLNC mapping or NC conding function, number and identification of sources, signals for channel estimation, signals for timing synchronisation and carrier frequency offset compensation must be reported to the place where the signal is being processed and/or source data are being recovered.

The previous chapters showed an example of such an approach based on a signalling data modulated on orthogonal CAZAC sequences. A question has naturally arisen, why a system that wants to overcome orthogonal separation of transmissions should insist on orthogonal signalling? In this chapter I would like to present an initial attempt utilising non-orthogonal sequences. Especially, I focus on non-orthogonal signals used for number of source recovery as well as for channel estimation. Advantages and disadvantages of the proposed solution are shown and discussed. Evaluation contains both numerical simulation and hardware verification.

A work presented in this chapter is based on my published as well as unpublished research. From published one I would like to especially mention [126, 127].

6.1 Problem statement

WPLNC directly processes symbols from superposed constellation. A particular shape of the superposed constellation depends mainly on the number of sources, the type of the source constellations and on the channel. Especially, channel attenuation and phase rotation significantly determine its resulting shape. Because of this dependency, there must exist a way to recover the information from the constellation shape. For example, the number of nodes can be deduced from the number of constellation points, or rather heaps of points due to the noise, in the superposed constellation – two superposed BPSKs cannot produce more then four and, if the signals are strong enough, not less than three constellation points regardless the channel phase, sup-

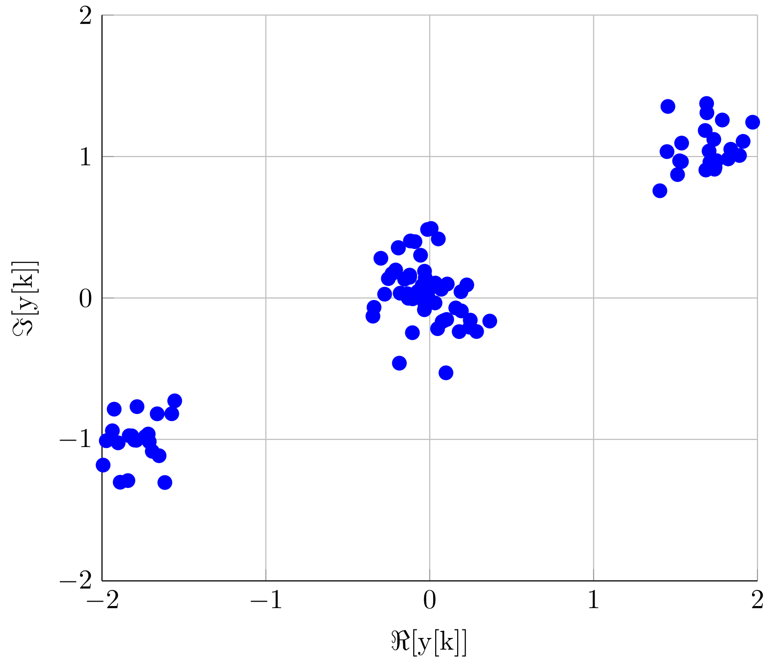

Figure 6.1: Superposition of two BPSK signals leading to 3 groups of points in the superposed constellation.

posing the channel is time constant at least during the processing time. For illustration see figures 6.1 and 6.2. Similar conclusion can be drawn for higher number of superposed, if not by exact calculation, then at least in form of a rule of thumb.

Information recovery from constellation shapes is closely related to automatic recognition of the used modulation scheme [128]. This technique is greatly elaborated in point-to-point scenarios, where modulation formats can be recovered under timing/frequency offset/channel imperfections in both AWGN and fading channels. An comprehensive study of classification methods can be found in [129]. This technique is frequently used with intelligent and cognitive radios. Although it is mostly used in single source scenarios, there are also some multi-source cases [130, 131] where it is usually used in combination with multiple antenna arrays and with help of by other estimators such as angle-of-arrival.

Given the concept of the wireless cloud network I look for an algorithm that:

- Estimates the number of the sources contained in the received superposition.
- Estimates the channels among the sources and receiving node.
- Does not depend on orthogonal sequences and their dedication to the sources.

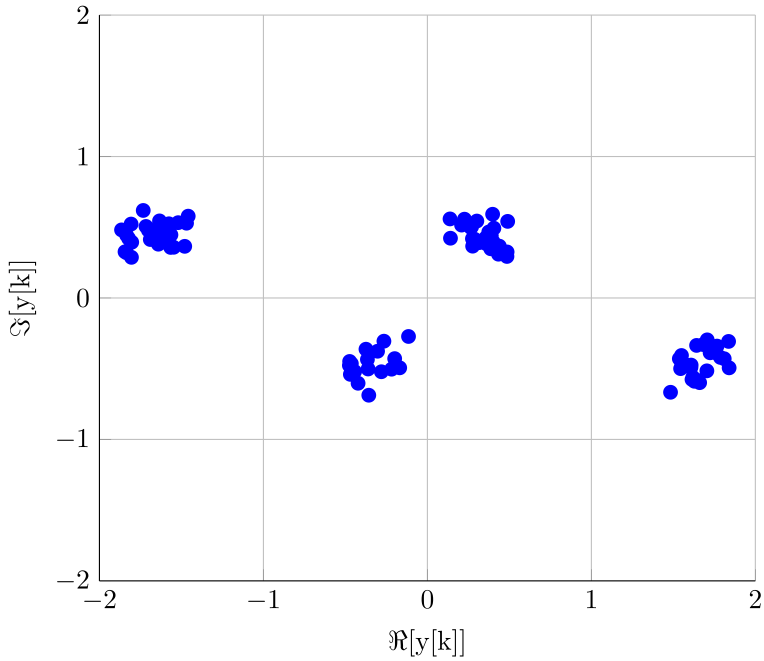

Figure 6.2: Superposition of two BPSK signals leading to 4 groups points in the superposed constellation.

- Is blind – does not exactly know the signal structure used by the sources, although a modulation format used, such as N-PSK, is known.
- Works easily with simple single antenna devices.
- Does not need immense computational power and/or help by any other estimator.
- Nevertheless provides solid performance in the scenario of interest.

These assumptions have emerged mainly from practical reasons resulting from simplicity of the sensor nodes and also from the conditions I laid on the distributed wireless cloud network and WPLNC processing, see chapter 2. The estimator should work directly with the superposed signals, without the necessity of source separation. Blindness is enforced by distributiveness of the network and selfishness of the users – each source would not like to produce its signal in cooperation with the others, it simply does not care about the others. So any form of cooperative creation of the sequences and/or any coordination among the sources must be avoided. Demands put on the estimator are also limited by capabilities of potentially simple devices used in the network, such as battery powered small simple devices that are typical in sensor networks, etc.

Since the task of blind identification of the number of sources and their channel estimation is quite challenging, the true and valid estimates will be provided only in limited set-ups. The estimator gains the information from the superposed constellation thus this constellation must be relatively simple for correct estimates. The number of sources that are allowed to participate in the superposed transitions is limited to just a few of nodes (up to 5 or so). Also to make the superposed constellation simple the individual source constellations should be simple, like an on-off keying (OOK) or a binary phase shift keying (BPSK). Also, a channel that is static during the reception of the signals is preferred. Although all these assumptions seems to be critical and fairly limiting, they can be fulfilled in practical systems, which will also be demonstrated later. Simple, however quite ineffective, constellations, such as OOK, are used only part time and for signalling purposes. Data payload can then use arbitrary modulation scheme. The limit on the number of the sources is natural. It can be hardly assumed that non-orthogonal superposition of signals from tens, hundreds or more nodes can be reliably processed by any means of signal processing.

It is important to note that the algorithm that will be presented is a very initial attempt to solve the presented issue. More elaborated ones could easily outperform it, especially in advanced and/or more severer scenarios. Despite its relative simplicity it provides a solid performance as is verified by numerical as well as hardware simulations described later. Also it is simple enough to run real-time with fairly low computational demand.

6.2 Network model

The network of interest consists of three elements

- Set of N_S sources $\mathcal{S} = \{S_1, S_2, \cdots, S_i, \cdots, S_{N_S}\}$,
- Set of N_R relays $\mathcal{R} = \{R_1, R_2, \cdots, R_j, \cdots, R_{N_R}\}$,
- Set of N_D destinations $\mathcal{D} = \{D_1, D_2, \cdots, D_k, \cdots, D_{N_D}\}$.

As usual the relays form wireless cloud network with all the properties as stated in chapter 2, sources are independent and selfish and destinations are nodes where source information shall be delivered. The number of destination nodes is not of a great interest here, all analysis can be simplified to existence of at least some destination for completeness.

For network topology see Fig.6.3, where connectivity of the nodes is shown by background colour of the nodes. Notice that some source nodes are connected to multiple relays. Also notice that only sources and relays are shown.

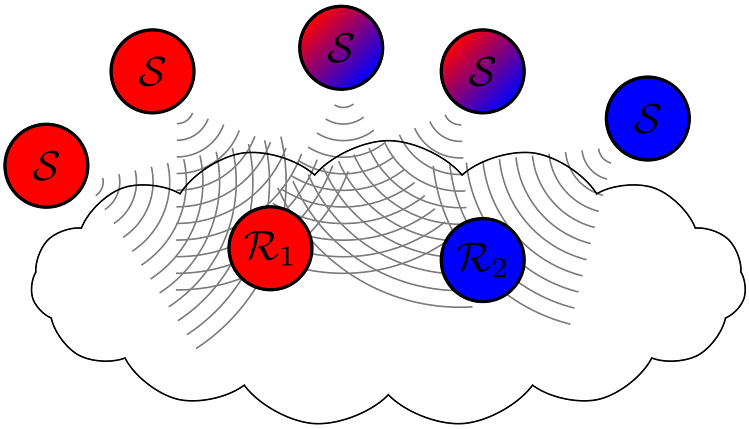

Figure 6.3: Cloud network with connectivity highlighted in colours.

6.3 Cloud initialisation procedure

In this chapter I mostly focus on the situation when the communications through the cloud network begins. During the following section I will propose an algorithm that serves to initialisation of the WPLNC communication in the cloud environment. The resulting algorithm is called a Cloud initialisation procedure (CIP).

The main obstacle is missing topology/connectivity/channel state knowledge at all the nodes at the beginning of the communication. The sources perform selfish operation irrespective to any other source, thus they are naturally blind to the other sources. The relays although forming a cloud network do not know about the presence and/or operation mode of the other relays and also know nothing about the sources/destinations. For simplicity I assume a sufficient time and frequency synchronisation is achieved among all nodes to support WLPNC mode of communication. Methods to achieve this in distributed way and impact of such methods on communication initialisation are beyond the scope of this thesis.

The source nodes enables cloud initialisation by transmission of properly selected signalisation/pilot sequences. It is important to highlight that these sequences are assumed to be non-orthogonal, generated locally without any a priori assignment or source node cooperation.

There are two fundamental issues associated with non-orthogonal pilots especially in distributed cloud networks:

- To generate sequences that are non-orthogonal, but can be distinguished each other, the obvious price is a length of such sequences. Basically, the longer the sequence the higher the probability that it will differ significantly from any other such sequence and thus can be recognised.

- Since the signalling sequences are created locally at the sources and are not globally shared among the nodes, because of distributed approach and to avoid overhead, any signalling processing may be blind to particular deterministic sequence generation method and/or sequence assignment to the source nodes.

From these two propositions it is obvious that applicable signalling sequences can be (probably not only) long enough random signals generated at the sources by (pseudo)random algorithms. These sequences serve as pilot signals and are prepend to normal data packet, this way they form a kind of a header related to CIP. There is no obvious need for any error protection of the pilot signals. Also modulation and coding scheme used for the signalisation impose nothing on modulation/coding scheme used for the rest of data packet. Any efficient and up-to-date technique can be used for data part, although some very simple method (as will be justified later) will be used for the header.

From the relay/cloud point of view the most important outcome of the cloud initialisation is to recover the following information:

- Number of sources transmitting to every single relay,
- Appropriate channels between the relay and the sources,
- Some form of identification of the sources.

These points are fundamental to set-up WLPNC processing by assigning proper WPLNC mappings.

For each individual relay I define a set of sources that are in the radio visibility with sufficiently strong channel such that this source can be received by the relay. I call this set a set of sources operated by relay R_j and denote it $\mathcal{S}_{R_j}$ and it is a subset of all the sources $\mathcal{S}$. A number of sources operated by relay R_j is denoted $|\mathcal{S}_{R_j}| = L_j$. A source node S_i can be operated by multiple relays (i.e. to belong to multiple sets of operated sources of different relays). In fact to fully utilise the advantages of WPLNC it is highly desired that the sources are connected to multiple relays. All the sources that are not members of the $\mathcal{S}_{R_j}$ have no or negligible impact on the relay R_j because their channels are weak.

The maximum number of the sources operated by one relay is limited. As was already mentioned it can be hardly imagined the one relay node can reliably process the signal from tens of the nodes. It will be drowned in the signals and interferences. In the presented analysis I limit L_j to $L_{max} = 4$. Although it may seem as quite a small number and really strict limit it still may provide immense advantage compared to classical orthogonal approach. Any relay that is capable to process 4 sources simultaneously may achieve 300% throughput improvement. There are also practical constraints such as natural WPLNC cardinality inflation limit, also the relay will face limited HW dynamics, near-far problem, etc. An example of two sets of the operated sources of two relays with non-empty intersection is depicted in Fig.6.3 where two colour background shading expresses that the sources belong to two sets of operated sources. Some of the sources in the figure belong to only single set.

The only observation available to the relay R_j at the beginning of the communication, during the initialisation phase, is the superposition of the signals from all operated sources. This superposition is given by:

$$y_{R_j}(t) = \sum_{i:S_i \in \mathcal{S}_{R_j}}^{L_j} h_{ij} s_i(t) + w_j(t), \tag{6.1}$$

where, for simplicity, I assume the single path constant complex channel h_{ij} from source S_i to relay R_j, $s_i(t)$ is signal transmitted by the source S_i and $w_j(t)$ is AWGN.

To avoid unnecessary time dependence the received signal can be easily expressed in the signal space domain. Here it forms a so called superposed constellation which is given by:

$$y_{R_j}[k] = \sum_{i:S_i \in \mathcal{S}_{R_j}}^{L_j} h_{ij} q_i[k] + w_j[k] \tag{6.2}$$

where k is used to index over the transmitted symbols and q_i are the source symbols transmitted by the source S_i. Symbols q_i belong to the source alphabet - OOK, N-PSK, ...

The particular shape of the superposed constellation depends on:

- The number of operated sources,
- Source node alphabet - such as OOK, N-PSK, N-ASK, etc.,
- Individual source to relay channels,
- Noise - expressed by signal to noise ratio.

An example of superposed constellations formed by various source alphabets and for various channel realisations is shown in Fig.6.4 to 6.7.

To recover the necessary information from the received superposed constellation, the constellation must:

- Be simple enough - which means low number of operated sources, each using simple alphabet such as OOK or BPSK. The higher order alphabets, although having better properties for data transfer, will make the cloud initialisation colossally complex. And of course they usage for data transmission after the initialisation phase is not prohibited.
- Be full or almost full - any points missing from the superposed constellation because it simply was not transmitted may lead to incorrect estimation of necessary parameters. For example, any half (two points) of the QPSK constellation can be easily interchanged for BPSK and so on. This also emphasizes the need of long enough random sequences for correct estimates.
- Have sufficient SNR - a constellation that is deep buried in the noise can be hardly used for estimation of parameters needed.

