THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par : *l'Institut Supérieur de l'Aéronautique et de l'Espace (ISAE)*

Présentée et soutenue le *25/11/2016* **par :**

Karine ZIDANE

Improving Synchronous Random Access Schemes for Satellite Communications

Techniques d'amélioration des performances des méthodes d'accès aléatoire synchrones pour les communications par satellite

JURY

Riccardo DE GAUDENZI	Directeur de Recherche ESA	Rapporteur
Catherine DOUILLARD	Professeur	Rapporteur
Giuseppe COCCO	Docteur	Examinateur
Laurent FRANCK	Professeur	Examinateur
Marie-Laure BOUCHERET	Professeur	Examinateur
Jérôme LACAN	Professeur	Directeur de Thèse
Mathieu GINESTE	Docteur	Membre du jury
Caroline BES	Docteur	Membre du jury

École doctorale et spécialité :

MITT : Domaine STIC : Réseaux, Télécoms, Systèmes et Architecture

Unité de Recherche :

Equipe d'accueil ISAE-ONERA MOIS

Directeur de Thèse :

Jérôme LACAN

Rapporteurs :

Riccardo DE GAUDENZI et *Catherine DOUILLARD*

“Trust in dreams, for in them
is the hidden gate to eternity.”

— Kahlil Gibran

To the loving memory of my Grandfather. . .

ACKNOWLEDGEMENTS

> "Il y a des gens formidables qu'on rencontre au mauvais moment. Et il y a des gens qui sont formidables parce qu'on les rencontre au bon moment."
>
> — David Foenkinos

First and foremost, I would like to thank my advisor, Jérôme Lacan, for his support, his encouragements and his optimism. Thank you Jérôme for the motivating discussions we had, for the brilliant ideas and the positiveness you transmitted to me during the most difficult times. It has been a great pleasure to work with you since my Master's internship until the end of my thesis.

I gratefully acknowledge the funding received towards my PhD from CNES and Thales Alenia Space (TAS, Toulouse). Thanks to my advisor at TAS, Mathieu Gineste, for all his time and his suggestions. And especially, for his helpful remarks regarding the practical implementation of my research, and the system assumptions. Another thank you to Caroline Bes, my advisor at CNES, for her constant support and motivation.

My thesis committee also had a great role in making this work more valuable. In particular, I would like to thank my thesis referees, Riccardo de Gaudenzi and Catherine Douillard for the time they allocated to read my manuscript and for their precious reviews and comments. Also, thank you to Giuseppe Cocco and Laurent Franck for being part of my thesis committee and for their constructive remarks during the defense. I would like to thank the president of this committee Marie-Laure Boucheret, with whom I had many helpful exchanges especially during the first year of my thesis.

During the last years, I spent some months working at the telecommunications department in TAS, with a great team of colleagues: Cedric, Fabrice, Mathieu D., Thibault and Renaud. I would like to thank them a lot for the interesting discussions we shared, and for frequently hanging out together in Toulouse.

I dedicate this part to thank all my colleagues and friends who helped and supported me during all these years in Toulouse. First, in the DISC research department at ISAE-SUPAERO, I say thank you to the professors Manu, Fabrice and Ahlem for the interesting stories we exchanged at lunch hours, back when we were at ENSICA until we moved to ISAE. And thank

you Manu for your amazing Jazz/Rock concerts that I always enjoyed! Many thanks to the professors Tanguy and Stéphanie for giving me the chance to have a great teaching experience at ISAE. A special acknowledgment goes to my dear colleagues Rami and Victor for the great fun moments we shared, and for the big support they gave me until the end. Also, many thanks to all my colleagues at ISAE: John, Bastien, Henrique, Anais, Ahmed, Antoine, Fred, Gwilherm, and Tuan.

I would like to thank all my colleagues and friends at TéSA labs. A big thank you to my current office mates Raoul, Selma and Sylvain, especially for their great support before my thesis defense. Having you as co-workers makes a day at work much happier. Thank you as well to the director of TéSA, Corinne Mailhes, for her cheerful encouragements.

And as Toulouse is one of the best cities in the world to meet amazing people, I would like to thank all my dearest friends whom I met during the last three years. First to my best friend Angela, one of the most amazing persons I have ever met, thank you! Another thank you to my close friends, Hussein, Eva, Diana, Tannous, Vicente, Rossi, Nicolas, Hugo, Chi, Hannes, Emil, and Rafa. Thanks as well to the lovely lebanese families I met in Toulouse, who treated me as their real daughter.

Of course, I do not forget my childhood friends who supported me, even across the distances. To my best friend Dima, a big thank you for your constant encouragement and cheerful wishes. Maria, Yasmine, Laura, and Ranime, thank you a lot for always cheering me up.

I finish with my family, the symbol of love, hope, ambition and energy. Thanking you will never be enough, and without you I would have never been in this position. In each step of my life, you helped me and respected my decisions. You have been an endless source of giving without seeking anything in return. To the most amazing family, my beautiful mom Marwa, my wise dad Abed, my lovely sister Perla and my kind brother Hazem, thank you and may God bless you always. Many thanks to the rest of my family who I love, Maha, Ghassan, Marwan, Amal, Jamal and my Grandmother. A final enormous thank you to my inspiring Grandfather, I will be ever grateful for his encouragements, and I am sorry that he has not lived to see me graduate.

Toulouse, 13 December 2016 K. Z.

ABSTRACT

With the need to provide the Internet access to deprived areas and to cope with constantly enlarging satellite networks, enhancing satellite communications becomes a crucial challenge. In this context, the use of Random Access (RA) techniques combined with dedicated access on the satellite return link, can improve the system performance. However conventional RA techniques like Aloha and Slotted Aloha suffer from a high packet loss rate caused by destructive packet collisions. For this reason, those techniques are not well-suited for data transmission in satellite communications. Therefore, researchers have been studying and proposing new RA techniques that can cope with packet collisions and decrease the packet loss ratio. In particular, recent RA techniques involving information redundancy and successive interference cancellation, have shown some promising performance gains.

With such methods that can function in high load regimes and resolve packets with high collisions, channel estimation is not an evident task. As a first contribution in this dissertation, we describe an improved channel estimation scheme for packets in collision in new RA methods in satellite communications. And we analyse the impact of residual channel estimation errors on the performance of interference cancellation. The results obtained show a performance degradation compared to the perfect channel knowledge case, but provide a performance enhancement compared to existing channel estimation algorithms.

Another contribution of this thesis is presenting a method called Multi-Replica Decoding using Correlation based Localisation (MARSALA). MARSALA is a new decoding technique for a recent synchronous RA method called Contention Resolution Diversity Slotted Aloha (CRDSA). Based on packets replication and successive interference cancellation, CRDSA enables to significantly enhance the performance of legacy RA techniques. However, if CRDSA is unable to resolve additional packets due to high levels of collision, MARSALA is applied. At the receiver side, MARSALA takes advantage of correlation procedures to localise the replicas of a given packet, then combines the replicas in order to obtain a better Signal to Noise plus Interference Ratio. Nevertheless, the performance of MARSALA is highly dependent on replicas synchronisation in timing and phase, otherwise replicas combination would not be constructive. In this dissertation, we describe an overall framework of MARSALA including replicas timing and phase estimation and compensation, then channel estimation for the resulting signal. This dissertation also provides an analytical model for the performance degradation of MARSALA due to imperfect replicas combination and channel estimation. In addition, several enhancement schemes for MARSALA are proposed like Maximum Ratio

Combining, packets power unbalance, and various coding schemes. Finally, we show that by choosing the optimal design configuration for MARSALA, the performance gain can be significantly enhanced.

Key words: Satellite communications, random access techniques, media access protocols, channel estimation, successive interference cancellation, maximum ratio combining, DVBRCS2, synchronisation

RÉSUMÉ

L'optimisation des communications par satellite devient un enjeu crucial pour fournir un accès Internet aux zones blanches et/ou défavorisées et pour supporter des réseaux à grande échelle. Dans ce contexte, l'utilisation des techniques d'accès aléatoires sur le lien retour permet d'améliorer les performances de ces systèmes. Cependant, les techniques d'accès aléatoire classiques comme 'Aloha' et 'Slotted Aloha' ne sont pas optimales pour la transmission de données sur le lien retour. En effet, ces techniques présentent un taux élevé de pertes de paquets suite aux collisions. Par conséquent, des études récentes ont proposé de nouvelles méthodes d'accès aléatoire pour résoudre les collisions entre les paquets et ainsi, améliorer les performances. En particulier, ces méthodes se basent sur la redondance de l'information et l'annulation successive des interférences.

Dans ces systèmes, l'estimation de canal sur le lien retour est un problème difficile en raison du haut niveau de collisions de paquets. Dans une première contribution dans cette thèse, nous décrivons une technique améliorée d'estimation de canal pour les paquets en collision. Par ailleurs, nous analysons l'impact des erreurs résiduelles d'estimation de canal sur la performance des annulations successives des interférences. Même si les résultats obtenus sont encore légèrement inférieurs au cas de connaissance parfaite du canal, on observe une amélioration significative des performances par rapport aux algorithmes d'estimation de canal existants.

Une autre contribution de cette thèse présente une méthode appelée 'Multi-Replica Decoding using Correlation based Localisation' (MARSALA). Celle-ci est une nouvelle technique de décodage pour la méthode d'accès aléatoire synchrone 'Contention Résolution diversité Slotted Aloha' (CRDSA), qui est basée sur les principe de réplication de paquets et d'annulation successive des interférences. Comparée aux méthodes d'accès aléatoire traditionnelles, CRDSA permet d'améliorer considérablement les performances. Toutefois, le débit offert par CRDSA peut être limité à cause des fortes collisions de paquets. L'utilisation de MARSALA par le récepteur permet d'améliorer les résultats en appliquant des techniques de corrélation temporelles pour localiser et combiner les répliques d'un paquet donné. Cette procédure aboutit à des gains en termes de débit et de taux d'erreurs paquets. Néanmoins, le gain offert par MARSALA est fortement dépendant de la synchronisation en temps et en phase des répliques d'un même paquet. Dans cette thèse, nous détaillons le fonctionnement de MARSALA afin de corriger la désynchronisation en temps et en phase entre les répliques. De plus, nous évaluons l'impact de la combinaison imparfaite des répliques sur les performances, en fournissant un modèle analytique ainsi que des résultats de simulation. En outre, plusieurs schémas d'optimisation de MARSALA sont proposés tels que le principe du 'Maximum Ratio Combining', ou la transmission des paquets à des puissances différentes. Utilisées conjointement, ces différentes propositions permettent d'obtenir une amélioration très significative des performances. Enfin, nous montrons qu'en choisissant la configuration optimale pour MARSALA, le gain de performance est considérablement amélioré.

Mots clefs : Communications par satellite, méthodes d'accès aléatoires, protocoles d'accès multiple, estimation de canal, suppression successive des interférences, DVBRCS2, synchronisation

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

Introduction

1.1 Motivation & goals

In today's world, the internet network has widely expanded its services and has become extremely important in many fields: from primordial aspects such as education and healthcare, to economic and social aspects such as business, government, media and social communications. Despite the huge expansion of the internet, 4.4 billion people around the world still did not have internet access in 2013, according to *The Washington Post* [2]. That is almost three times the population of China and more than half of the world's population. Internet access is not only deprived in third world countries suffering from poverty, war or economic crisis; but also in many rural settings with poor infrastructure and low population densities. In such places, internet service providers simply do not have enough economic incentives to deploy infrastructures that provide internet access.

In the context of providing internet coverage for the deprived areas without deploying complex terrestrial infrastructure, big internet companies such as *Google* and *Facebook* have already invested in satellite networks in order to win the chance to sell high-speed and cheap satellite internet, worldwide (O3b, OneWeb). *Google*, for instance, said to be willing to invest 1 billion dollars in satellites to spread internet access [3]. *Facebook* CEO, Mark Zuckerberg also showed the same interest through his project initiative *Internet.org*. The real big challenge in such projects is the ability to ensure the characteristics of 'high-speed' and 'low-cost' services in satellite internet access. As a matter of fact, Geostationary (GEO), Low-Earth Orbit (LEO) and Medium-Earth Orbit (MEO) satellites can provide world coverage. However, compared to terrestrial networks, they suffer from higher signal propagation delays (around 500 milliseconds for GEO and 20 milliseconds for LEO) and system delays (access and cross-link delays). Indeed, latency in satellite networks matter and can significantly affect on the user experience, even if the communication is tolerant to such latencies (web browsing, mail delivery, file sharing). In fact, several studies [4] showed that higher page load time lead to a higher rate of website abandonment. Besides, a study in [5] proved that if an e-commerce site is making 100000 dollars per day, each 1 second page delay could potentially cost 2.5 million dollars over a one year period. Therefore, regarding satellite communications, researchers and industries have been trying to minimise the network delays whether in GEO, LEO or MEO satellite systems. In order to organise the communications in such networks, research and industrial entities have defined many standards for satellite networks such as the Digital Video Broadcasting Standards for the forward link (DVB-S [6], DVB-S2 [7], DVB-S2X [8]) and the return link (DVB-RCS [9], DVB-RCS2 [10]), as well as the S-Band Mobile Interactive Multimedia (S-MIM [11]) standard and the Consultative Committee for Space Data Systems (CCSDS [12]) and many others. In the context of this thesis, we will only consider GEO satellite systems.

1.1.1 First problem statement

Not only reducing communications latencies is the main challenge in satellite networks, but also the ability to cope with the huge and rapid growth of such networks and the constantly increasing number of user terminals. Services like the satellite internet and the satellite phone are attracting more and more users. Other applications such as the Internet of Things (IoT), Machine-to-Machine (M2M) and SCADA (Supervisory Control and Data Acquisition) are also arising and implying a wider number of terminals on the network. According to analysts and engineers in *Northern Sky Research*[1] and *Thuraya*[2], the IoT would not survive without satellite communications. Everything from vehicles, body sensors, temperature sensors, industrial equipments, household appliances, electricity and gas meters, weather stations and many others will be connected. Thus, the number of devices connected will grow in an impressive way. Researchers expect that by 2020, this number will increase up to 26 million units, which represents 30 times the number of connected devices in 2009. In such networks, satellite technologies play a major role in providing access to the newly deployed services across industries and geographical borders. To summarise, the main problems in satellite networks are the need to cope with the constantly enlarging networks and to provide high throughput, low latencies and reliable communications for all users. Motivated by these problems, in this dissertation, we are interested in proposing solutions through a well-investigated subject in the domain of satellite communications: Random Access (RA) methods.

Why RA methods are a solution?

As a matter of fact, in GEO satellite networks, the minimum latency for a data packet to be transmitted by a user and for a response to be received by this same user, is 500 milliseconds. This end-to-end signal propagation delay is also referred to as the Round Trip Time (RTT).

[1] the Northern Sky Research (NSR) is an international market research and consulting firm specialising in telecommunications technology.

[2] Thuraya is an international mobile satellite phone provider based in the United Arab Emirates.

Additional delays could be encountered such as access control, retransmission and processing delays. Also, latency is added when many round trips are needed to establish connections and resource allocations before the actual data transmission. As detailed in Chapter 2, in order to organise such communications shared between multiple users, Media Access Control (MAC) protocols are required on the return link (i.e. the link from the user terminals to the satellite or the gateway). The users can access the network using Demand Assignment Multiple Access (DAMA) techniques or RA techniques. In DAMA, resource allocation requests are required prior to data transmission, and each user is assigned one or several frequencies and timeslots on which it can transmit its data. On the contrary, in RA techniques, the users can access the shared media at randomly chosen time and frequency, thus reducing communication latencies but increasing the risk of packet collisions. In addition, RA methods are well suited for sporadic and bursty internet traffic profiles with long silent periods or short data packets, such as the HTTP traffic. For this kind of scenarios, using DAMA techniques alone has proved to be inefficient and under-utilising for the satellite resources [13]. Therefore, using RA techniques combined with DAMA on the satellite return link presents a promising solution to such problems and motivates for a lot of research in this field. A detailed list of the legacy and recent RA methods will be given in Chapter 2.

1.1.2 Second problem statement

The problem with RA is in its uncoordinated nature, given that packet collisions can occur on the communication channel for users who have chosen to transmit at the same time. In legacy RA methods such as ALOHA [14] and Slotted ALOHA (SA) [15], packet collisions were often destructive. Given that the collisions rate on a communication channel is relatively high, then many packets are lost and delays are increased due to packets retransmissions. For this reason, researchers have proposed several enhanced RA techniques in the literature. Those techniques can be synchronous, also called slotted (i.e. the communication time axis is divided into timeslots, and each user can transmit packets only at the beginning of one timeslot) or asynchronous, also called unslotted (i.e. the time axis can be accessed by any user at any instant and partial collisions between packets can occur). The advantage of using asynchronous techniques over the synchronous ones, is that they do not require exchanges of control packets to ensure synchronisation between users at the timeslot or frame [3] level. However, what motivates the use of the synchronous techniques is the fact that they are more practical in terms of the detection of packets on a frame. Therefore, in this dissertation, we will focus on a recent synchronous RA technique for satellite communications [16] and ways to enhance its performance.

Among legacy RA methods we can cite the widely spread ALOHA technique and its slotted version SA. Another version of SA is Diversity Slotted ALOHA (DSA) [17], in which each user sends several copies (also called replicas) of the same packet on different timeslots of the frame. DSA permits to increase the probability of receiving at least one replica without collisions. Still

[3] A frame is a set of timeslots organised over one frequency band.

the performance of ALOHA, SA and DSA on the satellite return link is very poor in normal to high load regimes and induces a high level of packet losses. For this reason, De Gaudenzi et al. published a RA method in 2007 called Contention Resolution Diversity Slotted Aloha (CRDSA) [16]. This method combines packet replication with the concept of Successive Interference Cancellation (SIC) at the receiver side. In CRDSA, each user transmits several copies of the same packet on different timeslots with each copy containing signalling information about the locations of its other copies on the frame. At the receiver side, if a packet is correctly received, it is removed from its timeslot and the decoded signalling field is used to localise its replicas on the frame and remove them iteratively. The frame is re-scanned several times in order to recover packets after interference cancellation. With CRDSA, the performance is significantly enhanced compared to DSA. For instance, for a target Packet Loss Ratio (PLR) equal to 10^{-2} and using two replicas per packet and a Forward Error Correction (FEC) code of rate 1/2, the performance of CRDSA in terms of successfully recovered packets on a frame is 5-fold higher than DSA. An irregular version of CRDSA was presented in [18], where the number of replicas per packet follows an optimised probability distribution.

Other recent synchronous RA protocols that also use the SIC principal, have been proposed in the literature. On the one hand, we cite Coded Slotted Aloha (CSA) [19, 20]. In CSA each terminal divides the packet into k fragments, then uses erasure coding (n,k) to create n fragments and transmit them on distinct timeslots of the frame. On the other hand, we cite Multi-Slot Coded Aloha (MuSCA) [21]. Instead of resorting to packets replication, MuSCA encodes each packet with a strong FEC code, then divides the codeword into several fragments. To each fragment, a coded signalling field is added for the purpose of fragments localisation. At the receiver side, first the signalling fields of all packets are decoded and the fragments of a same packet are localised. In a second step, the payload fragments are combined together in order to decode the corresponding codeword. Once a codeword is successfully decoded, the SIC process is applied similarly to CRDSA. MuSCA shows important performance gains compared to the existing synchronous RA methods. However, the limitation of MuSCA is the significant overhead resulting from the redundancy used to encode the signalling field.

Our goal

Our goal is to further enhance the RA performance without adding any signalling overhead. Although important work and significant performance gains have been achieved in the field of satellite synchronous RA, and particularly with CRDSA, there remains some open questions: *if a packet could not be decoded with CRDSA, can we find a way to localise this packet's replicas before decoding?* And if the replicas are localised, *can we recover the information they contain?* These are the questions that we will try to answer in Chapters 4, 5 and 6.

1.2 Contributions

The major contribution of this thesis is the proposition and the evaluation of a new technique called Multi-Replica Decoding using Correlation based Initialisation (MARSALA). This technique significantly enhances the RA performance, while using the same packet structure as in CRDSA (i.e. no additional signalling overhead). Other contributions of this thesis focus on resolving practical issues in RA such as channel estimation and replicas synchronisation. A detailed list of the overall contributions is given below.

1.2.1 Improved channel estimation for interference cancellation in RA methods for satellite communications

In order to perform accurate interference cancellation, accurate channel estimation must be done. In previous studies of MuSCA and other random access methods, the channel impact on the received packets has been considered perfectly known when interference cancellation is performed. For this reason, we propose a channel estimation technique for superimposed packets colliding on a timeslot. We describe an algorithm based on Expectation-Maximisation (EM)[22] combining auto-correlation initialisation and Pilot Symbol Assisted Modulation (PSAM) [23]. With the proposed algorithm, we jointly estimate the channel parameters of users in collision, such as channel attenuation, frequency offsets, timing offsets and phase shifts. Then, we evaluate the impact of residual channel estimation errors on the Packet Error Rate (PER) of the remaining packet after interference cancellation.

1.2.2 Joint estimation and decoding for RA methods in satellite communications

Our second contribution is the introduction of the concept of Joint Estimation and Decoding (JED) [24] to RA methods used in satellite communications. Of course, JED is well known in the literature, but evaluating its impact in terms of residual channel estimation errors after interference cancellation has not been studied before. In this dissertation, we combine the proposed EM based channel estimation technique with JED: we estimate the channel parameters in a first iteration. Then, we perform demodulation and decoding, and we use the decoded bits to re-estimate the channel parameters in the next iteration. This operation is repeated iteratively in order to enhance the performance of the channel estimation.

1.2.3 MARSALA RA scheme for satellite communications

We present a new decoding technique for CRDSA called MARSALA. This scheme enables replicas localisation based on signal correlation and replicas combination in order to boost the Signal to Noise plus Interference Ratio (SNIR) when CRDSA fails to correctly decode a packet. We propose to use MARSALA and CRDSA jointly. In other words, when a packet cannot be decoded by CRDSA due to strong collisions encountered by all the replicas, then MARSALA is applied. The first novelty aspect in MARSALA is its ability to localise the replicas of a given packet prior to any decoding operation and without adding signalling overhead. The second novelty aspect in MARSALA is replicas combining before decoding. A list of contributions related to the evaluation of MARSALA in real channel conditions is given below.

- **Estimation of timing offsets and phase shifts between packet replicas:** a major contribution added to MARSALA is describing a method for coherent replicas combination. In fact, replicas received on separate timeslots are affected by different timing offsets and phase shifts due to carrier frequency and phase variations. However, the attenuation is supposed to be constant on all the replicas given the limited frame duration. In order to ensure coherent replicas combination, we describe a method to estimate and compensate the synchronisation errors such as timing offsets and phase shifts between replicas, prior to their combination.
- **An analytical model for performance degradation of MARSALA in real channel conditions:** as stated previously, the correction of timing offsets and phase shifts between the replicas, based on estimated values, can result in imperfectly coherent replicas combination. In this dissertation, we contribute in defining an analytical model for the performance degradation caused by imperfect replicas combining in MARSALA.

1.2.4 Enhancement of MARSALA with Maximum Ratio Combining, coding schemes, and packets power unbalance

We propose techniques to enhance the performance of MARSALA on the RA channel. In particular, we apply the concept of Maximum Ratio Combining (MRC) to MARSALA and to exploit its performance with packets power unbalance and several coding schemes such as turbo codes of DVB-RCS2, 3GPP [25] and CCSDS. These contributions are listed below.

- **MARSALA with MRC:** MRC is widely known in the literature especially applied to Multiple-Input Multiple-Output (MIMO) and Multiple-Input Single-Output (MISO) systems. Given that MARSALA is also based on a diversity transmission technique, in our work, we propose to use MRC in the decoding procedure of MARSALA. Thus, each replica is multiplied by a coefficient proportional to its SNIR. Then, the weighted replicas are combined together before demodulation and decoding. We show via simulations, that the performance of MARSALA with MRC is enhanced compared to MARSALA scheme with equal gain combining.
- **MARSALA with packets power unbalance and coding schemes:** the impact of packets power unbalance has been studied for many existing RA methods but not for MARSALA. Motivated by the significant performance gain achieved when introducing packets power unbalance to existing RA methods, in this dissertation, we contribute in evaluating the same impact on MARSALA. First, we evaluate MARSALA with lognormal packets power distributions, in order to compare the performance to previous results in this field.

  In the same context, it has been shown in previous research [26] and recent work done in the European Space Agency (ESA), that with power control techniques applied at the transmitters side, an optimal packets power distribution for an optimal RA performance can be computed for each RA technique. In this dissertation, we analyse the impact of several Probability Density Functions (PDFs) for packets power that could be used with power control techniques, in MARSALA. Then, we propose a configuration for optimal performance.

In addition, the effect of using different coding schemes such as DVB-RCS2 and 3GPP has been evaluated for CRDSA [27, 28] but not for MARSALA. The authors have shown that using a 3GPP turbo code results in a better performance compared to DVB-RCS2. They have explained that this result is obtained because the 3GPP decoding performance in the PER region between $[0.9, 1]$ is better. However, the performance gains compared to using the DVB-RCS2 turbo code are not significant. Inspired from the same study, in this dissertation, we compare the performance of MARSALA with three different turbo coding schemes: DVB-RCS2, 3GPP and CCSDS. And we show that unlike CRDSA, the performance gain is significantly affected by the choice of the turbo-encoder.

1.3 Thesis organisation

The rest of this dissertation is organised as follows.

In Chapter 2, we provide the background necessary to understand MAC protocols, dedicated access and RA techniques. This background is also crucial to understand the importance of using RA methods on the satellite return link. In addition, we provide a state of the art related to the existing RA methods, and particularly the recent RA schemes used in satellite communications. A brief literature on channel estimation techniques is also given, in order to have a better understanding of the next chapter.

In Chapter 3, we explain how channel estimation errors can affect the performance of interference cancellation in recent RA methods. We describe an improved channel estimation method for packets in collision. Then, we evaluate the impact of the proposed channel estimation method on the PER via simulations.

In Chapter 4, we present the main steps of MARSALA RA scheme and we explain the system assumptions. We also give a numerical example and first simulation results of MARSALA with perfect Channel State Information (CSI).

In Chapter 5, we give the system hypothesis considered for MARSALA in real channel conditions. Then, taking into account the channel impairments, we detail every step from replicas localisation to replicas combining and demodulation and decoding. In particular, we describe an analytical model for the impact of synchronisation errors on replicas combining. We validate our analytical study through simulations in real channel conditions.

In Chapter 6, we present several enhancement schemes for MARSALA in real channel conditions, and we evaluate the performance results via simulations.

Finally, we conclude in Chapter 7 with a discussion of the overall results as well as the limitations of our contributions. We also open perspectives for future work.

2

Background & Related Work

The purpose of this chapter is to provide the reader with a general understanding of Random Access methods used in satellite communications. We start by providing a background on sharing a communication channel between multiple users, using Media Access Protocols. Next, we describe more particularly how these protocols are employed to access the satellite return link. In this context, we present two schemes: Dedicated Access and Random Access. Then, we describe in more details recent Random Access methods for satellite communications, and the techniques which they apply to achieve a more reliable and efficient communication. We also highlight on prior work related to channel estimation in recent Random Access methods and we describe its relevance to this thesis.

2.1 Sharing a channel: Media Access Protocols

In many communication systems (wired or wireless), several user nodes tend to access the same channel and communicate with the same destination node (intermediate or final). Therefore, certain protocols are required to organise how such a common communication channel can be shared [29]. Given their role, these protocols are termed Media Access (MAC) protocols. In this section we describe how MAC protocols can be used for time sharing, frequency sharing, as well as code sharing. We also highlight the advantages of combining both time and frequency sharing techniques, particularly for satellite networks.

2.1.1 Time Division Multiple Access

One solution to organise transmissions on a shared channel, is to use the same frequency band for all users, but to allocate a portion of time for each user. This MAC protocol based on time-axis sharing between multiple nodes is called Time Division Multiple Access (TDMA) [30, 31]. The time axis is divided into several timeslots of a fixed duration, and each user can access a dedicated timeslot periodically as shown in Figure 2.1.

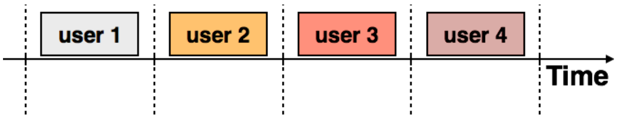


Figure 2.1 – A TDMA example: Time axis shared between 4 users.

Each user node can transmit its data packet within one timeslot. Thus, the collisions between packets transmitted by different nodes are avoided. In general, a centralised resource allocator is responsible of the fixed timeslots allocation procedure for the different users, such as a base station in a cellular network or a network control center in a satellite communications network. The advantages of using a TDMA system are:

- Fairness, because all the nodes can be allocated the same number of timeslots and thus the same number of transmission attempts;
- No packet collisions, because only one node can transmit a packet on one timeslot.
- Given that the time axis is not limited, the number of users sharing the channel is also not limited. However, this number should be controlled to avoid large delays.
- Given that the frequency band is not divided between the multiple users, TDMA requires less modems and digital filters at the transmitter and receiver sides.

Drawbacks of TDMA are:

- Under-Utilisation of the resources if the timeslots allocation is fixed and the traffic is bursty and sporadic;
- Larger end to end delays, if the number of users sharing the same channel increases.
- Higher transmitted peak power is required by each user due to the limited timeslots duration.

2.1.2 Frequency Division Multiple Access

Another multiple access technique based on frequency sharing instead of time sharing, is called Frequency Division Multiple Access (FDMA). In FDMA, instead of sharing the communication channel in time as done in TDMA, the users share the frequency band. Each user is allocated a part of the frequency band as illustrated in Figure 2.2. The frequency band is divided between the different users in a way that there is little or no interference between nodes transmitting at the same time. The advantages of FDMA reside in the reduction of transmission delays given its asynchronous nature (i.e. no timeslots synchronisation is required) and the reduction of the required peak power compared to TDMA (i.e. the power is divided by the number of timeslots). However, the number of users sharing the media is limited to the frequency resources available. Besides, FDMA requires high performing digital filters because

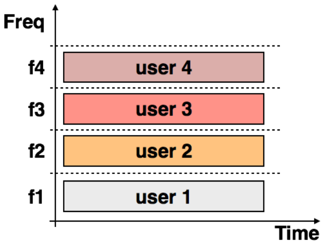


Figure 2.2 – FDMA example: Frequency band shared between 4 users.

of higher signals sensitivity to frequency and timing offsets. An advanced form of FDMA is Orthogonal Frequency Division Multiple Access (OFDMA) scheme [32]. In OFDMA, the whole frequency band is divided into orthogonal sub-carriers, and each node may use several sub-carriers depending on its radio channel conditions. Then, several nodes can access the same timeslot but using different sub-carriers.

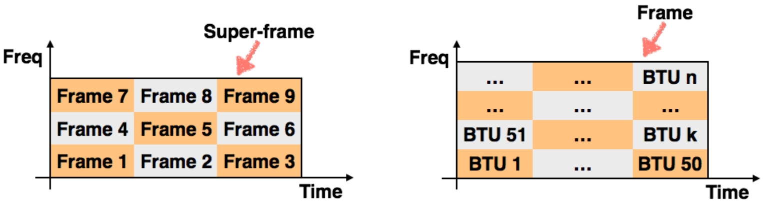
2.1.3 Multi-Frequency Time Division Multiple Access

An alternative protocol combining FDMA and TDMA is Multi-Frequency Time Division Multiple Access (MF-TDMA). MF-TDMA is the multiple access scheme widely used in today's satellite communications. It is mentioned in satellite standards DVB-RCS and DVB-RCS2 [9, 10]. MF-TDMA combines the advantages of both FDMA and TDMA, by allowing the user nodes to transmit at different frequency bands and different timeslots. Thus, MF-TDMA can lead to lower-cost terminals by requiring lower transmitting power compared to TDMA and fewer modems compared to FDMA.

In particular, the return link in satellite networks (i.e., the link from the terminals to the gateway) is structured following the MF-TDMA scheme. Each timeslot is defined with a carrier frequency, a bandwidth, a start time and a duration. To organise the transmission of the packets corresponding to different users, the largest entity defined in DVB-RCS2 is the super-frame. The super-frame is composed of frames and each frame is composed of timeslots. As shown in Figure 2.3a, the super-frame covers a certain frequency bandwidth and it is divided into several frames. The super-frame duration is system dependent. The frame structure is illustrated in Figure 2.3b. A frame can have a duration less or equal to a super-frame, and it can cover several frequency sub-bands. Each timeslot on a frequency sub-band is called a Bandwidth Time Unit (BTU), and each BTU is defined with a fixed number of symbols.

2.1.4 Code Division Multiple Access

Beside timing and frequency sharing techniques, other multiple access protocols such as Code Division Multiple Access (CDMA) are based on spread spectrum techniques. In other words, CDMA allows the user nodes to transmit their packets simultaneously over the same carrier frequency, but utilising different spreading codes [33]. By spreading codes, we mean unique



(a) Super-frame structure.

(b) Frame structure.

Figure 2.3 – Super-frame and frame structures according to the DVB-RCS/RCS2 standards.

codes for each user, with a higher channel signalling rate than the actual rate of the transmitted bit streams, and thus a wider radio spectrum. Spreading codes are used in CDMA to multiply the data streams of each user in order to distinguish among the packets corresponding to different users at the receiver. Each user in CDMA chooses a code among a set of orthogonal codes (for synchronous CDMA) or pseudo-random codes (for asynchronous CDMA). The receiver can distinguish each data-stream received from a particular user by correlating the received signal with its corresponding code sequence. The advantage of CDMA is that the transmitted power required by each user can be significantly reduced given the non-slotted transmission over a wider bandwidth.

2.2 Sharing the Return Link in Satellite Communications

Previously, we presented the MF-TDMA scheme used to divide the satellite resources between multiple terminals. In this section, we give a background on the techniques used to access the return link based on MF-TDMA. The background presented is mainly based on the DVB-RCS2 standard [10]. First, we describe the satellite system hypothesis as well as the network topology. Then, we provide a description of dedicated access and random access techniques.

2.2.1 Network topology

We consider the network topology of a transport star network system, as defined in the DVB-RCS2 system level specifications [34] (see Figure 2.4). The main entities of this system are:

- A transparent geostationary satellite segment to provide the link between the user terminals and the gateway. The Return Link (RL) is the communication link to transmit information from the user terminals to the gateway and the forward link is used to send packets from the gateway to user terminals;
- Terrestrial satellite terminals (STs) constituting the user nodes in a satellite communications system;
- A gateway that plays the role of a hub to interconnect satellite systems with the terrestrial networks. In parallel, it can also play the role of a Network Control Center (NCC) that performs the network control and management functions.

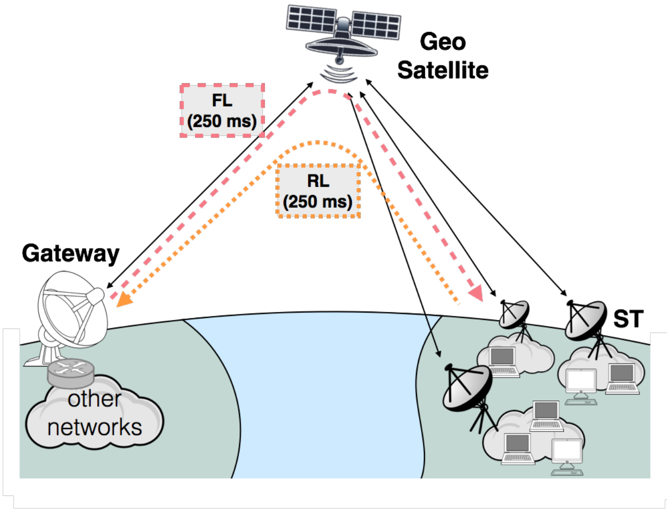


Figure 2.4 – Satellite network topology.

2.2.2 System hypothesis

We consider a high throughput broadband GEO satellite system with directive antennas. The uplink transmissions use the Ka-band (27.5-31 GHz). We suppose that the terrestrial terminals are fixed, and transmitting in a clear sky scenario. The modulation and coding schemes used correspond to the waveforms described in the DVB-RCS2 standard [10], although other waveforms are also considered in other chapters of this thesis. We suppose that power control techniques can be performed at the terminal side. Moreover, the sporadic services offered by such systems can be HTTP (sporadic and bursty internet traffic), FTP (bursty file downloads) or SCADA (Supervisory Control And Data Acquisition)/M2M (Machine to Machine) services.

2.2.3 RL medium access control

As shown in Figure 2.4, the return link is shared among multiple user nodes. Therefore MAC protocols are required to organise signal transmissions on the shared return link and avoid signal interferences, packets losses and retransmissions. In fact, the Round Trip Time (RTT) in satellite communications is relatively high due to the long propagation delays (around 250 ms both on the return link and the forward link for a geostationary satellite). Therefore, packet retransmissions in satellite networks induce significantly high delays and may not be practical for many applications.

As stated previously, the RL is organised based on MF-TDMA. We use the term ‘resources’ to denote the timeslots and frequency bands that can be shared among the user terminals. Two access schemes are used on the RL channel: Demand Assignment Multiple Access (DAMA) [35] and Random Access (RA). In some communication scenarios, both DAMA and RA techniques can be combined. The NCC indicates to the terminals which resources are available for dedicated access as well as random access. Based on resources availability and the type of traffic to be transmitted, each terminal sends Capacity Requests (CR) to the NCC. Then, the NCC makes the corresponding allocation for dedicated access (i.e., for a single terminal) or random access (i.e., a group of terminals). In the following, we detail the use cases associated to DAMA and RA and how both techniques can be combined on a satellite return link.

DAMA techniques

DAMA techniques allow sharing the RL channel among multiple users on a demand basis. Each user is assigned a given set of carrier frequency bands and timeslots following one of the six allocation methods listed below as defined in the DVB-RSC/RCS2 standards.

- Constant Rate Assignment (CRA): Rate capacity provided in full for each allocation. In other words, the user subscribes to a certain required constant rate, and it is offered this constant rate automatically at logon, without sending any capacity requests. CRA should be used for traffic which requires a bandwidth guarantee and a low loss, low latency and low jitter service, at the cost of low efficiency when not fully used.
- Rate Based Dynamic Capacity (RBDC): A rate capacity requested dynamically by the user and released upon the reception of another RBDC request or after a certain timeout.
- Volume Based Dynamic Capacity (VBDC): A volume capacity requested dynamically by the users. Such requests are cumulative. The request indicates a number of required timeslots. Thus, those kinds of requests are suitable for traffic that can tolerate jitter.
- Absolute Volume Based Dynamic Capacity (AVBDC): Same as VDBC but the volume requests are not cumulative.
- Free Capacity Assignment (FCA): Or 'Spare' system capacity, is a volume capacity that is unused and therefore could be assigned for a certain traffic without involving any CR from the users to the NCC. A packet arriving at the user station can be immediately transmitted over a free-assigned channel. However, the availability of free capacity is highly variable and depends on the population of user nodes sharing the RL channel. Therefore, FCA is more suitable for low traffic conditions. For scenarios with both heavy and low traffic loads, this scheme is usually combined with demand based capacity and called Combined Free-Demand Assignment Multiple Access (CF-DAMA) [36].

RA techniques

One main usage of RA methods (also called contention protocols) is sending capacity requests for resource allocations on the return link. Each contention protocol offers a specific maximum throughput. Therefore, if the requests arrival rate exceeds a certain limit, many capacity requests may be lost and backlogged, and some users may wait indefinitely to have allocated resources. Yet, for a PLR= 10^{-2}, the maximum throughput offered by legacy RA techniques varies between only 0.1% and 0.5%. Given this extremely limited throughput, the performance of contention protocols on the return link shall be further enhanced.

Other RA channel use cases stated in the DVB-RCS/RCS2 standards, are listed as follows:

- RA cold start: This case corresponds to when the user terminal is logged onto the network but is initially idle. For instance, CRA control slots following a period with no traffic. Using RA in this case is more efficient because it avoids the DA request/allocation cycle.
- RA-DAMA top-up: To reduce the jitter impact resulting from a sudden increase in the traffic, the RA channel can be used to provide extra capacity until DA capacity is received.
- RA-DAMA backup: For instance when a CR is lost.
- RA for Supervisory Control and Data Acquisition (SCADA): RA is able to accommodate the large number of terminals required in many SCADA systems. RA is also better suited for transmissions of short packets.

A detailed list of legacy and recent RA techniques for satellite communications is given in the next section.

Joint use of DAMA and RA

If the system can take into account specific higher and lower layer requirements, the integration of RA and DAMA techniques can improve the experienced end to end delays and the overall efficiency. In particular, combining DAMA and RA techniques is more practical and efficient for traffic profiles with a mix of long data transmissions and short data transmissions. In this context, enhancing the RA throughput is even more important to avoid data losses and retransmissions.

Based on the DVB-RCS2 standard, the traffic eligibility for DAMA or RA channels is decided according to the specified requirements coming from the higher layers. Then, the limitations coming from the lower layers are taken into account. For instance, a long network layer (L3) packet will result in a large number of lower layer (L2) packets. Therefore, if those L2 packets are sent over the RA channel, their loss probability increases (given the RA channel higher loss rate due to collisions) and the probability of losing the L3 packet becomes more important. Therefore, the choice of a DA or RA channel should take into account L3 packets size. Hence, the joint use of RA and DAMA on the return link can result in positive effects on the communication delays and a better resources utilisation. Nonetheless, the possibility of collisions on the RA channel may produce, in some cases, unpredictability in the packets loss ratio. But this disadvantage of contention protocols is diminished by a reduction of transmission delays. In addition, the introduction of recent efficient RA techniques for satellite communications, allows to obtain a substantial throughput enhancement.

2.3 RA methods used in satellite communications

In the following section, we begin with a brief introduction to understand the importance of using recent RA techniques on the satellite return link. Then, we present the main metrics necessary for the evaluation of such systems. In addition, we provide a detailed list of the synchronous and asynchronous RA proposed recently for satellite communications.

2.3.1 Introduction

During the last decade, the use of satellite technologies for internet and telephony services has widely expanded. International leader companies such as *Google*, *Facebook*, *Amazon* and many others have been suggesting new satellite networks deployments in order to bring and accelerate data connectivity for many users deprived of the economic and social benefits of the internet. Also, the importance of satellite communications has been significantly visible in Machine-to-Machine (M2M) applications. The increased demand of M2M technologies has encouraged satellite service providers to integrate M2M in satellite systems. Hence, satellite networks have been significantly growing in order to handle huge demands of internet connectivity, telephony services, and M2M applications. One of the main challenges for satellite service providers and network operators is providing connectivity to as many demanding users as possible, while efficiently using the bandwidth resources and responding in the minimum achievable delays, depending on each application.

As previously presented, to set up a communication and access the satellite resources on the return link, the user terminals may use DAMA or RA techniques. DAMA methods require an exchange of capacity requests and resource assignments between the terminals and the NCC, so that each terminal is assigned one carrier frequency and one timeslot. As for RA techniques, many users can share the same frequency band and even the same timeslot, at the cost of a probability for packet collisions. However, in systems with large populations of users where the communications involve transmissions of very short packets, sporadic internet traffic or long idle periods, the use of DAMA or CF-DAMA techniques is not optimal [13]. In this context, the use of more efficient RA techniques combined with DAMA for data transmission on the return link is of interest. Legacy RA protocols such as Aloha and Slotted Aloha are currently being used on the return link for transmissions of signalling packets, logons, capacity requests and short data packets.

In the following sections, we provide a background on the performance of these techniques, and we explain why they are not practical for data transmissions. In order to make RA protocols more suitable for data transmissions over the satellite return link, and to further enhance the MAC layer performance and decrease the Packet Loss Ratio (PLR), several new RA techniques have been proposed in the literature. These recent RA techniques propose to cope with packets collisions on the physical layer by using the principle of Successive Interference Cancellation (SIC). They can be synchronous (slotted) or asynchronous (non slotted), and based on redundancy transmission or spread-spectrum techniques, depending on the system requirements and the applications. In the following, we provide a detailed state of the art on the synchronous and asynchronous RA techniques proposed for satellite communications. But first, let us explain the metrics required to measure the performance of such multiple access systems.

2.3.2 Metrics

The metrics used to evaluate the performance of most recent RA protocols designed for satellite communications are: the normalised MAC-layer load (λ in packets per slot and G in bits per symbol), the normalised MAC-layer throughput (T in in bits/symbol) and the MAC-layer Packet Loss Ratio (PLR). In some cases, particularly in asynchronous RA schemes, the communication end-to-end delay is also evaluated.

We consider a Constant Bit Rate (CBR) traffic profile. Thus, the normalised MAC-layer load expressed in packets/slot is denoted by λ and computed as follows:

$$\lambda = \left(\frac{N_u N_b}{N_s} \right) \frac{1}{N_b} \quad \text{(packets/slot)}, \tag{2.1}$$

with N_u being the total number of users transmitting one packet each, on the duration of one frame. N_s is the total number of slots on a frame and N_b is the number of packet replicas (or packet fragments) transmitted by each user on different timeslots of the frame. λ is normalised to the number of packet replicas N_b, so that different RA schemes using different values of N_b can be fairly compared. In addition, in order to compare different systems using different types of modulation and coding schemes, we use the normalised MAC-layer load G expressed in bits/symbol. G is computed as shown below:

$$G = \lambda R log_2(M) \quad \text{(bits/symbol)}, \tag{2.2}$$

with M being the modulation order (for example, $M = 4$ for Quadrature Phase Shift Keying), and R being the coding rate. The normalised MAC-throughput T corresponding to a certain load G can be expressed as:

$$T = G(1 - \text{PLR}(G)) \quad \text{(bits/symbol)}, \tag{2.3}$$

where PLR(G) is the Packet Loss Ratio i.e. the percentage of lost packets on a frame for a given load G and a given Signal to Noise Ratio (SNR).

2.3.3 Legacy RA protocols

We can distinguish two kinds of contention protocols also called Random Access (RA) protocols: the slotted and the non-slotted.

Non-slotted contention protocols: Aloha

In non-slotted contention protocols, the time axis can be accessed by any user terminal at any instant and not just at the beginning of a timeslot. Whenever a user has information to transmit, it can send it without waiting for resource allocation procedures. Among traditional non-slotted contention protocols, we can cite the famous protocol 'Aloha' [14], which was proposed by Norman Abramson in the 60s at the university of Hawaii. The basic principle of the Aloha protocol as shown in Figure 2.5, is the following:

- If a user has a backlogged packet, it sends it at a random time instant.
- At the receiver side, if the packet has encountered collisions from other users, it is considered destructed and non decoded.

Given the destructive collisions inducing packet losses, the Aloha protocol is not well suited for systems with a relatively high population of users. Therefore, it is mostly used for signalling transmissions, logons and resource allocation requests.

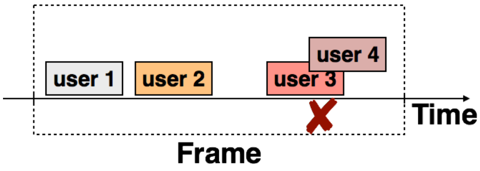


Figure 2.5 – Contention access protocols: Aloha.

The performance of Aloha is analysed in terms of the achieved MAC layer throughput, i.e., the average rate of successfully decoded packets over the total transmitted packets on one frame duration. It is often assumed that packet arrivals in Aloha occur according to a Poisson process with an average value λ_n expressed in received packets per second. The average number of packet arrivals over one packet duration T_p is denoted by $\lambda = \lambda_n T_p$. It also refers to the average normalised MAC layer load. The probability of having k packet arrivals in a duration of T_p seconds is expressed as

$$P(\text{k packet arrivals in } T_p \text{ seconds}) = \frac{\lambda^k}{k!}e^{-\lambda} \quad . \tag{2.4}$$

If a given packet arrives at an arbitrary instant t_0, the probability that this packet does not suffer from a collision is the probability that no packet arrivals occur in the interval $[t_0 - T_p, t_0 + T_p]$. Therefore, in Aloha systems, the probability that a packet is received without collisions is given by

$$P_{success} = P(0 \text{ packet arrivals in } 2T_p \text{ seconds}) = e^{-2\lambda} \quad . \tag{2.5}$$

Then, the throughput of Aloha, is the average number of packet arrivals over a duration T_p multiplied by $P_{success}$. And it is given by

$$T_{Aloha} = \lambda P_{success} = \lambda e^{-2\lambda} \quad . \tag{2.6}$$

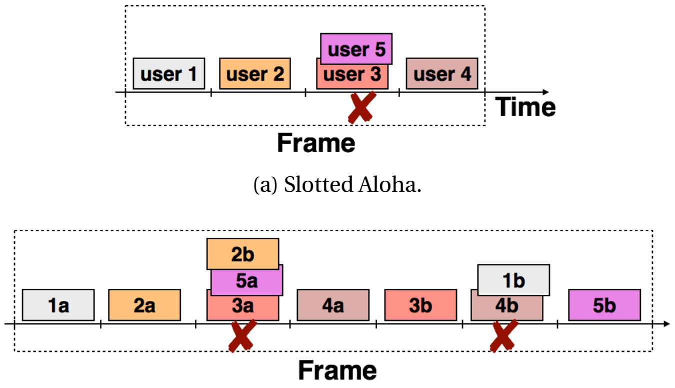
The maximum theoretical throughput achieved with Aloha is computed as shown below

$$\frac{dT}{d\lambda} = e^{-2\lambda} - 2\lambda e^{-2\lambda} = 0 \Rightarrow \lambda_{max} = \frac{1}{2} \Rightarrow T_{max} = \frac{1}{2}e^{-1} = 0.1839 \quad . \tag{2.7}$$

Obviously, the maximum throughput of Aloha is relatively low. In particular for a target PLR of 10^{-3} generally required for satellite scenarios, the Aloha throughput does not exceed 10^{-3} bits/symbol.

Slotted contention protocols: Slotted Aloha & Diversity Slotted Aloha

In slotted contention protocols, each user node should wait for the beginning of one timeslot in order to send its packet. However a timeslot is not already allocated to only one user as in TDMA-DAMA. On the contrary, any user node can randomly choose a timeslot whenever it has information to transmit. Thus, no prior resource allocation is required, but timeslot-level synchronisation is needed between the different users. Among legacy slotted contention protocols we can cite Slotted Aloha [15] and Diversity Slotted Aloha [17, 37] (see Figure 2.6).



(a) Slotted Aloha.

(b) Diversity Slotted Aloha.

Figure 2.6 – Contention access protocols: Slotted Aloha & Diversity Slotted Aloha

.

Slotted Aloha (SA) is a modified version of Aloha where the time axis is divided into timeslots. In SA the probability of partial collisions between packets is removed and only full collisions can still occur (for instance, in Figure 2.6a, user 5 decides to transmit its packet in the same timeslot as user 3). Therefore, the successful decoding probability of a given packet is the probability that no other packets have been received in the same timeslot. Then the probability of receiving a packet without collisions in SA is expressed as shown below

$$P_{success} = e^{-\lambda} \quad . \tag{2.8}$$

And the theoretical throughput of SA is

$$T_{SA} = \lambda P_{success} = \lambda e^{-\lambda} \quad . \tag{2.9}$$

The maximum theoretical throughput for SA is computed as follows

$$\frac{dT}{d\lambda} = e^{-\lambda} - \lambda e^{-\lambda} = 0 \Rightarrow \lambda_{max} = 1 \Rightarrow T_{max} = e^{-1} = 0.3679 \quad . \tag{2.10}$$

Thus, the maximum throughput achieved with SA is doubled compared to the throughput of Aloha. However, the performance of SA is still poor for satellite scenarios because of the high PLR.

Diversity Slotted Aloha is an alternative version of SA with packets replication. As shown in Figure 2.6b, each user node can transmit multiple replicas of the same packet on randomly chosen timeslots of the frame. At the receiver side, each replica of a given packet has a slightly higher probability of being received without collisions, due to the increased timeslots diversity.

Given that the average channel load is now multiplied by N_b, the probability of receiving any packet replica without collisions is $e^{-N_b\lambda}$. Then the probability of receiving at least one replica of a given packet without collisions is (1-Probability of receiving all replicas with collisions) as shown below

$$P_{success} = 1 - (1 - e^{-N_b\lambda})^{N_b} \quad . \tag{2.11}$$

The throughput obtained in this case is

$$T_{DSA} = \lambda P_{success} = \lambda \left(1 - (1 - e^{-N_b\lambda})^{N_b}\right) \quad . \tag{2.12}$$

Figure 2.7 shows a comparison of the MAC layer throughput achieved with Aloha, SA and DSA. The notation *DSA-N_b* is used to denote DSA with different number of replicas per packet. Given the results observed in Figure 2.7, the maximum throughput obtained with DSA-2 is higher than Aloha, SA, DSA-3 and DSA-4, for a MAC channel load $\lambda < 0.5$. However, the performance degrades rapidly for higher channel loads. Figure 2.7 also shows that the maximum throughput achieved with all the RA techniques presented previously is relatively low and induces high levels of packet losses. Therefore, legacy RA methods such as Aloha, SA and DSA are not good candidates to be used for applications that do not tolerate large packet retransmissions delays. For this reason, researchers have been investigating in new RA methods for satellite communications, that can cope with packet collisions and increase the MAC layer throughput.

2.3.4 Contention Resolution Diversity Slotted Aloha

Contention Resolution Diversity Slotted Aloha (CRDSA) [16] is an enhanced version of DSA proposed at the European Space Agency (ESA) in 2007. CRDSA has been included in the DVB-RCS2 standard. It is a synchronous RA technique based on packets replication and the SIC process. As shown in Figure 2.8, in CRDSA, each terminal sends two or more replicas of the same packet on randomly chosen timeslots of the frame. Each packet contains a signalling field with pointers to the locations of its replicas. At the receiver side, the SIC process is

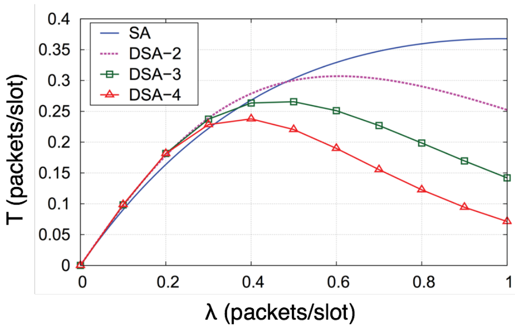


Figure 2.7 – Analytical throughput vs normalised channel load for SA, and DSA (source: [1]).

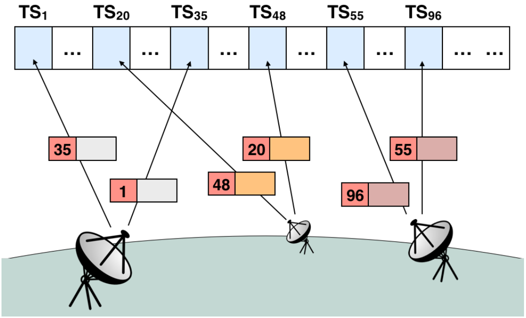


Figure 2.8 – CRDSA transmission scheme with $N_b = 2$ replicas per user.

applied for each successfully decoded packet. In other words, when one packet is decoded successfully, it is removed from the frame, then the decoded pointers are used to localise and remove its replicas. Hence, the interference contribution on the remaining non-decoded packets is reduced. The frame is stored at the receiver side and scanned iteratively until no more packets can be resolved, or after a number of iterations is reached. Figure 2.9 illustrates an example of a frame treated with CRDSA, in the case of equi-powered packets. Figure 2.9a and Figure 2.9b show the packets successfully decoded in the first CRDSA iteration at the receiver side. First, the packet replica (3b) is decoded successfully in timeslot 4 (TS_4) because it is received without collisions. Its replica (3a) is localised with the decoded pointer then reconstructed and removed from TS_2. In a similar way, the packet replica of user 1 on TS_5 is decoded successfully and removed from the frame as well as its copy received on TS_1. In a second CRDSA iteration (see Figure 2.9c and Figure 2.9d), the packets corresponding to users 2 and 4 are successively decoded and removed from the frame. Thus, the SIC process is repeated until all the packets are decoded successfully.

It is worth noting that without SIC, a DSA scheme is obtained and the decoding process stops only after decoding the replicas (1b) and (3b). The remaining packets on the frame are lost. The authors of CRDSA showed that with a QPSK modulation, a Forward Error Correction (FEC) code of rate 1/3, and $E_s/N_0 = 10$ dB, replicas experiencing one packet collision can be resolved successfully. In that case, with $N_b = 3$ replicas per packet, the maximal throughput of CRDSA can reach 1.2 packets/slot which is equivalent to an efficiency of 0.8 bits/symbol.

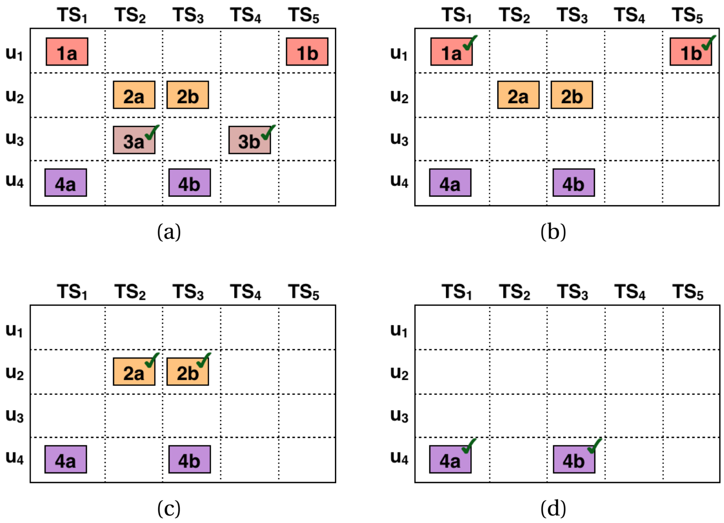


Figure 2.9 – CRDSA example: 4 users (u) sharing a frame of 5 timeslots (TS).

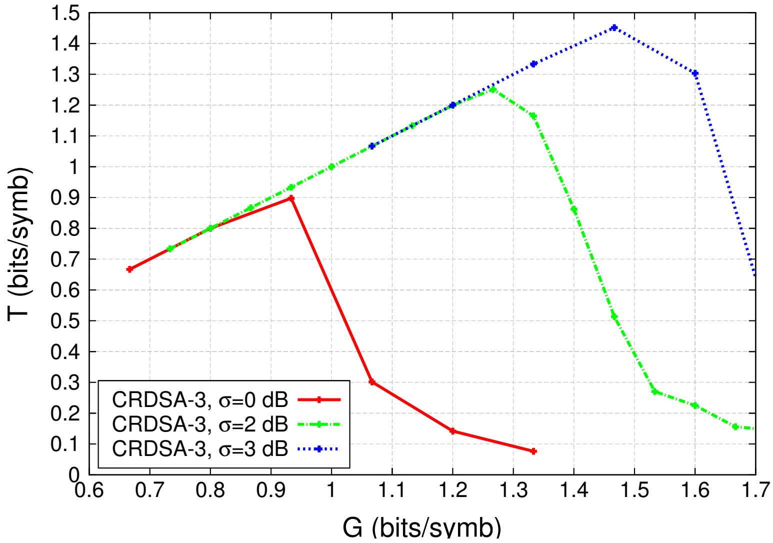
CRDSA with packets power unbalance

The first version of CRDSA considered packets received at the same power level (i.e., equi-powered). However, it has been shown in [38, 27, 13], that packets power unbalance can play a major role in enhancing the performance of RA schemes such as: SA, DSA, and CRDSA. When a collision occurs, the fact that some packets have higher power levels than others enables the receiver to capture the strongest packets [15, 39]. The so-called 'capture effect' is the ability to correctly receive a signal even if it has been interfered by other signals. Exploiting the capture effect allows to resolve packet collisions, increase network throughput and reduce communication delays.

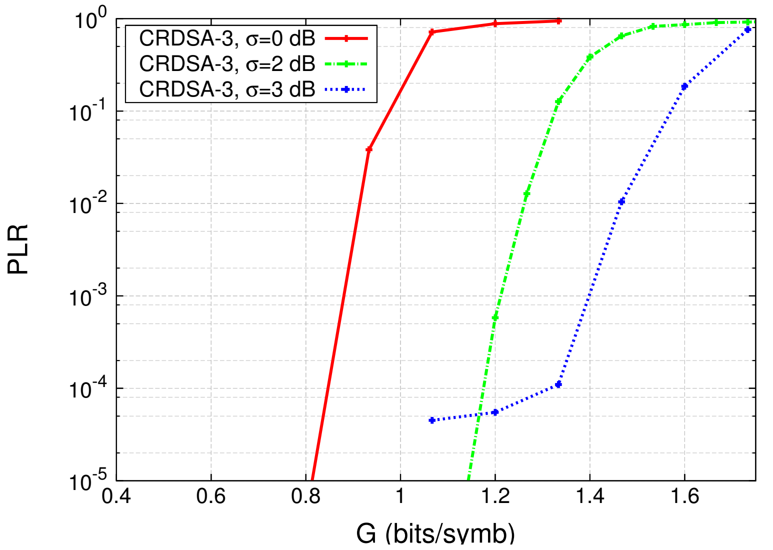
The authors of CRDSA evaluated the impact of received packets power unbalance on the performance in [38, 27], and showed that the performance of CRDSA with log-normally distributed packets power is significantly enhanced compared to the equi-powered packets case. Figure 2.10 depicts the performance of CRDSA with $N_b = 3$ replicas per packet, in terms of normalised MAC throughput in bits/symbol and PLR, with several values of the standard deviation σ for the truncated lognormal packets power distribution ($\sigma = 0$ dB refers to the case of equi-powered packets). The results are shown using a QPSK modulation with a 3GPP turbo code [25] of rate 1/3 and $E_s/N_0 = 10$ dB. An error floor appears in Figure 2.10b with $\sigma = 3$ dB, because there is a higher probability of packets received at lower values of E_s/N_0.