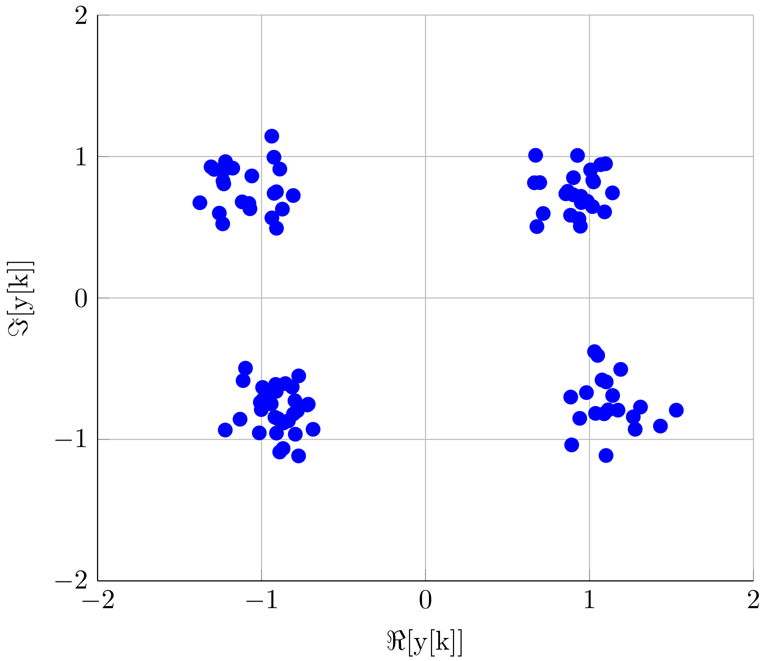

Figure 6.4: Superposition of two BPSK signals for some channel realisation.

Generally this problem, of recovering the information, falls into the area of a machine learning and/or pattern recognition [132]. There is a vast area of mathematical as well as technical solutions to infer the necessary information. An exhaustive overview of likelihood based methods is provided in [128, 129], where the problem is generally claimed as automatic modulation classification. Several other methods based on recovering specific modulation features, such as cyclostationarity, are also discussed therein. From the pattern recognition perspective the issue falls into a branch of an unsupervised learning, where various clustering methods are successfully applied. This chapter is based on adoption of a K-means algorithm [133]. Other clustering methods such as expectation-maximisation algorithm [134] can be also used. This algorithm should be especially useful because it works well in the case of mixture of normally distributed data, which is a common case in noisy wireless channels. Nevertheless the K-means was adopted here because of its overhelming simplicity. Far more advanced algorithms based for example on artificial neural networks [135] can and are obviously applied [136].

There is a wide range of automatic modulation classification applications, including classification of linear modulation schemes of N-PSK or N-ASK families [137], area of multi-antenna communications [138], adaptive OFDM schemes [139], even relaying schemes [140]. There are already existing efficient FPGA based implementations of classification algorithms [141]. Although modulation classification is applied in many areas of communication systems, to my best knowledge, there is no application of it to the multi-source WPLNC based area prior to my work. If there is any such multi-source application then it is not in the WLPNC context and usually using multiple antenna

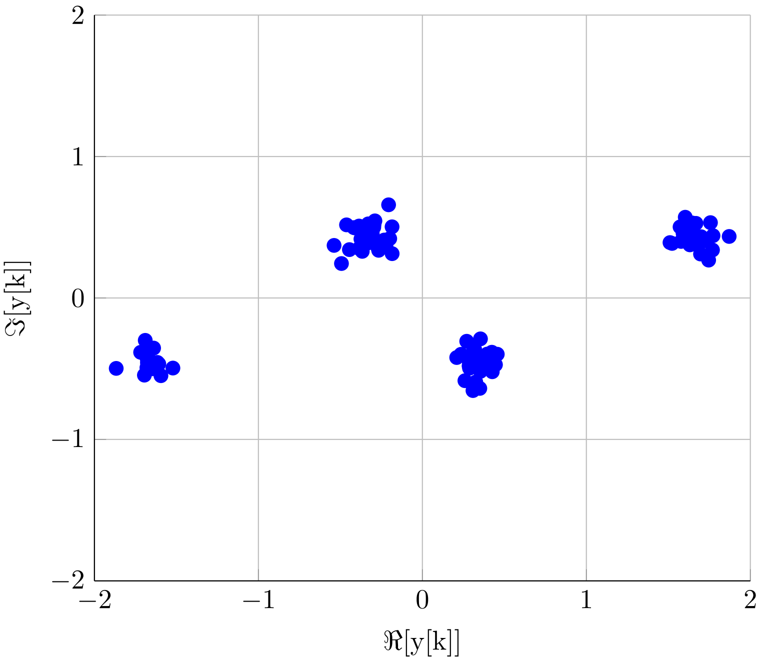

Figure 6.5: Superposition of two BPSK signals for different channel realisation.

transceivers [131]. If there is an application in the relay network, then mostly single source one with relay employing Amplify-and-Forward scheme, again usually combined with MIMO technique [140].

Since all the necessary information shall be inferred from the received signal, usually expressed in signal space domain, clustering methods can be applied. The clustering algorithm groups data to be clustered into collections of similar properties. In terms of signal space, they can be naturally grouped based on mutual distances. Superposed constellation thus can be divided into groups that can be used for parameter estimation by the receiver as I will show later.

Having the assumption of simple node construction, limited computation power, limited battery life, etc. in mind, there is a need for really simple algorithm. I have ad-hoc chosen a blind clustering method called K-means algorithm [133] which is simple enough to meet the criteria, however, still able to provide solid performance in realistic cases.

K-means algorithm creates clusters in analysed data by grouping the data based on a given metric. Starting from initial random guess centres of the futures clusters are computed as centres of mass (called centroids or barycentres). The process is iteratively repeated until a steady state is achieved and/or stop condition is met, basically, when new centres of cluster mass are not moving.

K-means clustering algorithm applied on the problem to be solved here is described by the pseudocode 4.

The algorithm returns the position of the l centroids $\{c^{(1)}, \cdots, c^{(i)} \cdots, c^{(l)}\}$ and the identification to which cluster each received point $y[k]$ belongs to.

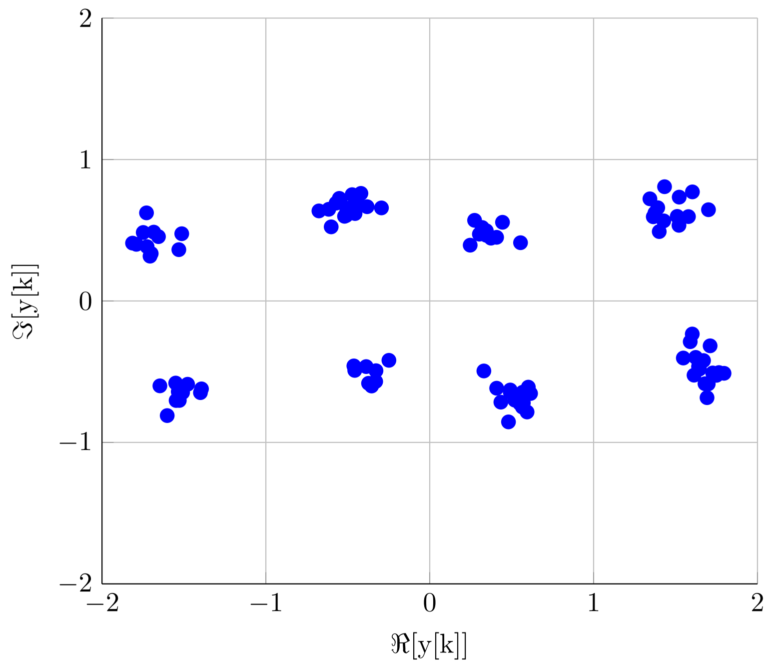

Figure 6.6: Superposition of BPSK and QPSK signal for some channel realisation.

Algorithm 4 K-means

Place l points (initial centroids) randomly into the space of all received symbols $y_{R_j}[k]$ from Eq.6.2
while stop condition is not met **do**
  Assign each received point $y_{R_j}[k]$ to the closest centroid
  Compute new centroids as barycentres
end while

In the case of the signal space domain the Euclidean distance, or rather its square, is naturally used as a metric. Each point $y[k]$ is assigned to a cluster i with centroid $c^{(i)}$ based on:

$$\min_i |y_{R_j}[k] - c^{(i)}|^2 \tag{6.3}$$

When the point clustering is done a new position of the centroids is computed as a center of mass of existing cluster. New centroid $c_+^{(i)}$ of the cluster i is:

$$c_+^{(i)} = \text{mean}(y^i_{R_j}[k]), \tag{6.4}$$

where $y^i_{R_j}[k]$ means those points $y_{R_j}[k]$ that actually belong to cluster i, i.e. have the closest centroid $c^{(i)}$.

Although the K-means algorithm guarantees the termination of the clustering process the optimal solution is not guaranteed at all. The algorithm may converge to local optimum [133].

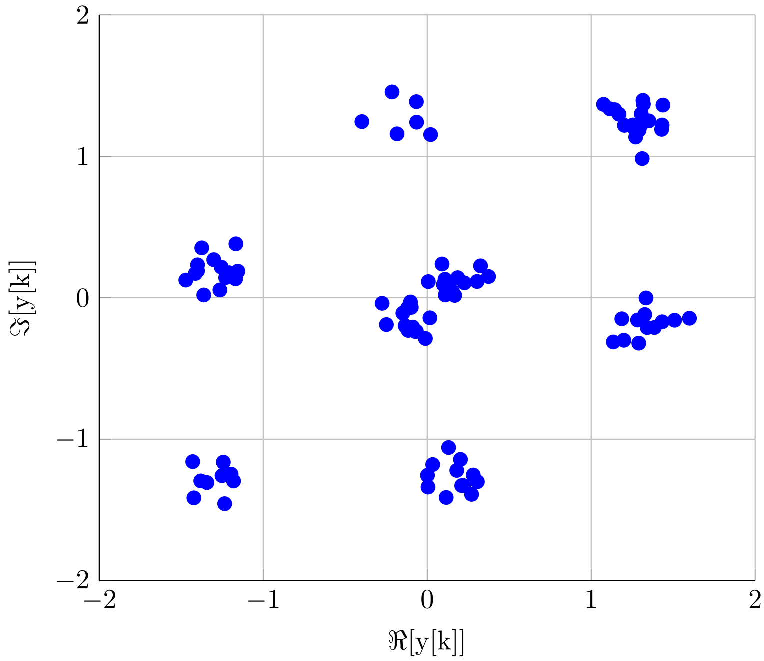

Figure 6.7: Superposition of BPSK and QPSK signal for different channel realisation.

One run (six iterations) of the K-means is illustrated on Figs.6.8 - 6.13. Notice mainly how centroid positions and division of the nodes into the individual clusters change with iterations.

There are two basic obstacles that must be resolved before the K-means is used for CIP. Firstly, K-means may converged to local optimum clustering only, there may be significantly better results, that is not achieved in current run. The resulting clustering mainly depends on starting centroids positions and on data to be clustered [133]. To overcome this the K-means can be started multiple times with different starting points placement. The best (the one minimising remaining inter-cluster distances) resulting clustering is then selected. With enough repetitions there is good enough chance that optimal or almost optimal clustering is found. Secondly, the initial number of clusters in unknown. The number of clusters is mainly given by the number of the sources contained in the superposed constellation, all particular source to relay channels and noise. It can be quite easily shown that in the case of two OOK or BPSK sources only three or four clusters can be expected. Three clusters are observed in this case when the channels h_1, h_2 are (anti-)collinear – having similar magnitude and angular difference either 0 or π radians, Fig.6.14. Four clusters are expected in all the remaining channel states. The number of clusters is dependent also on the noise level (expressed by SNR), basically the higher the noise the more spread are the resulting clusters and four case clustering at high SNR can be easily incorrectly estimated to be three clusters when noise gets higher (see Fig.6.15 and compare with Fig.6.16 – the same channel realisation with two different SNRs). Similar assumptions can be

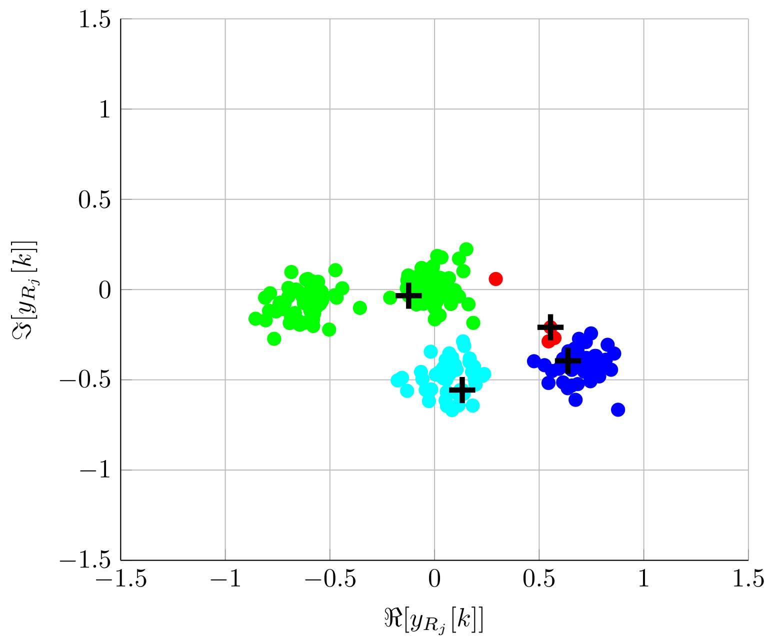

Figure 6.8: K-means iteration 1

made for higher number of the sources, when the alphabets are simple.

As a result of aforementioned obstacles and properties of K-means algorithm a Cloud Initialisation Procedure was created. It is described by the pseudocode 5.

Algorithm 5 Cloud Initialisation Procedure

```
Collect y_{R_j}[k] k ∈ {1, ··· , n_0}
Set the threshold d_th based on the SNR
Set the number of the repetitions of the k-means r_MAX
for l = 1 → 2^{L_MAX} do
    for r = 1 → r_MAX do
        perform the K-means over y_{R_j}[k] ∀k with l clusters
    end for
    remember the clustering with the minimal d_sum(l)
end for
l_best ← Find the first l with d_sum(l) ≤ d_th
if d_sum(l) > d_th ∀l then
    l_best = 2^{L_max}
end if
return \hat{L}_j = ⌈log_2(l_best)⌉
return {c}_best = {c_best^(1), ··· c_best^(l_best)}
```

There are several parameters related to the CIP.

- r_{MAX} is a number of repetitions of K-means, which is needed due to local convergence of it.

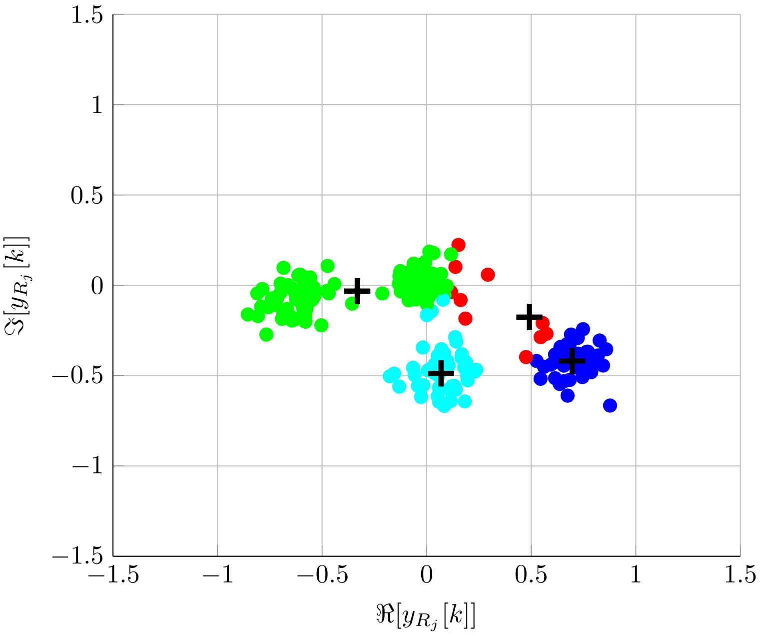

Figure 6.9: K-means iteration 2

- $2^{L_{max}}$ is the maximum number of clusters. In this case it is given by alphabet cardinality (OOK or BPSK assumed here) powered to maximal number of the sources operated by the relay. Obviously, $2^{L_{max}}$ cannot be exceeded under the presented assumptions. The true number of the clusters may be lower for particular channel/noise conditions as illustrated on Fig.6.1.
- $d_{sum}(l)$ is sum of resulting inter-cluster distances, that are given by:

$$d_{sum}(l) = \sum_{i=1}^{l} \sum_{y^i_{R_j}[k]} |y_{R_j}[k] - c^{(i)}|^2. \tag{6.5}$$

- d_{th} is a threshold value which is used to recognise good enough clustering. Obviously the best (with the minimal inter-cluster distances) clustering is one that has as much centroids as data points to be clustered and each point coincides with one centroid. From some number of the clusters adding a new centroid gives no additional information, just already correctly estimated clusters are subdivided. The value of d_{th} is a function of SNR and can be set by means of look-up table.

The outcome of the CIP is the best clustering l_{best} which is the first clustering with resulting $d_{sum}(l)$ below the threshold d_{th}. This optimal clustering is then converted into an estimate of number of source operated $\hat{L}_j$ by ceiling operation $\lceil \log_2(l_{best}) \rceil$. The ceiling function is used since the number of clusters in the superposed constellation can be lower than 2^{L_j}, for binary source alphabets, because of overlaps, but never goes too much low.

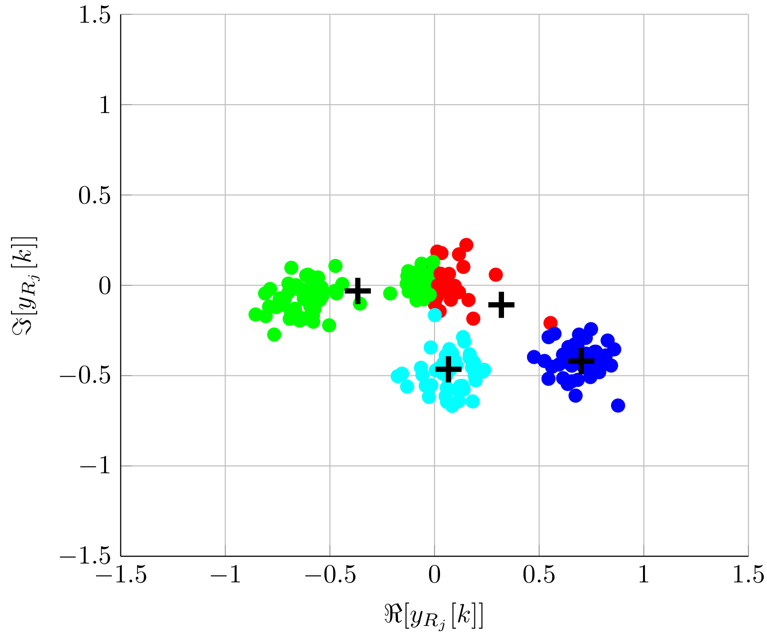

Figure 6.10: K-means iteration 3

The positions of the centroids $\{c\}_{best} = \left\{c_{best}^{(1)}, \cdots c_{best}^{(l_{best})}\right\}$ are also recorded and can be later used for channel estimation. Notice that positions of centroids represent estimates of the ideal, noiseless, superposed constellation.

From the estimation of the best position of the centroids $\{c\}_{best}$ the receiver is able to estimate the channels. Received superposed constellation is formed by all possible linear combinations of all channels, where combination coefficients are symbols from the source alphabets. In two source case the superposed constellation is:

$$y_{R_j}[k] = h_{1j}q_1[k] + h_{2j}q_2[k], \tag{6.6}$$

where h_{ij} are appropriate channels, q_i are source alphabet symbols and the complete superposed constellation is achieved when all pairs of q_is are used.

Estimation of the channel states can be described by the simple matrix equation

$$\mathbb{A}\mathbf{h}_j = \mathbf{b}, \tag{6.7}$$

where $\mathbb{A}$ is a $(2^{\hat{L}_j} - 1) \times \hat{L}_j$ matrix of all possible $\hat{L}_j$-tuples formed by all possible combinations of source alphabet symbols, $\mathbf{h}_j = [h_{1j}, \cdots, h_{\hat{L}_j j}]^T$ is a vector of unknown channel states from $\hat{L}_j$ sources and the right-hand side vector $\mathbf{b}$ is an unknown ordering (possibly with repetitions of some elements) of the centroid positions $\{c\}_{best}$.

To estimate the channels it is necessary to find an appropriate ordering of the right-hand side vector $\mathbf{b}$. I illustrate this on a simple $L_j = 2$ example with OOK used as a common source alphabet, see Fig.6.17. By application of Algorithm 5 I obtain the correct estimate of the number of the sources

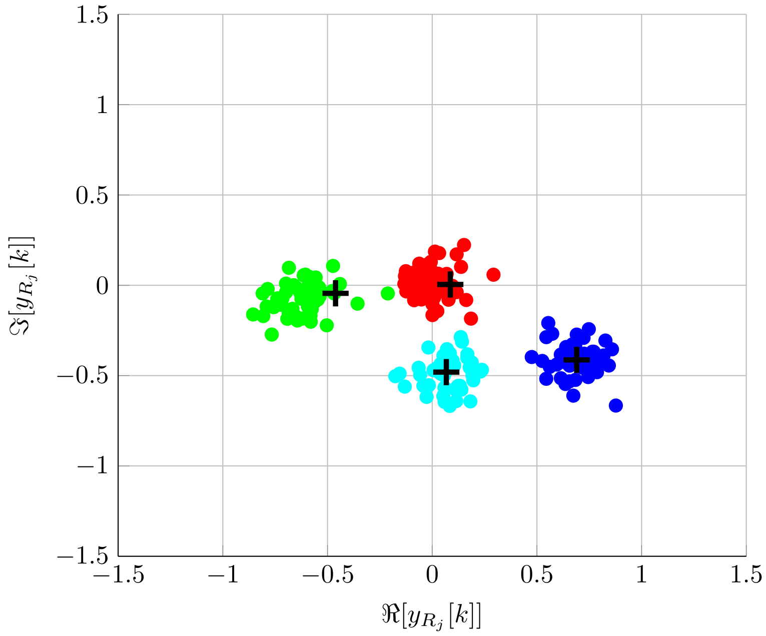

Figure 6.11: K-means iteration 4

$\hat{L}_j = L_j = 2$ because the best clustering is the one with $l = 4$. I also obtain the positions of the centroids $\{c^{(1)}, c^{(2)}, c^{(3)}, c^{(4)}\}$ (red crosses in Fig.6.17 and compare them to the correct channels – green crosses). One of them, let us say $c^{(4)}$, corresponds to the transmission of the zero OOK symbols and bears no information about the channel state.

In this example the particular form of Eq.(6.7) is

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} h_{1j} \\ h_{2j} \end{pmatrix} = \begin{pmatrix} c^{(u)} \\ c^{(v)} \\ c^{(w)} \end{pmatrix} . \tag{6.8}$$

The goal is to find a proper assignment between the estimated cluster centroids $\{c^{(1)}, c^{(2)}, c^{(3)}\}$ and its ordering $\{c^{(u)}, c^{(v)}, c^{(w)}\}$. From Eq.(6.8) it is obvious that we seek a pair of centroids that summed together gives the third one. If the solution is for example $c^{(1)} + c^{(2)} = c^{(3)}$ then $c^{(1)}$ and $c^{(2)}$ equals to h_{1j} and h_{2j}. Note that there is an ambiguity because the solution is not able to distinguish which channel parametrization belongs to which source. Although this ambiguity can be neglected for symmetric WPLNC mapping it comprise a significant drawback of blind clustering in general. Similar procedure can be tractably extended to more than two sources and other simple source alphabets such as BPSK. However the assumption of the symmetric WPLNC mappings is very strict, especially when the mapping shall be matched to the channels for SER improvements, as discussed in section 4.6. The ambiguity in the channel estimation capability has to be investigated by other methods.

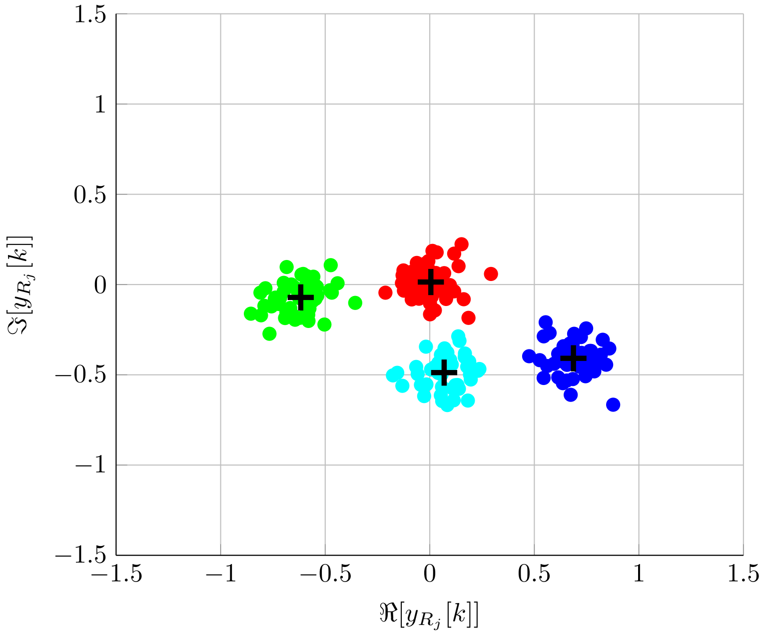

Figure 6.12: K-means iteration 5

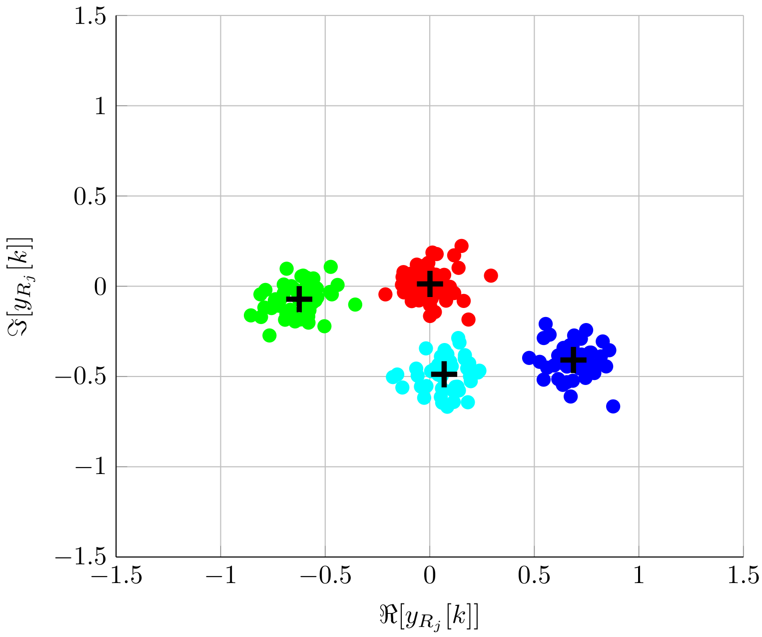

Figure 6.13: K-means iteration 6

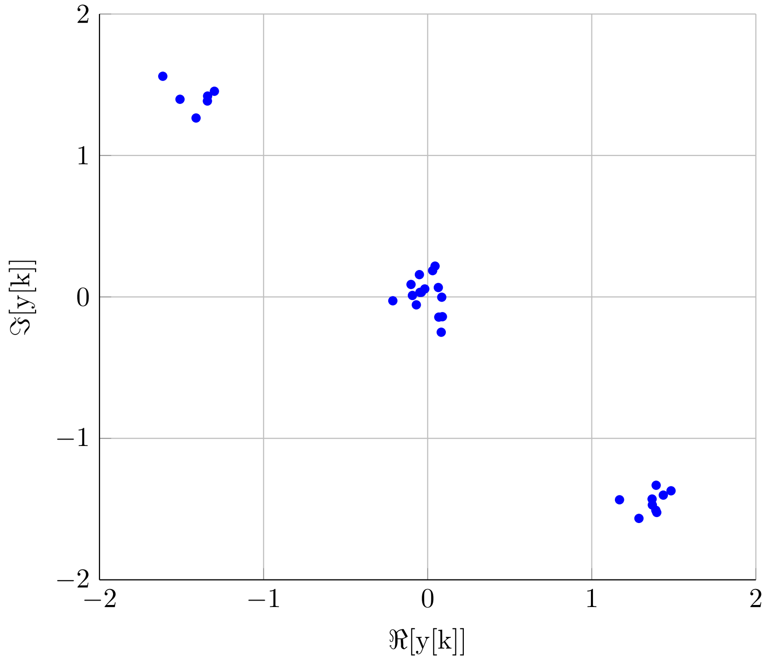

Figure 6.14: Superposed constellation for two BPSK signals with anti-collinear channels

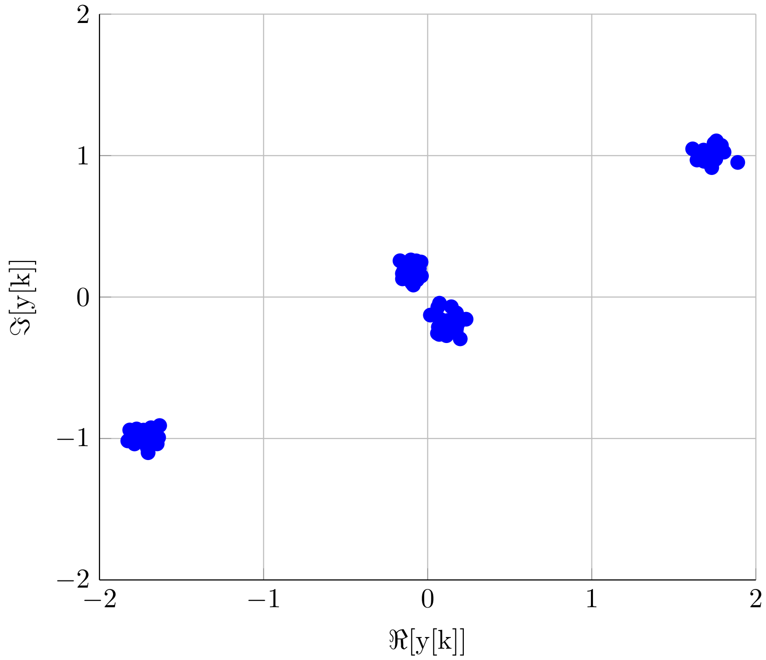

Figure 6.15: Superposed constellation for two BPSK signals in high SNR regime.