Irregular Repetition Slotted Aloha

Another variant of CRDSA with an irregular number of replicas per packet is Irregular Repetition Slotted Aloha (IRSA) [18]. In IRSA, an optimal distribution for the number of replicas transmitted by each terminal has been derived by exploiting the analogy between the SIC



(a)



(b)

Figure 2.10 – CRDSA performance with $E_s/N_0 = 10$ dB and $N_b = 3$ replicas. QPSK modulation, 3GPP turbo code $R = 1/3$. (a) Throughput. (b) PLR.

process and iterative erasure decoding of graph-based codes [40, 41]. In order to maximise the efficiency on the RA channel, the SIC process in CRDSA is described with a bipartite graph. IRSA enhances the performance of CRDSA in terms of throughput but its performance for a packet loss ratio around 10^{-3} is lower when compared to CRDSA with 3 or 4 replicas.

2.3.5 Coded Slotted Aloha

Coded Slotted Aloha (CSA) is another slotted RA scheme which was patented by Liva in 2011 [42], and introduced in [19, 20]. Its achievable throughput was investigated in [43]. CSA is a generalisation of IRSA, that encodes packets prior to transmission, rather than simply replicating them. CSA also applies the SIC principle for resolving collisions. Figure 2.11 shows the operations performed on each packet before transmission on a frame employing CSA. First, the packet is divided into k information fragments. Then the k fragments are encoded via a local packet-oriented code which generates n encoded fragments ($n > k$). The code rate in this case is $R = \frac{k}{n}$. To each encoded fragment, signalling information is added in order to point at the other fragments of the same packet on the frame. At the receiver side, if the packet fragments of a given user are received in collision, they are considered lost (hence, the analogy with erasure decoding). However, the receiver may recover the information received from other fragments of the same packet that have not experienced collisions. Thus, the packet can be decoded and its interference contribution is subtracted from the frame (i.e., all the corresponding fragments are removed). In [43], it is shown that the asymptotic throughput (i.e., the throughput obtained when $N_u \to \infty$ and $N_s \to \infty$) can reach up to 0.9 packets per slot with an erasure code of rate $R = 2/7$.

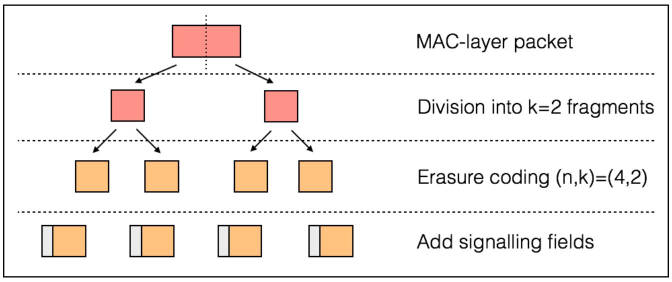


Figure 2.11 – CSA transmission scheme.

2.3.6 Multi-Slot Coded Aloha

Multi-Slot Coded Aloha (MuSCA) [21] is another recent synchronous RA method proposed in 2010 by H.C. Bui et al. Figure 2.12 illustrates the operations done at the transmitter side in MuSCA. First, the transmitter encodes the packet with a robust FEC code of rate R. The codeword is bit-interleaved and modulated. Then the transmitter divides the obtained codeword into multiple fragments. To each fragment, a separate signalling field is added and used for the purpose of fragments localisation on the frame. In order to be able to localise the fragments of a given packet even when collisions occur, the signalling field is encoded with a robust Reed-Muller code of rate R_s.

At the receiver side, two phases can be distinguished. The first phase is the signalling fields decoding and the second is the useful information decoding. First the decoder scans the frame and attempts to decode the signalling fields. Then, it performs SIC on the decoded signalling

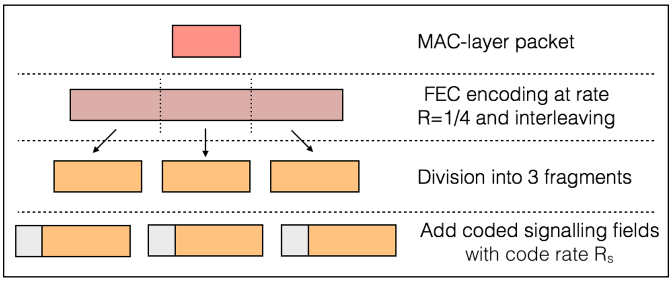

MAC-layer packet
FEC encoding at rate R=1/4 and interleaving
Division into 3 fragments
Add coded signalling fields with code rate R$_s$

Figure 2.12 – MuSCA transmission scheme.

fields. In other words, whenever a signalling field for a given packet is successfully decoded, the signalling fields for the same packet on other timeslots are reconstructed and subtracted from the frame. Once all the signalling fields for all the packets on the frame are decoded, the receiver knows the locations of all the fragments as well as the level of collisions on each timeslot. Therefore the receiver can proceed to the next phase of useful information decoding. For each packet, its corresponding fragments are collected and the codeword is reconstructed then demodulated, de-interleaved and decoded. Of course, SIC is also performed on the decoded payload data.

Figure 2.13 and Figure 2.14 show an example of the two-phases decoding process at the receiver side in MuSCA. In Figure 2.13a, the decoder finds the packet of user 2 on slot 5 free of collisions, so it decodes its corresponding signalling field successfully and removes the signalling parts of its fragments in slot 2 and slot 8. The packet of user 3 in slot 2 becomes in collision with only one other packet. Therefore, the receiver can decode its signalling field successfully given the robust Reed-Muller code used. Once decoded, the other signalling fields in slot 3 and 6 can be removed. The decoder continues this process iteratively until all the signalling fields are decoded.

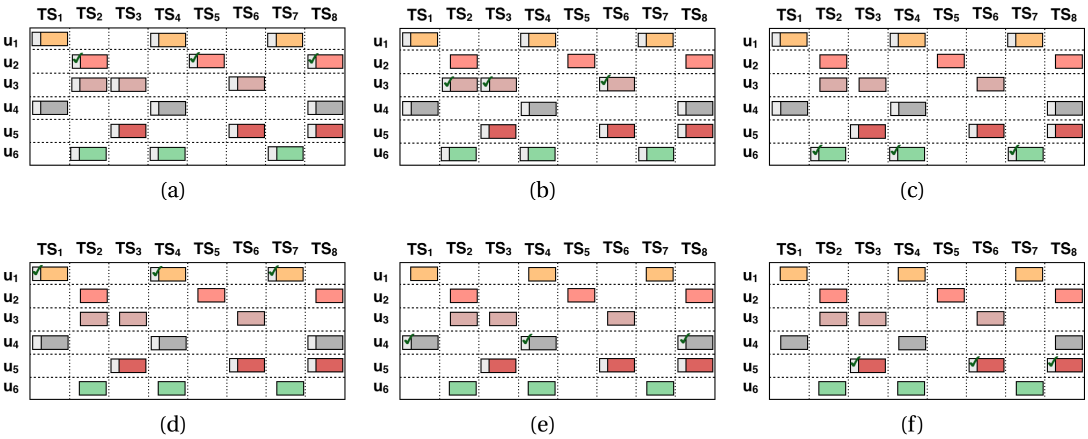
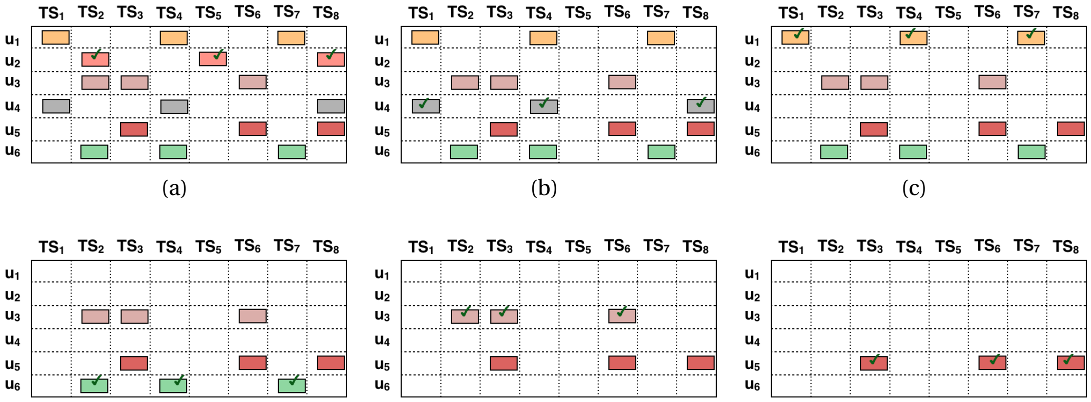


Figure 2.13 – MuSCA example: signalling fields decoding phase. 6 users (u) sharing a frame of 8 timeslots (TS).

Figure 2.14 depicts the useful information decoding phase. The decoder starts by choosing the packet which is less interfered on all its fragments. In the case of Figure 2.14a, it starts by decoding the packet of user 2. The fragments of the packet are collected from slots 2, 5 and 8, then the codeword is reconstructed, demodulated and decoded. Given the robust FEC code used on the payload part of each packet, the successful decoding probability can be considered relatively high. Once decoded successfully, the packet and all its fragments are removed from slots 2, 5 and 8. Then, the decoder tries to resolve the packet fragments of user 5. If the decoding attempt fails, then the decoder passes to user 4 and so on until all the packets on the frame are successfully retrieved.



(a) (b) (c)

(d) (e) (f)

Figure 2.14 – MuSCA example: useful information decoding phase. 6 users (u) sharing a frame of 8 timeslots (TS).

Figure 2.15 shows the performance of MuSCA in terms of throughput T (in packets/slot) obtained with a QPSK modulated payload and a Consultative Committee for Space Data Systems (CCSDS) FEC code of rate $R = 1/6$ and a number of fragments per packet N_f equal to 3. Several levels of E_s/N_0 are compared. The results are shown for a system with perfect channel state knowledge. It is observed that the performance is significantly enhanced compared to the RA techniques presented previously. The maximum throughput obtained is higher than 1 packet/slot starting from $E_s/N_0 = 1$ dB. The authors of MuSCA have also proposed an irregular version of the scheme called Irregular MuSCA [44], where each user sends a random number of packet fragments on the frame. The maximum throughput achieved with Irregular MuSCA could reach 1.4 packets/slot for an optimal probability distribution of the number of fragments sent by each user. However, in both regular and irregular versions of MuSCA, the cost for the better performance is the increased signalling overhead required to localise the packets fragments on the frame.

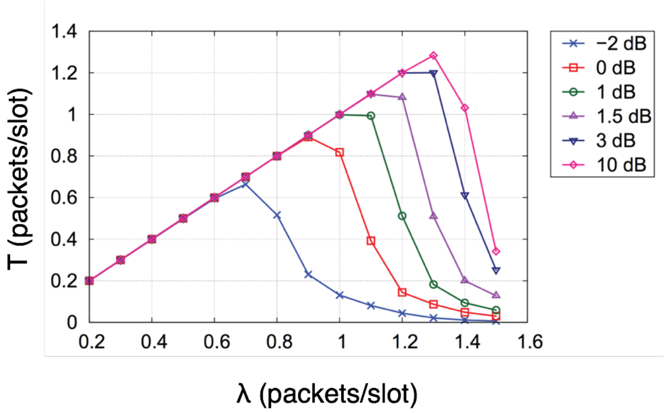


Figure 2.15 – Throughput T vs. channel load λ for MuSCA in perfect channel conditions with several values of E_s/N_0. QPSK modulation, FEC code $R = 1/6$, $N_f = 3$ fragments per packet, packet length 456 bits, and $N_s = 100$ slots (Source: [1]).

2.3.7 Enhanced Spread Spectrum Aloha

Among the asynchronous RA methods recently proposed in the literature, we cite Enhanced Spread Spectrum Aloha (E-SSA) [45, 46]. First, let us give a brief state of the art on the legacy RA method Spread Spectrum Aloha (SSA) [47]. The concept of SSA is inspired from code spreading in CDMA. As stated in Section 2.1.4, CDMA allows sharing the media between several transmitters by using 'code division'. In order to separate signals from different users at the receiver, each user is assigned a unique binary code with a higher rate (R_c) than the actual data bit rate (R_b). The spreading factor is denoted by g, with $g = \frac{R_c}{R_b}$.

In SSA, the authors showed that using different codes in a CDMA system is not necessary to distinguish between the users. Instead, a large value of the spreading factor g can be used as the probability of collision with a spreading sequence aligned at chip level increases when the spreading factor is reduced. A subtraction algorithm is also applied in SSA. In other words, upon the reception of several signals at the same time, the receiver determines the strongest or earliest signal, then demodulates and decodes it and subtracts it from the frame. Then the same process is repeated on the remaining signals of the other users. The fact that the signals of different users are not synchronised, enables to distinguish between them by correlation with the corresponding spreading code.

E-SSA is a recent version of SSA that introduces a sliding window based approach combined with iterative interference cancellation (IC) on each window. As shown in Figure 2.16, the receiver stores the signal received on one window duration W (usually W is equal to the duration of 3 packets), then when the E-SSA process on one window is finished, the receiver slides the actual window by a predefined step ΔW.

On each window duration, the detector in E-SSA performs the following operations iteratively until N_{max} iterations are reached:

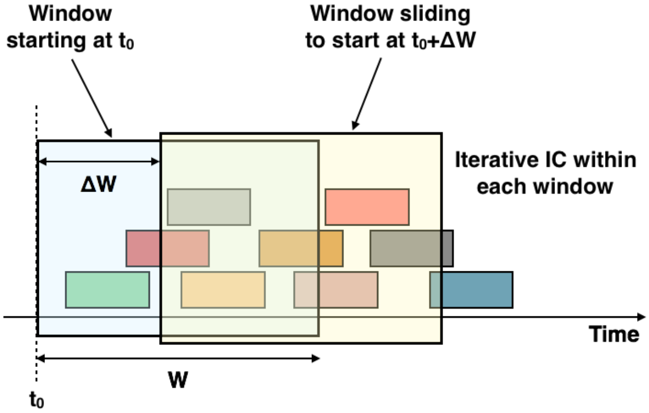


Figure 2.16 – E-SSA sliding window and iterative IC algorithm.

1. Detection of the packet with the strongest Signal to Noise plus Interference Ratio (SNIR);
2. Channel estimation then demodulation and decoding of the strongest packet;
3. If decoding is successful after Cyclic Redundancy Check (CRC):
   (a) Re-encoding and modulation of the decoded packet;
   (b) Interference cancellation.

The E-SSA performance was evaluated in [45]. The authors showed that the throughput with E-SSA can reach up to 1.7 bits/symbol for a target PLR of 10^{-4} with a log-normal packets power distribution ($\sigma = 2$ dB), BPSK modulation, 3GPP turbo code of rate 1/3 and a spreading factor $g = 256$. This result presents a significant gain compared to SSA which achieves a maximal throughput of only 0.5 bits/symbol.

2.3.8 Enhanced Contention Resolution Aloha

Enhanced Contention Resolution Aloha (ECRA) is another asynchronous RA protocol proposed in 2013 [48]. As its name reveals, it is an enhanced version of a previously presented RA technique called Contention Resolution Aloha (CRA) [49]. In CRA, the authors have proposed a relaxation of the slots boundaries of CRDSA by allowing asynchronous packets transmissions on one frame. However, the concept of frame-level synchronisation is still present. ECRA adds the idea of replicas 'best parts' combination to CRA. As a matter of fact, given the asynchronous nature of the communication, replicas corresponding to a same packet may experience interference on different parts of the packet as shown in Figure 2.17.

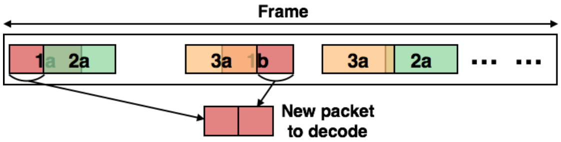


Figure 2.17 – ECRA scheme.

The example in the figure shows that the first replica of user 1 experiences collisions only on the right-most part of the packet, however the second replica is interfered only on the left part. In the case of CRA, both replicas can be non-decodable if the interference power is too high. For this reason, ECRA proposes a solution to this problem by combining the non-interfered symbols of the replicas into a new packet. Thus, it is obvious that the new packet would have a higher successful decoding probability. The decoding process of ECRA can be detailed as follows:

1. The frame is stored at the receiver side and the SIC process is applied. The frame is scanned iteratively in order to detect decodable packets. Whenever a packet is decoded successfully, it is removed from the frame and the decoded pointers are used to localise its replicas and remove them as well;
2. When no further packets could be decoded on the frame, ECRA intervenes in order to attempt to decode the remaining packets using the following procedure:
   (a) If some parts of a given packet encounter interference in all the replicas, then the parts (or the symbols) encountering the lowest interference power are used to construct a new packet. Therefore, ECRA shall perform symbol-by-symbol SNIR estimation in order to correctly select the parts of the replicas to combine;
   (b) If the new constructed packet is successfully decoded, the packet and its replicas are removed from the frame.

The authors of ECRA showed that it can achieve a maximum normalised throughput of 1.2 bits/symbol with $N_b = 2$ replicas, a QPSK 1/2 modulation coding scheme and $\frac{E_s}{N_0} = 10$ dB.

2.3.9 Asynchronous Contention Resolution Diversity Aloha

Another asynchronous RA method recently proposed in the literature is Asynchronous Contention Resolution Diversity Aloha (ACRDA) [50]. ACRDA is a modified asynchronous version of CRDSA. The operations at the transmitter and the receiver sides have similarities with both CRDSA and E-SSA in order to cope with asynchronous transmissions. In ACRDA, timeslots and frame boundaries are not defined in reference to the global timeline at the centralised gateway (i.e., NCC) demodulator. Instead, the delimitation of timeslots and frame are local to each transmitter and completely asynchronous among different transmitters. Thus, unlike CRA, frame-level synchronisation among users is not needed. The term 'virtual frame' (VF) is used to refer to the local frame at each transmitter. Each VF contains N_{slots} and each slot has a duration T_{slot}, so that the duration of a VF is $T_{VF} = N_{slots} T_{slot}$. Figure 2.18 illustrates the reception of 3 asynchronous VFs corresponding to different transmitters with completely independent timing offsets at the receiver. If all the transmitters have the same timing offset, then the classical CRDSA scheme is obtained. The ACRDA scheme at the transmitter side is detailed as follows.

1. Before transmitting a packet on the RA channel, N_b replicas are generated and N_b timeslots are randomly selected within the duration of one VF;

2. Similarly to CRDSA, the information concerning the location of the other replicas is added to each packet. In the case of ACRDA, the location information is the timeslot offset relative to the start of the current packet;
3. The start time of a VF is chosen randomly at the transmitter side, and no wide centralised synchronisation is needed;
4. A preamble containing a known sequence common to all transmitters is added to the beginning of each packet replica. This common preamble is used for packets detection and channel estimation at the receiver side;
5. Each packet replica is transmitted on the selected timeslots of the local VF.

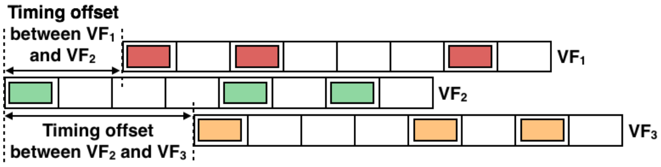

Timing offset between VF₁ and VF₂
VF₁
VF₂
Timing offset between VF₂ and VF₃
VF₃

Figure 2.18 – ACRDA virtual frames scheme.

At the receiver side, the same window-based memory processing as done in E-SSA is applied as it was shown in Figure 2.16. The operations of ACRDA at the receiver side are detailed below.

1. The received signal is down-converted, filtered and sampled;
2. For each sliding window,
   (a) The signal covering a duration of W VFs is stored in the receiver memory (in general $W = 3$ is assumed);
   (b) The ACRDA process is repeated iteratively on each window, as explained below:
      i. First, the common packet preamble is searched using a cross-correlation matched to the preamble sequence;
      ii. Each time a preamble sequence is detected, the packet demodulation and decoding is attempted;
      iii. If a packet is successfully decoded, channel estimation is performed using the full packet content. Then the packet is removed from the frame;
      iv. The successfully decoded packet is also used to locate its other replicas, reconstruct their corresponding signals and remove them from the frame;
      v. If the currently decoded packet points to a replica that is not in the current window, the packet information is stored until the sliding window finds the corresponding replica.
   (c) Once the ACRDA process on one window is completed, the window is shifted towards the next $\Delta W T_{VF}$.

The authors in [50] have concluded that ACRDA performs slightly better than CRDSA in terms of throughput and PLR, particularly with $N_b = 2$ replicas per packet. However, in the asynchronous mode, the implementation complexity at the receiver side is higher. The performance simulations done in [50] have considered a QPSK modulation and a 3GPP FEC code of rate $R = 1/3$ and $E_s/N_0 = 10$ dB. It has been shown that the maximum normalised throughput is achieved with $N_b = 2$ replicas and it can reach 0.9 bit/symbol for a $PLR < 10^{-4}$. With a lognormal packets power distribution of $\sigma = 3$ dB, the throughput can increase up to 1.5 bits/symbol for a $PLR < 10^{-4}$. At the same time, the authors have shown that significant gains are achieved with ACRDA in terms of packets transmission delays.

2.4 Background on Channel Estimation for recent RA methods

In previous sections, we presented the various techniques used to organise a shared channel between several users. We also explained the importance of developing more efficient and reliable RA techniques for the satellite return link. In this section, we address one of the main practical problems related to this field: channel estimation for superimposed received signals.

2.4.1 Brief introduction

In fact, most of the recent RA methods proposed in satellite communications use the SIC principle. When a packet is successfully received on a given timeslot of the frame, the receiver can reconstruct its signal and remove it from the corresponding timeslot, in order to reduce the degree of packet collisions on this timeslot. In diversity based RA techniques such as CRDSA, IRSA, CSA, and MuSCA, the reconstructed packet should also be removed from the timeslots containing its other replicas or other fragments.

The problem in signal cancellation from a given timeslot is that it requires channel knowledge for the signal to be removed on this particular timeslot. If the Channel State Information (CSI) is not taken into consideration at the moment of signal cancellation, then the signal would not be perfectly removed and residual channel estimation errors would be added to the other packets present on the same timeslot. However, in a real system, the receiver does not have prior knowledge of the CSI for each user on each timeslot. Therefore, channel parameters should be accurately estimated, otherwise the successfully decoded packets on a given timeslot would not be correctly removed and important residual channel estimation errors could be added to the remaining packets.

In CRDSA, the problem of channel estimation and their effect on the SIC process was studied in [51]. The authors used data-aided channel estimation techniques in order to estimate the amplitude, the frequency shift, the time shift and the varying phase of replicas on separate timeslots. In particular, the authors in [51] exploited the known data acquired on the successful retrieval of a given replica, in order to estimate the channel parameters corresponding to its copies on other timeslots.

However, in other RA methods such as MuSCA, the maximum throughput is significantly enhanced. Thus the degree of packet collisions resolved on a frame can be relatively high and the probability of receiving contention-free packets is reduced. In such systems able to bear high channel loads, a main challenge is the estimation of channel parameters in case of superimposed signals, in order to achieve a performance close to the perfect CSI case.

2.4.2 Signal & channel model assumptions

We consider a discrete signal vector $\boldsymbol{y}$ of length L, received within the duration of one timeslot, containing N_{coll} packets in collision and oversampled with an oversampling factor Q. The i^{th} element $y(i)$ of the vector $\boldsymbol{y}$ is expressed as shown below:

$$y(i) = \sum_{k=1}^{N_{coll}} h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT - \tau_k T) + n(i) \quad , \tag{2.13}$$

where:

- T and $T_s = T/Q$ are respectively the symbol period and the oversampling period;
- $i = 0, 1, ..., LQ - 1$ and $n = 0, 1, ..., L$ refer respectively to the T_s-spaced and the T-spaced samples of the discrete signal y;
- $x_k(n)$ refers to the n^{th} symbol transmitted in the k^{th} packet;
- τ_k is the timing offset experienced by the k^{th} packet. τ_k is supposed to be uniformly distributed in $[-0.5T, 0.5T]$.
- g stands for the shaping filter function;
- n is the complex Additive White Gaussian Noise (AWGN) process with variance σ_n^2;
- h_k is the complex channel coefficient affecting on the k^{th} packet and modelled as given in [52] and as shown below:

$$h_k(i) = A_k e^{j(2\pi\Delta f_k i T_s + \phi_k)} \quad , \tag{2.14}$$

  with:
  - A_k being a log-normally distributed random variable modelling the channel amplitude for the k^{th} packet, assumed to remain constant over one frame duration.
  - Δf_k being the frequency offset relative to the k^{th} packet, assumed constant over the frame duration. Δf is considered uniformly distributed in $[0, \Delta f_{max}]$ with Δf_{max} equal to 1% of the symbol rate (with respect to the DVB-RCS2 guidelines [28]);
  - ϕ_k referring to the phase shift of the k^{th} packet. It is assumed to remain constant only over the duration of one timeslot. ϕ_k is a random variable drawn independently for each packet on each timeslot, from a uniform distribution in $[0, 2\pi]$.

The structure of the packet sent by each user is shown in Figure 2.19. Guard intervals delimit

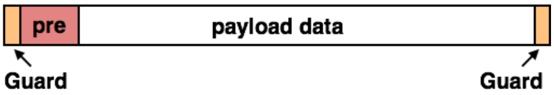


Figure 2.19 – Packet structure with preamble and guard intervals.

the beginning and the end of each packet, in order to ensure that distinct transmissions on distinct timeslots do not interfere with one another. A preamble (denoted by the vector $\boldsymbol{x_{pre}^k}$) is added to the beginning of a packet transmitted by user k. The preamble is a unique orthogonal sequence modulated with Binary Phase-Shift Keying (BPSK), known at the receiver node and used for the purpose of channel estimation. The length in symbols of the preamble sequence is L_{pre}. The payload part is denoted with $\boldsymbol{x_{pay}^k}$ and has a length equal to L_{pay} symbols.

2.4.3 Expectation-Maximisation algorithm for channel estimation

As previously stated, in recent slotted RA schemes, the maximum throughput is significantly enhanced on the RA channel and the degree of packet collisions is relatively high on all the timeslots of a frame. In such systems, the receiver is supposed to detect the users present on one timeslot by employing multi-user detection techniques (for instance, using the coded signalling field in MuSCA or ECRA). Then, the receiver performs channel estimation for the desired signal on each timeslot containing a replica or a packet fragment. In CRDSA, the receiver can perform channel estimation for a replica on a given timeslot, and usually this replica is free of collisions or in collision with one other packet (in equi-powered packets scenarios). Then, the receiver uses the retrieved payload data to estimate the channel parameters of its replicas on other timeslots. However, unlike CRDSA, in MuSCA and ECRA, the information necessary to decode a packet is needed from all its replicas or fragments received on separate timeslots. Therefore, channel estimation for a desired signal should be performed on each timeslot containing a replica (or a packet fragment), before demodulation and decoding, and in presence of different levels of packet collisions.

In this context, the Expectation-Maximisation (EM) algorithm presents a useful tool to estimate the channel parameters when receiving multiple signals in collision on the same timeslot. The EM algorithm is a two-step iterative estimation method proposed in [53] to perform maximum-likelihood estimation. In [22], this algorithm is applied for signal parameters estimation in the case of superimposed received signals. In [52], the EM algorithm is used for channel estimation applied to a Network Coded Diversity Protocol (NDCP). NDCP is presented in [52] as a scheme performing Physical-Layer Network Coding (PNC) with finite fields, also called Galois fields (GF). PNC [54] is a method to resolve signals collisions at the symbol level with the assumption of symbol, frequency and phase synchronism. First, the receiver tries to decode the bit-wise XOR of the received signals, then it can deduce the transmitted bits of each user. At the transmitter side, elements from extended GFs are used as coefficients to multiply different parts of a same packet and allow resolution of linear equations and packets recovery at the receiver. Of course, the effect of the channel on the received messages on each timeslot should be taken into account. Therefore, in [52], the EM algorithm is applied for channel estimation of superimposed signals.

In this case, the observed signal samples from which the channel parameters are estimated are the elements of the received signal vector $\boldsymbol{y}$ containing N_{coll} packets in collision. In a first step called “Expectation” (E-step), the preamble sequence part is extracted from the received signal $\boldsymbol{y}$ and the resulting discrete signal vector is denoted by $\boldsymbol{y_{pre}}$. Then, for each packet k, a vector $\boldsymbol{p_k}$ is computed as described in the following: the preamble sequence vector $\boldsymbol{x^k_{pre}}$ for each packet k, is multiplied by its estimated channel coefficient $\widehat{h_k}$ (computed in the previous iteration), and added to an arbitrary percentage (β_k) of the difference between the received training sequence $\boldsymbol{y_{pre}}$ and the sum of N_{coll} known training sequences multiplied by their estimated channels. The expression of the n^{th} element of the vector $\boldsymbol{p_k}$ at each E-step iteration (m) is shown below.

E-step at m^{th} iteration

$$p_k^{(m)}(n) = x^k_{pre}(n)\widehat{h_k}(n)^{(m-1)} + \beta_k \left[y_{pre}(n) - \sum_{l=1}^{N_{coll}} x^l_{pre}(n)\widehat{h_l}(n)^{(m-1)} \right] \quad . \tag{2.15}$$

In a second step called Maximisation (M-step), the channel parameters for each packet k ($\widehat{A_k}$, $\widehat{\Delta f_k}$ and $\widehat{\phi_k}$) are estimated at each m^{th} iteration. For this purpose, the Mean Square Error (MSE) is minimised, between the reconstructed vector $\boldsymbol{p_k^{(m)}}$ in the E-step of the same iteration, and a reconstructed sequence using the channel parameters to estimate (A', $\Delta f'$ and ϕ'). The operation done at the M-step is shown below.
M-step at m^{th} iteration

$$\left\{\widehat{A_k}, \widehat{\Delta f_k}, \widehat{\phi_k}\right\} = \underset{A', \Delta f', \phi'}{\operatorname{argmin}} \sum_{i=0}^{L_{pre}-1} \left| x^k_{pre}(n) p_k^{(m)}(n) - A' e^{j(2\pi\Delta f' nT + \phi')} \right|^2 \quad . \tag{2.16}$$

Limitations of the presented EM approach in [52] for the NDCP scheme

One of the limitations of this EM approach used in [52] is that only preambles at the beginning of each packet are used in the estimation algorithm. However, using grouped training symbols only the beginning and the end of a packet, makes it difficult to estimate the variable parameters such as the phase of the signal. Especially that the frequency offset is supposed to vary between 0 and 1% of the symbol rate. Therefore, in order to track the phase of a signal before it turns around 2π rd, it is important to estimate the channel at the beginning as well as the middle and the end of a packet duration. Moreover, the channel parameters for the EM algorithm applied in [52] are initialised randomly at the first iteration $m = 0$. Nevertheless, random initialisation of the EM algorithm has been proved to be inaccurate in [55], and to affect on the convergence of the algorithm and the correctness of the estimated values. Therefore, it is important to find a parameters initialisation method in order to avoid the divergence of the results from the desired values.

2.4.4 Other data-aided channel estimation techniques

As explained previously, the EM algorithm enables to estimate jointly the channel parameters of several superimposed signals. In the following, we give a brief presentation on data-aided (DA) techniques used to estimate the channel parameters for each user separately. DA estimators enable to estimate the channel parameters by computing the correlation between the received signal and the known data at the receiver (i.e., known training symbols like the preamble and the postamble symbols). In [51], correlation based DA estimators were used to derive the channel parameters of the desired signal using known training sequences in a first step. In a second step, the payload data acquired from a given replica is used to refine the first channel estimation and remove the other replicas from the frame.