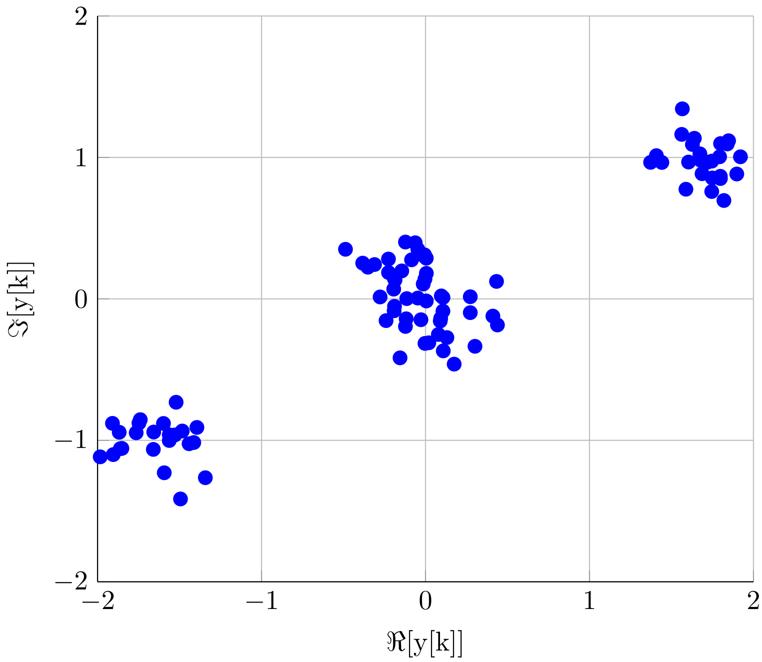

Figure 6.16: Superposed constellation for two BPSK signals in lower SNR regime.

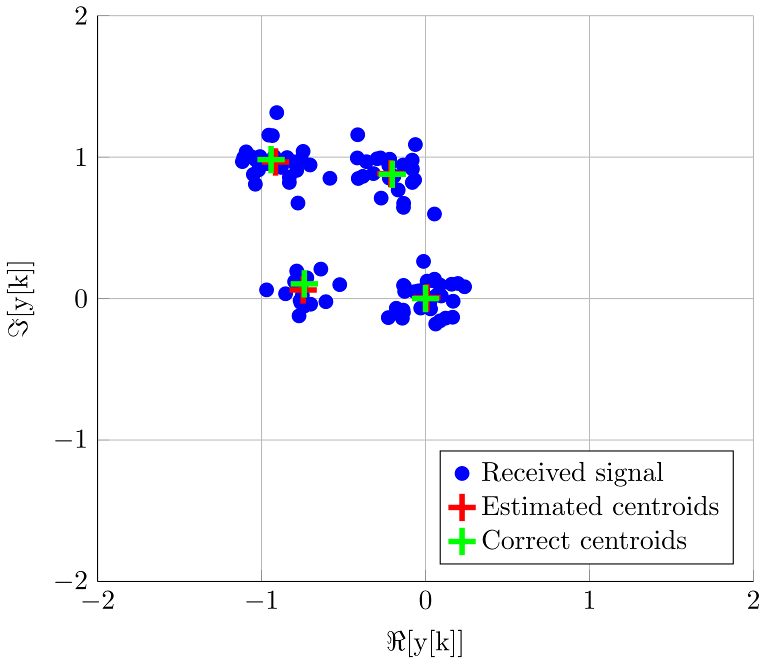

Figure 6.17: CIP channel estimation for two OOK signals.

6.4 Numerical results

First of all it is necessary to define a type and a length of the non-orthogonal sequences to be generated by the sources and to be used for initial estimation by the relays. From aforementioned requirements the sequence must be long enough to guarantee that the whole superposed constellation will be produced and the resulting superposed constellation shall be simple enough, with minimum points. Each source produces non-orthogonal sequence randomly without any specific setting or sequence pre-assignment.

The simplest constellation will be produced for OOK source alphabet since there will be always an all-zero point when none of the sources is in fact transmitting. Alternatively BPSK can be used since it still produces relatively simple superposed constellation. In many numerical examples here I will persist in using OOK alphabet although with real-world testing the BPSK will be used, since it has more suitable properties for transmission. It is important to note and highlight that modulation scheme used for CIP does not impose anything on later data transmission phase, any better suitable modulation and coding can be used.

To evaluate the necessary length of the sequence I define an event E meaning that the relay observes each constellation point of the superposed constellation at least once. We want to find such a length of the sequence n_0 that guarantees $\Pr\{E\} \to 1$. This probability can be evaluated analytically by an inclusion-exclusion principle [142].

$$\Pr(L, n) = 1 - \sum_{i=1}^{2^L - 1} (-1)^{i+1} \binom{2^L - 1}{i} \left(\frac{2^L - i}{2^L} \right)^n, \tag{6.9}$$

where $\Pr(L, n)$ is probability of having a complete superposed constellation of L sources and OOK sequence length n.

Probability of the event E for various sequence length and various number of operated sources is plotted in Fig.6.18. It was obtained by numerical simulation that perfectly matched Eq.(6.9). One can see that the sequence length $n_0 = 150$ is sufficient for $\Pr\{E\} \to 1$ up to four (my bound on L_{MAX}) sources operated by a relay. For four sources and $n_0 = 150$ we have $\Pr\{E\} = 0.999$. Such a sequence length may be seen as unbearably long, wasting of transmission resources. However, this is an inherent penalty to be paid for blind and distributed approach. Also, CIP algorithm is repeated only from time to time, based on dynamics of the cloud network, relay movement and/or node (dis)connection. As such this length may be tolerable.

An example of superposed constellation that misses some of the points is shown in Fig.6.19. This example shows superposition of two BPSK signals, each 30 symbols long. Even though the sequence length 30 should be enough, there is a non zero, although negligible, probability, that the constellation will not be full. With the length of the sequence this probability obviously tends more and more to zero. 30 symbol superposed constellation usually looks as in Fig.6.20.

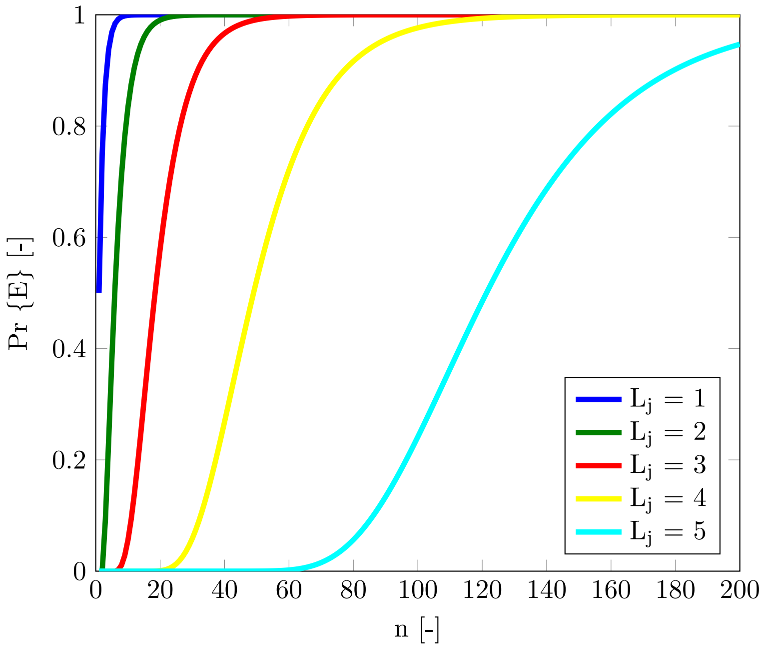

Figure 6.18: Probability of full constellation.

The proposed CIP algorithm 5 is first implemented in MATLAB and its properties are numerically evaluated in various scenarios (number of sources, random channels, impact of the threshold level, etc.). The simulations mainly test the abilities of the CIP to correctly estimate the number of the communicating sources under random channel parametrizations. All simulations were performed with the following parameters: the random OOK sequence length $n_0 = 150$, the number of the repetitions of the K-means algorithm $r_{MAX} = 5$ and the maximum number of the sources received by the relay $L_{MAX} = 4$. Only a limited imbalance among the amplitudes of the channel states is assumed such that $|h_{ij}|$ $\forall i, j$ is a random variable with the uniform distribution on the closed interval $[0.5, 1]$. The channel phases $\measuredangle h_{ij}$ $\forall i, j$ are random variables with the uniform distribution on the interval $[0, 2\pi)$. The amplitude bound is a form of threshold to decide if the particular source belongs to the set of the sources operated by the relay R_j or not. Different statistics for both amplitude and phase are obviously possible. However, a lower bound for amplitude value will be in place regardless the channel statistics. It is hard to apply sensible WPLNC function when the particular sources are highly imbalanced as I showed in section 4.6.

In the following I define a signal to noise ratio (SNR) at the relay R_j by $\gamma_j = E[|Q_j|]^2/\sigma_w^2$, where $E[|Q_j|]^2$ is the energy of the superposition constellation which is a function of the number of the sources L_j, set of channel realizations $\{h_{ij}\}_{i:S_i \in \mathcal{S}_j}$ and the individual source channel alphabets $\{q_i\}$ $\forall i : S_i \in \mathcal{S}_j$. Operator $E[\cdot]$ denotes the expectation.

Figs.6.21-6.24 show the example results after the clustering of the received superposition of the randomly generated 150 symbols long OOK sequences

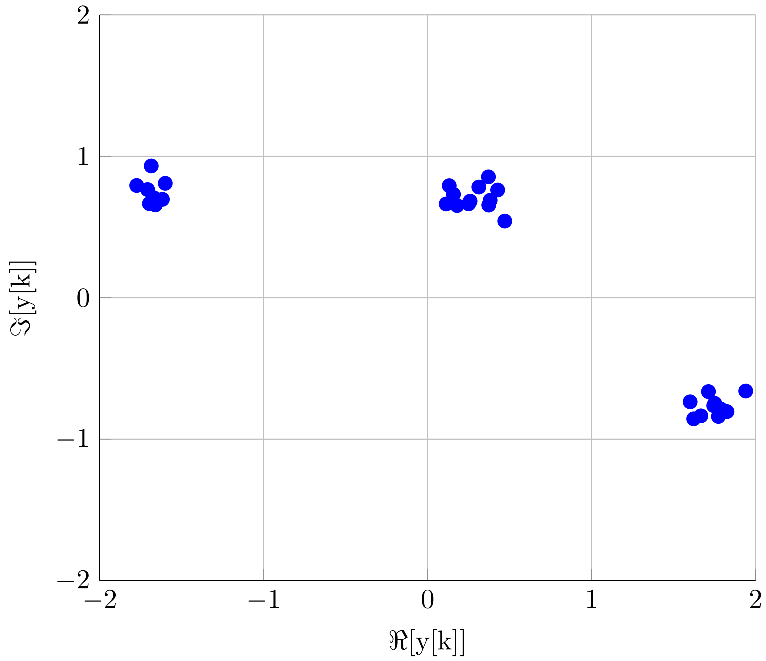

Figure 6.19: Incomplete superposed constellation.

with two, three, respectively, four sources for random channel parametrizations. Fig.6.21 shows the correct clustering in the case of two sources. The incorrect clustering is depicted in Fig.6.22. Here the channel parametrizations, which is approximately anti-collinear, cause two points of the superposed constellation to fall close to each other and thus to be clustered within one cluster. It is important to note that the number of the operated sources is correctly estimated in this situation due to ceiling $\hat{L}_j = \lceil \log_2(l_{best}) \rceil$. On the other hand the incorrect clustering will complicate the channel estimation procedure and also the channel estimation error will increase. Fig.6.23 shows the correct clustering of the signal from three operated sources. And finally Fig.6.24 shows the most difficult case for clustering – clustering of four sources, which was incorrect in this case.

The performance of the CIP algorithm is significantly determined by the choice of the threshold value d_{th}. Fig.6.25 shows the probability of the correct estimation of the number of the operated sources parametrized by the threshold value, i.e. $\Pr\{L_j = \hat{L}_j | d_{th}\}$. The proposed algorithm achieves approximately 97% probability of the correct detection of the number of the operated sources at the high SNR regime. It is important to highlight that this is not equivalent to any of errors of the communication phase such as bit or symbol error. When the number of the sources is incorrectly estimated then it is possible that some of the sources will be ignored. This is not an issue for dense cloud networks where source interactions with multiple relays are beneficial. There will be an issue if the ignored source is connected to just one relay, i.e. belongs into just one set $\mathcal{S}_{R_j}$. On the other hand, the source is usually ignored only if it has a weak channel to the relay, if the other

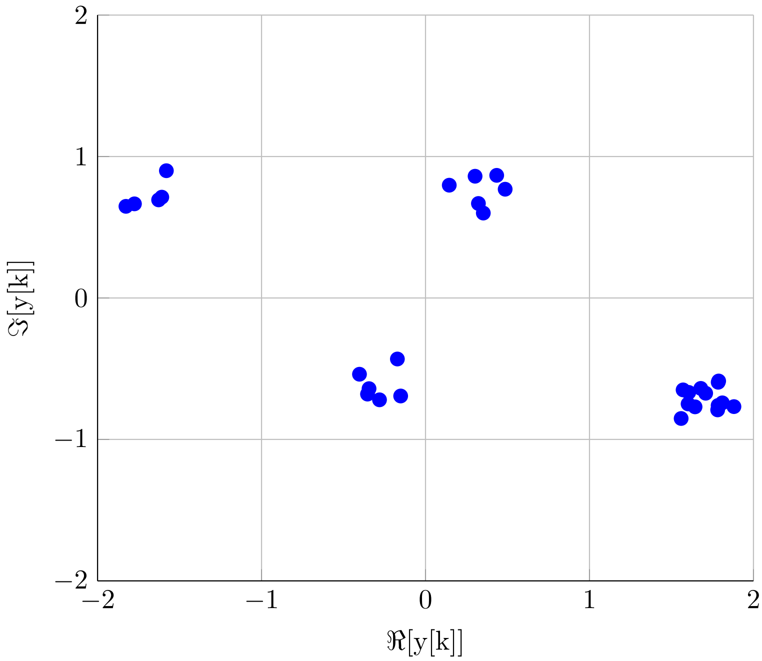

Figure 6.20: Complete superposed constellation.

sources have significantly stronger channel and/or in bad noise conditions. It is natural, that under these conditions the source may be ignored even with classical PHY solutions and even in point-to-point case.

The optimal, providing the highest probability of correct estimation, threshold value d_{th} is close to 3 in wide range of high SNR region. At the lower SNRs the high probability of correct estimation can be achieved by adaptive selection of the threshold value according to the actual SNR. Obviously the algorithm performance is vulnerable to noise.

An exhaustive numerical simulation was carried to find out properties of CIP algorithm, such as dependence on random sequence length, threshold value settings etc. A number of trials in random channel with varying number of sources (from 1 to 4) were performed. Various properties are shown in Figs.6.26 to 6.32.

First of all an impact of the random sequence length is analysed. Figs.6.26 to 6.29 show, for 1 up to 4 sources, respectively, the performance of CIP algorithm for different sequence lengths, always for the best threshold value for each SNR point. It can be generalised, that for high SNR values a high probability (around 97% for the worst case of 4 sources) can be achieved with sufficiently long sequences. In low SNR regions the shorter (verified on the range of 40 to 160 symbols) the sequence the better the performance. Obviously the sequence could not be too short. Superposed constellation with missing points must be avoided. Since the noise is high in this regime the scatter of the constellation points is bigger and thus false clusters are likely to be produced when the sequence is too long. On the other hand in the high SNR regime the longer the sequence the higher the performance.