Before channel estimation, the received signal vector $\tilde{\boldsymbol{y}}$ is matched-filtered and the resulting discrete signal is denoted by $\tilde{\boldsymbol{y}}$. Many Carrier Frequency Offset (CFO) estimators have been described in the literature from which we can cite [56, 57]. Following [56], for each user k, the CFO can be estimated using the preamble sequence of the k^{th} user, denoted by $\boldsymbol{x^k_{pre}}$. An example for performing preamble data-aided CFO estimation for a user k is shown in Equation (2.17). The frequency offset estimate $\widehat{\Delta f}_k$ is the value that maximises the square amplitude of the cross-correlation between the preamble part of the received signal $\tilde{\boldsymbol{y}}_{\boldsymbol{pre}}$ and the known preamble sequence of the k^{th} user $\boldsymbol{x^k_{pre}}$, multiplied by the exponential of a tentative value Δf.

$$\widehat{\Delta f}_k = \underset{\Delta f}{\operatorname{argmax}} \left| \sum_{m=-(QL-1)}^{m=QL-1} \tilde{y}_{pre}(i) \left(x^k_{pre}(i-m) \right)^* e^{-j2\pi \Delta f i T_s} \right|^2 . \tag{2.17}$$

As for the timing offset estimation for each user k, it can be done following algorithms like [58], [59], or [60]. Following [60] and using the preamble sequence and the estimated CFO of each user, the timing offset estimate $\widehat{\tau}_k$ for the k^{th} user can be computed as shown in the following equation:

$$\widehat{\tau}_k = \underset{i}{\operatorname{argmax}} \left| \sum_{m=-(QL-1)}^{m=QL-1} \tilde{y}_{pre}(i) \left(x^k_{pre}(i-m) \right)^* e^{-j2\pi \widehat{\Delta f}_k i T_s} \right|^2 . \tag{2.18}$$

The phase shift estimate $\widehat{\phi}_k$ can be derived based on [61] as the angle of the peak correlation obtained in Equation (2.18). In the next chapter, we will propose an improved channel estimation technique for RA in satellite communications, combining the EM algorithm with the correlation-based data aided techniques presented above.

2.4.5 Cramer-Rao Lower Bounds

In order to analyse the effectiveness of any estimation algorithm, it is of interest to compute the corresponding Cramer-Rao Lower Bounds (CRLBs). In estimation theory, the CRLBs define theoretical lower bounds on the variance of estimation errors of deterministic parameters. In the following, we present a brief summary of the literature related to the CRLBs.

Brief Literature for one parameter case

Let us consider observed measurements of a random variable x distributed according to a Probability density Function (PDF) $f(x;\theta)$, with θ being an unknown parameter. $f(x;\theta)$ also represents the likelihood function between x and θ. According to the literature [62, 63], the variance of the estimated value $\hat{\theta}$ can always be bounded by the inverse of the Fisher information $I(\theta)$ [64]. Fisher information is defined as the amount of information provided by the random variable x about θ. It is denoted by $I(\theta)$ and expressed as shown below:

$$I(\theta) = E\left[l'(x|\theta)^2\right] = \int_x \left[l'(x|\theta)\right]^2 f(x;\theta)dx \quad , \tag{2.19}$$

with $l'(x|\theta)$ being the derivative of the log-likelihood function $l(x|\theta) = log\left(f(x;\theta)\right)$ and (log) being the natural logarithm operator. $I(\theta)$ can be also computed as:

$$I(\theta) = E\left[l'(x|\theta)^2\right] = var_\theta\left[l'(x|\theta)\right] \quad . \tag{2.20}$$

To demonstrate the result shown in Equation (2.20), $l'(x|\theta)$ can be expressed as:

$$l'(x|\theta) = \frac{\partial\left(log\left(f(x;\theta)\right)\right)}{\partial\theta} = \frac{f'(x;\theta)}{f(x;\theta)} \quad , \tag{2.21}$$

and $E\left[l'(x|\theta)\right]$ can be expressed as:

$$E\left[l'(x|\theta)\right] = \int_x l'(x|\theta)f(x;\theta)dx = \int_x f'(x;\theta)dx = \frac{\partial}{\partial\theta}\int_x f(x;\theta)dx = 0 \quad . \tag{2.22}$$

If the observed measurements constitute a vector $\boldsymbol{X} = [x_1, x_2, \cdots, x_n]$ containing n samples following the same PDF, then the Fisher Information in this random sample of size n is equal to n times the Fisher information in a single observation: $I_n(\theta) = nI(\theta)$. The joint PDF of $\boldsymbol{X}$ is denoted by $f_n(\boldsymbol{x};\theta)$.

Now that we gave a brief summary on the Fisher information and its role in describing the amount of information contained in observed random samples about a given parameter, we can explain the derivation of the CRLBs as follows. Suppose that we want to estimate the parameter θ based on the observed samples $\boldsymbol{X} = [x_1, x_2, \cdots, x_n]$. Let $\hat{\theta} = r(x_1, x_2, \cdots, x_n)$ be an arbitrary estimator of θ, with a mean $E\left[\hat{\theta}\right] = m(\theta)$. By deriving the covariance between $\hat{\theta}$ and $l'_n(\boldsymbol{x}|\theta)$ and using the Cauchy-Schwartz inequality, it is demonstrated that the lower bound of variance of an arbitrary estimator $\hat{\theta}$ is:

$$var\left(\hat{\theta}\right) \geq \frac{\left[m'(\theta)\right]^2}{nI(\theta)} \quad . \tag{2.23}$$

This inequality is called the Cramer-Rao inequality. If $\hat{\theta}$ is an unbiased estimator then $m(\theta) = \theta$ and $m'(\theta) = 1$. Thus, the CRLB can be simplified to $\frac{1}{nI(\theta)}$.

Brief literature for multiple parameters case

In this case, multiple parameters have to be estimated from an observed random variable x, i.e x follows a PDF $f(x;\boldsymbol{\theta})$ with $\boldsymbol{\theta} = (\theta_1, \cdots, \theta_k)^T$ being the parameters vector to estimate. The log-likelihood function is a vector denoted by $l(\boldsymbol{\theta}) = log f(x|\boldsymbol{\theta})$. Its first order derivative is a vector of length k expressed as:

$$\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} = \left(\frac{\partial l(\boldsymbol{\theta})}{\partial \theta_1}, \cdots, \frac{\partial l(\boldsymbol{\theta})}{\partial \theta_k}\right)^T \quad , \tag{2.24}$$

with $(.)^T$ being the transpose operator. The second order derivative is a $k \times k$ matrix:

$$\frac{\partial^2 l(\boldsymbol{\theta})}{\partial \theta^2} = \left[\frac{\partial^2 l(\boldsymbol{\theta})}{\partial \theta_i \partial \theta_j}\right]_{i,j=1,\cdots,k} \quad . \tag{2.25}$$

The Fisher information matrix in this case is a symmetric matrix and it is written as:

$$I(\boldsymbol{\theta}) = E\left[\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\left(\frac{\partial l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^T\right] \quad . \tag{2.26}$$

2.5 Summary & conclusion

The use of MAC protocols in multi-user communication systems is a necessity to organise the access to the resources. TDMA protocols used with DAMA ensure reliability in the delivery of the information transmitted by each user without collisions. At the same time, this could be time consuming and inefficient in certain types of communication scenarios. Such communication scenarios are characterised with sporadic traffic, very short packets transmissions and long idle periods. In this context, contention protocols present a more flexible solution. However, this solution induces a higher probability of packet losses due to collisions.

The DVB-RCS/RCS2 standards for the return link of satellite networks, use MF-TDMA. Dedicated access and random access are used on the return link channels. These access techniques are chosen depending on the communication scenarios (i.e., service required, application, type of traffic, etc...). In particular, in low traffic regimes, RA methods used for data transmissions on the return link, can lead to lower end-to-end delays. However, the problem with legacy RA protocols for satellite communications is the high packet collisions rate. Given the high propagation delays nature of satellite networks, packet losses and retransmissions must be avoided. Therefore, the need for new RA protocols that can cope with packets collisions has presented a huge motivation for many researchers. Recent RA methods proposed in the literature have been in competition, in order to provide the highest throughput and efficiency with the lowest delays and packet losses. Most of these recent protocols rely on one essential principle at the receiver: successive interference cancellation.

Significant performance enhancements have been acquired. Nevertheless, practical issues such as channel estimation, correct interference cancellation, signals detection and many others are raised with the recently proposed RA protocols. *What are the possible solutions to deal with such problems?* And *is there a way to further boost the performance of the existing RA protocols and make them more reliable for data transmissions?* Such questions can be concluded from this chapter and will be answered in the following chapters of this thesis.

3

Channel Estimation for Recent RA Methods

As stated in the previous chapter, recent RA methods based on SIC can significantly increase the throughput and reduce packet losses on the return link. It is often assumed that perfect knowledge of the interference is available at the receiver. However, in practice, the interference term has to be accurately estimated to avoid performance degradation. As pointed out in the previous chapter, several channel estimation techniques have been proposed lately in the literature for the case of superimposed signals. In this chapter, we first detail the performance issues resulting from inaccurate channel estimation in recent RA schemes. Then, we propose enhancements to a channel estimation algorithm based on Expectation-Maximisation, by resorting to: autocorrelation for parameters initialisation, pilot symbols assisted modulation, and joint estimation and decoding. We compare the mean square estimation errors obtained to the Cramer-Rao theoretical lower bounds. Finally, we evaluate via simulations, the impact of the proposed channel estimation algorithm on the packet error rate. Final results show that the performance loss after imperfect interference cancellation using the proposed channel estimation method, does not exceed 0.3 dB. The work presented in this chapter has been already published in IEEE proceedings [65] and [66].

3.1 Introduction

3.1.1 Problem statement

In every communication system, channel estimation is a required step at the receiver side. Depending on the communication scenario, the channel between the transmitter and the receiver causes variations of the transmitted signal's amplitude, phase and frequency. Therefore, at the receiver side, channel estimation should be performed before demodulation and decoding. Otherwise, the decoding process would not take into account the channel impairments and the resulting decoded information would be erroneous. A basic general illustration of a block diagram showing the main operations performed at the receiver side after signal reception, is given in Figure 3.1.

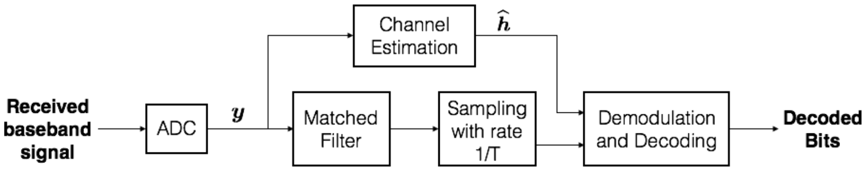


Figure 3.1 – Main operations performed at the receiver side upon reception of a transmitted signal.

The vector $\boldsymbol{y}$ represents the received signal at the output of the Analog to Digital Converter (ADC), oversampled at a rate $\frac{1}{T_s} = \frac{Q}{T}$ (T being the symbol period and Q the oversampling factor). $\boldsymbol{y}$ can be expressed as: $\boldsymbol{y} = \boldsymbol{hx} + \boldsymbol{n}$, where $\boldsymbol{h}$ is the channel coefficient vector, $\boldsymbol{x}$ is the transmitted discrete signal sequence and $\boldsymbol{n}$ is the vector representing the AWGN term. After computing the estimated channel coefficient vector $\widehat{\boldsymbol{h}}$, the received discrete signal is matched filtered and sampled at the symbol period T. Then, demodulation and decoding are performed. In this context, the accuracy of the channel estimation algorithm plays an important role in correct detection of the transmitted symbols. In other words, the channel parameters should be estimated for each user accurately in order to correctly detect and demodulate its transmitted symbols and successfully decode the useful information.

In addition to this problem, in recent RA methods based on SIC, inaccurate channel estimation leads to imperfect interference cancellation. In this case, channel estimation should be accurate not only to achieve correct demodulation, but also to perform accurate interference cancellation and leave minimal residual errors on the remaining signals. Given that recent RA protocols apply the principle of SIC, correct interference cancellation requires accurate channel estimation for the interference signal to be removed. Unless the estimated channel coefficient is perfectly representative of the real channel conditions, 'residual estimation errors' will appear after removing the interference signal from its corresponding timeslot.

Therefore, the problem to be addressed in this chapter is the impact of residual channel estimation errors on synchronous RA methods with SIC. In order to illustrate this problem, let us consider the example in Figure 3.2 showing 5 timeslots of a frame shared between 3 users using a recent slotted RA scheme. In the case of the CRDSA RA scheme, the notations (a) and (b) refer to the first and the second replica of a same packet. However, if the RA scheme used is MuSCA, then the same notations correspond to fragments of the same packet. It is observed in Figure 3.2a that the packets of user 1 and user 2 are in collision on slot 1. We suppose that the receiver decodes the packet (2b) successfully on timeslot 4 and needs to remove the signals corresponding to (2a) and (2b) from timeslots 1 and 4, respectively. If the receiver has perfect knowledge of the Channel State Information (CSI) for user 2 on timeslots 1 and 4, the resulting timeslot after interference cancellation is shown in Figure 3.2b. Thus, the packet (1a) becomes free of collisions and its probability for successful decoding increases. However, in a real system, the receiver does not have perfect CSI knowledge, and the channel coefficients of user 2 on timeslots 1 and 4 should be estimated before interference cancellation. In this case, the resulting timeslot after interference cancellation is shown in Figure 3.2c. Residual channel estimation errors ($e_2 = h_2 - \widehat{h}_2$) are added to the packet (1a), with h_2 and $\widehat{h}_2$ being respectively the real and estimated channel coefficient of user 2 on timeslot 1. Therefore, the quality of channel estimation for user 2 on timeslot 1 can affect on the demodulation performance of other remaining packets transmitted on the same timeslot.

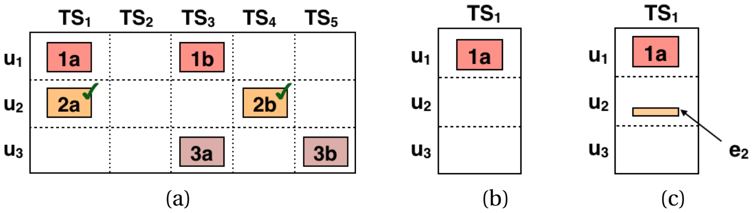


Figure 3.2 – Example showing perfect and imperfect interference cancellation on a frame. $N_u = 3$ users (u) and $N_s = 5$ timeslots (TS).

3.1.2 System assumptions

We consider a timeslot denoted by (TS) received on the uplink frame in a communication scenario with fixed user terminals. We suppose that TS contains K packets in collision, transmitted by K different users. The channel coefficient for each user on (TS) is denoted by h_k and has the same parameters defined in Equation (2.14) of the previous chapter, with k being an integer varying in $[1, K]$. We recall from Equation (2.14) that the channel coefficient h_k for the k^{th} user is expressed as $h_k(i) = A_k e^{j(2\pi\Delta f_k i T_s + \phi_k)}$, with A_k being a log-normally distributed random variable modelling the channel amplitude, Δf_k being the frequency offset uniformly distributed in $[0, \Delta f_{max}]$ ($\Delta f_{max} = 1\%$ of the symbol rate) and ϕ_k referring to the phase shift following a uniform distribution in $[0, 2\pi]$. In a first stage, we will consider that the users are synchronised at the symbol level, so that timing offset τ_k can be neglected. Later in this chapter, the timing offset will be taken into consideration in the channel estimation algorithm. Guard intervals are used to delimit the beginning and the end of each packet, and training symbols (preamble and postamble) are added to each packet and used for the purpose of channel estimation. We suppose that the receiver can detect the users present on (TS) by resorting to multi-user detection techniques [67]. For example, in MuSCA the users can be detected on each timeslot using the coded signalling field on each packet. The received signal on (TS) after pulse shaping with a square-root raised cosine and oversampling is given by:

$$y(i) = \sum_{k=1}^{K} \underbrace{h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT)}_{s_k(i)} + w(i) \quad , \tag{3.1}$$

where:

- T and $T_s = \frac{T}{Q}$ are respectively the symbol period and the oversampling period;
- $i = 0, 1, \cdots, LQ - 1$ and $n = 0, 1, \cdots, L$ refer to T_s-spaced and T-spaced samples respectively, with L being the length of the entire packet in symbols;
- $x_k(n)$ refers to the n^{th} symbol sent by user k;
- g stands for the square root raised cosine pulse function used as a shaping filter;
- w is a discrete complex additive white Gaussian noise process of variance σ_w^2.

Due to prior decoding on other timeslots, we suppose that the receiver already knows the information transmitted by all the users on (TS) except the information corresponding to the signal $\boldsymbol{s_1}$ (see Figure3.3). Thus, the receiver knows the interference symbols sequences $\boldsymbol{x_2}$, $\boldsymbol{x_3}$,..., and $\boldsymbol{x_K}$. The goal is to be able to demodulate and decode the signal $\boldsymbol{s_1}$ in presence of the residual channel estimation errors left by the other signals after interference cancellation.

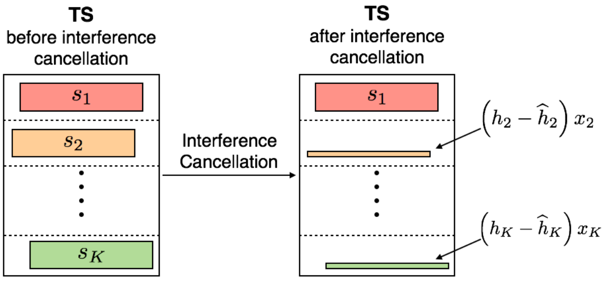

TS
before interference cancellation
s_1
s_2
s_K
Interference Cancellation
TS
after interference cancellation
s_1
$\left(h_2 - \widehat{h}_2\right) x_2$
$\left(h_K - \widehat{h}_K\right) x_K$

Figure 3.3 – Interference cancellation with residual channel estimation errors on one timeslot TS.

Therefore, the receiver needs to compute the channel estimates $\widehat{\boldsymbol{h_1}}$, $\widehat{\boldsymbol{h_2}}$, $\widehat{\boldsymbol{h_3}}$, $\cdots$, and $\widehat{\boldsymbol{h_K}}$, then suppress the interference signals from $\boldsymbol{y}$ in order to obtain the estimated discrete signal vector $\widehat{\boldsymbol{s_1}}$ as follows

$$\widehat{s}_1(i) = h_1(i) \sum_{n=0}^{L-1} x_1(n) g(iT_s - nT) + \sum_{k=2}^{K} \left[\left(h_k(i) - \widehat{h}_k(i)\right) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT) \right] + w(i) \tag{3.2}$$

After interference cancellation and in presence of residual estimation errors, the estimated signal $\widehat{\boldsymbol{s_1}}$ is matched filtered then sampled at optimum sampling instant with the symbol period T, and the resulting estimated symbols are expressed as

$$\widehat{s}_1(n) = h_1(n) x_1(n) + \sum_{k=2}^{K} \left[\left(h_k(n) - \widehat{h}_k(n)\right) x_k(n) \right] + w(n) \quad . \tag{3.3}$$

$\widehat{s}_1$ is demodulated using a log-MAP demodulator [68] then decoded and suppressed from the frame. In the following, we will present our proposed channel estimation algorithm, that allows to reduce the residual channel estimation errors $\left(\boldsymbol{h_k} - \widehat{\boldsymbol{h_k}}\right)$ and thus enhance the successful demodulation and decoding probability for the estimated symbols of user 1.

3.2 Proposed channel estimation algorithm

Our goal is to find an algorithm able to jointly estimate the channel coefficients $\widehat{\boldsymbol{h_1}}$, $\widehat{\boldsymbol{h}}_{\boldsymbol{2}}$, $\widehat{\boldsymbol{h}}_{\boldsymbol{3}}$, $\cdots$, and $\widehat{\boldsymbol{h_K}}$, given that the replicas of $\boldsymbol{s_1}$ are not yet decoded on any other timeslot. For this purpose, the EM algorithm is a good solution, since it enables to jointly estimate the amplitude, frequency offset and phase of several signals in collision on one timeslot. Our second objective in this section, is to propose enhancements to the EM channel estimator in order to reduce the impact of imperfect interference cancellation on the decoding of the remaining signal $\boldsymbol{s_1}$.

As explained in Section 2.4.3, the EM algorithm is a two-step iterative estimation method. In the literature [52], the EM algorithm has been applied on the preamble part of the signal and the channel parameters initialisation has been random. In order to take advantage of the impact of the communication channel on the transmitted signals at the beginning and the end of a timeslot, we propose to use the 'postamble' sequence at the end of each packet as shown in the packet structure in Figure 3.4. Similarly to the preamble, the postamble for each user k (denoted by the vector $\boldsymbol{x^k_{post}}$) is a unique orthogonal sequence of length L_{post} symbols, modulated with BPSK and known at the receiver. We use the term 'training sequence' to refer to the vector concatenating the preamble and the postamble parts of a packet. The notation $\boldsymbol{x^k_{tr}}$ corresponds to the known training symbols sequence of the k^{th} packet. The length in symbols of each training sequence is $L_{tr} = L_{pre} + L_{post}$.

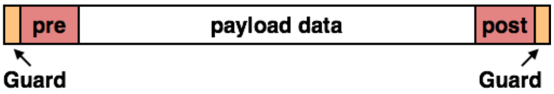


Figure 3.4 – Packet structure with preamble and postamble and guard intervals.

To enhance the channel estimation technique based on EM described in Section 2.4.3, we find it reasonable to apply the EM algorithm not only on the preamble symbols, but also on the postamble symbols. The observed samples correspond to the training sequence part extracted from the received signal $\boldsymbol{y}$ and they constitute the vector $\boldsymbol{y_{tr}}$. In the case of K colliding packets on one timeslot, the EM equations for each iteration 'm' become:

E-step: for $k = 1, ..., K$

$$p_k^{(m)}(n) = x_{tr}^k(n)\widehat{h}_k(n)^{(m-1)} + \beta_k \left[y_{tr}(n) - \sum_{l=1}^{K} x_{tr}^l(n)\widehat{h}_l(n)^{(m-1)} \right] \quad . \tag{3.4}$$

M-step: for $k = 1, ..., K$

$$\left\{\widehat{A}_k, \widehat{\Delta f}_k, \widehat{\phi}_k\right\} = \underset{A', \Delta f', \phi'}{\operatorname{argmin}} \sum_{n \in \Theta} \left| x_{tr}^k(n) p_k^{(m)}(n) - A' e^{j(2\pi \Delta f' nT + \phi')} \right|^2 \quad , \tag{3.5}$$

with Θ being the set of the preamble and postamble integer indexes inside a packet, $\Theta = \{0, \cdots, L_{pre} - 1, L_{pre} + L_{pay}, \cdots, L_{pre} + L_{pay} + L_{post} - 1\}$.

3.2.1 Combining EM with autocorrelation initialisation

Beside adding the postamble use to the EM algorithm, we also propose to initialise the channel parameters for the first iteration by applying an autocorrelation based technique. In fact, it has been shown in prior research [55], that random initialisation of the channel coefficients in the EM algorithm can lead to inaccurate results. In [55], the author focuses on the application of the EM algorithm within the model based clustering framework. Received data is coming from a mixture density and the goal is to be able to separate this data into subsets of density functions, with each subset having specific density function parameters. For this purpose, the author in [55] proposed to use the EM algorithm in order to estimate the density functions available in the observed data set. Inspired from the study in [55], we choose to compare the effectiveness of the EM channel estimation method using random initialisation as well as initialisation based on known symbols autocorrelation.

Initialisation by autocorrelation

For the first iteration of the EM algorithm ($m = 0$) and for each user k, we compute the initial parameters estimates of the amplitude $\widehat{A}_k^{(0)}$ and the phase $\widehat{\phi}_k^{(0)}$ with an autocorrelation-based method, using the known preamble sequences $\boldsymbol{x_{pre}^k}$ of each user k. The initial estimates of the channel amplitude $\widehat{A_k}^{(0)}$ and the phase shift $\widehat{\phi_k}^{(0)}$ corresponding to user k are derived as shown in Equations (3.6) below.

$$\widehat{A_k}^{(0)} = \frac{\left| \boldsymbol{y_{pre}} \times \left(\boldsymbol{x_{pre}^k}\right)^T \right|}{L_{pre}} \quad , \qquad \widehat{\phi_k}^{(0)} = \angle\left(\boldsymbol{y_{pre}} \times \left(\boldsymbol{x_{pre}^k}\right)^T \right) \quad , \tag{3.6}$$

with $(\times)$ referring to the vector multiplication operator, $(\cdot)^T$ denoting the transpose version of a vector and $(\angle)$ representing the angle symbol.

First results & observations

In the following we evaluate throughout simulations the results obtained with the proposed enhancement scheme for channel estimation based on postamble integration and autocorrelation initialisation for the EM algorithm. Therefore, we compute the Packet Error Rate (PER) obtained after interference cancellation, demodulation and decoding of the desired signal.

In the simulations we consider the scenario shown in Figure 3.2 on timeslot 1. The received signal $\boldsymbol{y}$ contains two signals in collision s_1 and s_2. Due to prior decoding, we suppose that the receiver knows the symbols constituting s_2. We perform channel estimation for both signals s_1 and s_2 on TS1, using the proposed EM algorithm with autocorrelation initialisation. Using the estimated channel coefficient $\widehat{h}_2$ and the known symbols of user 2, we reconstruct the signal s_2 and remove it from $\boldsymbol{y}$. Then, using the estimated channel coefficient $\widehat{h}_1$ of user 1, we demodulate and decode the remaining signal in $\boldsymbol{y}$ and we analyse the PER for user 1.

Without loss of generality, we assume that the channel gain A_k of user k is normalised to one. We use as preambles and postambles Walsh-Hadamard BPSK modulated words of length 80 and 48 symbols, respectively. The results are obtained with a CCSDS turbo code of rate $R = 1/2$ and a payload length equal to 460 symbols modulated with QPSK. The overhead added by the preamble and the postamble constitutes 27% of the payload length in symbols. This overhead percentage is taken approximately equal to the case of the waveform id 1 defined in the DVB-RCS2 guidelines [28]. The number of iterations needed to achieve convergence of the EM algorithm is 3. Figure 3.5 shows the PER in function of E_s/N_0 achieved with different channel estimation methods. Three simulation scenarios are compared: **(1)** a perfect CSI scenario, **(2)** a scenario using the EM algorithm applied only on preambles and postambles with random initialisation and **(3)** a scenario using the proposed channel estimation scheme applying EM on the preamble and the postamble combined with autocorrelation initialisation. In this particular scenario, two additional reference curves are considered: one with perfect knowledge of the frequency offset Δf, and another with perfect knowledge of the phase shift ϕ.

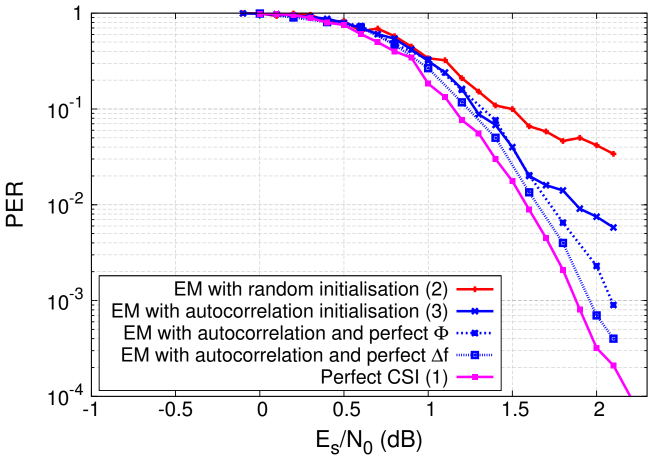


Figure 3.5 – PER vs. E_s/N_0 achieved after interference cancellation in different channel estimation scenarios: with random EM initialisation and with autocorrelation-based EM initialisation. Modcod used: CCSDS turbo code of rate 1/2 and QPSK. $L = 460$ symbols.

We can observe from Figure 3.5 that for the same E_s/N_0 ratio, channel estimation using EM combined with autocorrelation initialisation results in lower PER than the traditional EM with random initialisation (PER = $7*10^{-3}$ instead of PER = $5*10^{-2}$ for $E_s/N_0 = 2$ dB). However the degradation compared to the perfect CSI case starts increasing when $E_s/N_0 = 1.6$ dB. We can also observe that when Δf is assumed perfectly know, the performance is significantly enhanced. Thus, we can conclude that the estimation error on the frequency offset is the main reason for the performance degradation.

3.2.2 Estimation using pilot symbol assisted modulation

Figure 3.5 shows that a more accurate estimation of the frequency offset can lead to better PER results. To be coherent with the DVB-RCS/RCS2 standards in terms of channel assumptions, we suppose that the frequency offset Δf affecting a signal over the duration of a frame is a random variable uniformly distributed between 0 and 10^{-2} $(1/T_s)$. Given that the frequency offset causes the phase of the signal to vary linearly with time, then the orthogonal preamble and postamble sequences of different users can lose their orthogonality property. Therefore, it would be more useful to estimate the frequency offset on small sequences of orthogonal training symbols distributed inside the packet.

In this context, we propose to refine the estimation of Δf and to preserve the orthogonality of the training sequences, by using Pilot Symbol Assisted Modulation (PSAM) [23, 69] integrated with the EM algorithm. The idea is to divide the preamble and the postamble into smaller parts and to distribute them inside the data burst. In fact, PSAM consists in periodically inserting known symbols, called pilot symbols, inside the data sequence, as shown in the packet structure illustrated in Figure 3.6. The training symbols of a total length L_{tr}, are divided between shorter preamble and postamble and N_p pilot blocks of length L_p separated by M data symbols, each. Similarly to the preamble and the postamble, the pilot blocks are Walsh-Hadamard words modulated with BPSK. We will show that, by using PSAM jointly with the EM algorithm, the performance of channel estimation can be significantly enhanced.

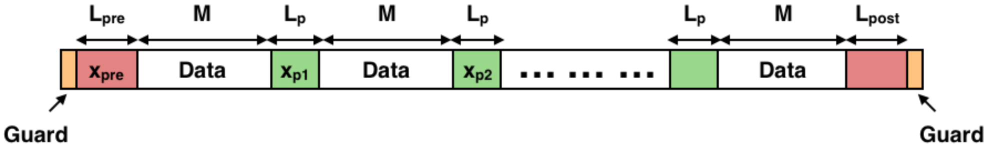


Figure 3.6 – Packet structure with preamble, pilots, postamble and guard intervals.

With PSAM, the EM algorithm is applied, not only on the preamble and postamble parts but also on the pilot blocks inside the packet. The set of training symbols indexes in Equation (3.5) becomes: $\Theta = \{0, \cdots, L_{pre}-1, \cdots, L_{pre}+M, \cdots, L_{pre}+M+L_p-1, \cdots, L_{pre}+2M+L_p, \cdots, L_{pre}+2M+2L_p-1, \cdots, L_{pre}+(N_p+1)M+N_pL_p, \cdots, L-1\}$ (we recall that L is the total length of a packet in symbols). The known training symbols sequence for each user k becomes: $\boldsymbol{x}_{tr}^{k} = \left[\boldsymbol{x}_{pre}^{k}, \boldsymbol{x}_{p_1}^{k}, \boldsymbol{x}_{p_2}^{k}, \cdots, \boldsymbol{x}_{p_{N_p}}^{k}, \boldsymbol{x}_{post}^{k}\right]$, with $\boldsymbol{x}_{p_i}^{k}$ being the i^{th} pilot sequence of the k^{th} packet. We propose to benefit from PSAM in order to also compute an estimate for the initial value of $\Delta f_k^{(0)}$ for each user k in the first iteration ($m = 0$) of the EM algorithm. $\Delta f_k^{(0)}$ can be computed as follows

$$\Delta f_k^{(0)} = \frac{\angle\left(\boldsymbol{y}_{\boldsymbol{pre}}^{\boldsymbol{k}} \times \left(\boldsymbol{x}_{\boldsymbol{pre}}^{\boldsymbol{k}}\right)^T\right) - \angle\left(\boldsymbol{y}_{\boldsymbol{p}_1}^{\boldsymbol{k}} \times \left(\boldsymbol{x}_{\boldsymbol{p}_1}^{\boldsymbol{k}}\right)^T\right)}{2\pi(L_p + M)} \quad , \tag{3.7}$$

with $\boldsymbol{y}_{\boldsymbol{p}_1}^{\boldsymbol{k}}$ being the part of the received signal containing the first pilot sequence.

Simulation results with PSAM

We evaluate, via simulations, the performance of the proposed channel estimation technique combining EM with autocorrelation initialisation and PSAM. The same simulation scenario described previously for Figure 3.5 is considered. For the PSAM configuration, we uniformly distribute $N_p = 9$ pilot blocks inside the data burst, each of length $L_p = 12$ symbols. We reduce the preamble and postamble lengths to $L_{pre} = 40$ symbols and $L_{post} = 12$ symbols, respectively. Thus, the total training symbols length with PSAM is $L_{tr_{PSAM}} = 160$. It is worth noting that using PSAM induces an additional overhead compared to the previous packet structure with only a preamble and a postamble. The additional overhead is $(L_{tr_{PSAM}} - L_{tr})/L = 7\%$.

In Figure 3.7, we compare the PER curves obtained using the proposed EM algorithm with and without PSAM. It is observed that with PSAM the PER floor at 10^{-2} is mitigated. The degradation in E_s/N_0 compared to perfect CSI is constant around 0.1 dB for PER $\in [10^{-1}, 10^{-4}]$.

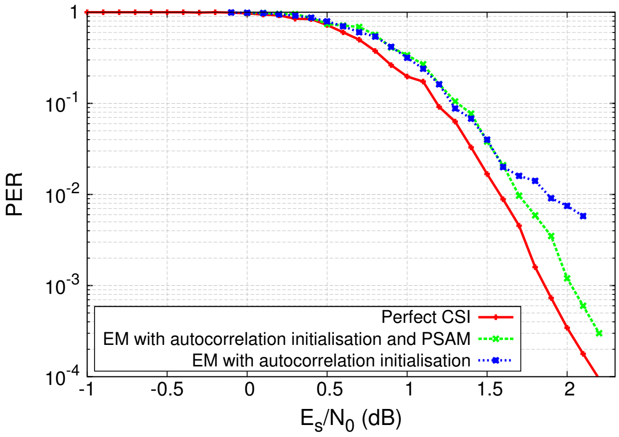


Figure 3.7 – PER vs. E_s/N_0 achieved after interference cancellation with and without PSAM. Modcod used: CCSDS turbo code of rate 1/2 and QPSK. $L = 460$ symbols.

3.2.3 Timing offsets consideration

At this stage of the chapter, we will consider that the timing offset τ_k for each packet of user k is different and not negligible. In other words, the signals in collision on one timeslot are not aligned at the symbol level. The objective is to study the performance of the EM algorithm for channel estimation when the received signals on one timeslot are misaligned. In this case, the received discrete signal $\boldsymbol{y}$ previously detailed in Equation (3.1) becomes:

$$y(i) = \sum_{k=1}^{K} \underbrace{h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT - \tau_k T)}_{s_k(i)} + w(i) \quad , \tag{3.8}$$

with τ_k referring to the timing offset relative to the signal $\boldsymbol{s_k}$ sent by user k. We suppose that τ_k is a discrete random variable uniformly distributed in $\left[0, \frac{1}{Q}, \frac{2}{Q}, \cdots, \frac{Q-1}{Q}\right]$. In this case, to remove the previously decoded signals in collision with s_1, the receiver shall not only compute the estimate of their channel coefficient $\widehat{\boldsymbol{h_k}}$ but also their corresponding timing offset $\widehat{\tau}_k$. The resulting estimated signal $\widehat{\boldsymbol{s}}_{\boldsymbol{1}}$ after interference cancellation, becomes:

$$\begin{aligned}\widehat{s}_1(i) &= h_1(i) \sum_{n=0}^{L-1} x_1(n) g(iT_s - nT - \tau_1 T) + w(i) \\ &+ \sum_{k=2}^{K} \left[h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT - \tau_k T) - \widehat{h}_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT - \widehat{\tau}_k T) \right] \quad . \end{aligned} \tag{3.9}$$

The receiver uses the estimated timing offset of user 1 in order to match filter it with the delayed filter $g\left(-(iT_s - \widehat{\tau}_1 T)\right)$ and then samples it at the symbol rate T. The problem is that when match-filtering according to $\widehat{\tau}_1$, the signals in $\boldsymbol{y}$ not corresponding to user 1 will contain Inter-Symbol Interference and will distort the results of the maximisation step of the EM. As shown in the equation below, the resulting estimated symbols are:

$$\begin{aligned}\widehat{s}_1(n) &= h_1(n) \sum_{n=0}^{L-1} x_1(n) q\left(nT + (\tau_1 - \widehat{\tau}_1)T\right) + w(T) \\ &+ \sum_{k=2}^{K} \left[h_k(i) \sum_{n=0}^{L-1} x_k(n) q(nT + (\tau_k - \widehat{\tau}_1)T) - \widehat{h}_k(i) \sum_{n=0}^{L-1} x_k(n) q(nT + (\widehat{\tau}_k - \widehat{\tau}_1)T) \right] \quad , \end{aligned} \tag{3.10}$$

with q being the raised cosine function resulting from the convolution $g(t) \otimes g^*(-t)$. For simplicity reasons, we suppose that the timing offset of user 1 is the reference timing, i.e., $\tau_1 = 0$, and the timing offsets for the other users are relative to τ_1. To obtain coherent channel estimation results, the estimation of timing offsets for each user shall be taken into account in the EM algorithm. As stated previously in this section, the problem is that if the received signal for each user is sampled at inaccurate sampling times, the samples relative to each user will be attenuated and Inter-Symbol Interference (ISI) terms will be added. For this purpose, we propose integrating timing offset estimation in the EM iterations as explained in the following.

Timing offset estimation

Our approach proposes to estimate the timing offset for each user at each iteration 'm' of the EM algorithm. Because of the unknown frequency offsets at the first EM iteration ($m = 0$), the timing offsets are estimated using only the preamble part of the received signal. In other words, in the first EM iteration, we compute the cross-correlation between the received signal preamble part $\boldsymbol{y_{pre}}$ and the preamble sequence $\boldsymbol{x_{pre}^{k}}$ of user k shaped filtered with the filter g. The initial timing offset estimate $\widehat{\tau_k}^{(0)}$ is computed as the time that maximises the square amplitude of the cross-correlation function. It can be expressed as follows:

$$\widehat{\tau_k}^{(0)} = \underset{\tau}{\text{argmax}} \left| \sum_{i=0}^{QL_{pre}-1} y_{pre}(i) \sum_{n=0}^{L-1} \left(x_{pre}^{k}(i) g(iT_s - nT - \tau T) \right)^{*} \right|^{2} \quad . \tag{3.11}$$

For the iteration $m > 0$, we compensate the impact of the frequency offset $\widehat{\Delta f}_k^{(m-1)}$ derived from the previous iteration $(m-1)$. Then, we re-compute the timing offset $\widehat{\tau_k}^{(m)}$ of user k based on the cross-correlation between the training symbols part of the received signal $\boldsymbol{y_{tr}}$ and the training sequence $\boldsymbol{x_{tr}^{k}}$ of user k shaped filtered with the filter function g. The estimation of $\widehat{\tau_k}^{(m)}$ at the iteration $m > 0$ is expressed as follows:

$$\widehat{\tau_k}^{(m)} = \underset{\tau}{\text{argmax}} \left| \sum_{i=0}^{QL_{tr}-1} y_{tr}(i) e^{-j2\pi\widehat{\Delta f_k}^{(m-1)} iT_s} \sum_{n=0}^{L-1} \left(x_{tr}^{k}(i) g(iT_s - nT - \tau T) \right)^{*} \right|^{2} \quad . \tag{3.12}$$

Channel parameters estimation

At each iteration 'm' of the EM algorithm, and for each user k, we first estimate the timing offset $\widehat{\tau_k}^{(m)}$. Then we enter the received signal $\boldsymbol{y}$ into a delayed matched filter $g\left(-(iT_s - \widehat{\tau_k}^{(m)} T)\right)$ and we sample it at the symbol rate. The resulting discrete signal is denoted by $\boldsymbol{y^{(m)}_{\widehat{\tau_k}}}$. This signal is used instead of $\boldsymbol{y}$ in the following 'E' and 'M' steps of the EM algorithm.

For the initial iteration, the channel parameters are estimated using cross-correlation with the preamble symbols in a similar way as shown in Equations (3.6). However, the difference here is that the preamble part of the discrete signal $\boldsymbol{y^{(0)}_{\widehat{\tau_k}}}$ denoted by $\boldsymbol{y^{(0)}_{\widehat{\tau_k},pre}}$ (obtained after timing offset estimation) is used instead of $\boldsymbol{y_{pre}}$ (before timing estimation). After channel parameters initialisation, for each iteration 'm', and for each user k, we go through the following steps:

1. We estimate the timing offset $\widehat{\tau_k}^{(m)}$ as shown in Equation (8.11) and we derive the discrete signal $\boldsymbol{y^{(m)}_{\widehat{\tau_k}}}$;
2. **E-step**

$$p_k^{(m)}(n) = x_{tr}^{k}(n)\widehat{h}_k(n)^{(m-1)} + \beta_k \left[y^{(m)}_{\widehat{\tau}_k,tr}(n) - \sum_{l=1}^{K} \widehat{h}_l(n)^{(m-1)} \sum_{n=0}^{L-1} x_{tr}^{l}(n) q(nT + (\widehat{\tau}_k - \widehat{\tau}_l)T) \right] \quad , \tag{3.13}$$

with $\boldsymbol{y^{(m)}_{\widehat{\tau_k},tr}}$ being the training sequence part of the discrete signal $\boldsymbol{y^{(m)}_{\widehat{\tau_k}}}$.

3. **M-step**

$$\left\{\widehat{A_k}, \widehat{\Delta f_k}, \widehat{\phi_k}\right\} = \underset{A',\Delta f',\phi'}{\text{argmin}} \sum_{n\in\Theta} \left| x_{tr}^{k}(n) p_k^{(m)}(n) - A' e^{j(2\pi\Delta f' nT + \phi')} \right|^{2} \quad . \tag{3.14}$$

3.2.4 Joint estimation & decoding approach

Timing offsets among the received signals on a given timeslot have two main consequences: on the one hand, the signal samples for each user are attenuated due to imperfect sampling instants. On the other hand, Inter-Symbol Interference (ISI) terms are added to the interference signals and the noise term. Thus, the performance of the EM algorithm for channel estimation is degraded. Given that the EM-based channel estimation relies on the knowledge of the training symbols for each user, this estimation can be enhanced by introducing longer training sequences. The problem is that longer training sequences induce larger overhead on the useful information transmitted. For this reason, we propose another solution inspired from the existing literature [24, 70, 71, 72] and based on a Joint Estimation and Decoding (JED) approach. Our contribution is the integration of JED with the EM algorithm for the context of iterative interference cancellation in recent RA methods.

It is explained in [24, 70, 71] that after performing channel estimation for a particular signal, based on the known training symbols, JED enables to enhance the channel estimates by using a feedback from the decoder output. In other words, the channel parameters are estimated using training symbols in a first step. Then the received signal is demodulated and decoded. The resulting bits (even erroneous bits) are fed back to the channel estimator, which then produces a better channel estimate, due to the longer known symbols sequence. This process is iterative until good channel estimation performance is achieved. Given that the output of the decoder used in the iterative estimation process is a sequence of decoded bits, we call it a 'Hard Decision' feedback. Other alternatives in the literature [70, 71] proposed to use the soft values at the output of the decoder rather than the hard values, in this case the scheme is called 'Soft-Decision' feedback. However, in this work, we will only study the performance of JED with hard-decision feedback on the recent RA techniques.

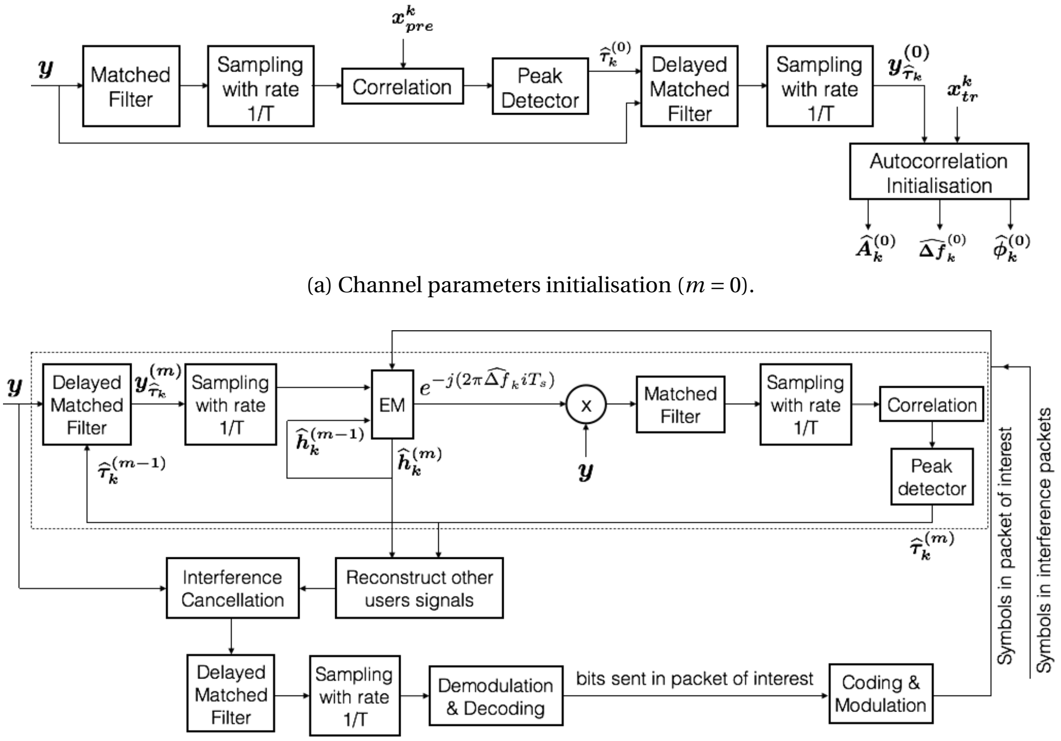
In the context of RA methods based on SIC, we propose to apply the proposed channel estimation algorithm based on EM and combine it with iterative JED, as explained below:

1. In the first JED iteration, the channel parameters are estimated using EM (with 3 iterations) with autocorrelation initialisation and PSAM;
2. Using the resulting channel estimates, the interference signals are reconstructed and removed from the actual timeslot. Practically, this first estimation of the channel parameters is not perfect and residual channel estimation errors are added to the signal of interest;
3. Despite the presence of residual channel estimation errors, the signal of interest is sent to the decoder;
4. If the decoding is not successful, the resulting decoded bits, although not all correct, are fed back to the channel estimator in the next iteration;
5. At this stage, the channel estimator uses the decoded bits to reconstruct the symbol sequence of the signal of interest. Then this symbols sequence is used in the channel estimation algorithm instead of the training symbols. The resulting channel parameters

are used to re-attempt demodulation and decoding of the signal of interest.

6. The receiver repeats steps 2-5 using not only the training symbols, but also the decoded payload data, making the channel parameters estimation more accurate.

The overall channel estimation scheme proposed combining EM with autocorrelation initialisation and PSAM as well as JED is illustrated in Figure 3.8. First, the parameters initialisation procedure is detailed in Figure 3.8a. Then, The operations performed for channel estimation at the m^{th} iteration are shown in Figure 3.8b. The simulation results in terms of PER obtained with JED and timing offsets are presented in Section 3.4.



(a) Channel parameters initialisation ($m = 0$).

(b) Channel parameters estimation ($m > 0$).

Figure 3.8 – Overall channel estimation scheme proposed combining EM with autocorrelation initialisation and PSAM as well as JED.

3.3 Mean Square Errors validation with Cramer-Rao Lower Bounds

In this section, we compute via Monte Carlo simulations the Mean Square Errors (MSEs) between the estimated and real channel parameters. The estimation technique used is the one described in the previous section: Estimation of amplitude, frequency offset, phase shift and timing offset based on the EM algorithm with PSAM and autocorrelation initialisation, combined with joint estimation and decoding. We first compute the Cramer-Rao Lower Bounds (CRLBs) for joint estimation of multiple channel parameters. A brief overview on the CRLBs was presented in Chapter 2. The CRLBs are theoretical bounds derived in the literature to mark out the minimum MSEs that can be achieved with any estimation technique. Then, we derive the MSEs of the estimated channel parameters and compare them to the CRLBs.

3.3.1 CRLBs for joint estimation of multiple channel parameters

In the context of this chapter, the CRLBs shall be derived for channel parameters estimated based on the training symbols for each user. In prior research [73], the CRLBs for joint estimation of multiple channel impairments have been derived for an Amplify and Forward (AF) two-way relaying network. In this section, we use a similar approach as the one described in [73] in order to derive the CRLBs for joint estimation of amplitudes, frequency offsets, phase shifts and timing offsets of multiple signals in collision on one timeslot. We express the received training part of the discrete signal on one timeslot in its column vector form, as shown below:

$$\boldsymbol{y} = \boldsymbol{\Omega}\boldsymbol{\alpha} + \boldsymbol{W} \quad , \tag{3.15}$$

where $\boldsymbol{\Omega}$, $\boldsymbol{\alpha}$ and $\boldsymbol{W}$ are defined as follows:

- $\boldsymbol{\Omega} = \left[\boldsymbol{\Lambda_1 G_1 x_{tr}^1}, \cdots, \boldsymbol{\Lambda_K G_K x_{tr}^K}\right]$ is an $L_{tr}Q \times K$ matrix, with

  $$\boldsymbol{\Lambda_k} = \text{diag}\left(\left[e^{j(2\pi\Delta f_k(0)T_s)}, \cdots, e^{j(2\pi\Delta f_k(LQ-1)T_s)}\right]\right)$$

  being an $L_{tr}Q \times L_{tr}Q$ diagonal matrix relative to the k^{th} user. And $\boldsymbol{G_k}$ being an $L_{tr}Q \times L_{tr}$ matrix defining the pulse shaping square root raised cosine filter such that $[\boldsymbol{G_k}]_{i,n} = g(iT_s + nT + \tau_k T)$;
- $\boldsymbol{\alpha} = [\alpha_1, \cdots, \alpha_K]^T$ is a column vector of length K, with $\alpha_k = A_k e^{j\phi_k}$;
- $\boldsymbol{W}$ is the complex noise column vector of length QL_{tr}.

Following [73], the discrete signal $\boldsymbol{y}$ can be approximated to a circularly symmetric complex Gaussian random vector with mean $\boldsymbol{\mu} = \boldsymbol{\Omega\alpha}$ and a covariance matrix $\boldsymbol{C} = \sigma_w^2 \boldsymbol{I}_{LQ}$ with $\boldsymbol{I}_{LQ}$ being the identity matrix of size $(LQ \times LQ)$. We suppose that the timing offsets are estimated separately in an earlier stage than the rest of the channel parameters. Therefore, the column parameter vector $\boldsymbol{\theta}$ of length $(3K)$, corresponding to the channel parameters to estimate is expressed as follows:

$$\boldsymbol{\theta} = \left[\Re\{\boldsymbol{\alpha}\}^T, \Im\{\boldsymbol{\alpha}\}^T, \Delta f_1, \cdots, \Delta f_K\right] \quad . \tag{3.16}$$

Based on [62], the discrete signal $\boldsymbol{y}$ belongs to the complex Gaussian case: $\boldsymbol{y} \sim \mathcal{N}\left(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{C}(\boldsymbol{\theta})\right)$, with both the mean and the covariance depending on $\boldsymbol{\theta}$. Then, as detailed in Appendix 15C of [62], the $(3K \times 3K)$ complex Fisher information matrix for a complex Gaussian PDF is given by:

$$I(\boldsymbol{\theta})_{l,q} = 2\Re\left[\frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\theta})}{\partial \theta_l} \boldsymbol{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_q}\right] + tr\left[\boldsymbol{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_l} \boldsymbol{C}^{-1}(\boldsymbol{\theta}) \frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_q}\right] \quad , \tag{3.17}$$

where $tr(.)$ refers to the trace operator and the superscript $(.)^H$ denotes the conjugate transpose operator. Given that $\boldsymbol{C} = \sigma_w^2 \boldsymbol{I}_{LQ}$, then $\frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_l} = \frac{\partial \boldsymbol{C}(\boldsymbol{\theta})}{\partial \theta_q} = 0$. Therefore, Fisher's information matrix can be written as follows:

$$I(\theta)_{l,q} = \frac{2}{\sigma_w^2} \Re \left[\frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\theta})}{\partial \theta_l} \frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_q} \right] = \frac{2}{\sigma_w^2} \left[\Re \left(\frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\theta})}{\partial \theta_l} \right) \Re \left(\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_q} \right) - \Im \left(\frac{\partial \boldsymbol{\mu}^H(\boldsymbol{\theta})}{\partial \theta_l} \right) \Im \left(\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \theta_q} \right) \right] \quad . \tag{3.18}$$

Thus, the final expression of the Fisher information matrix for the parameters vector $\boldsymbol{\theta}$ is expressed below:

$$I(\boldsymbol{\theta}) = \frac{2}{\sigma_w^2} \begin{bmatrix} \Re\left(\boldsymbol{\Omega}^H \boldsymbol{\Omega}\right) & -\Im\left(\boldsymbol{\Omega}^H \boldsymbol{\Omega}\right) & -\Im\left(\boldsymbol{\Omega}^H \boldsymbol{D}\boldsymbol{\Phi}\boldsymbol{x_{tr}}\right) \\ \Im\left(\boldsymbol{\Omega}^H \boldsymbol{\Omega}\right) & \Re\left(\boldsymbol{\Omega}^H \boldsymbol{\Omega}\right) & \Re\left(\boldsymbol{\Omega}^H \boldsymbol{D}\boldsymbol{\Phi}\boldsymbol{x_{tr}}\right) \\ \Im\left(\boldsymbol{x}_{\boldsymbol{tr}}^H \boldsymbol{\Phi}^H \boldsymbol{D}\boldsymbol{\Omega}\right) & \Im\left(\boldsymbol{x}_{\boldsymbol{tr}}^H \boldsymbol{\Phi}^H \boldsymbol{D}\boldsymbol{\Omega}\right) & \Re\left(\boldsymbol{x}_{\boldsymbol{tr}}^H \boldsymbol{\Phi}^H \boldsymbol{D}^2 \boldsymbol{\Phi}\boldsymbol{x_{tr}}\right) \end{bmatrix} \quad . \tag{3.19}$$

The vector of the CRLBs for the estimation of $\boldsymbol{\theta}$ is the vector containing the diagonal elements of the inverse of the information matrix I. It should be noted that the CRLB for the estimation of $\boldsymbol{\alpha}$ is the sum of the CRLBs for the estimation of the real and imaginary parts of $\boldsymbol{\alpha}$.

3.3.2 Channel estimation MSEs

In order to compare the performance of the actual estimator with the derived CRLBs, we compute the MSEs for the estimated channel parameters of each user based on simulations results and using the previously proposed estimation algorithm. For each user k, the MSEs (e_{α_k}) and $(e_{\Delta f_k})$ for the estimation of α_k and Δf_k are computed as shown in Equations (3.20) and (3.21), respectively.

$$\text{MSE}\left(\alpha_k\right) = E\left[\left|e_{\alpha_k}\right|^2\right] = E\left[\left|A_k e^{j\phi_k} - \hat{A}_k e^{j\hat{\phi}_k}\right|^2\right] \quad , \tag{3.20}$$

$$\text{MSE}\left(\Delta f_k\right) = E\left[\left|e_{\Delta f_k}\right|^2\right] = E\left[\left|\Delta f_k - \widehat{\Delta f}_k\right|^2\right] \quad . \tag{3.21}$$

We consider a simulation scenario with $K = 2$ users in collision on one timeslot. The packet of user 2 has been decoded successfully and the packet of user 1 needs to be decoded after removing the packet of user 2. Therefore, the channel estimation for user 2 shall be evaluated in order to conclude the impact of the residual channel estimation errors on user 1. Both Equations (3.20) and (3.21) are used to plot the MSEs for α_2 and Δf_2 of user 2 in Figure 3.9 and Figure 3.10, respectively. Both figures show a comparison between the CRLBs and the MSEs of the channel estimation for user 2. For the MSE curves, confidence intervals equal to $\left[\text{MSE} - \sigma_e^2, \text{MSE} + \sigma_e^2\right]$ are also shown, with σ_e^2 being the variance of the estimation error. Similar results have been obtained as well for user 1 for the same E_s/N_0 range considered.

It can be observed from Figure 3.9 that the MSE is 18% higher than the CRLB for the estimation of α_2. Moreover, in Figure 3.10 it is shown that the MSE for the estimation of Δf_2 is 75% higher from the corresponding CRLB. In order to show the effect of the degradation of MSEs versus the CRLBs on the decoding of the remaining packet, we derive the average Signal to Noise

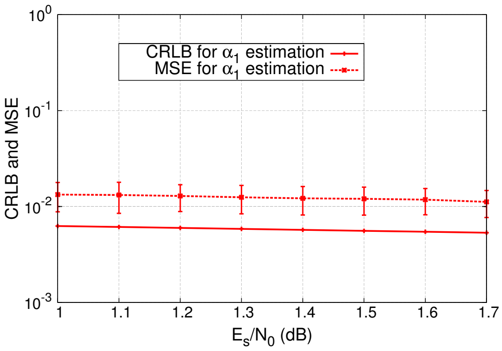


Figure 3.9 – CRLB and MSE for estimation of $\alpha_1 = A_1 e^{j\phi_1}$.

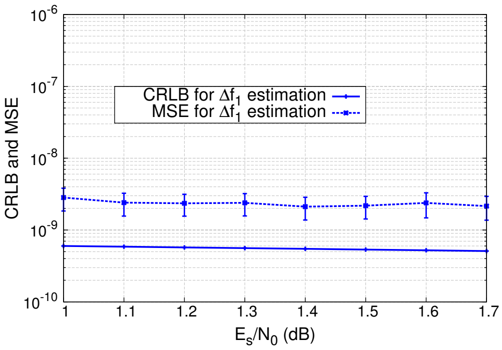


Figure 3.10 – CRLB and MSE for estimation of Δf_1.

Ratio for user 1 (SNR_1) with residual estimation errors, as shown below:

$$\mathrm{SNR}_1 = \frac{P_s}{N_0 + P_{e_2}} = \frac{E\left[\left|\hat{h}_1\right|^2\right]}{N_0 + P_{e_2}} \quad , \tag{3.22}$$

where $P_s = E\left[\left|\hat{h}_1\right|\right]$ is the power of the signal of user 1 when the energy of each transmitted symbol is normalised, N_0 is the AWGN power spectral density and P_{e_2} is the power of the residual estimation errors of user 2 detailed as follows:

$$\begin{aligned} P_{e_2} =& E\left[\left|\widehat{h}_2 - h_2\right|^2\right] \\ =& E\left[\left|\widehat{\alpha}_2 e^{j2\pi\widehat{\Delta f}_2 nT} - \alpha_2 e^{j2\pi\Delta f_2 nT}\right|^2\right] \\ =& A_2^2 E\left[\left|\frac{\widehat{A}_2}{A_2} e^{j2\pi(\widehat{\Delta f}_2 - \Delta f_2)nT + (\widehat{\phi}_2 - \phi_2)} - 1\right|^2\right] \quad . \end{aligned} \tag{3.23}$$

Based on the numerical values obtained in Figure 3.10, we can assume that, in practice, the effect of $(\widehat{\Delta f}_2 - \Delta f_2)$ can be neglected over the packet length (620 symbols in our case). Then we can suppose that $P_{e_2} = \mathrm{MSE}(\alpha_2)$. For instance, a numerical application at $E_s/N_0 = 1$ dB would give a degradation of SNR_1 of 0.2 dB compared to the case with no residual estimation errors.

3.4 Experimental results with residual channel estimation errors

In this section, we compute the Packet Error Rate (PER) obtained after demodulation and decoding of the signal s_1 of user 1 in presence of residual channel estimation errors. We also compare the results with respect to the perfect Channel State Information (CSI) case. The same simulation parameters as in Section 3.2.2 are considered, and joint estimation and decoding is performed. The JED iterations are repeated up to 3 times. Several simulation scenarios with various numbers of interference signals, and various attenuation and timing offsets schemes are compared.

3.4.1 One interference case

We consider two users in collision on the same timeslot. We suppose that the channel amplitudes A_1 and A_2 are either normalised to 1 or taking different values. The timing offsets τ_1 and τ_2 can have discrete values in $\left[0, \frac{1}{Q}, \frac{2}{Q}, \cdots, \frac{Q-1}{Q}\right]$. Figure 3.11 shows the PER obtained in 5 simulation scenarios:

- Perfect CSI scenario;
- Proposed Channel Estimation (CE) technique: EM with autocorrelation initialisation, PSAM and JED with an assumption of normalised channel amplitudes $A_1 = A_2 = 1$ and negligible timing offsets (synchronous);
- Proposed Channel Estimation (CE) technique with different channel attenuations (A_1 uniformly distributed in $[0.7, 1]$ and $A_2 = 1$) and negligible timing offsets (synchronous);
- Proposed Channel Estimation (CE) technique with $A_1 = A_2 = 1$ and misaligned received packets (Asynchronous);
- CE using EM with autocorrelation initialisation and PSAM without JED.

The figure shows that with JED, the PER performance degradation in comparison to the perfect CSI case is around 0.1 dB when $E_s/N_0 \leq 2$ dB, and around 0.3 dB when $E_s/N_0 > 2$ dB. We can observe that the PER results are coherent with the numerical SNR degradation derived from the MSEs values in Section 3.3.2. Figure 3.11 also shows that in the case of two superimposed packets with different channel attenuations, the channel estimation induces negligible degradation on the PER after interference cancellation.

3.4.2 More than one interference

Now let us consider the case where several packets are in collision on the same timeslot, and we decode the packet of interest after iterative interference cancellation of all the others, and in presence of cumulative residual channel estimation errors. The results shown in Figure 3.12 are given for the synchronous packets case (negligible timing offsets among packets in collision). Figure 3.12 plots the PER after cancellation of up to 4 interference signals all having equal power. We can observe that the PER degradation is less than 0.05 dB.