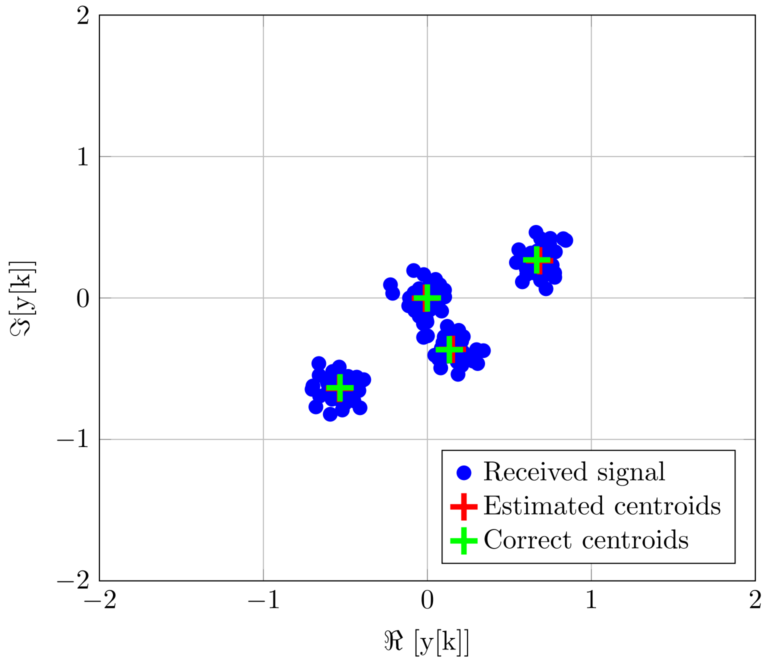

Figure 6.21: Correct clustering of two sources.

This is the most evident in 4 source case shown in Fig.6.29. Relatively high success rate of CIP in 4 source case in low SNR regime is caused by current implementation and should be mostly considered as a false alarm. These results are later deleted from the study.

An impact of the threshold d_{th} selection is visible in Figs.6.30 and 6.31. Both use 100 symbols long sequences. Evidently, the threshold value must be matched to the actual SNR to provide sufficient performance, this can be done for example by a look up table. Threshold values around 2 to 3 are mostly optimal. The better performance, especially in 4 source case, can be achieved by sequences longer than 100 symbols.

Finally, Fig.6.32 shows optimal tuning of the CIP algorithm - sequence length and threshold value. Probability of correct estimation higher than 95% can be achieved even in 4 source case. Notice that low SNR values in 4 source case are deleted due to high false alarm rate.

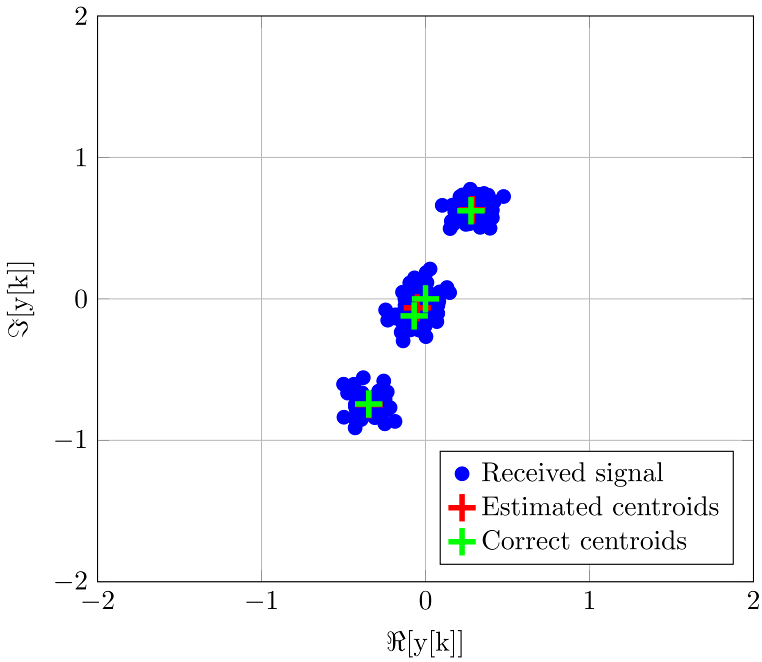

Figure 6.22: Incorrect clustering of two sources.

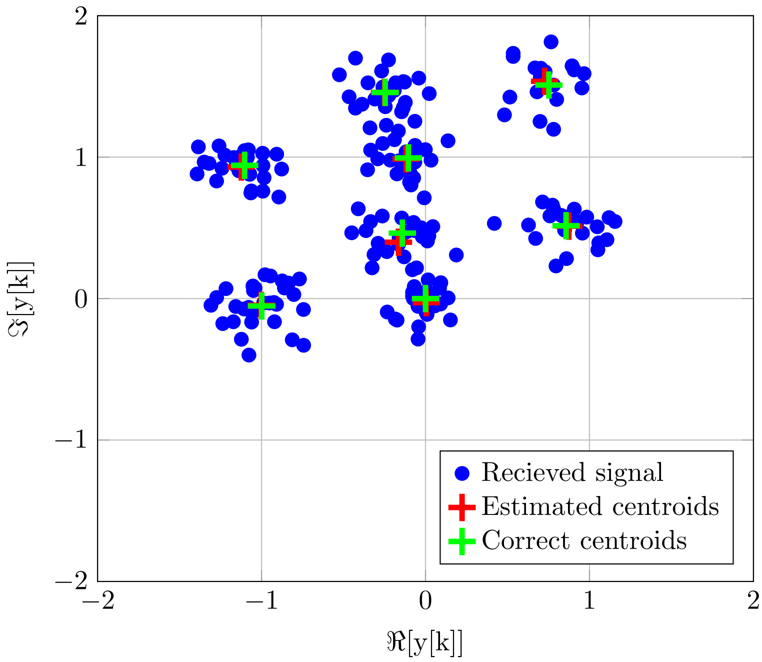

Figure 6.23: Correct clustering of three sources.

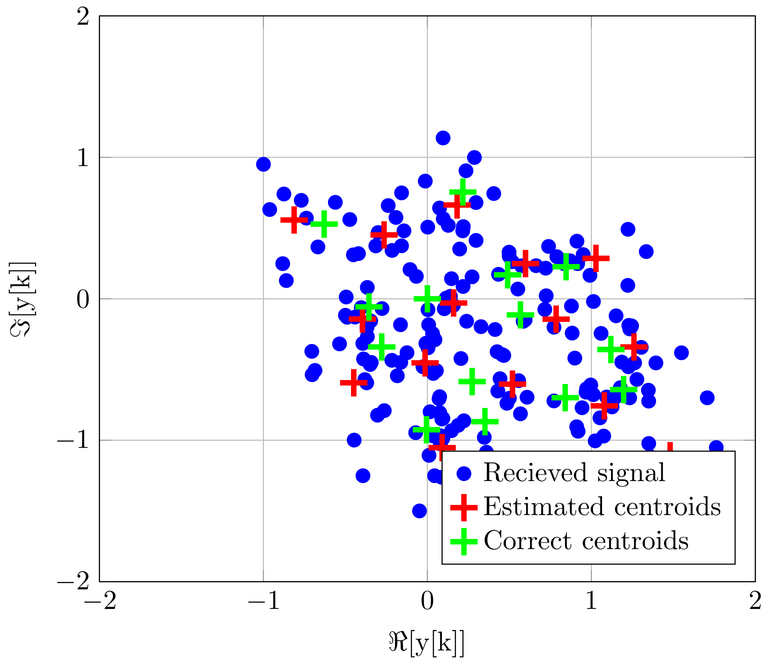

Figure 6.24: Incorrect clustering of four sources.

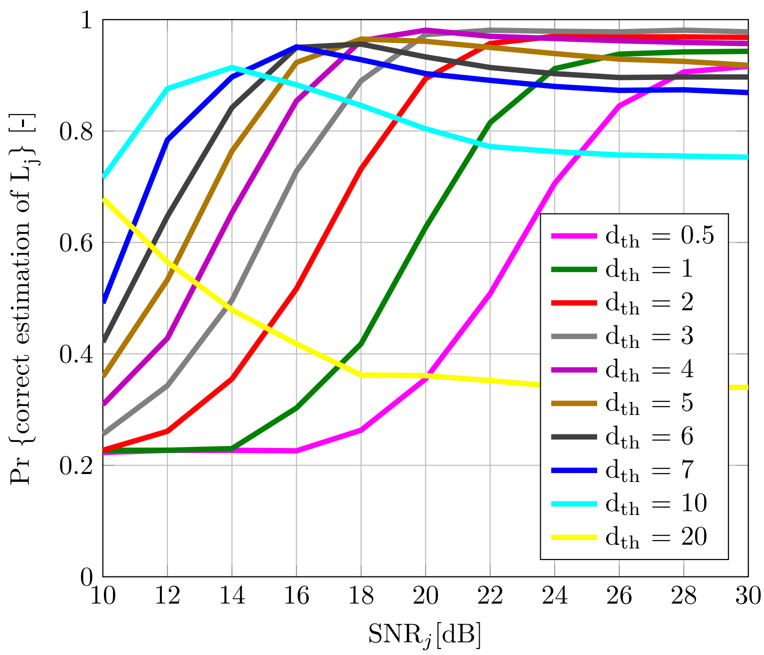

Figure 6.25: Probability of correct estimation of the number of transmitting sources - numerical evaluation.

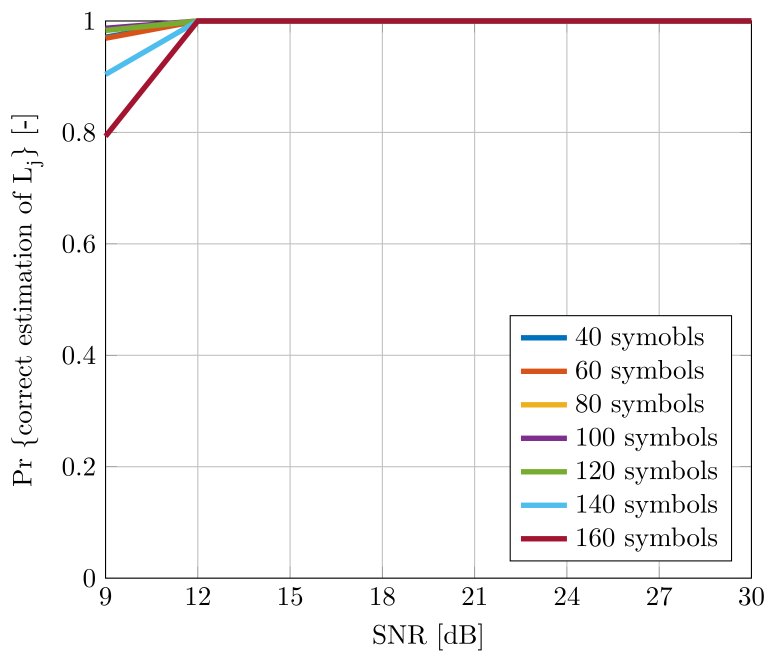

Figure 6.26: Probability of correct estimation for one source - varying sequence length, optimal d_{th}.

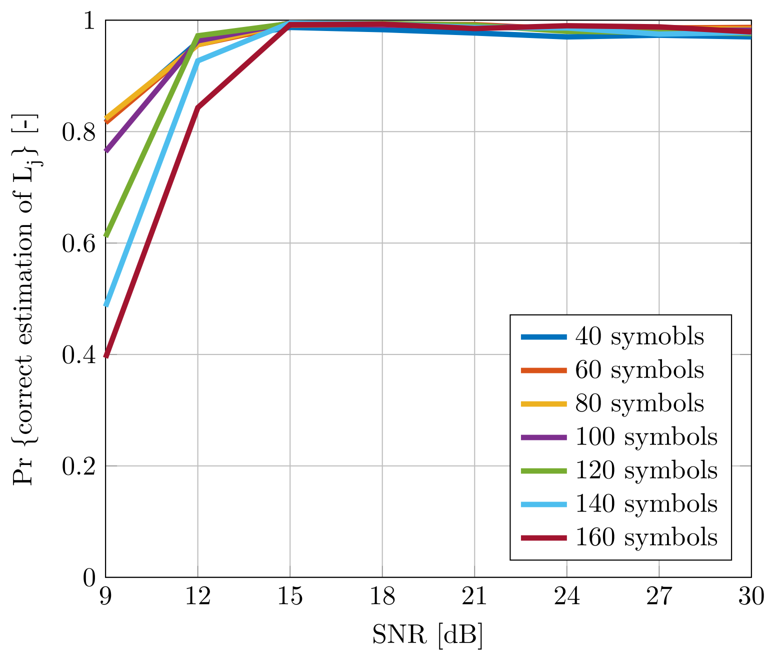

Figure 6.27: Probability of correct estimation for two sources - varying sequence length, optimal d_{th}.

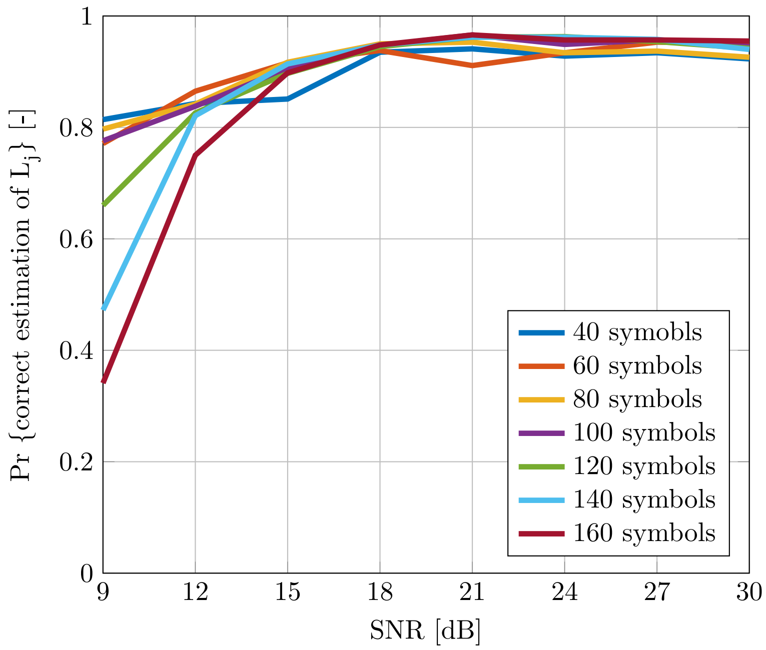

Figure 6.28: Probability of correct estimation for three sources - varying sequence length, optimal d_{th}.

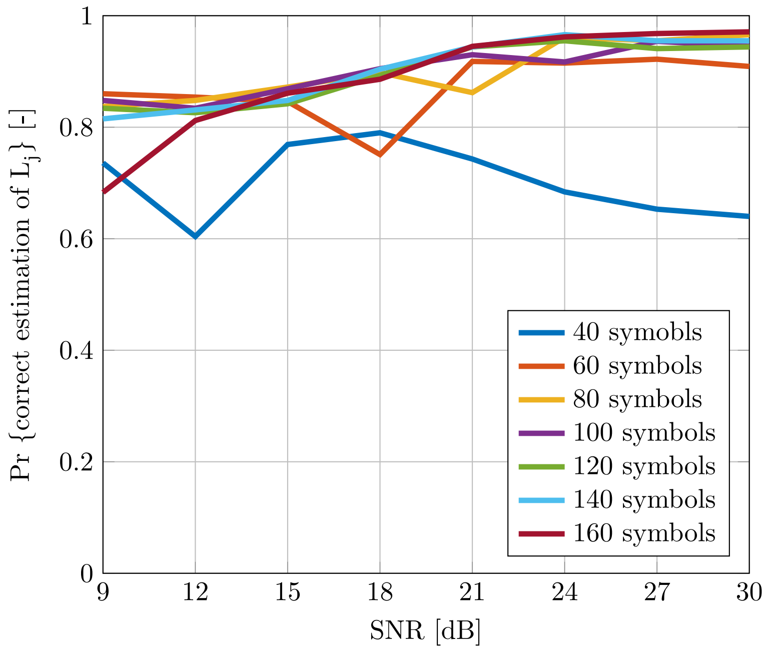

Figure 6.29: Probability of correct estimation for four sources - varying sequence length, optimal d_{th}.