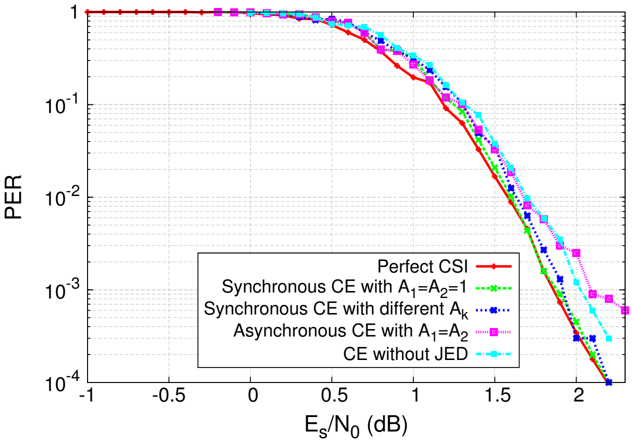


Figure 3.11 – PER vs. E_s/N_0 with channel estimation and interference cancellation in case of one interference packet. Modcod: CCSDS turbo code $R = 1/2$ and QPSK. $L = 460$ symbols.

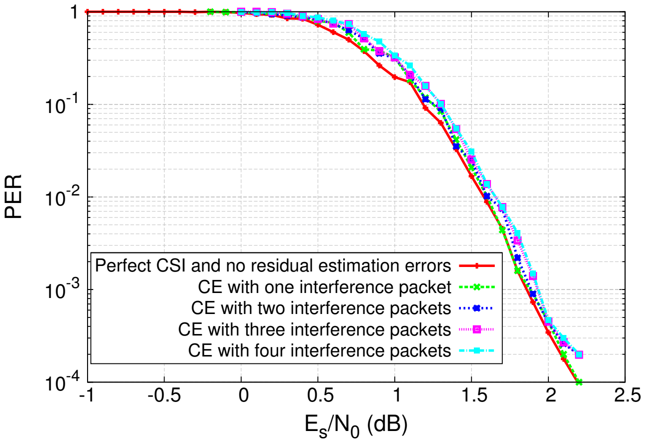


Figure 3.12 – PER vs. E_s/N_0 with channel estimation and interference cancellation in case of more than one interference. Modcod: CCSDS turbo code $R = 1/2$ and QPSK. $L = 460$ symbols.

3.5 Summary & conclusion

In this chapter, we investigated the impact of cumulative residual channel estimation errors after interference cancellation. We proposed an enhanced channel estimation technique to jointly estimate the channel parameters of several packets in collision on one timeslot. The proposed technique combines the EM algorithm with autocorrelation initialisation and the integration of timing offsets estimation, as well as using PSAM. We also proposed to apply a joint estimation and decoding approach to enhance the SIC performance. With the proposed algorithm, we were able to jointly estimate different channel parameters while keeping a relatively low performance loss. We also showed that the MSEs curves obtained are parallel to the CRLBs. Furthermore, we noticed that with the consideration of timing offsets, additional loss in the PER curve is observed and we can conclude that the performance of the estimation algorithm is not optimal in this case. Therefore, in future work, it shall be considered to use more accurate timing offsets estimators prior to the other channel parameters estimation. In addition, the trade-off of performance versus complexity shall be studied, since the quality of the channel estimation is enhanced with the number of EM and JED iterations performed.

The work presented in this chapter permitted to study the problem of channel estimation for RA methods in low SNIR and high load regimes. Therefore, this study is useful to propose and evaluate new techniques that enhance the throughput and reduce packet losses over the RA channel. For this objective, the following chapter presents a new data reception technique over a random access channel, that enables to significantly increase the throughput of existing RA methods, while maintaining a relatively low packet loss ratio.

4

Multi-Replica Decoding using Correlation based Localisation (MARSALA)

In the previous chapter, we evaluated the practical issues related to residual channel estimation errors in recent RA techniques such as CRDSA and MuSCA. We showed that the impact of residual channel estimation errors on the performance can be decreased when using the proposed estimation algorithm. Based on those results, we now propose in this chapter a new method for packet reception over a RA channel that enables to enhance the throughput and reduce the packet loss ratio. This method is called Multi-replicA decoding using corRelation baSed locALisAtion (MARSALA). It uses the same transmission scheme as in CRDSA where each user sends several replicas of the same packet over the frame. MARSALA presents a new decoding technique for CRDSA that localises all the replicas of a packet using a correlation based method, then combines them in order to decode the data. With MARSALA, the maximal normalised throughput is significantly enhanced. The work presented in this chapter was published in [74].

4.1 Introduction

As pointed out in Chapter 2, recent RA methods proposed for satellite communications can enhance the throughput and reduce packets losses and transmission delays, compared to legacy RA methods. However, a main challenge in recent RA schemes is recovering signals in collision arriving simultaneously at the receiver. In fact, the information loss due to signal collisions is one of the principal causes of the low throughput on RA channels. To handle this problem, recent RA methods use Physical Layer Network Coding (PNC) [54] and Successive Interference Cancellation (SIC). Among these methods, we cited in Chapter 2 CRDSA, a RA method included as optional in the DVB-RCS2 standard [10]. We recall that in CRDSA and its recent versions [16, 27], each user transmits two or more replicas of the same packet on randomly chosen timeslots on the frame. Each replica contains a signalling field used to localise its copy. At the receiver side, if a packet is successfully decoded, its replicas are localised on the frame. Then, SIC is performed, so that the packet and its corresponding replicas are removed from the frame. This process is repeated iteratively over the same frame until no more packets can be recovered, or after a number of iterations is reached. As shown in Chapter 2, in the case of equi-powered packets, CRDSA can achieve a throughput of 0.55 bits/symbol with transmissions of 2 replicas per packet and 0.7 bits/symbol with 3 replicas per packet, at $E_s/N_0 = 10$ dB.

Another recent RA technique also cited in Chapter 2 is MuSCA [21]. It proposes to significantly enhance the throughput by resorting to packets strong FEC encoding and codeword fragmentation before transmission on the RA channel. MuSCA is able to achieve a normalised throughput greater than 1.3 bits/symbol. However, the weakness in MuSCA is that it requires the use of encoded signalling fields to localise the fragments of a same packet, resulting in additional overhead on the useful information and system modifications at the receiver side.

In order to reduce the signalling overhead and enhance the throughput, we propose Multi-replicA decoding using corRelation baSed locALisAtion (MARSALA) [75] as a new method enabling to significantly increase the throughput compared to previous RA methods such as CRDSA, IRSA and CSA. In contrary to MuSCA, with MARSALA, we are able to decode a packet by combining all its replicas without any additional overhead compared to CRDSA. MARSALA does not result in any system modifications on the CRDSA transmitter side. The only implementation complexity added is at the receiver side, and it is mainly induced by the channel estimation and the correlation based localisation. Therefore, MARSALA can conform to the requirements of the DVB-RCS2 standard. In the following sections of this chapter, we explain precisely how packet replicas can be localised on the frame using a simple correlation based technique. Then, we detail how packet replicas are combined and decoded even when they are in collision with packets sent by other users. In addition, we highlight on the channel estimation operations that should be done in order to ensure a good performance of MARSALA. Finally, we show a first evaluation of the normalised throughput achieved with MARSALA and compare it to CRDSA in the case of equi-powered packets.

4.2 System assumptions

Figure 4.1 shows the system model. We consider N_u fixed terrestrial terminals sharing one or several carrier frequency bands on the RA channel of an uplink of a geostationary satellite communication system. The considered carrier is logically divided into frames, themselves divided into N_s timeslots as described in the MF-TDMA return link structure of the DVB-RCS2 standard. For simplification purpose, we will consider in the following that only one carrier is shared among several satellite terminals. We denote by T_F the duration of one frame. Each terminal transmits N_b replicas of the same packet on randomly chosen timeslots of the frame. We suppose that each terminal must wait for the beginning of the next frame in order to send a different packet. We assume that there is no direct link between the terminals. We consider an Additive White Gaussian Noise (AWGN) channel model. The AWGN power spectral density denoted by N_0 is constant over one T_F . We consider equi-powered packets at the receiver side.

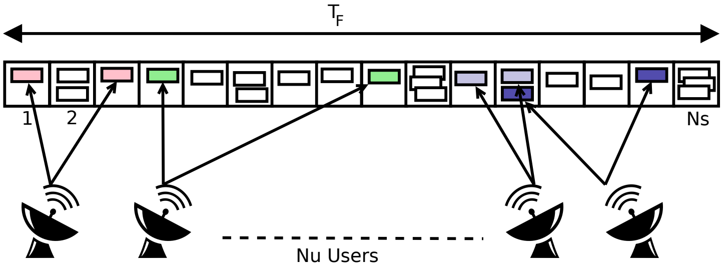
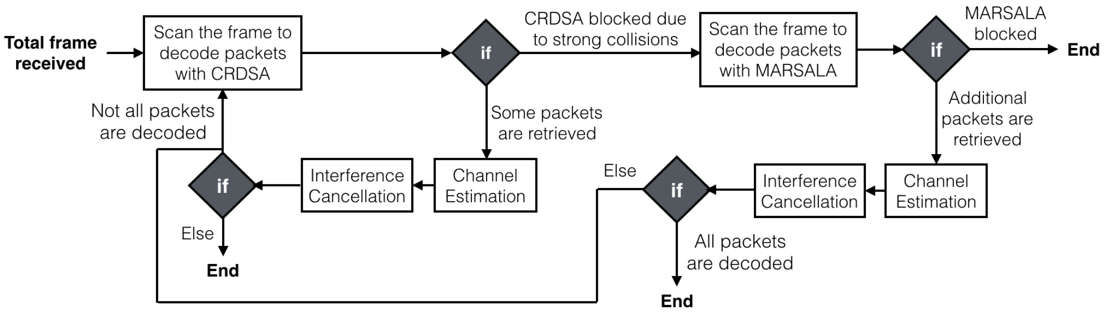


Figure 4.1 – System model: an uplink frame of N_s timeslots shared among N_u terminals, each transmitting $N_b = 2$ replicas per packet.

Before transmission on the RA channel, each packet is encoded with a turbo code of rate R and modulated with a modulation of order M. A preamble, a postamble and pilot symbols are added to the modulated codeword for the purpose of channel estimation. Fields with pointers to the replicas positions on the frame are also added to the packet structure. The total burst length is equal to L symbols. To avoid inter-symbol interference, the symbols sequence constituting each packet enters a shaping filter with a square root raised cosine function. At the receiver side, the signal is oversampled by a factor Q then it enters a matched filter to optimally filter out noise before being sampled at the symbol period.

In this work, we consider a system combining CRDSA and MARSALA as shown in Figure 4.2 . The transmitter side is the same as the one described in [16] for CRDSA. The system modifications are only made at the receiver side. After a complete frame is stored, each timeslot is treated with CRDSA in order to retrieve non-decoded packets. Whenever a packet is successfully decoded, the SIC process is applied to remove the decoded packet and its corresponding replicas from the frame. After scanning the entire frame, the same packets detection and SIC process is repeated iteratively in order to recover more packets after the previous interference cancellation. However, in high load regimes, some packets might not be recovered with CRDSA due to strong collisions, even after several SIC iterations. At this point, MARSALA is applied. MARSALA chooses one timeslot as a reference timeslot (TS_{ref}) and attempts to localise the replicas of the packets available on this timeslot by using a correlation technique. Once the replicas of a given packet on TS_{ref} are localised, they are combined together, then decoded. Similarly to CRDSA, if a packet is successfully decoded with MARSALA, the packet and its replicas are removed from the frame. MARSALA can be applied until at least one packet is successfully recovered, then the receiver can switch back to re-scan the frame using CRDSA. Thus, MARSALA can play a major role in releasing CRDSA non-decoded packets from collisions and trigger additional SIC iterations. In the next section, we will describe in details the several steps of the MARSALA scheme.



Total frame received
Scan the frame to decode packets with CRDSA
if
CRDSA blocked due to strong collisions
Scan the frame to decode packets with MARSALA
if
MARSALA blocked
End
Additional packets are retrieved
Some packets are retrieved
Not all packets are decoded
if
Interference Cancellation
Channel Estimation
Else
End
Else
if
Interference Cancellation
Channel Estimation
All packets are decoded
End

Figure 4.2 – Frame processing scheme combining CRDSA and MARSALA at the receiver side.

4.3 MARSALA detailed scheme at the receiver side

As explained in the previous section, when CRDSA is blocked, MARSALA is applied in order to recover additional packets and trigger more SIC iterations. The proposed localisation and decoding scheme in MARSALA operates according to the following steps:

1. **Replicas Localisation**
   Localisation of N_b replicas corresponding to a same packet.
2. For each set of N_b localised replicas:
   (a) **Channel parameters estimation and adjustment**
   Estimation and correction of the following parameters: phase shifts, timing offsets and frequency offsets;
   (b) **Signal Combination**
   Coherent combination of the N_b replicas of a given packet with corrected channel parameters;
   (c) **Demodulation and decoding of the resulting packet**.
   (d) **Interference cancellation**.
3. Go back to (1) until one or several packets are successfully decoded.

4.3.1 Replicas Localisation

The replicas localisation entity at the receiver side allows to localise all the replicas corresponding to a same packet on the received frame. This operation is based on signal time-domain correlation. First, the receiver identifies an arbitrary reference timeslot containing packets in collision and refers to it by TS_k, where k is an integer index ($k \in [1, N_s]$). It uses the signal received within this timeslot as a reference signal denoted by $x(t)$, with t being the discrete time vector, i.e. $t = \left[(k-1)T_{slot}, (k-1)T_{slot} + \frac{T_s}{Q}, ..., kT_{slot}\right]$. T_{slot} refers to the duration of one timeslot and T_s denotes the symbol period. In other words, $x(t)$ is the sum of all signals received on the timeslot TS_k.

After the identification of one reference signal $x(t)$, the receiver computes the correlation between $x(t)$ and the signal received on the rest of the frame, denoted by $y(t)$. An example of $x(t)$ and $y(t)$ is given in Figure 4.3, where $k = 2$, i.e. $x(t)$ is the signal received on the second timeslot. The result of the cross-correlation between $x(t)$ and $y(t)$ presents a number of correlation peaks N_p. This number depends on the number of users that transmitted a replica on TS_k as well as the number of replicas per packet N_b. For instance, if TS_k contains N_{coll} packets in collision, the number of cross-correlation peaks observed should be less or equal to $(N_b - 1) \times N_{coll}$. A correlation between signals on two distinct timeslots presents a peak if the correlation amplitude is above a predefined threshold (function max_peaks in Algorithm 1). Finally, the correlation peaks are used to identify all the timeslots TS_m, $(m \in [1, N_s]; m \neq k)$, containing a packet replica of one of the packets received on TS_k.

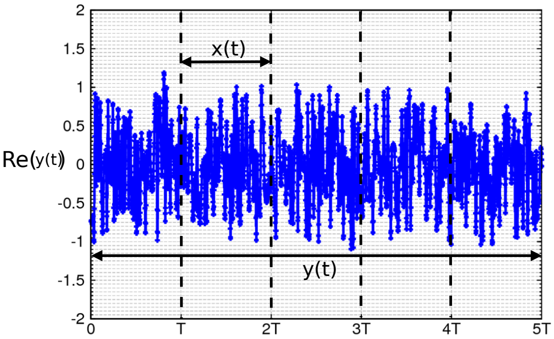


Figure 4.3 – Example of the received signal on one timeslot: $x(t)$, and the received signal on the entire frame: $y(t)$, with $N_s = 5$ timeslots and $T = T_{slot}$.

Association of the localised replicas to a given packet on the reference timeslot

In the previous step, correlation peaks detected the timeslots containing the replicas of all the packets received on the reference timeslot TS_k. However, in order to combine the replicas of a same packet, it is important to identify which set of localised timeslots contain information about the same packet. Therefore, in this step, we describe how to associate each set of $(N_b - 1)$ correlation peaks to the replicas of only one particular packet received on TS_k.

Obviously, if $N_b = 2$ replicas per packet, we can directly associate any of the correlation peaks detected to the replica of a packet on TS_k. However, if $N_b > 2$ replicas per packet, we should find a way to associate a set of $(N_b - 1)$ correlation peaks to the replicas of a packet received on TS_k. The method proposed is detailed as follows:

1. We associate the highest correlation peak detected to the second replica of one packet available on TS_k.
2. We combine the signal on TS_k with the signal corresponding to the second replica, as described later in the ‘Signal combination’ step.
3. We compute the cross-correlation between the resulting combined signal and the signals received on the timeslots containing the replicas detected previously with the first correlation procedure.
4. The maximum amplitude peaks resulting from this cross-correlation procedure correspond to the locations of the other replicas.

Below is a pseudo code explaining the procedure for replicas association to a given packet.

```
Let $\boldsymbol{s_{rep_1}}$ denote the signal received on $TS_{ref}$;
for each timeslot $k \neq TS_{ref}$ do
    $CrossCorr1(k) = cross\_correlation(s_{rep_1}, \text{signal on} TS_k)$;
end
$peaks1 = max\_peaks(CrossCorr1, N_p)$;
$max(peaks1) \Rightarrow TS_{rep_2} \Rightarrow \boldsymbol{s_{rep_2}}$;
Correct the timing offset and the phase shift between $\boldsymbol{s_{rep_1}}$ and $\boldsymbol{s_{rep_2}}$;
Combine $\boldsymbol{s_{rep_1}}$ and $\boldsymbol{s_{rep_2}}$;
for $k \in peaks1$ & $TS_{rep_k} \neq TS_{rep_2}$ do
    $CrossCorr2(k) = cross\_correlation(\boldsymbol{s_{rep_1}} + \boldsymbol{s_{rep_2}}, \text{signal on} TS_{rep_k})$;
end
$peaks2 = max\_peaks(CrossCorr2, N_p)$;
$max(peaks2) \Rightarrow TS_{rep_3} \Rightarrow \boldsymbol{s_{rep_3}}$;
```

Algorithm 1: How to associate replicas to a given packet

It is worth noting that the ‘Replicas localisation’ step constitutes one of the main differences between MARSALA and all CRDSA versions. MARSALA enables the localisation of the replicas without having to decode the signalling information. Even before any demodulation or decoding operation, the correlation peaks detected are used to identify the timeslots containing the replicas of the same packet. Once the replicas of a same packet are localised, the next step is to coherently combine these replicas together.

4.3.2 Channel parameters estimation and adjustment

One main problem with replicas combination is that the replicas received on distinct timeslots have different phases and timing offsets. In fact, in real transmission conditions, the packet replicas transmitted by the same user experience different channel impairments because they are sent on different timeslots of the frame. In particular, carrier frequency offsets and phase variations, as well as timing offsets among distinct replicas must be estimated and corrected prior to any signal combination operation.

To estimate the timing offset of each localised packet replica, the use of a timing estimator such as the Oerder & Meyr synchronisation algorithm for timing recovery [76] is not practical for MARSALA. In fact, the algorithm in [76] presents a method to estimate the timing offset based on measuring the angle of the Power Spectral Density (PSD) function of the corresponding signal. Such an algorithm is not optimal in the context of MARSALA due to the high collisions environment as well as the frequency and phase variations.

In addition, the estimation of the phase for each symbol using the Viterbi algorithm [77] is not well-suited for MARSALA because of the very low SNIR of each packet replica before replicas combination.

Therefore, we detail in Chapter 5 a method to jointly estimate the timing offsets and phase variations among the replicas inspired from the algorithm of time and frequency offset estimation in Orthogonal Frequency Division Multiplexing (OFDM) systems [57]. The algorithm in [57] performs correlation procedures on the repeated pilot sequences in a OFDM burst in order to estimate the carrier frequency and timing offsets. Whereas in MARSALA, the repeated sequences are the packet replicas corresponding to a same user.

At the end of these synchronisation operations, we obtain N_b signals corresponding to the replicas of a same packet, which are coherent in terms of timing and phase. Thus, the receiver can proceed with signal combination as explained in the following subsection.

4.3.3 Signal combination

After channel estimation and adjustment for the localised packet replicas, we obtain N_b coherent signals corresponding to N_b replicas of the same user. Each signal is interfered by a different number of other users signals. The receiver performs the combination of these N_b signals. The power P_{eq} of the resulting signal is

$$P_{eq} = P(s_1 + s_2 + ... + s_{N_b}) + P(I_1 + I_2 + ... + I_{N_b}) + \sum_{i=1}^{N_b} N_0 \quad , \tag{4.1}$$

where s_1, s_2, $\cdots$, and s_{N_b} represent the signals relative to N_b replicas of a same packet after phase and timing correction, I_1, I_2 , $\cdots$, and I_{N_b} represent the interference signals on the replicas timeslots and N_0 is the power spectral density of AWGN. Given that the signals s_1, s_2, $\cdots$ and s_{N_b} contain the same information and are almost coherent in phase and timing, while interference and noise are uncorrelated, we can write P_{eq} as follows

$$P_{eq} = P(N_b s_1) + \sum_{i=1}^{N_b} P(I_i) + \sum_{i=1}^{N_b} N_0 = N_b^2 P(s_1) + \sum_{i=1}^{N_b} P(I_i) + N_b N_0 \quad . \tag{4.2}$$

The goal is to obtain an equivalent Signal to Noise plus Interference Ratio (SNIR) for a localised set of N_b replicas after signal combination, higher than the SNIR for one interfered replica without signal combination. Once the signals are combined, the channel parameters relative to the combined signals are estimated (A and Δf), and the receiver attempts to demodulate and decode the desired signal. If demodulation and decoding is successful, the receiver performs interference cancellation, and removes the corresponding packet and its replicas from the corresponding timeslots on the frame (interference cancellation).

4.4 Numerical Example

Figure 4.4 illustrates an example of a received frame, where $N_s = 8$ timeslots, $N_u = 8$ users and $N_b = 3$ replicas per packet. We suppose that the modcod used is a CCSDS turbo code of rate 1/2, the modulation is QPSK and all the packets are transmitted with equal power. The received packets are affected by random phase, timing and frequency offsets. The frequency offset Δf is different for each user but it is considered to remain constant for one user on the duration of one frame. Δf can have a value between 0 to 1% of the symbol rate $1/T$. For each user, the phase offset ϕ has a value between 0 and 2π and it is supposed to change from one timeslot to another. It is worth noting that this is a simplified assumption, given that in practice, phase noise creates random fluctuations over a single packet duration. According to the DVB-RCS2 guidelines [28], the phase noise is lower than -16 dBc/Hz for a symbol rate≥ 128 kBaud (with dBc/Hz denoting phase noise power relative to the carrier contained in a 1 Hz bandwidth centred at different offsets from the carrier). Given that those fluctuations are relatively low compared to packet collisions distortions, the results presented in this work assume a negligible impact of phase noise, and open perspectives for evaluating it in future work.

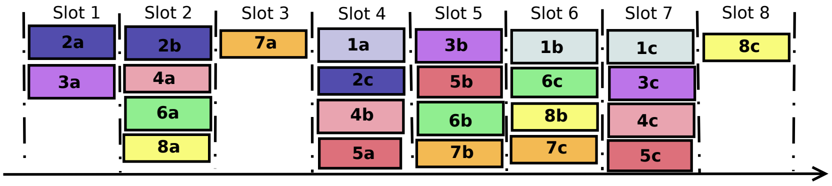


Figure 4.4 – A frame with 2 clean packets on timeslots 3 and 8. $N_s = 8$ slots, $N_u = 8$ users, and $N_b = 3$ replicas per user.

At the receiver side, the frame is scanned. As shown in Figure 4.4, the packets $7a$ and $8c$ transmitted by users 7 and 8, on timeslots 3 and 8 respectively, are supposed to be decoded successfully because they are contention-free packets. Then, they are removed from the frame as well as their replicas ($7b$, $7c$) and ($8a$, $8b$). After interference cancellation, the new frame configuration is shown in Figure 4.5. All the remaining packet replicas are in collision. Given that all the packets are equi-powered, this situation might result in a high Packet Error Rate (PER) in RA schemes such as CRDSA and IRSA, especially if the FEC code rate is not sufficiently low. Therefore, the decoder can proceed with MARSALA in order to attempt to decode additional packets.

The receiver identifies timeslot 1 (TS_1) as a reference timeslot. In Figure 4.5, TS_1 contains two packets in collision ($2a$) and ($3a$) transmitted by user 2 and user 3, respectively. Their replicas

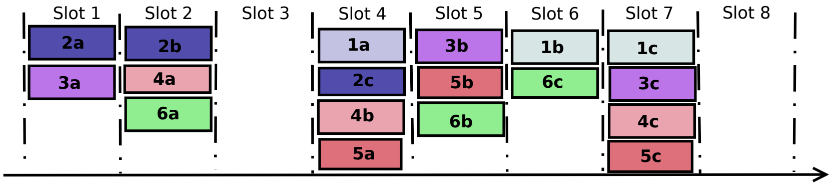


Figure 4.5 – The frame after cancellation of clean replicas.

are received on other timeslots of the frame as follows: $(2b)$ on TS_2, $(2c)$ on TS_4, $(3b)$ on TS_5 and $(3c)$ on TS_7. Our objective is to localise the replicas of the packets transmitted on TS_1 over the rest of the frame. Therefore, the localisation entity computes the cross-correlation between the signal received on TS_1 $(x(t))$ and the signal received on the rest of the frame $(y(t))$. The signals $x(t)$ and $y(t)$ can be written as shown in Equations (4.3) and (4.4) respectively.

$$x(t) = s_2(t)e^{j(\phi_2+2\pi\Delta f_2 t)} + \underbrace{s_3(t)e^{j\phi_3+2\pi\Delta f_3 t} + n_1(t)}_{w(t)} \quad , \tag{4.3}$$

where s_2 and s_3 are the signals corresponding to the packets sent by user 2 and user 3, respectively. ϕ_i and Δf_i denote respectively, the phase shift and the frequency offset relative to the first packet replica sent by user i. $n_1(t)$ represents the additive white gaussian noise on $TS1$, and $w(t)$ encompasses both the interference plus noise terms on $TS1$.

$$\begin{aligned} y(t) = s_2(t-d_2)e^{j(\phi_2'+2\pi\Delta f_2 t)} &+ s_2(t-d_4)e^{j(\phi_2''+2\pi\Delta f_2 t)} \\ &+ \underbrace{s_3(t-d_5)e^{j(\phi_3'+2\pi\Delta f_3 t)} + s_3(t-d_7)e^{j(\phi_3''+2\pi\Delta f_3 t)} + s(t)}_{z(t)} \quad , \end{aligned} \tag{4.4}$$

where $s(t)$ represents the signal encompassing the total noise plus interference signals received on the rest of the frame, other than the signals transmitted by users 2 and 3. ϕ_i' and ϕ_i'' denote the phase offsets of the second and third replica sent by user i. While d_i represents the time delay between the packet replicas on the i^{th} timeslot, and we suppose that the first packet replica has a reference time delay equal to zero.

Let us focus on the signal transmitted by user 2. The same evaluation can be done for user 3 or any other user on the frame. In Eq. (4.5) and Eq. (4.6), we express respectively $x(t)$ and $y(t)$ in function of the signal sent by user 2, $w(t)$ and $z(t)$.

$$x(t) = s_2(t)e^{j(\phi_2+2\pi\Delta f_2 t)} + w(t) \quad , \tag{4.5}$$

$$y(t) = s_2(t-d_2)e^{j(\phi_2'+2\pi\Delta f_2 t)} + s_2(t-d_4)e^{j(\phi_2''+2\pi\Delta f_2 t)} + z(t) \quad . \tag{4.6}$$

The time domain cross-correlation $R_{Y,X}$ between $x(t)$ and $y(t)$ can be written as follows

$$
\begin{aligned}
R_{Y,X}(\tau) = \int_t y(t)x^*(t-\tau)dt \\
&= \int_t s_2(t-d_2)e^{j(\phi_2'+2\pi\Delta f_2 t)}\, s_2^*(t-\tau)e^{-j(\phi_2+2\pi\Delta f_2(t-\tau))} \\
&+ \int_t s_2(t-d_4)e^{j(\phi_2''+2\pi\Delta f_2 t)}\, s_2^*(t-\tau)e^{-j(\phi_2+2\pi\Delta f_2(t-\tau))} \\
&\qquad + R_{z,s_2}(\tau)+R_{z,w}(\tau)+R_{s_2,w}(\tau) \quad , \qquad (4.7)
\end{aligned}
$$

with $(R_{z,s_2}+R_{z,w}+R_{s_2,w})$ being the sum of the cross-correlations between the uncorrelated terms in $x(t)$ and $y(t)$. Clearly for user 2, the peak amplitudes of $R_{Y,X}(\tau)$ are obtained for $\tau = d_2$ and $\tau = d_4$. If the same evaluation is done for user 3, the results will show correlation peaks for $\tau = d_5$ and $\tau = d_7$. The correlation amplitudes obtained are illustrated in Figure 4.6. The positions of the correlation peaks show that time slots 2, 4, 5 and 7 contain replicas of the packets present on TS_1.

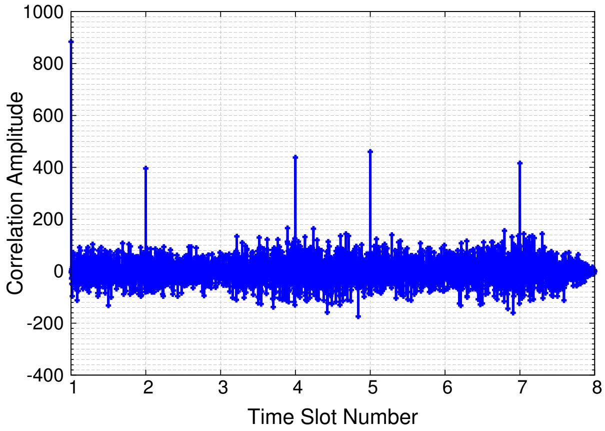


Figure 4.6 – Correlation of the signal received on slot 1 and the rest of the timeslots of the frame.

As explained earlier in the previous section, in order to identify which time slots correspond to the set of replicas sent by the same user, each signal received on a timeslot detected by a correlation peak, is correlated with the other timeslots detected. Following Section 4.3.2 and as explained in detail in Chapter 5, once the packet replicas of user 2 are localised, the receiver estimates and compensates the timing offsets $\Delta\tau_2$ and $\Delta\tau_4$ as well as the phase shifts ϕ_2' and ϕ_2'' among the replicas(see Figure 4.7). Consequently, the three almost coherent signals can be combined prior their demodulation. In the rest of this example, we suppose that the parameters correction is perfect so the combined replicas are perfectly coherent in timing and phase.

Let us consider the following scenario for user 2. We suppose that all the packets transmitted within the duration of one frame are equi-powered. Therefore, the power of any interference

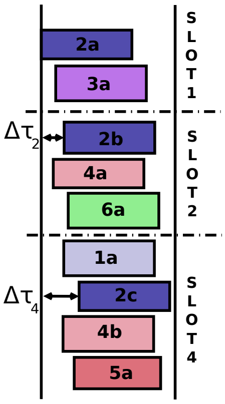


Figure 4.7 – Reception of packet replicas of user 2 with a timing offset $\tau = 0$ and a clock drift $\Delta\tau_i$ on each time slot i

packet is equal to the power of the packet transmitted by user 2. For instance, on timeslot 1, the power of the interference term $P(I_1) = P(s_2)$, and on timeslot 2, $P(I_2) = 2P(s_2)$. We suppose that user 2 uses QPSK modulation and a CCSDS turbocode [12] with a coding rate 1/2. The AWGN has the same power spectral density N_0 on the entire frame duration. The Signal to Noise Ratio for user 2 $\left(SNR = \frac{P(s_2)}{N_0}\right)$ is supposed to be equal to 2 dB. With the numerical assumptions considered, the SNIR for user 2 on timeslots 1, 2 and 4, can be written as shown in Equations (4.8), (4.9) and (4.11) respectively.

$$SNIR_{2,1} = \frac{P(s_2)}{P(s_3) + N_0} = \left.\frac{1}{1 + \left(\frac{P(s_2)}{N_0}\right)^{-1}}\right|_{dB} = -2.12 \text{ dB} \Rightarrow \text{PER} = 100\% \quad , \tag{4.8}$$

$$SNIR_{2,2} = \left.\frac{1}{2 + \left(\frac{P(s_2)}{N_0}\right)^{-1}}\right|_{dB} = -4.2 \text{ dB} \Rightarrow \text{PER} = 100\% \quad , \tag{4.9}$$

$$SNIR_{2,4} = \left.\frac{1}{3 + \left(\frac{P(s_2)}{N_0}\right)^{-1}}\right|_{dB} = -5.6 \text{ dB} \Rightarrow \text{PER} = 100\% \quad . \tag{4.10}$$

As it was shown in Figure 3.5 of Chapter 3, the PER obtained with a CCSDS turbo code of rate 1/2 and a QPSK modulation at $E_s/N_0 < 0$ dB is equal to 100%. Therefore, the packet transmitted by user 2 is not decoded successfully on any of the timeslots 1, 2 or 4. However, after applying MARSALA and combining the 3 packet replicas of user 2, the equivalent SNIR becomes

$$\begin{aligned} SNIR_{eq} =& \frac{N_b^2 P(s_2)}{\sum_{i=1}^{3} iN_0 + P(I_i)} \\ =& \left. \frac{3}{2+\left(\frac{P(s_2)}{N_0}\right)^{-1}} \right|_{dB} = 0.57 \text{ dB} \Rightarrow \text{PER} = 63.73\% \quad . \end{aligned} \tag{4.11}$$

Thus, the probability for the information transmitted by user 2 to be demodulated and decoded successfully increases from 0% to 36.27% when combining all three packet replicas of user 2. In the following section, we will show how even such a small increase in the successful decoding probability can lead to a significant throughput enhancement with MARSALA.

4.5 Simulation Results with perfect channel state information

To evaluate the performance in terms of throughput of MARSALA, we realise Monte Carlo simulations and compare the results with the throughput obtained by CRDSA. In our simulations, we suppose that all the packets are received with equal power, so that the channel SNR at the receiver side is the same for all users. To make a first evaluation of the performance that can be achieved with MARSALA, we consider that the channel estimation and the interference cancellation are ideal, i.e. perfect Channel State Information (CSI) is available at the receiver. We also assume that the combined replicas are coherent in timing and phase. In the simulations, we still consider a frame composed of N_s time slots with N_u users attempting to transmit replicas within a frame duration. The traffic profile assumed is Constant Bit Rate (CBR).

We recall that the normalised load (G) is measured in packets per slot and expressed as $G = \frac{N_u}{N_s}$, and the normalised throughput (T) is given by $T = G * (1 - PLR(G))$, where PLR(G) is the probability that a packet is not decoded successfully for a given G and a given SNR. To ease the recognition of several MARSALA and CRDSA versions, we denote by MARSALA-2 and CRDSA-2 the MARSALA and CRDSA systems where each user transmits 2 replicas of the same packet. The same notation is taken for MARSALA-3 and CRDSA-3.

Figure 4.8 and Figure 4.9 show the simulation results achieved with a frame size equal to 100 timeslots. The modulation used is QPSK. For a first evaluation of MARSALA, we use a CCSDS turbo code of rate 1/2. On the one hand, Figure 4.8 compares the performance in terms of normalised throughput between MARSALA and CRDSA, both with an SNR equal to 2 dB. The chosen SNR can achieve a PER close to 10^{-5} in a collision-free transmission. In other words, in a system with dedicated access and no packet collisions, the ideal SNR for which the packets should be received is around 2 dB (with CCSDS-QPSK 1/2). We observe from Figure 4.8 that the throughput gain between MARSALA and CRDSA is significant. The maximum normalised throughput of MARSALA-2 and MARSALA-3 is 0.8 and 1.1 packets/slot respectively, while the throughput achieved with CRDSA is almost 1.5 times lower.

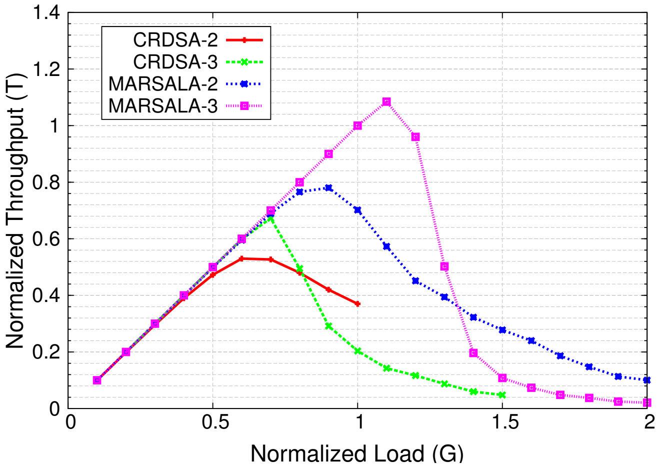


Figure 4.8 – T (packets/slot) vs. G(packets/slot) with CRDSA-2,-3 and MARSALA-2,-3; CCSDS turbo code of rate 1/2 with QPSK modulation, E_s/N_0 =2 dB, N_s = 100 slots. Perfect CSI.

On the other hand, Figure 4.9 shows the normalised throughput curves of MARSALA-3 for $E_s/N_0 = -2$, 0, 2, 5 and 10 dB. For each E_s/N_0, we vary the normalised load until a maximum normalised throughput is achieved. We observe that, even for a relatively low E_s/N_0 (0 dB), MARSALA enables to reach a throughput up to 0.7 packets/slot. For a higher E_s/N_0, the peak of throughput significantly increases and reaches a value of 1.4 packets/slot at an E_s/N_0 equal to 10 dB.

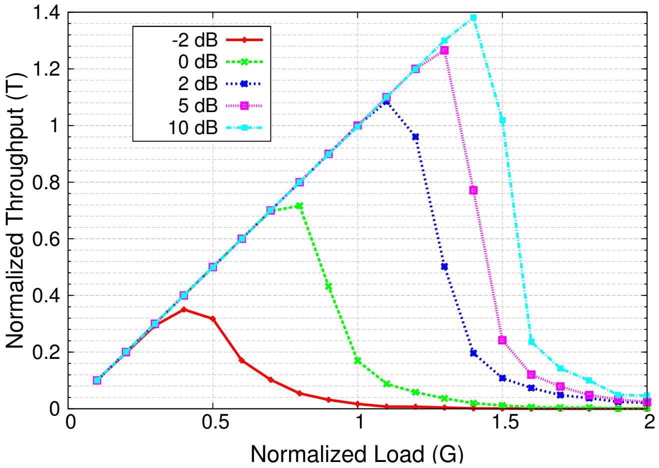


Figure 4.9 – T(packets/slot) vs. G(packets/slot) with MARSALA-3 for several values of E_s/N_0; CCSDS turbo code of rate 1/2 with QPSK modulation, N_s = 100 slots. Perfect CSI.

4.6 Summary & conclusion

In this chapter, we presented MARSALA, as a new decoding technique for CRDSA. The idea behind MARSALA is to propose a solution to the deadlock problem of CRDSA, by performing correlation operations within the received frame, and signal combination on the timeslots containing replicas of a same packet. With the simulation assumptions of perfect channel state information, MARSALA proved to achieve a significantly higher throughput than CRDSA with a CCSDS turbo code of rate 1/2 and a QPSK modulation.

In the following chapter, we will evaluate the effect of imperfect correction of phase and timing when the replicas of a same user are combined. Later, we will also study the gain of MARSALA with other modcods like QPSK 1/3 and other coding schemes such as DVB-RCS2 and 3GPP turbo codes. Given that in this chapter only MARSALA with equi-powered packets was evaluated, in the next chapters we will also study its behaviour in presence of unbalanced packets power.

5

Evaluation of MARSALA in Real Channel Conditions

In the previous chapter, we presented MARSALA as a new decoding technique for CRDSA designed for the satellite return link. It follows the same multiple transmission and interference cancellation scheme as CRDSA. In addition, at the receiver side, MARSALA uses correlation procedures to localise replicas of a same packet so as to coherently combine them. The work presented in the previous chapter showed significant performance gains with an assumption of ideal channel state information and coherent combining of the several replicas of a given packet. However, in a real system, synchronisation errors such as timing offsets and phase shifts between the replicas on separate timeslots will result in less constructive combining of the received signals. The main goal of this chapter is to describe a method to estimate and compensate for the timing and phase differences between the replicas, prior to their combination. The impact of signal misalignment due to timing offsets and phase shifts estimation errors is modelled and evaluated analytically. Based on the analytical results, the performance of MARSALA in realistic channel conditions is assessed via simulations, and compared to CRDSA in various scenarios. The work presented in this chapter was accepted for publication in [78].

5.1 Introduction & system hypothesis

5.1.1 Introduction

In the context of enhancing RA performance on the satellite return link, MARSALA was proposed in the previous chapter, as a solution to increase the throughput achieved with CRDSA without adding signalling overhead. Basically, MARSALA presents a new decoding technique for CRDSA based on replicas localisation using autocorrelation. In particular, it takes advantage of correlation procedures between the signals received on separate timeslots, in order to locate the replicas of one packet even when all of them are undergoing a collision and their pointers are not decodable. Furthermore, the coherent combination of the identified packet replicas allows to enhance the Signal to Noise plus Interference Ratio (SNIR), in order to recover additional packets and trigger more SIC iterations. The transmitter side in MARSALA is the same as in CRDSA, the only modifications to take into account are at the receiver side. In addition, MARSALA does not require any waveform modifications to the DVB-RCS2 standard.

In Chapter 3, we analysed the effect of residual channel estimation errors on the performance of RA methods that use the SIC principle. In MARSALA, an additional critical task is to estimate and compensate for the synchronisation differences between the replicas on separate timeslots, so as to allow their coherent combination. This step is very important to maximise the combination gain. Therefore, two main contributions are presented in this chapter:

- A method to estimate and compensate the timing offsets between packet replicas, as well as the phase shifts caused by carrier frequency and phase variations.
- The definition of an analytical model for the SNIR degradation caused by imperfect replicas combination due to timing and phase estimation errors. This model will be used to evaluate the performance degradation of MARSALA in terms of throughput and PLR in realistic channel conditions.

5.1.2 System Hypothesis

In this section, we recall the same system model as the one described in Section 4.2 of the previous chapter. The frame on the uplink covers one frequency band shared among N_u users and divided into N_s timeslots with a duration T_{slot} each. The users are supposed to be fixed terrestrial terminals. Each user transmits N_b copies of the same packet to a destination node (a satellite or a gateway) over randomly chosen timeslots within the duration of one frame (T_F). We suppose that, to send other packets, the user must wait until the beginning of the next frame. In this chapter, we still assume that all the received packets are equi-powered.

Each user experiences independent random timing offsets, frequency offsets and phase shifts. The timing offset is uniformly distributed in $[-T, T]$ and varies from one timeslot to another, with T being the symbol period. Due to the limited duration of the frame, the carrier frequency offset is supposed to remain constant for the packet replicas of the same user and it is uniformly distributed in $[0, \frac{0.01}{T}]$. The phase shift is uniformly distributed in $[-\pi, \pi]$ and may vary from one timeslot to another. We consider an Additive White Gaussian Noise (AWGN) channel model. The AWGN power spectral density denoted by N_0 is constant over T_F. The channel amplitude is supposed normalised to 1. Each packet is encoded with the DVB-RCS2 turbo-encoder for linear modulation, of rate R. The resulting code word is modulated with a modulation of order M. A preamble and a postamble are added at the beginning and at the end of each packet, and pilot symbols are distributed inside the packet for the purpose of channel estimation. The total packet length is equal to L symbols. Before transmission on the RA channel, the symbols corresponding to each packet enter a shaping filter with a square root raised cosine function. At the receiver side, the frame is processed using CRDSA. When all the packets are in a non-decodable situation, the receiver applies MARSALA in order to attempt to recover additional packets and possibly trigger more SIC iterations.

Let us consider N_b received signals, r_1 and r_i (with $i \in [2, N_b]$), corresponding to N_b replicas of a same packet transmitted on separate timeslots of the frame. Figure 5.1 illustrates how those signals are transmitted and received on the frame.

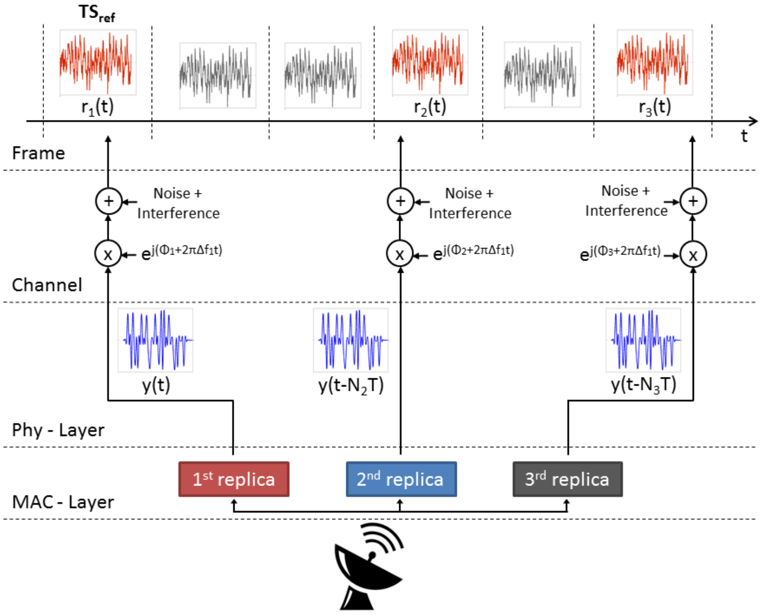


Figure 5.1 – Illustration of the received signals $r_1(t)$, $r_2(t)$ and $r_3(t)$ corresponding to three replicas of a same packet transmitted on separate timeslots on the frame.

As explained in the previous chapter, in MARSALA, the receiver identifies one timeslot as a reference timeslot, that we denote by TS_{ref}. This reference timeslot contains the first replica of an arbitrary packet plus other interference packets. Therefore, the signal received on TS_{ref} can be expressed as follows

$$r_1(t) = y(t + \tau_1) e^{j(\phi_1 + 2\pi\Delta f_1 t)} + n_1(t) + \zeta_1(t) \quad , \tag{5.1}$$

with y being the useful signal corresponding to the first replica of an arbitrary packet. τ_1, ϕ_1, and Δf_1 denote respectively the timing offset, the phase shift, and the frequency offset relative to the signal y on TS_{ref}. n_1 is the AWGN term and ζ_1 represents the sum of the interference signals on TS_{ref}. The signal y can be detailed as follows

$$y(t) = \sum_{k=0}^{L-1} a_k h(t - kT) \quad , \tag{5.2}$$

with a_k being the k^{th} symbol transmitted in y, and h denoting the shaping filter function. The signal r_i received on other timeslots than TS_{ref} and containing the i^{th} replica of y, can be expressed as follows

$$r_i(t) = y(t - N_i T + \tau_i) e^{j(\phi_i + 2\pi\Delta f_1 t)} + n_i(t) + \zeta_i(t) \quad , \tag{5.3}$$

where N_i is the number of symbols separating the first replica of the useful packet on TS_{ref} from its i^{th} replica on another timeslot of the frame. τ_i and ϕ_i refer respectively to the timing offset and the phase shift relative to y on the timeslot containing the i^{th} packet replica. n_i and ζ_i are respectively, the AWGN term and the sum of the interference signals on the timeslot containing the i^{th} replica.

5.2 Replicas synchronisation procedure

This section aims to describe a synchronisation scheme for the received packet replicas in MARSALA. The main objective is to be able to perform coherent replicas combination. In the following, we first recall the method for replicas localisation using a correlation based technique. Then, we describe the procedure to estimate and compensate the timing offsets and phase shifts between localised replicas.

5.2.1 Replicas localisation using a correlation based technique

To localise the replicas of a given packet, the receiver computes the cross-correlation between r_1 and each signal received on each remaining timeslot of the frame. The highest $(N_b - 1)$ correlation peaks should indicate the locations of the replicas of a packet present on TS_{ref}.

The signal r_i described in Equation (5.3), can be expressed in function of r_1, n_1, ζ_1, n_i and ζ_i as shown below

$$r_i(t) = r_1(t - N_i T + \Delta\tau_{i,1}) e^{j\Delta\phi_{i,1}} + n_{tot}(t) \quad , \tag{5.4}$$

where $N_i T - \Delta\tau_{i,1} = N_i T - (\tau_i - \tau_1)$ is the timing offset between the packet on TS_{ref} and its i^{th} replica. $\Delta\phi_{i,1} = \phi_i - \phi_1 + 2\pi\Delta f_1(N_i T - \ \Delta\tau_{i,1})$ is the phase shift between the two replicas. n_{tot} is a signal including noise and interference signals expressed as follows:

$$n_{tot}(t) = [n_i(t) + \zeta_i(t)] - [n_1(t - N_i T + \Delta\tau_{i,1}) + \zeta_1(t - N_i T + \Delta\tau_{i,1})] e^{j\Delta\phi_{i,1}} \quad . \tag{5.5}$$

Correlation of the reference signal with the signal on the other timeslots

Without loss of generality, we will only express the cross-correlation function between r_1 and r_i. Since both signals are supposed to have finite power, the time domain correlation between r_i and r_1 , denoted by $R_{r_i,r_1}(\tau)$, is expressed as shown below,

$$\begin{aligned} R_{r_i,r_1}(\tau) &= \int_0^{T_{slot}} r_i(t) r_1^*(t-\tau) dt \\ &= \int_0^{T_{slot}} r_1(t - N_i T + \Delta\tau_{i,1}) r_1^*(t-\tau) e^{j\Delta\phi_{i,1}} dt + \int_0^{T_{slot}} n_{tot}(t) r_1^*(t-\tau) dt \\ &= R_{r_1}\left(\tau - (N_i T - \Delta\tau_{i,1})\right) e^{j\Delta\phi_{i,1}} + R_{n_{tot},r_1}(\tau) \quad , \end{aligned} \tag{5.6}$$

with R_{r_1} denoting the autocorrelation function of r_1 and R_{n_{tot},r_1} being the cross-correlation function between n_{tot} and r_1. The operator $(\cdot)^*$ denotes the complex conjugate.

Correlation peaks detection

The maximum correlation peaks can be detected on the timeslots containing the replicas of the packet corresponding to the desired signal y. In other words, the absolute value of the correlation function $R_{r_i,r_1}(\tau)$ reaches its maximum when $\tau = \tau_{max,i} = N_i T - (\tau_i - \tau_1)$, which represents the timing offset between the first and the i^{th} replica of the same packet. The peak correlation amplitude in this case is equal to:

$$\left|R_{r_i,r_1}(\tau_{max,i})\right| \approx \left|R_{r_1}(0)\right| + \left|R_{n_{tot},r_1}(\tau_{max,i})\right| \quad . \tag{5.7}$$

This replicas localisation procedure is similar to the popular non coherent correlation detection method used for frame synchronisation [79, 80]. As shown in [81], this approach can be used to detect the location of a Synchronisation Word (SW) inside a frame. In MARSALA, replicas localisation follows the same principle except that the correlation is performed between the signals on separate timeslots to detect the packet replicas. It was shown in Fig. 9 in [81] that the error probability for peak detectors using non-coherent correlation and a SW of length 64 symbols, is around 10^{-4} for E_s/N_0 values as low as -1 dB. In MARSALA, this probability is further decreased since the entire signal on TS_{ref} is used to detect packet replicas. Hence, a much longer symbols sequence than the case of frame synchronisation is used (for instance, if the DVB-RCS2 waveform id 3 is used, the packet length is equal to 536 symbols).

Thus, the correlation peaks allow to determine the timeslots containing replicas of a same packet in MARSALA. In the case of $N_b = 2$ replicas, the signals on the identified timeslots are combined two-by-two. Whereas, in the case of $N_b > 2$ replicas, the timeslots are associated to each packet as described in Section 4.3.1 of Chapter 4. Then, the identified replicas for a same packet are combined.

5.2.2 Estimation and Compensation of the timing offsets and phase shifts between replicas

At this stage, we can assume that replicas of a given packet on TS_{ref} are localised. However, given that the replicas are received on distinct timeslots, they have different timing offsets and phase shifts due to carrier frequency and phase variations. For instance, in order to combine r_1 and r_i coherently, proper timing and phase synchronisation between the replicas is required. Therefore, before proceeding to replicas combination, the receiver in MARSALA must estimate and compensate the timing offset $\tau_{max_i} = N_i T - (\tau_i - \tau_1)$ and the phase shift $\Delta\phi_{i,1} = \phi_i - \phi_1 + 2\pi\Delta f_1(N_i T + \ \Delta\tau_{i,1})$ between the received replicas. Given that τ_{max_i} and $\Delta\phi_{i,1}$ are supposed to remain constant over the duration of one packet, they can be computed using the cross-correlation function between both signals r_1 and r_i.

The estimate of the timing offset between the replicas can be obtained as the value of τ that maximises the amplitude of $R_{i,1}$ in (5.6). Nonetheless, given that the received signal is oversampled with an oversampling factor Q, if τ_{max_i} is not an integer multiple of $\frac{T}{Q}$, the timing offset estimate will be detected with a timing error err_i. The timing error can be modelled with a random variable uniformly distributed in $[-T/2Q, T/2Q]$. Then, the estimate of the timing offset using correlation can be written as shown in the following:

$$\hat{\tau}_{max_i} = \underset{\tau}{\operatorname{argmax}}\left|R_{r_i,r_1}(\tau)\right| = N_i T_s - (\tau_i - \tau_1) + err_i \quad . \tag{5.8}$$

The value of the cross-correlation function $R_{r_i,r_1}(\tau)$ for $\tau = \hat{\tau}_{max_i}$ is

$$\begin{aligned}
R_{r_i,r_1}(\hat{\tau}_{max_i}) =& R_{r_1}(err_i)\, e^{j\Delta\phi_{i,1}} + R_{n_{tot},r_1}(\hat{\tau}_{max_i}) \\
=& \int_0^{T_{slot}} y(t) y^*(t - err_i) e^{j(\Delta f_1 err_i + \Delta\phi_{i,1})} dt + R_{tot} \\
=& \int_0^{T_{slot}} \sum_{k=0}^{L-1} a_k h(t - kT) \sum_{k=0}^{L-1} a_k^* h^*(t - kT - err_i) dt + R_{tot} \\
=& \sum_{k=0}^{L-1} |a_k|^2 \int_0^{T_{slot}} h(t - kT) h^*(t - kT - err_i) e^{j(\Delta f_1 err_i + \Delta\phi_{i,1})} dt \\
&+ \int_0^{T_{slot}} \sum_{k=0}^{L-1} \sum_{k' \neq k}^{L-1} a_k a_{k'}^* h(t - kT) h^*(t - k'T - err_i) e^{j(\Delta f_1 err_i + \Delta\phi_{i,1})} dt + R_{tot} \\
\simeq& LP(a_k) \left[\frac{H(err_i)}{T}\right] e^{j(\Delta f_1 err_i + \Delta\phi_{i,1})} + R_{tot} \quad ,
\end{aligned} \tag{5.9}$$

with $H(t)$ and $h(t)$ denoting respectively the raised cosine filter function and its square root version. $P(a_k)$ is the power of the transmitted symbol a_k, which we will consider normalised to one for all values of $k \in [0, L-1]$. The total cross-correlation terms denoted by R_{tot} are detailed as follows:

$$R_{tot} = R_{y,(n_1+\zeta_1)}(\widehat{\tau}_{max_i}) + R_{(n_1+\zeta_1),y}(\widehat{\tau}_{max_i}) + R_{(n_1+\zeta_1),(n_1+\zeta_1)}(\widehat{\tau}_{max_i}) + R_{n_{tot},r_1}(\widehat{\tau}_{max_i}) \quad . \tag{5.10}$$

Thus, the estimate of the phase shift $\widehat{\Delta\phi}_{i,1}$ can be computed as the angle of $R_{r_i,r_1}(\widehat{\tau}_{max_i})$. As shown in Equation (5.11), $\widehat{\Delta\phi}_{i,1}$ contains a phase error ϕ_{err_i} that depends on $\Delta f_1 err_i$, $H(err_i)$ and the cross-correlation functions between noise, interference and the useful signal.

$$\widehat{\Delta\phi}_i = \text{angle}\left(R_{r_i,r_1}(\widehat{\tau}_{max_i})\right) = \Delta\phi_i + \phi_{err_i} \quad . \tag{5.11}$$

It can be observed from the second term of Equation (8.19), that in order to minimize the error ϕ_{err_i}, we shall choose a reference signal r_1 containing the lowest interference level. Therefore, since all the replicas of a same packet are equi-powered, then among the signals corresponding to replicas of a same packet, the signal having the lowest total received power on one timeslot shall have the lowest interference power of ζ_1. In other words, in order to estimate $\Delta\phi_{i,1}$ after replicas localisation, the receiver shall choose r_1 as the received signal on one localised timeslot with the lowest power level.

Once the timing offset and phase shift between replicas are estimated, they are compensated as explained in the following. To correct the timing offset of the i^{th} replica with respect to the replica on TS_{ref}, the signal r_i is oversampled with a factor Q at time instants $t = n\frac{T}{Q} + \widehat{\tau}_{max_i}$, with n being an integer index varying from 0 to $(QL-1)$. Then, to correct the phase shift, the resulting samples are multiplied by $e^{-j\widehat{\Delta\phi}_i}$. The n^{th} sample of r_i obtained after timing and phase corrections is expressed as follows:

$$r_i(n) = y\left(n\frac{T}{Q} + \tau_1 + err_i\right)e^{j\left(\phi_1+\phi_{err_i}+2\pi\Delta f_1 n\frac{T}{Q}\right)} + \left[n_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right) + \zeta_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right)\right]e^{-j\widehat{\Delta\phi}_{i,1}} \quad . \tag{5.12}$$

Let us also note that the discrete form of the signal r_1 obtained after oversampling with a factor Q, can be expressed as shown below:

$$r_1(n) = y\left(n\frac{T}{Q} + \tau_1\right)e^{j\left(\phi_1+2\pi\Delta f_1 n\frac{T}{Q}\right)} + n_1\left(n\frac{T}{Q}\right) + \zeta_1\left(n\frac{T}{Q}\right) \quad . \tag{5.13}$$

5.3 Replicas combination & channel estimation

After estimation and compensation of timing and phase according to the scheme described above, replicas combination is performed. Then, the channel parameters τ_1, ϕ_1 and Δf_1 shall be estimated before demodulation and decoding of the combined signal.

5.3.1 Replicas combination scheme

Replicas combination consists in the summation of r_1 (From Equation (8.23)) and r_i (from Equation (8.22)) for $i = [2, \cdots, N_b]$. The n^{th} sample of the resulting discrete signal r_{sum} is expressed as shown below:

$$r_{sum}(n) = y_{sum}(n) e^{j\left(\phi_1 + 2\pi\Delta f_1 n \frac{T}{Q}\right)} + z_1\left(n\frac{T}{Q}\right) + \sum_{i=2}^{N_b} z_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right) e^{-j\widehat{\Delta\phi}_i} \quad , \tag{5.14}$$

where z_1 and z_i refer to the noise plus interference term in signals r_1 and r_i, respectively. The signal y_{sum} is detailed as follows:

$$y_{sum}(n) = y\left(n\frac{T}{Q} + \tau_1\right) + \sum_{i=2}^{N_b} y\left(n\frac{T}{Q} + \tau_1 + err_i\right) e^{j\phi_{err_i}} \quad . \tag{5.15}$$

5.3.2 Channel estimation

Before demodulation and decoding, the receiver shall perform channel estimation for the useful signal y_{sum}. The challenge is to be able to estimate the channel parameters τ_1, Δf_1 and ϕ_1, in low SNIR conditions and without a priori decoded information. Therefore, we propose to use an algorithm employing data-aided channel estimation and joint estimation and decoding (JED). Data-aided channel estimation uses sequences of training symbols (i.e. preamble, postamble and pilots), as a form of pseudo-random codes known at the receiver for each user. The channel estimation procedure is described in Figure 5.2.

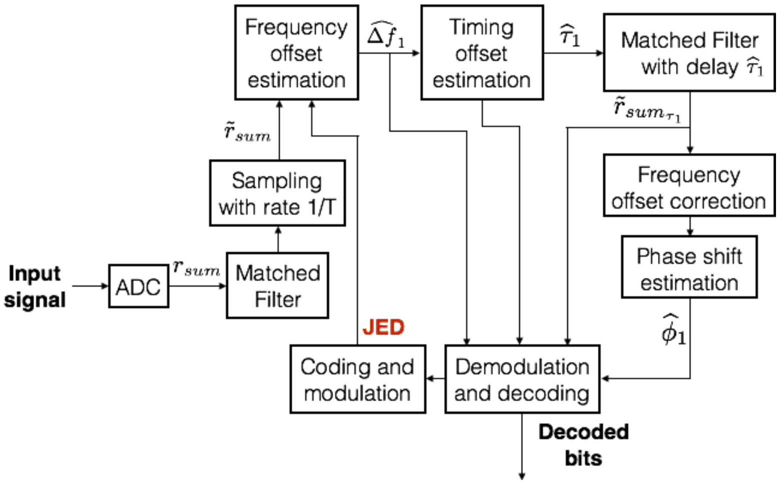


Figure 5.2 – Channel estimation scheme to estimate τ_1, Δf_1 and ϕ_1 for the combined replicas.