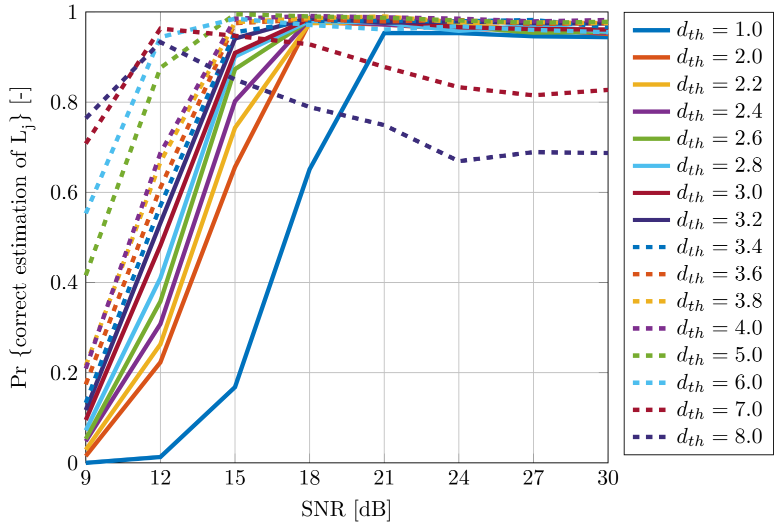

Figure 6.30: Probability of correct estimation for two sources - sequence length 100, varying d_{th}.

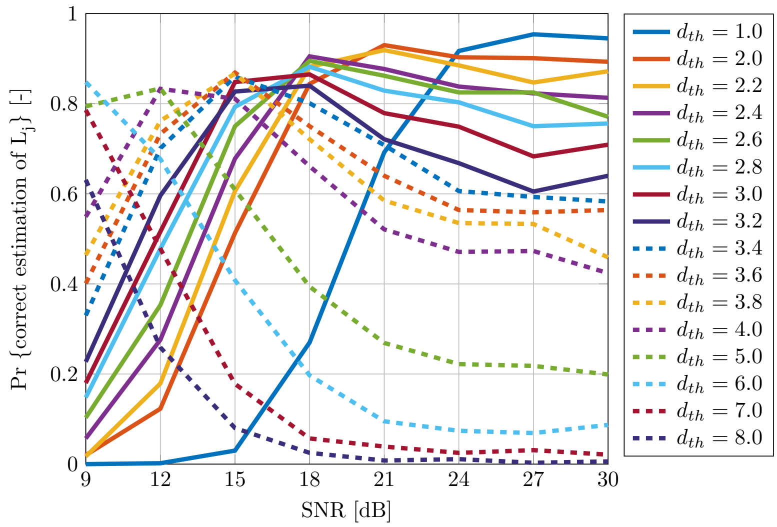

Figure 6.31: Probability of correct estimation for four sources - sequence length 100, varying d_{th}.

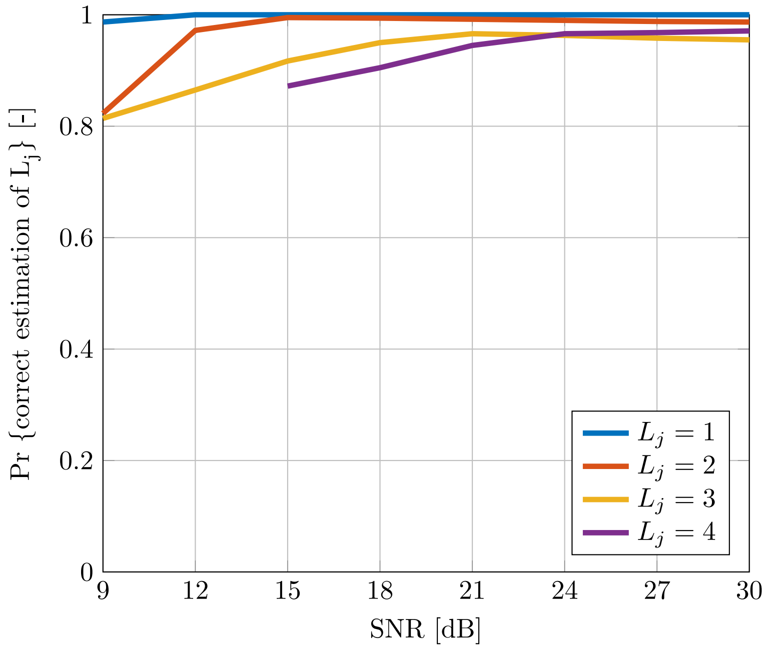

Figure 6.32: Probability of correct estimation - optimal parameter settings.

6.5 Hardware implementation and real-world testing

The hardware performance of the proposed clustering algorithm is evaluated again using Ettus Research Universal Software Radio Peripherals (USRPs) based test-bed with similar, although largely simplified, version for advanced DLA demonstration, see chapter 5.

A significant change, relative to theoretical and numerical analysis, is employment of BPSK modulation instead of OOK by the sources. This is because OOK has not got suitable properties for OFDM transmission, especially Peak to Average Power Ratio (PAPR) is an issue with OOK. On the other hand BPSK as a source alphabet in the considered topology still provides relatively simple superposed constellations.

Verification is done on a 3-node, i.e. a simple one relay cloud with two sources, which is shown in compact form in Fig.6.33. The USRPs each have a single 3dBi omni-directional antenna (at 2.4 - 2.5 GHz and 4.9 - 5.85 GHz). The two USRPs on the left-hand side are the sources (S_A lower, S_B upper) and the USRP on the right-hand side is the receiving relay (R_R). The first transmitter (S_A) USRP is connected to a powerful Core i7 laptop, running a single instance of GNURadio, by a Gigabit Ethernet link. The second transmitter (S_B) USRP is connected to the first by the MIMO expansion cable. This provides the second transmitter with baseband data from GNURadio as well as time and frequency synchronization signals from the first transmitter. This means that sample-level synchronization of the transmitted signals from the two transmitters can be assured. It should be noted that this cable is for baseband data and synchronization only and not radio-frequency waveform/data transmission or reception. The receiver USRP (R_R) is connected to a second laptop running GNURadio by a Gigabit Ethernet link.

Orthogonal pilot signals and frame structure as used within advanced DLA demonstration are reused to achieve Carrier Frequency and Timing Offset corrections. Also the orthogonal signals for channel estimation are used. The CIP capability of channel estimation is not used because of inherent ambiguity. Nevertheless the CIP is used for estimation of number of communicating sources and is solely performed on overlapped data from payload part, no information from orthogonal synchronisation words, nor from channel tracking pilots is used.

Various tests and verification were performed using the OFDM implementation test-bed. Over-the-wire and over-the-air tests were both carried out. I present the results only for the most interesting case - the over-the-air transmission.

The receiving node is oblivious to the number of nodes in this set-up. It has to recover it from the received superposed constellations using the CIP algorithm. Figs.6.34 and 6.35. show the output of the CIP algorithm for sequences of 100 BPSK symbols two for different Tx gains (and thus different SNRs). Note how the shape of the superposed constellation changes with

Figure 6.33: Compact laboratory placement of the network nodes for the CIP verification experiment.

SNR and with different channel realizations. In both cases the CIP algorithm successfully recovers the number of sources (two in both cases). The positions of the centroids, which is another output of the CIP algorithm, correspond correctly to the shape of the superposed constellation. Thus the channel estimation would be possible for both cases although ambiguous.

Since the performance of the CIP algorithm depends on the shape of the superposed constellation (and thus on channel realizations) and on a number of CIP input parameters. The average CIP behaviour was analysed under various channel conditions and threshold d_{th} optimisation was done. For each SNR a number of independent measurements were taken. SNR is expressed in terms of estimated values by the receiving node rather than true output radio-frequency power. Thus there may exist a shift between SNR used for numerical results and SNR used for experiments. Exact calibration, requiring specific measurements equipment should be necessary, but was not done. Nevertheless the measurement campaign provides useful insight into real-world behaviour regardless the potential SNR shift.

Source CIP random sequences of $n_0 = 100$ BPSK symbols were used. The sequence length is longer than necessary for two source case but it provides good results and is viable even using real-time GNURadio implementation of CIP at the receiving relay (as opposed to off-line post-processing). The window length also allows the addition of more sources and thus the analysis of more complicated topologies. During the data processing it was observed that the number of K-means runs $r_{max} = 15$ provides satisfactory results with a tractable processing time.

The most critical parameter determining the quality of the CIP results is the threshold value d_{th}. It defines when to consider the K-means clustering to be good enough to provide correct estimates based on the residual inter-cluster distances. Too low d_{th} values perform badly in low SNR regions since the superposed constellation points affected by strong noise (that should be considered as just a one cluster) are incorrectly divided in two or more

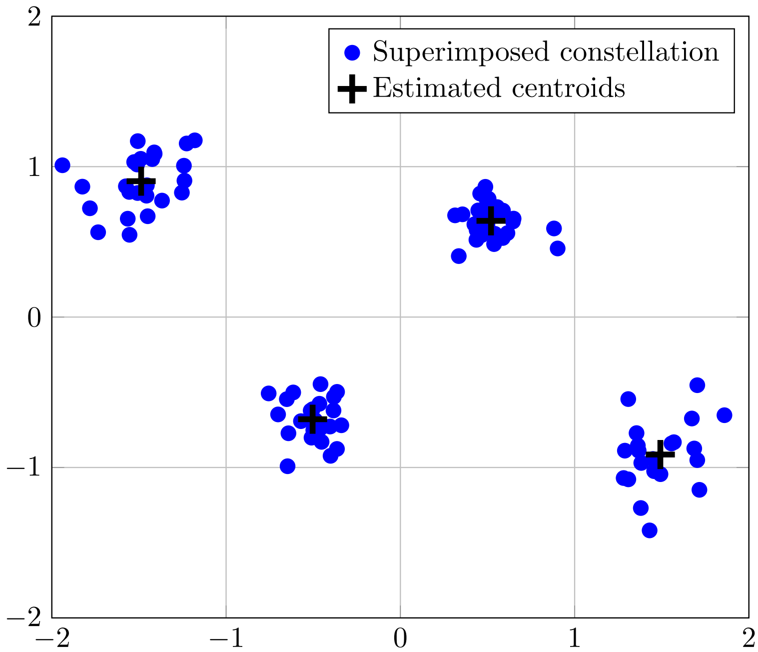

Figure 6.34: CIP results for SNR = 20dB - hardware evaluation.

clusters. On the other hand too high values in high SNR regime puts two or more superposed constellation points (that should be separated) into one cluster. Currently a closed form formula for d_{th} selection given the SNR is unknown, however it can be experimentally verified and provided in a form of look-up table.

The probability of correct estimation of the number of operated sources L_j as a function of the threshold value d_{th} is evaluated for various Tx gains (resulting in various SNRs) for several random and independent channel realizations. The focus here is on the situation when the receiving node distinguishes between the set-up with one or two active transmitters. The results are presented in Fig.6.36. This figure can be compared to the similar one (Fig.6.25) obtained by the numerical simulation. Notice that for hardware verification a significantly larger scale for threshold values was used, mind the logarithm scale of the x-axis. The optimal threshold value can be determined for each SNR scenario. The general trend is to choose higher d_{th} for lower SNR. High probability, 100% or very close to 100%, of correct estimation can be achieved for each case by fine tuning of d_{th}. The effect of threshold overshoot can be seen in Fig.6.36, with ever increasing d_{th} the CIP algorithm will underestimate the number of sources (create less cluster than it should, because incorrect clustering is selected too soon due to badly chosen threshold) and thus the probability of correct estimation will decrease.

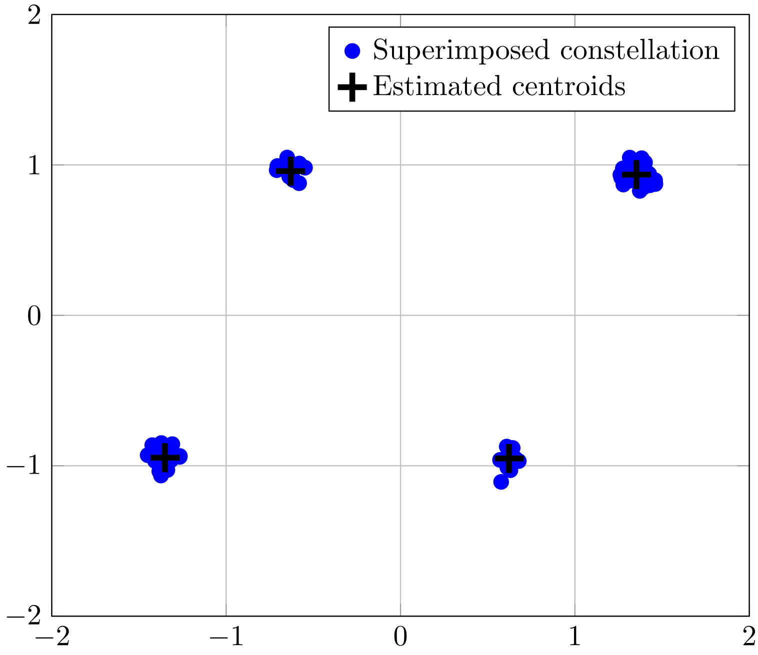

Figure 6.35: CIP results for SNR = 35dB - hardware evaluation.

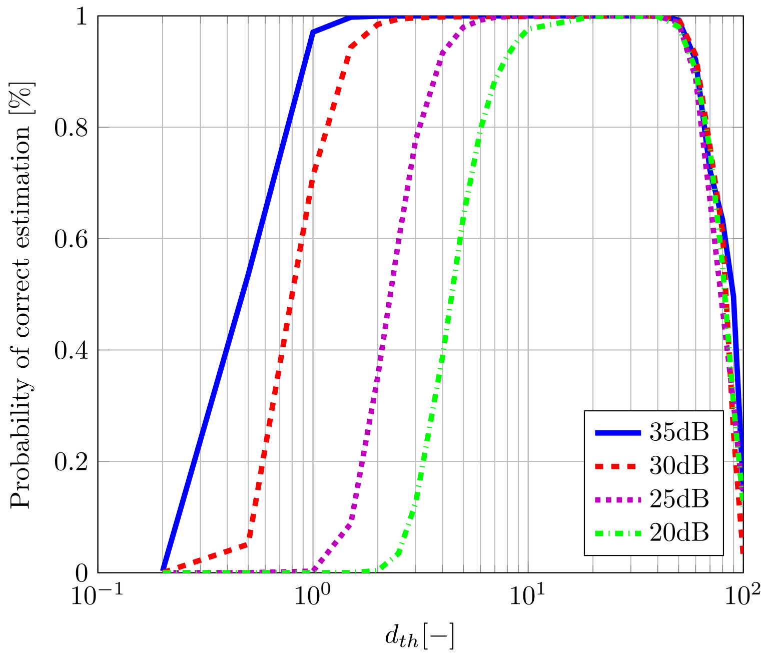

Figure 6.36: Probability of correct estimation of the number of transmitting sources - hardware evaluation.

6.6 Summary

The blind algorithm, Cloud Initialisation Procedure, that provides the relay nodes the necessary information to set-up WPLNC phase of communication is presented. The algorithm is based on blind clustering of non-orthogonal random sequences produced by independent, non-cooperating sources. The most effort is laid on simplicity of the CIP algorithm. Any potential inter-node cooperation is minimised, simple transmission schemes are assumed and low computational demanding operations are used. This effort for simplicity on the other hand may bring suboptimal performance. The output of the algorithm is twofold estimate of:

- the number of the sources that can be received by the relay,
- the particular sources to relay channels.

The CIP algorithm is verified by both numerical simulations and hardware experiments. Both show highly promising results. Most importantly, the algorithm is able to correctly estimate the number of communicating sources with probability close to one, under correct settings of key parameters. The channel estimation capability is somewhat questionable, mainly because of inherent resolution ambiguity.

Although the techniques of pattern recognition and automatic modulation classification are applied in vast technical areas, even to modulation type and channel estimation, to my best knowledge, supported by exhaustive literature search, there is no similar effort taken in the case of multi-source communications. The CIP algorithm is the only one that tries to solve the presented obstacles connected with the initialisation of WPLNC communication. Its most important task is that it paves the way towards more advanced algorithms.

Related approach that also avoids orthogonal signalisation and assumes channel state as unknown to be estimated is named Channel Class [143]. Also this algorithm has its particular limitations and is evaluated by numerical simulation in very simplistic scenario only.

Chapter 7

Conclusions

7.1 Summary of contribution

Several algorithms for distributed WPLNC operation of cloud networks are proposed, analysed and discussed. Two main areas are covered – distributed optimal allocation of WPLNC mapping to individual relays and non-orthogonal signalisation.

Using the algorithms presented in this thesis the cloud network is equipped with communication protocol that is able to optimally allocate WPLNC mappings to individual relay nodes. The allocation enables all destinations to recover source data from several received WPLNC encoded signals. Thus it enables terminal to terminal communication over the cloud network. The allocation also optimises some defined utility function of all the relays. Many time the utility used is the relay output cardinality, however, channel related utility – symbol error rate – is also provided and discussed. Using the proposed algorithm each relay is able to chose the proper WPLNC mapping and is able to adapt it immediately when situation in its neighbourhood changes. By this mechanism the network can smoothly and reliably react to node failures, their disconnections or reconnections back to the network.

The relay nodes are also able by the implementation of the proposed protocol to interact with non-standard relays. Particularly maliciously behaving relays that tries to harm and block the other nodes or relays that change their behaviour, become selfish, due to emptying battery.

Some initial effort is given to non-orthogonal signalisation schemes. With the proposed scheme the relay node is able to deduce the number of sources contained in the incoming signal. Using classification algorithms it is also able to estimate wireless channels of these sources, although not fully and unambiguously. Both of these estimates are necessary to initiate WPLNC communication. This algorithm helps to overcome traditional dependency of non-orthogonal WPLNC on orthogonal signalisation.

One of the key part of the thesis is the verification of proposed algorithms using the cloud network test-bed. Whenever it was possible the algorithms were tested in real environment using software defined radios. A care was taken to come as close as possible to real application scenarios. Quite advanced and developed self-organising cloud demonstrator was build.

7.2 Future research

Although a lot of work was done to join cloud based network with WPLNC technique there are still many areas that need further research. From the point of view of presented algorithms assigning WPLNC mappings to the nodes even more complex network should be analysed and suitable protocols designed. Especially non-layered networks should be in the main interest.

Practical protocols for implementation of distributed WPLNC allocating mechanisms are missing for advanced scenarios of malicious or battery dependent behaviour of the relays. Presumably they should be implemented as additional extensions of advanced DLA algorithm.

More elaborated mathematical tool from the area of game theory can be used to devise even more complicated schemes. For example repeated games and games including nodes' reputation can be used to not only identify but also mitigate maliciously behaving relays.

There were several simplifying assumptions taken to ease initial analysis. These simplification shall be removed to create more realistic scenarios. Particularly, WPLNC assignment shall be joined with advanced forward error correction techniques. Distributed algorithm for transmission activity assignment is significant missing piece. There are also some issues connected with protocol implementation such as signals and preamble designs to make existing preambles more flexible and less restricting.

Almost not elaborated is development of non-orthogonal signalisation to be used with non-orthogonal WPLNC. Although CIP presented in this thesis represents some initial effort a fully deployable scheme is unknown.

Bibliography

[1] "Cisco visual networking index:forecast and methodology, 2015–2020," White Paper, Cisco, Jun. 2016.

[2] A. Osseiran, J. F. Monserrat, and P. Marsch, *5G Mobile and Wireless Communications Technology*, 1st ed. New York, NY, USA: Cambridge University Press, 2016.

[3] L. Srivastava, P. Biggs, T. Kelly, Y. Lozanova, L. P. Chavolla, J. Ponder, R. Sagalbayeva, S. Skvortsova, and C. Sund, "ITU internet report 2005: The internet of things," International Telecommunication Union, Tech. Rep., 2005.

[4] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, Q4 2015.

[5] M. R. Palattella, M. Dohler, A. Grieco, G. Rizzo, J. Torsner, T. Engel, and L. Ladid, "Internet of things in the 5g era: Enablers, architecture, and business models," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 3, pp. 510–527, March 2016.

[6] A. Dohr, R. Modre-Opsrian, M. Drobics, D. Hayn, and G. Schreier, "The internet of things for ambient assisted living," in *2010 Seventh International Conference on Information Technology: New Generations*, April 2010, pp. 804–809.

[7] H. Hartenstein and L. P. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, June 2008.

[8] C. Liu, K. T. Chau, D. Wu, and S. Gao, "Opportunities and challenges of vehicle-to-home, vehicle-to-vehicle, and vehicle-to-grid technologies," *Proceedings of the IEEE*, vol. 101, no. 11, pp. 2409–2427, Nov 2013.

[9] L. D. Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov 2014.

[10] Z. Bi, L. D. Xu, and C. Wang, "Internet of things for enterprise systems of modern manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1537–1546, May 2014.

[11] P. Rashidi and D. J. Cook, "Keeping the resident in the loop: Adapting the smart home to the user," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 39, no. 5, pp. 949–959, Sept 2009.

[12] M. Starsinic, "System architecture challenges in the home m2m network," in *2010 IEEE Long Island Systems, Applications and Technology Conference*, May 2010, pp. 1–7.

[13] D. m. Han and J. h. Lim, "Smart home energy management system using ieee 802.15.4 and zigbee," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 3, pp. 1403–1410, Aug 2010.

[14] M. R. Alam, M. B. I. Reaz, and M. A. M. Ali, "A review of smart homes - past, present, and future," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1190–1203, Nov 2012.

[15] D. Snoonian, "Smart buildings," *IEEE Spectrum*, vol. 40, no. 8, pp. 18–23, Aug 2003.

[16] T. Weng and Y. Agarwal, "From buildings to smart buildings; sensing and actuation to improve energy efficiency," *IEEE Design Test of Computers*, vol. 29, no. 4, pp. 36–44, Aug 2012.

[17] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, "A survey on smart grid communication infrastructures: Motivations, requirements and challenges," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 5–20, Q1 2013.

[18] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of things for smart cities," *IEEE Internet of Things Journal*, vol. 1, no. 1, pp. 22–32, Feb 2014.

[19] B. Warneke, M. Last, B. Liebowitz, and K. S. J. Pister, "Smart dust: communicating with a cubic-millimeter computer," *Computer*, vol. 34, no. 1, pp. 44–51, Jan 2001.

[20] I. F. Akyildiz, M. Pierobon, S. Balasubramaniam, and Y. Koucheryavy, "The internet of bio-nano things," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 32–40, March 2015.

[21] S. Andreev, O. Galinina, A. Pyattaev, M. Gerasimenko, T. Tirronen, J. Torsner, J. Sachs, M. Dohler, and Y. Koucheryavy, "Understanding the iot connectivity landscape: a contemporary m2m radio technology roadmap," *IEEE Communications Magazine*, vol. 53, no. 9, pp. 32–40, September 2015.

[22] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud computing and grid computing 360-degree compared," in *Grid Computing Environments Workshop, 2008. GCE '08*, 2008, pp. 1–10.

[23] M. Vouk, "Cloud computing - issues, research and implementations," in *Information Technology Interfaces, 2008. ITI 2008. 30th International Conference on*, 2008, pp. 31–40.

[24] B. Rimal, E. Choi, and I. Lumb, "A taxonomy and survey of cloud computing systems," in *INC, IMS and IDC, 2009. NCM '09. Fifth International Joint Conference on*, 2009, pp. 44–51.

[25] L. Youseff, M. Butrico, and D. Da Silva, "Toward a unified ontology of cloud computing," in *Grid Computing Environments Workshop, 2008. GCE '08*, 2008, pp. 1–10.

[26] "DIWINE - Dense cooperative wireless cloud network," http://www.diwine-project.eu, accessed: 2016-11-21.

[27] O. G. Aliu, A. Imran, M. A. Imran, and B. Evans, "A survey of self organisation in future cellular networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 1, pp. 336–361, First 2013.

[28] O. Simeone, U. Spagnolini, Y. Bar-Ness, and S. H. Strogatz, "Distributed synchronization in wireless networks," *Signal Processing Magazine, IEEE*, vol. 25, no. 5, pp. 81–97, 2008.

[29] M. A. Alvarez, B. Azari, and U. Spagnolini, "Time and frequency self-synchronization in dense cooperative network," in *2014 48th Asilomar Conference on Signals, Systems and Computers*, Nov 2014, pp. 1811–1815.

[30] P. Kamalinejad, C. Mahapatra, Z. Sheng, S. Mirabbasi, V. C. M. Leung, and Y. L. Guan, "Wireless energy harvesting for the internet of things," *IEEE Communications Magazine*, vol. 53, no. 6, pp. 102–108, June 2015.

[31] O. Georgiou, K. Mimis, D. Halls, W. H. Thompson, and D. Gibbins, "How many wi-fi aps does it take to light a lightbulb?" *IEEE Access*, vol. 4, pp. 3732–3746, 2016.

[32] H. Jayakumar, K. Lee, W. S. Lee, A. Raha, Y. Kim, and V. Raghunathan, "Powering the internet of things," in *2014 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, Aug 2014, pp. 375–380.

[33] E. Tsimbalo, X. Fafoutis, and R. J. Piechocki, "Crc error correction in iot applications," *IEEE Transactions on Industrial Informatics*, vol. PP, no. 99, pp. 1–1, 2016.

[34] D. Medhi and K. Ramasamy, *Network Routing: Algorithms, Protocols, and Architectures*, ser. The Morgan Kaufmann Series in Networking. Elsevier Science, 2010.

[35] C. Huitema, *Routing in the Internet*, ser. CCIE Resource Library. Prentice Hall PTR, 2000.

[36] J. Al-Karaki and A. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6–28, 2004.

[37] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *Information Theory, IEEE Transactions on*, vol. 46, no. 4, pp. 1204 –1216, Jul. 2000.

[38] M. Medard and A. Sprintson, *Network Coding: Fundamentals and Applications*, ser. Academic Press. Academic/Elsevier, 2012.

[39] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network." *Canadian Journal of Mathematics*, vol. 8, pp. 399–404, 1956.

[40] S.-Y. Li, R. Yeung, and N. Cai, "Linear network coding," *Information Theory, IEEE Transactions on*, vol. 49, no. 2, pp. 371–381, 2003.

[41] R. Koetter and M. Medard, "An algebraic approach to network coding," *Networking, IEEE/ACM Transactions on*, vol. 11, no. 5, pp. 782–795, 2003.

[42] R. Dougherty, C. Freiling, and K. Zeger, "Insufficiency of linear coding in network information flow," *Information Theory, IEEE Transactions on*, vol. 51, no. 8, pp. 2745–2759, 2005.

[43] S. Y. R. Li and Q. T. Sun, "Network coding theory via commutative algebra," *IEEE Transactions on Information Theory*, vol. 57, no. 1, pp. 403–415, Jan 2011.

[44] R. W. Yeung and N. Cai, "Network error correction, i: Basic concepts and upper bounds," *Commun. Inf. Syst.*, vol. 6, no. 1, pp. 19–35, 2006.

[45] N. Cai and R. W. Yeung, "Network error correction, ii: Lower bounds," *Commun. Inf. Syst.*, vol. 6, no. 1, pp. 37–54, 2006.

[46] S. Katti, D. Katabi, H. Balakrishnan, and M. Medard, "Symbol-Level Network Coding for Wireless Mesh Networks," in *ACM SIGCOMM*, Seattle, WA, August 2008.

[47] N. Cai and R. Yeung, "Secure network coding," in *Information Theory, 2002. Proceedings. 2002 IEEE International Symposium on*, 2002, pp. 323–328.

[48] A. Jiang, "Network coding for joint storage and transmission with minimum cost," in *Information Theory, 2006 IEEE International Symposium on*, 2006, pp. 1359–1363.

[49] C. Fragouli, J. Widmer, and J.-Y. Le Boudec, "Efficient broadcasting using network coding," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 2, pp. 450–463, 2008.

[50] D. Lucani, F. Fitzek, M. Medard, and M. Stojanovic, "Network coding for data dissemination: it is not what you know, but what your neighbors don't know," in *Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks, 2009. WiOPT 2009. 7th International Symposium on*, 2009, pp. 1–8.

[51] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 3, pp. 497–510, 2008.

[52] M. Pedersen and F. Fitzek, "Implementation and performance evaluation of network coding for cooperative mobile devices," in *Communications Workshops, 2008. ICC Workshops '08. IEEE International Conference on*, 2008, pp. 91–96.

[53] H. Shojania and B. Li, "Random network coding on the iphone: fact or fiction?" in *Proceedings of the 18th international workshop on Network and operating systems support for digital audio and video*, ser. NOSSDAV '09. New York, NY, USA: ACM, 2009, pp. 37–42.

[54] R. Bassoli, H. Marques, J. Rodriguez, K. W. Shum, and R. Tafazolli, "Network coding theory: A survey," *IEEE Communications Surveys Tutorials*, vol. 15, no. 4, pp. 1950–1978, 2013.

[55] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, and L. M. G. M. Tolhuizen, "Polynomial time algorithms for multicast network code construction," *IEEE Transactions on Information Theory*, vol. 51, no. 6, pp. 1973–1982, June 2005.

[56] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *Information Theory, IEEE Transactions on*, vol. 52, no. 10, pp. 4413–4430, 2006.

[57] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: physical-layer network coding," in *Proceedings of the 12th annual international conference on Mobile computing and networking*, ser. MobiCom '06. New York, NY, USA: ACM, 2006, pp. 358–365.

[58] T. Koike-Akino, P. Popovski, and V. Tarokh, "Optimized constellations for two-way wireless relaying with physical network coding," *Selected Areas in Communications, IEEE Journal on*, vol. 27, no. 5, pp. 773–787, 2009.

[59] H. Yang, W. Meng, B. Li, and G. Wang, "Physical layer implementation of network coding in two-way relay networks," in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 671–675.

[60] S. Katti, S. Gollakota, and D. Katabi, "Embracing wireless interference: analog network coding," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 397–408, Aug. 2007.

[61] Z.-Q. Luo, "Universal decentralized estimation in a bandwidth constrained sensor network," *Information Theory, IEEE Transactions on*, vol. 51, no. 6, pp. 2210–2219, 2005.

[62] M.-K. Oh, X. Ma, G. Giannakis, and D.-J. Park, "Cooperative synchronization and channel estimation in wireless sensor networks," *Communications and Networks, Journal of*, vol. 7, no. 3, pp. 284–293, 2005.

[63] S. Ramanan and J. Walsh, "Distributed estimation of channel gains in wireless sensor networks," *Signal Processing, IEEE Transactions on*, vol. 58, no. 6, pp. 3097–3107, 2010.

[64] S. C. Liew, S. Zhang, and L. Lu, "Physical-layer network coding: Tutorial, survey, and beyond," *Physical Communication*, vol. 6, pp. 4 – 42, 2013, network Coding and its Applications to Wireless Communications.

[65] S. Sahin and U. Aygolu, "Amplify-and-forward strategy with limited feedback in two-way relay channels," in *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*, 2011, pp. 1–5.

[66] K. Yasami, A. Razi, and A. Abedi, "Analysis of channel estimation error in physical layer network coding," *Communications Letters, IEEE*, vol. 15, no. 10, pp. 1029 –1031, Oct. 2011.

[67] S. Zhang, S.-C. Liew, and P. Lam, "On the synchronization of physical-layer network coding," in *Information Theory Workshop, 2006. ITW '06 Punta del Este. IEEE*, 2006, pp. 404–408.

[68] T. Uricar, "Wireless (physical layer) network coding design for parametric channels and systems with partial hierarchical side-information," Ph.D. dissertation, Czech Technical University in Prague, 2014.

[69] J. Laneman, D. Tse, and G. W. Wornell, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *Information Theory, IEEE Transactions on*, vol. 50, no. 12, pp. 3062–3080, 2004.

[70] G. Kramer, M. Gastpar, and P. Gupta, "Cooperative strategies and capacity theorems for relay networks," *Information Theory, IEEE Transactions on*, vol. 51, no. 9, pp. 3037–3063, 2005.

[71] T. Cover and A. Gamal, "Capacity theorems for the relay channel," *Information Theory, IEEE Transactions on*, vol. 25, no. 5, pp. 572–584, 1979.

[72] M. Khojastepour, A. Sabharwal, and B. Aazhang, “Lower bounds on the capacity of gaussian relay channel,” in *Proc. 38th Annu. Conf. Information Sciences and Systems (CISS)*, 2004, pp. 597–602.

[73] B. Rankov and A. Wittneben, “Achievable rate regions for the two-way relay channel,” in *Information Theory, 2006 IEEE International Symposium on*, 2006, pp. 1668–1672.

[74] P. Popovski and H. Yomo, “Physical network coding in two-way wireless relay channels,” in *Communications, 2007. ICC ’07. IEEE International Conference on*, 2007, pp. 707–712.

[75] J. Sykora and A. Burr, “Network coded modulation with partial side-information and hierarchical decode and forward relay sharing in multi-source wireless network,” in *Wireless Conference (EW), 2010 European*, Apr. 2010, pp. 639 –645.

[76] ——, “Layered design of hierarchical exclusive codebook and its capacity regions for HDF strategy in parametric wireless 2-WRC,” *Vehicular Technology, IEEE Transactions on*, vol. 60, no. 7, pp. 3241 –3252, Sep. 2011.

[77] T. Koike-Akino, P. Popovski, and V. Tarokh, “Denoising maps and constellations for wireless network coding in two-way relaying systems,” in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, 2008, pp. 1–5.

[78] B. Nazer and M. Gastpar, “Compute-and-forward: Harnessing interference through structured codes,” *Information Theory, IEEE Transactions on*, vol. 57, no. 10, pp. 6463 –6486, Oct. 2011.

[79] R. Zamir, “Lattices are everywhere,” in *Information Theory and Applications Workshop, 2009*, 2009, pp. 392–421.

[80] B. Hassibi and H. Vikalo, “On the sphere-decoding algorithm i. expected complexity,” *IEEE Transactions on Signal Processing*, vol. 53, no. 8, pp. 2806–2818, Aug 2005.

[81] N. Sommer, M. Feder, and O. Shalvi, “Low density lattice codes,” in *2006 IEEE International Symposium on Information Theory*, July 2006, pp. 88–92.

[82] Y. Yona and M. Feder, “Efficient decoding of low density lattice codes,” in *Electrical and Electronics Engineers in Israel, 2008. IEEEI 2008. IEEE 25th Convention of*, Dec 2008, pp. 484–488.

[83] J. Sykora and A. Burr, “Wireless network coding: The network aware PHY layer,” in *Proc. Int. Symp. of Wireless Communication Systems (ISWCS)*, York, United Kingdom, Sep. 2010, tutorial.

[84] C. E. Shannon, “A mathematical theory of communication,” *Bell system technical journal*, vol. 27, 1948.

[85] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[86] M. Salehi and J. Proakis, *Digital Communications*. McGraw-Hill Education, 2007.

[87] T. Hynek and J. Sykora, “Non-cooperative broadcast game for distributed decision map selection of relay wireless network coding processing,” in *Signal Processing Advances in Wireless Communications (SPAWC), 2013 IEEE 14th International Workshop on*, Jun. 2013.

[88] T. Hynek, D. Halls, and J. Sykora, “Hardware implementation of distributed learning algorithm for mapping selection for wireless physical layer network coding,” in *Communications (ICC), 2015 IEEE International Conference on*, June 2015, pp. 1–6.

[89] A. Burr and J. Sykora, “Cooperative wireless network coding for uplink transmission on hierarchical wireless networks,” in *Int. Symp. on Signals, Systems and Electronics*, Oct. 2012, pp. 1 – 5.

[90] J. Price and T. Javidi, “Network coding games with unicast flows,” *Selected Areas in Communications, IEEE Journal on*, vol. 26, no. 7, pp. 1302 –1316, Sep. 2008.

[91] X.-B. Liang, “Matrix games in the multicast networks: maximum information flows with network switching,” *Information Theory, IEEE Transactions on*, vol. 52, no. 6, pp. 2433 –2466, Jun. 2006.

[92] J. Marden and M. Effros, “A game theoretic approach to network coding,” in *Networking and Information Theory, 2009. ITW 2009. IEEE Information Theory Workshop on*, Jun. 2009, pp. 147 –151.

[93] C. Yang, *Game Theory Framework Applied to Wireless Communication Networks*, ser. Advances in Wireless Technologies and Telecommunication:. IGI Global, 2015.

[94] J. Zhou, T. H. Chuang, T. Dinc, and H. Krishnaswamy, “19.1 receiver with 20mhz bandwidth self-interference cancellation suitable for fdd, co-existence and full-duplex applications,” in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[95] J. Nash, “Non-cooperative games,” *The Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951.

[96] T. Hynek, “Game-theoretical approach to self-adaptation of wireless cloud networks,” in *18th International Student Conference on Electrical Engineering; Proceedings of*, May 2014, pp. 1–6.

[97] D. Monderer and L. Shapley, "Potential games," *Games and Economic Behavior*, vol. 14, pp. 124–143, 1996.

[98] Z. Han, D. Niyato, W. Saad, T. Baar, and A. Hjrungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*, 1st ed. New York, NY, USA: Cambridge University Press, 2012.

[99] A. B. MacKenzie and L. A. DaSilva, "Game theory for wireless engineers," *Synthesis Lectures on Communications*, vol. 1, no. 1, pp. 1–86, 2006.

[100] C. Fragouli, "Network coding: Beyond throughput benefits," *Proceedings of the IEEE*, vol. 99, no. 3, pp. 461–475, 2011.

[101] Ettus Research. N200/N210 product overview. [Online]. Available: https://www.ettus.com/content/files/07495_Ettus_N200-210_DS_Flyer_HR_1.pdf

[102] ——. OctoClock-G product overview. [Online]. Available: https://www.ettus.com/product/details/OctoClock-G

[103] J. Postel, "User Datagram Protocol," Internet Requests for Comments, RFC Editor, RFC 768, August 1980. [Online]. Available: https://www.rfc-editor.org/info/rfc768

[104] D. Chu, "Polyphase codes with good periodic correlation properties (corresp.)," *Information Theory, IEEE Transactions on*, vol. 18, no. 4, pp. 531–532, Jul 1972.

[105] T. Koike-Akino, P. Popovski, and V. Tarokh, "Optimized constellations for two-way wireless relaying with physical network coding," *Selected Areas in Communications, IEEE Journal on*, vol. 27, no. 5, pp. 773 –787, Jun. 2009.

[106] T. Hynek and J. Sykora, "Wireless physical layer network coding in potential presence of malicious relays - incomplete information game approach," *Electronics Letters*, vol. 51, no. 16, pp. 1292–1294, 2015.

[107] ——, "Sequential incomplete information game in relay networks based on wireless physical layer network coding," in *Multiple Access Communications: 8th International Workshop, MACOM 2015, Helsinki, Finland*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2015, pp. 47–56.

[108] M. A. Alvarez, W. Thompson, and U. Spagnolini, "Distributed time and frequency synchronization: Usrp hardware implementation," in *2015 IEEE International Conference on Communication Workshop (ICCW)*, June 2015, pp. 2157–2162.

[109] T. Schmidl and D. Cox, "Robust frequency and timing synchronization for OFDM," *Communications, IEEE Transactions on*, vol. 45, no. 12, pp. 1613–1621, 1997.

[110] D. Liu and J.-M. Chung, "Enhanced OFDM time and frequency synchronization through optimal code correlation," in *Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on*, vol. 1, 2002, pp. I–176–9 vol.1.

[111] T. Pollet, M. Van Bladel, and M. Moeneclaey, "BER sensitivity of OFDM systems to carrier frequency offset and Wiener phase noise," *Communications, IEEE Transactions on*, vol. 43, no. 234, pp. 191–193, 1995.

[112] C. Oberli and B. Daneshrad, "Maximum likelihood tracking algorithms for MIMO-OFDM," in *Communications, 2004 IEEE International Conference on*, vol. 4, 2004, pp. 2468–2472 Vol.4.

[113] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, Jul. 1970.

[114] B. Fan, D. G. Andersen, M. Kaminsky, and M. D. Mitzenmacher, "Cuckoo filter: Practically better than bloom," in *Proceedings of the 10th ACM International on Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '14. New York, NY, USA: ACM, 2014, pp. 75–88.

[115] L. Lu, T. Wang, S. C. Liew, and S. Zhang, "Implementation of physical-layer network coding," in *Communications (ICC), 2012 IEEE International Conference on*, June 2012, pp. 4734–4740.

[116] L. Lu, L. You, Q. Yang, T. Wang, M. Zhang, S. Zhang, and S. C. Liew, "Real-time implementation of physical-layer network coding," in *Proceedings of the Second Workshop on Software Radio Implementation Forum*, ser. SRIF '13. New York, NY, USA: ACM, 2013, pp. 71–76.

[117] M. H. Firooz, Z. Chen, S. Roy, and H. Liu, "Wireless network coding via modified 802.11 mac/phy: Design and implementation on sdr," *IEEE Journal on Selected Areas in Communications*, vol. 31, no. 8, pp. 1618–1628, August 2013.

[118] M. Wu, F. Ludwig, M. Woltering, D. Wuebben, A. Dekorsy, and S. Paul, "Analysis and implementation for physical-layer network coding with carrier frequency offset," in *Smart Antennas (WSA), 2014 18th International ITG Workshop on*, March 2014, pp. 1–8.

[119] T. Wang, S. C. Liew, and L. You, "Joint phase tracking and channel decoding for ofdm pnc: Algorithm and experimental evaluation," in *Proceedings of the 2014 ACM Workshop on Software Radio Implementation Forum*, ser. SRIF '14. New York, NY, USA: ACM, 2014, pp. 69–76.

[120] A. C. Marcum, J. V. Krogmeier, D. J. Love, and A. Sprintson, “Analysis and implementation of asynchronous physical layer network coding,” *IEEE Transactions on Wireless Communications*, vol. 14, no. 12, pp. 6595–6607, Dec 2015.

[121] W. Wang, M. Chatterjee, and K. Kwiat, “Coexistence with malicious nodes: A game theoretic approach,” in *Game Theory for Networks, 2009. GameNets ’09. International Conference on*, May 2009, pp. 277–286.

[122] S. Dehnie, H. Sencar, and N. Memon, “Detecting malicious behavior in cooperative diversity,” in *Information Sciences and Systems, 2007. CISS ’07. 41st Annual Conference on*, March 2007, pp. 895–899.

[123] M.-H. Chen, S.-C. Lin, and Y.-W. Hong, “A game theoretic approach for the cooperative network with the presence of malicious relays,” in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–5.

[124] J. C. Harsanyi, “Games with incomplete information played by "Bayesian" players, I-III part I. The Basic Model,” *Management Science*, vol. 14, no. 3, pp. 159–182, 1967.

[125] D. Fudenberg and J. Tirole, *Game Theory*. MIT Press, 1991.

[126] T. Hynek and J. Sykora, “Initialization procedure of wireless network coding with hierarchical decode and forward strategy in random connectivity networks,” in *Multiple Access Communications*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 13–24.

[127] T. Hynek, D. Halls, and J. Sykora, “Practical implementation of cloud initialization procedure for wireless physical layer network coding clouds,” in *European Wireless 2014; 20th European Wireless Conference; Proceedings of*, May 2014, pp. 1–6.

[128] O. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, “Blind modulation classification: a concept whose time has come,” in *Advances in Wired and Wireless Communication, 2005 IEEE/Sarnoff Symposium on*, April 2005, pp. 223–228.

[129] ——, “Survey of automatic modulation classification techniques: classical approaches and new trends,” *Communications, IET*, vol. 1, no. 2, pp. 137–156, 2007.

[130] O. Duval, A. Punchihewa, F. Gagnon, C. Despins, and V. Bhargava, “Blind multi-sources detection and localization for cognitive radio,” in *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, Nov 2008, pp. 1–5.

[131] Y. Liang and F. Yong, "Blind estimation of source number based on svm in wireless communication systems," in *Wireless Mobile and Computing (CCWMC 2009), IET International Communication Conference on*, Dec 2009, pp. 109–112.

[132] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.

[133] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[134] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal statistical society, Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[135] S. Haykin, *Neural Networks and Learning Machines*, ser. Neural networks and learning machines. Prentice Hall, 2009.

[136] M. M. Roganovic, A. M. Neskovic, and N. J. Neskovic, "Application of artificial neural networks in classification of digital modulations for software defined radio," in *EUROCON 2009, EUROCON '09. IEEE*, May 2009, pp. 1700–1706.

[137] M. Zamanian, A. A. Tadaion, and M. T. Sadeghi, "Modulation classification of linearly modulated signals in a cognitive radio network using constellation shape," in *Systems, Signal Processing and their Applications (WOSSPA), 2011 7th International Workshop on*, May 2011, pp. 13–16.

[138] M. S. Muhlhaus, M. Oner, O. A. Dobre, and F. K. Jondral, "A low complexity modulation classification algorithm for mimo systems," *IEEE Communications Letters*, vol. 17, no. 10, pp. 1881–1884, October 2013.

[139] L. Haring, Y. Chen, and A. Czylwik, "Automatic modulation classification methods for wireless ofdm systems in tdd mode," *IEEE Transactions on Communications*, vol. 58, no. 9, pp. 2480–2485, September 2010.

[140] W. B. Chikha, I. Dayoub, W. Hamouda, and R. Attia, "Modulation recognition for mimo relaying broadcast channels with direct link," *IEEE Wireless Communications Letters*, vol. 3, no. 1, pp. 50–53, February 2014.

[141] C. Cardoso, A. R. Castro, and A. Klautau, "An efficient fpga ip core for automatic modulation classification," *IEEE Embedded Systems Letters*, vol. 5, no. 3, pp. 42–45, Sept 2013.

[142] E. W. Weisstein. Inclusion-exclusion principle. From MathWorld—A Wolfram Web Resource. [Online]. Available: http://mathworld.wolfram.com/Inclusion-ExclusionPrinciple.html

[143] J. Sykora and A. Burr, "Network coded modulation for random channel class in wnc with hdf relaying strategy," *IEEE Communications Letters*, vol. 17, no. 5, pp. 818–821, May 2013.