First, a frequency offset estimation is done using the matched filtered version of r_{sum} denoted with $\tilde{r}_{sum}$ and the training symbols sequence y_{train} of user 1. The frequency estimate is computed as the value of Δf that maximises the square amplitude of the correlation peak obtained from the cross-correlation function shown below:

$$\widehat{\Delta f}_1 = \underset{\Delta f}{argmax} \left| \sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum}(n) y_{train}(n-m) e^{-j2\pi\Delta f n \frac{T}{Q}} \right|^2 \quad . \tag{5.16}$$

In order to estimate the timing offset τ_1, we compensate the frequency offset of $\tilde{r}_{sum}$ according to the estimated frequency offset $\widehat{\Delta f}_1$. Then the resulting signal, along with the training sequence, are used to estimate the timing offset τ_1 as shown in the following equation:

$$\widehat{\tau}_1 = \underset{n}{argmax} \left| \sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum}(n) y_{train}(n-m) e^{-j2\pi\widehat{\Delta f}_1 n \frac{T}{Q}} \right|^2 \quad . \tag{5.17}$$

Once τ_1 is estimated, the discrete signal r_{sum} is matched filtered by taking into account the estimated timing offset. We refer to the resulting discrete signal by $\tilde{r}_{sum_{\tau_1}}$. Then, the phase shift ϕ_1 is estimated by computing the angle of the peak cross-correlation as shown below:

$$n_{peak} = \underset{n}{argmax} \left| \underbrace{\sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum_{\tau_1}}(n) y_{train}(n-m) e^{-j2\pi\widehat{\Delta f}_1 n \frac{T}{Q}}}_{R_{\widehat{\phi}_1}(n)} \right|^2 \tag{5.18}$$

$$\widehat{\phi}_1 = angle\left(R_{\widehat{\phi}_1}(n_{peak})\right) \tag{5.19}$$

In order to refine the estimation of the channel parameters, joint estimation and decoding is applied. In other words, after the first channel estimation and demodulation and decoding procedure, the decoded bits are re-encoded and re-modulated to be used instead of the training symbols in an iterative channel estimation process.

5.3.3 Matched filter & downsampling

In this final step, the discrete signal r_{sum} is matched filtered with respect to the estimated timing offset $\widehat{\tau}_1$, and downsampled at instants $k'T_s$, with k' being an integer varying from 0 to $L-1$. After these operations, the resulting signal $\tilde{r}_{sum_{\tau_1}}$ is expressed as follows:

$$\tilde{r}_{sum_{\tau_1}}(k'T) = \tilde{y}_{sum_{\tau_1}}(k'T) e^{j(\phi_1 + 2\pi\Delta f_1 k'T)} + \tilde{z}_1(k'T) + \sum_{i=2}^{N_b} \tilde{z}_i(k'T + \widehat{\tau}_{max_i}) e^{-j\widehat{\Delta\phi}_{i,1}} \quad , \tag{5.20}$$

where $\tilde{(\cdot)}$ denotes the signal at the output of the matched filter, and $\tilde{y}_{sum_{\tau_1}}$ is detailed as follows:

$$\tilde{y}_{sum_{\tau_1}}(k'T) = \tilde{y}\left(k'T + \underbrace{(\tau_1 - \widehat{\tau}_1)}_{err_1}\right) + \sum_{i=2}^{N_b} \tilde{y}\left(k'T + \underbrace{(\tau_1 - \widehat{\tau}_1)}_{err_1} + err_i\right) e^{j\phi_{err_i}} \quad , \tag{5.21}$$

where $err_1 = \tau_1 - \widehat{\tau}_1$ is supposed to follow a uniform distribution in $\left[\frac{-T}{2Q}, \frac{T}{2Q}\right]$.

5.4 An Analytical Model for Replicas Combining with Synchronisation Errors

As seen in (5.21), the replicas combined in $\tilde{y}_{sum_{\tau_1}}$ do not have exactly the same timing offset and phase. In fact, after replicas synchronisation, residual timing offsets and phase shifts estimation errors (err_i and ϕ_{err_i}) are added to the combined replicas. In order to evaluate the performance of MARSALA with imperfect replicas combination, we define an analytical model describing the impact of imperfect replicas combination on the performance degradation of MARSALA, compared to the perfect Channel State Information (CSI) case.

We can conclude from (5.21), that due to the timing offset existing between replicas, the combined signal $\tilde{y}_{sum_{\tau_1}}$ can be divided into two terms: a desired signal denoted by $y_{sum,des}$ and an inter-symbol interference (ISI) term denoted by $y_{sum,isi}$. Both terms are detailed below:

$$y_{sum,des}(k'T) = a_{k'}H(err_1) + \sum_{i=2}^{N_b} a_{k'}H\underbrace{(err_1 + err_i)}_{err_{1,i}}e^{j\phi_{err_i}} \quad , \tag{5.22}$$

with $a_{k'}$ referring to the k'^{th} symbol corresponding to y and H being the raised cosine function. $err_{1,i} = err_1 + err_i$ follows a triangular distribution between $-T/Q$ and T/Q. As for the ISI term, we consider only the first three side lobes on either side of the raised cosine filter, since the interference symbol would be attenuated by more than 10 dB beyond these lobes. In fact, for the fourth interference symbol $sinc(4.5) = 0.07 \Rightarrow -11$ dB. Then, we can express the ISI term of y_{sum} as follows:

$$y_{sum,isi}(k'T) = \sum_{l=k'-3,l\neq k'}^{k'+3} a_l H\big((k'-l)T + err_1\big) + \sum_{i=2}^{N_b}\left(\sum_{l=k'-3,l\neq k'}^{k'+3} a_l H\big((k'-l)T + err_{1,i}\big)\right)e^{j\phi_{err_i}}. \tag{5.23}$$

We deduce from (5.20), (5.22) and (8.33), that the average equivalent SNIR of the combined signal is:

$$SNIR_{eq} = \frac{E\left[P\left(y_{sum,des}\right)\right]}{E\left[P\left(y_{sum,isi}\right)\right] + (I_1 + N_0) + \sum_{i=2}^{N_b}(I_i + N_0)} \quad , \tag{5.24}$$

with $P\left(y_{sum,des}\right)$, $P\left(y_{sum,isi}\right)$, I_1 and I_i being the power of $y_{sum,des}$, $y_{sum,isi}$, and the interference terms on TS_{ref} and on the timeslot containing the i^{th} replica, respectively.

5.4.1 Average power of the desired signal

Let us detail the term $P\left(y_{sum,des}\right)$ as shown below:

$$P(y_{sum,des}) = \frac{1}{L}\sum_{k'=0}^{L-1}\left|y_{sum,des}(k'T_s)\right|^2 \quad ,$$

$$\begin{aligned} P(y_{sum,des}) = H^2(err_1) + \sum_{i=2}^{N_b} H^2(err_{1,i}) \left[cos^2(\phi_{err_i}) + sin^2(\phi_{err_i})\right] \\ + 2\sum_{i,j\neq i}^{N_b} H(err_{1,i})H(err_{1,j})cos(\phi_{err_i})cos(\phi_{err_j}) \\ + 2\sum_{i,j\neq i}^{N_b} H(err_{1,i})H(err_{1,j})sin(\phi_{err_i})sin(\phi_{err_j}) + 2H(err_1)\sum_{i=2}^{N_b} H(err_{1,i})cos(\phi_{err_i}) \quad . \end{aligned} \tag{5.25}$$

Given that timing offsets and phase shifts in (8.35) are random variables, we compute the average power of the desired signal $E\left[P(y_{sum,des})\right]$. The average value of each of the terms in (8.35) is detailed below. We suppose that the raised cosine function H can be approximated to a sine cardinal ($sinc$) function on the interval $[-T, T]$, and $err_1 \sim \mathcal{U}\left(\frac{-T}{2Q}, \frac{T}{2Q}\right)$, then $E[H(err_1)]$ and $E[H^2(err_1)]$ can be computed as shown in (5.26) and (5.27) respectively.

$$E\left[H(err_1)\right] = \frac{2Q}{T}\int_0^{\frac{T}{2Q}} sinc\left(\pi\frac{err_1}{T_s}\right)d(err_1) = \frac{2Q}{\pi}Si\left(\frac{\pi}{2Q}\right) \quad , \tag{5.26}$$

with Si being the sine integral function.

$$E\left[H^2(err_1)\right] = \frac{2Q}{T}\int_0^{\frac{T}{2Q}} sinc^2\left(\pi\frac{err_1}{T}\right)d(err_1) = \frac{2Q}{\pi}\left(Si\left(\frac{\pi}{Q}\right) - \frac{sin^2\left(\frac{\pi}{2Q}\right)}{\frac{\pi}{2Q}}\right) \quad . \tag{5.27}$$

Assuming that err_1, $err_{1,i}$ and $err_{1,j}$ are independant random variables and given that $err_{1,i} \sim \Lambda\left(\frac{-T_s}{Q}, \frac{T_s}{Q}\right)$, then we can derive $E[H(err_{1,i})]$, $E[H^2(err_{1,i})]$ and $E[H(err_{1,i})H(err_{1,j})]$ as shown in (5.28), (5.29) and (5.30) respectively.

$$\begin{aligned} E\left[H(err_{1,i})\right] &= \frac{2M}{T}\int_{-\frac{T}{M}}^{0} \left(\frac{Merr_{1,i}}{T} + 1\right) sinc\left(\frac{\pi err_{1,i}}{T}\right) d(err_{1,i}) \\ &= \frac{2M}{\pi}\left(Si\left(\frac{\pi}{M}\right) + \frac{M}{\pi}\left(cos\left(\frac{\pi}{M}\right) - 1\right)\right) \quad . \end{aligned} \tag{5.28}$$

$$\begin{aligned} E\left[H^2(err_{1,i})\right] &= \frac{2M}{T}\int_{-\frac{T}{M}}^{0} \left(\frac{Merr_{1,i}}{T} + 1\right) sinc^2\left(\frac{\pi err_{1,i}}{T}\right) d(err_{1,i}) \\ &= \frac{2M}{\pi}\left(Si\left(\frac{2\pi}{M}\right) - \frac{sin^2\left(\frac{\pi}{M}\right)}{\frac{\pi}{2M}}\right) + \frac{M^2}{\pi^2}\left(Ci\left(\frac{2\pi}{M}\right) - \gamma + log\left(\frac{M}{2\pi}\right)\right) \quad , \end{aligned} \tag{5.29}$$

with Ci being the cosine integral function and γ the Euler-Mascheroni constant, $\gamma = 0.577216$.

$$E\left[H(err_{1,i})H(err_{1,j})\right] = \frac{4M^2}{\pi^2}\left(Si\left(\frac{\pi}{M}\right) + \frac{M}{\pi}\left(cos\left(\frac{\pi}{M}\right) - 1\right)\right)^2 \quad . \tag{5.30}$$

In order to compute the average value of the term $[cos(\phi_{err_i})cos(\phi_{err_j}) + sin(\phi_{err_i})sin(\phi_{err_j})]$, we shall estimate the Probability Distribution Function (PDF) of ϕ_{err_i} and ϕ_{err_j}, based on the phase shift estimation procedure described in the previous section. Figure 5.3 shows the PDF of ϕ_{err_i} obtained through simulations. Various simulation scenarios are considered with various number of equi-powered interference packets on each replica (one, two or three interference packets). The results show that ϕ_{err_i} can be approximated by a Gaussian variable of mean zero and variance $\sigma^2_{\phi_{err_i}}$, thus $\exp(j\phi_{err_i})$ follows a log-normal distribution. The average values of $cos(\phi_{err_i})$ and $sin(\phi_{err_i})$ are computed as shown below:

$$E[e^{j\phi_{err_i}}] = e^{-\frac{\sigma^2_{\phi_{err_i}}}{2}} \quad ; \quad E[cos(\phi_{err_i})] = e^{-\frac{\sigma^2_{\phi_{err_i}}}{2}} \quad ; \quad E[sin(\phi_{err_i})] = 0 \quad . \tag{5.31}$$

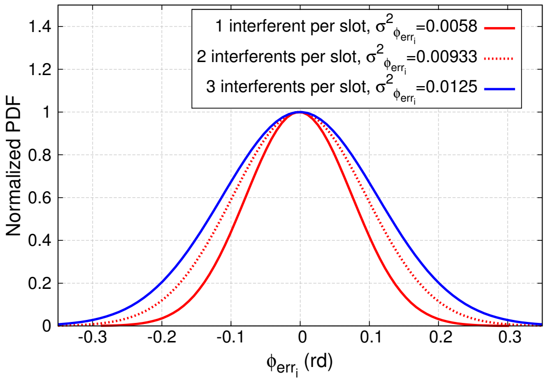


Figure 5.3 – Normalized PDF of ϕ_{err_i} for various numbers of interferents per timeslot.

5.4.2 Average power of the ISI term

To analyse the ISI term $y_{sum,isi}$ after signal combination, we choose to proceed as done in [82] for cooperative MISO systems with time synchronisation errors. Therefore we approximate the raised cosine pulse to a piecewise linear function with slopes m_l on each symbol instant $k'T$. The upper bound of the worst case of ISI is obtained when $a_l G((k'-l)T + err) = |m_l|\frac{err}{T}$ (i.e. when all the ISI symbols multiplied by the raised cosine function are of the same sign). Thus the worst case ISI term can be written as follows:

$$y_{sum,isi} = \underbrace{\left(\sum_{l=-3,l\neq 0}^{3} |m_l|\right)}_{\beta} \frac{|err_1|}{T} + \sum_{i=2}^{N_b} \underbrace{\left(\sum_{l=-3,l\neq 0}^{3} |m_l|\right)}_{\beta} \frac{|err_{1,i}|}{T} \quad . \tag{5.32}$$

Since $err_1 \sim \mathcal{U}\left(\frac{-T}{2Q},\frac{T}{2Q}\right)$ and $err_{1,i} \sim \Lambda\left(\frac{-T}{Q},\frac{T}{Q}\right)$, then $|err_1| \sim \mathcal{U}\left(0,\frac{T_s}{2Q}\right)$ and $|err_{1,i}| \sim \Lambda\left(0,\frac{T_s}{Q}\right)$. The mean and variance values of $y_{sum,isi}$ are given in (5.33) and (5.34) respectively.

$$E\left[y_{sum,isi}\right] = \frac{\beta}{4Q} + \sum_{i=2}^{N_b} \frac{\beta}{3Q} \quad , \tag{5.33}$$

$$var\left[y_{sum,isi}\right] = \frac{\beta^2}{48Q^2} + \sum_{i=2}^{N_b} \frac{\beta^2}{18Q^2} \quad . \tag{5.34}$$

Then the average power of the worst case ISI term is given below.

$$E\left[P\left(y_{sum,isi}\right)\right] = E\left[\left|y_{sum,isi}\right|^2\right] = var\left[s_{sum,isi}\right] + E^2\left[s_{sum,isi}\right] \quad . \tag{5.35}$$

5.5 Numerical results

In this section, we provide the numerical results obtained in the case of imperfect replicas combination in MARSALA. Based on the analytical model described in the previous section, the performance of MARSALA in real channel conditions is evaluated in terms of Packet Error Rate (PER) and MAC-layer throughput and Packet Loss Ratio (PLR).

5.5.1 Numerical evaluation of the equivalent SNIR with imperfect replicas combination

In this section, we analyse the numerical degradation of the average equivalent SNIR in MARSALA, caused by imperfect replicas combination. Table 5.1 and Table 5.2 show the average equivalent SNIR obtained in real channel conditions (denoted by $SNIR_{eq}$ and computed with Equation (8.34)), compared to the equivalent SNIR with no error model (denoted by $SNIR_{eq_{ref}}$), for $N_b = 2$ and $N_b = 3$ replicas respectively. The oversampling factor considered is $Q = 4$ and $E_s/N_0 = 7$ dB. Both cases with $\sigma^2_{\phi_{err_i}} = 0$ and $\sigma^2_{\phi_{err_i}} = 0.0125$ (worst case scenario for ϕ_{err_i}) are taken into account.

$SNIR_{eq_{ref}}$ (dB)	-2	-1	0
$SNIR_{eq}$ (dB), $\sigma^2_{\phi_{err_i}} = 0$	-2.24	-1.26	-0.3
$SNIR_{eq}$ (dB), $\sigma^2_{\phi_{err_i}} = 0.0125$	-2.25	-1.276	-0.31

Table 5.1 – Average anaytical $SNIR_{eq}$ degradation for imperfect replicas combination, with $N_b = 2$ replicas and $E_s/N_0 = 7$ dB.

We can notice that the results for $\sigma^2_{\phi_{err_i}} = 0$ and $\sigma^2_{\phi_{err_i}} = 0.0125$ are approximately the same. Then, we can conclude that the degradation is mainly caused by the timing offsets. We can also observe that the degradation with $N_b = 3$ replicas is slightly higher than $N_b = 2$ replicas (0.03 dB higher). This is due to the fact that the ISI power is increased when combining 3

$SNIR_{eq_{ref}}$ (dB)	-2	-1	0
$SNIR_{eq}$ (dB), $\sigma^2_{\phi_{err_i}} = 0$	-2.26	-1.29	-0.32
$SNIR_{eq}$ (dB), $\sigma^2_{\phi_{err_i}} = 0.0125$	-2.28	-1.31	-0.35

Table 5.2 – Average anaytical $SNIR_{eq}$ degradation for imperfect replicas combination, with $N_b = 3$ replicas and $E_s/N_0 = 7$ dB.

replicas instead of 2. The analytical bounds for the average SNIR degradation compared to the perfect CSI case are between $[0.2, 0.35]$ dB when the reference SNIR varies in the range $[-2, 0]$ dB.

To validate the analytical bounds obtained, we evaluate, via simulations, the Packet Error Rate (PER) achieved with MARSALA in real channel conditions. The numerical results are shown in Figure 5.4 for a particular scenario described as follows: Two timeslots with replicas of a same packet are combined, each containing 3 packets in collision (the useful packet + 2 interference packets). The timing offsets of the first and second replica respectively, are $\tau_1 = 0.25\frac{T}{Q}$ and $\tau_2 = -0.25\frac{T}{Q}$, so as to obtain the worst case value of $err_2 = \pm 0.5\frac{T}{Q}$. The replicas are affected by a different phase shift on each timeslot. The phase shifts and timing offsets between replicas are estimated and compensated as described in Section 5.2.2, then the replicas are combined and the channel parameters relative to the first replica are estimated as described in Section 5.3.2. The signal resulting from replicas combination is demodulated and decoded and the PER is computed for each $SNIR_{eq}$ value. In Figure 5.4, we use a QPSK modulation and DVB-RCS2 turbo code of rate $R = 1/3$ and the number of JED iterations for channel estimation in the real channel scenario is equal to 3. The PER obtained with the analytical average equivalent SNIR degradation, is also plotted. We can observe that the analytical results present an upper bound for the performance degradation caused by imperfect signal combination in MARSALA.

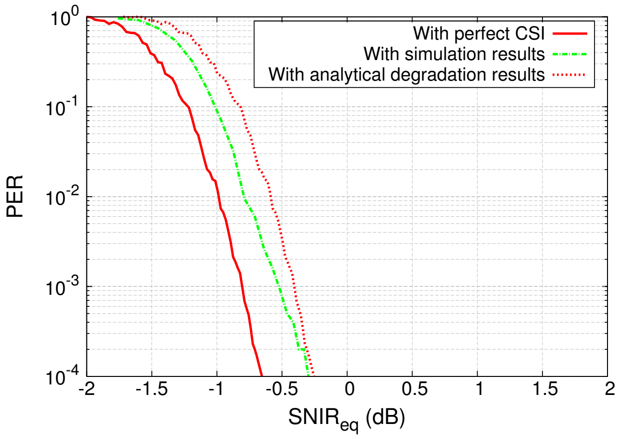


Figure 5.4 – Comparison between the analytical and the simulated PER obtained with real channel conditions in MARSALA, with $N_b = 2$ replicas.

5.5.2 Throughput and PLR simulation results

In the following, we use the numerical results of the analytical model proposed in this chapter, in order to evaluate the impact of synchronisation errors on the performance of MARSALA, in terms of throughput and PLR. In other words, in the experimental simulations, the equivalent SNIR obtained after replicas combination is computed based on Equation (8.34). In this scenario, all the packets are considered equi-powered. The number of terminals transmitting over one frame duration is N_u. Each frame is composed of $N_s = 100$ timeslots. The trafic profile tested is a Constant Bit Rate (CBR) profile. Residual channel estimation errors caused by imperfect interference cancellation have been taken into account. QPSK modulation and DVB-RCS2 turbo code for linear modulation of rate 1/3 are used in the simulation scenarios (denoted by waveform id 3 in the DVB-RCS2 guidelines [28]). The results obtained are compared with CRDSA in similar simulation scenarios. The scenario considered in our simulations is the following: once a frame is received, the receiver tries to decode a packet using CRDSA. If the decoding is not successful, the receiver applies MARSALA to localise and combine the replicas of this packet. This procedure is repeated iteratively, until no more packets can be decoded successfully. In order to compare several modulation and coding schemes, the normalised load (G) is expressed in bits per symbol and computed as

$$G = R \times \log_2(M) \times \frac{\lambda}{N_s} \quad ,$$

with R being the code rate and M the modulation order. The normalised throughput (T) is given by:

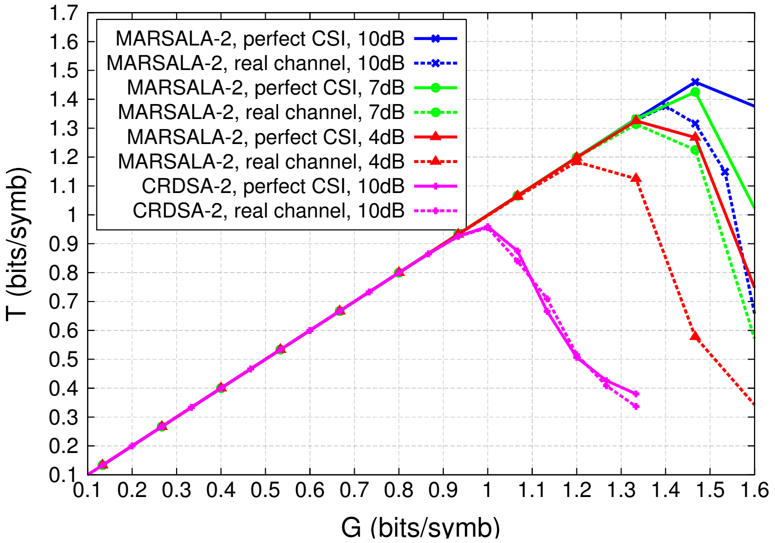
$$T = G \times (1 - PLR(G)) \quad ,$$

where PLR(G) is the probability that a packet is not decoded for a given G and a given SNIR.

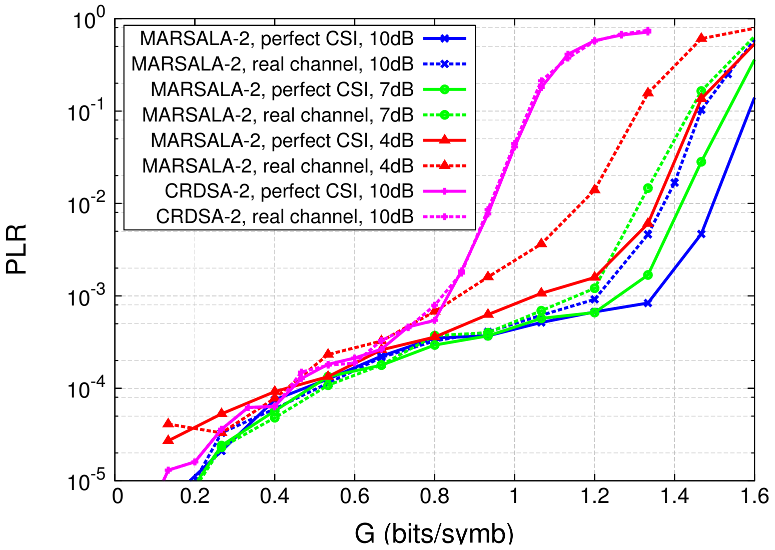
Based on the numerical results obtained in Section 5.5.1, Figure 5.5 and Figure 5.6 trace the throughput and the PLR obtained with MARSALA-2 and MARSALA-3. Two scenarios are compared: the case of perfect knowledge of Channel State Information (CSI) as well as the case of real channel conditions (simplified with the analytical model presented in this chapter). Both figures also show a comparison between the performance of MARSALA and CRDSA.

We can observe that, for MARSALA-2 in real channel conditions, the maximal throughput is degraded compared to the perfect CSI case, by around 12% to 14%, for $\frac{E_s}{N_0} = 10$ and 7 dB, and around 25% for $\frac{E_s}{N_0} = 4$ dB. Although the throughput is degraded in real channel conditions, but it is still 1.5 times higher than CRDSA in real channel conditions at $E_s/N_0 = 10$ dB. However, we can always observe a PLR floor between 10^{-4} and 10^{-3}. This is mainly caused by the so-called loop phenomenon when 2 packet replicas of a given user are in collision with the same 2 other replicas of another user.

For MARSALA-3, the degradation in real channel conditions is around 5% at $\frac{E_s}{N_0} = 10$ and 7 dB, and 10% at $\frac{E_s}{N_0} = 4$ dB. The degradation compared to perfect CSI is less significant for MARSALA-3 because the equivalent SNIR obtained with the combination of 3 replicas is



(a)



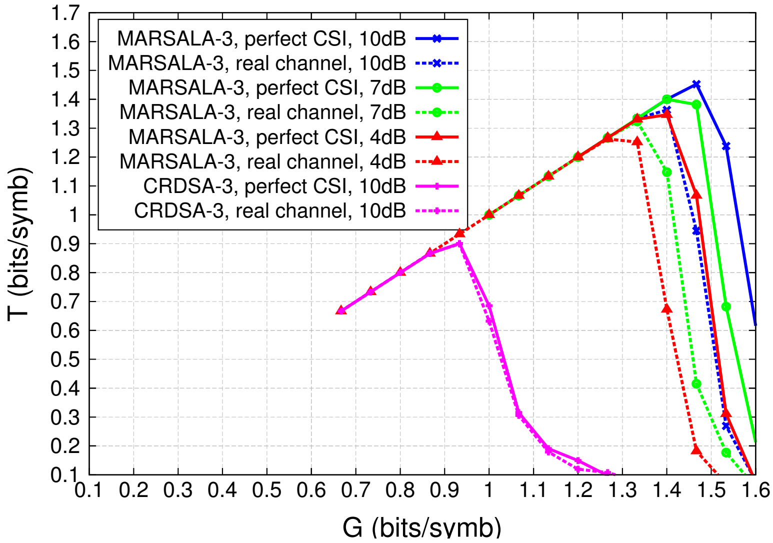
(b)

Figure 5.5 – MARSALA-2 performance in real channel conditions compared to perfect CSI. $E_s/N_0 = 4$, 7 and 10 dB; QPSK with DVB-RCS2 Turbocode $R = 1/3$. (a) Throughput. (b) PLR.

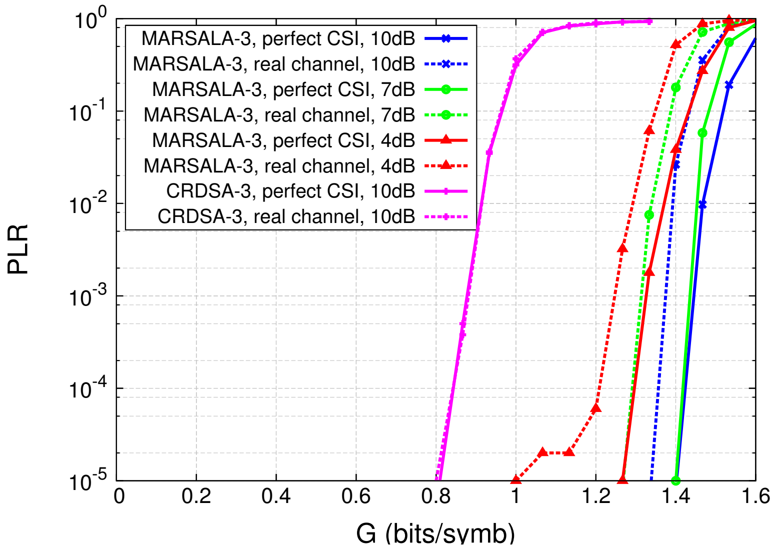
higher and permits more successful decoding than the case of 2 replicas, even in presence of synchronisation errors. The PLR floor at 10^{-3} in MARSALA-3 is mitigated because the probability of the loop phenomenon is lower.

5.6 Summary & conclusion

In this chapter, we designed a method to estimate and compensate the synchronisation errors between the replicas of a given packet, prior to signal combination in MARSALA. Based on



(a)



(b)

Figure 5.6 – MARSALA-3 performance in real channel conditions compared to perfect CSI for $E_s/N_0 = 4$, 7 and 10 dB. QPSK with DVB-RCS2 Turbocode $R = 1/3$. (a) Throughput. (b) PLR.

the proposed estimation method, we presented a detailed analytical model to evaluate the impact of signal misalignment on the combination performance in MARSALA. We showed that the correction of the timing offsets and phase shifts between the replicas of a given packet prior to their combination is a critical task in MARSALA, and it can degrade the performance, if not done accurately. However, even in presence of synchronisation and channel estimation errors, MARSALA still present significant performance gains compared to CRDSA. In order to further evaluate MARSALA combined with CRDSA, the next chapter will test the behaviour of MARSALA with a 3GPP turbo code and in the case of unbalanced packets power, as well as other performance enhancement schemes.

6

Enhancement Schemes for MARSALA

MARSALA was introduced in the previous chapters as a new decoding technique for CRDSA. We showed that even in presence of synchronisation errors between the replicas, MARSALA is able to significantly enhance the throughput and the PLR compared to many other RA schemes. In the previous chapter, we evaluated the performance of MARSALA in a particular scenario using the DVB-RCS2 coding scheme and considering equi-powered received packets. However, it has been shown in the literature that alternative coding schemes and unbalanced packets power can boost the performance of RA techniques. Therefore, in this chapter, we study the behaviour of MARSALA with various coding schemes and various packet power distributions, then we propose a configuration for optimal performance. This chapter also introduces an enhancement of MARSALA by adding the concept of Maximum Ratio Combining (MRC) to maximise the gain of replicas combination. We compare two different MRC techniques and we evaluate, via simulations, the gain achieved using MRC with several coding schemes and unbalanced packets. The simulation results demonstrate that the proposed enhancements to MARSALA show substantial performance gain in terms of throughput achieved for a target PLR. Part of the work presented in this chapter was accepted for publication in [83].

6.1 Introduction

As explained in Chapters 4 and 5, the need to maintain low signalling overhead and enhance the throughput motivated the proposition of MARSALA as a new decoding technique for CRDSA. MARSALA introduces a method for replicas localisation and decoding, in order to recover non-decoded packets in CRDSA. As explained in the previous chapters, the transmission scheme is the same as in CRDSA. However, some modifications should be taken into account at the receiver side. To localise the replicas of a given packet, MARSALA performs correlation of the signal received on one timeslot with the signals received on the other timeslots of the frame. When correlation peaks are detected, the replicas are combined to obtain a higher equivalent SNIR. To evaluate the performance of MARSALA in real channel conditions, a method to correct the timing offsets and phase shifts between the replicas before combination was presented in Chapter 5.

MARSALA showed good performance in a scenario of equi-powered packets using the DVB-RCS2 turbo code for linear modulation. However, it has been proven in previous studies, particularly for CRDSA and ESSA [27, 38, 45], that packets power unbalance significantly enhances the SIC performance. In addition, it has been shown that the choice of the coding scheme can also an impact on triggering more SIC iterations. For all those reasons, these enhancement schemes will be studied for MARSALA in this chapter. Overall, three new contributions are presented in this chapter, having the objective to enhance the performance of MARSALA RA scheme:

- Adding Maximum Ratio Combining (MRC) to MARSALA. Two techniques for applying MRC with MARSALA are proposed and evaluated via simulations.
- Evaluating MARSALA with packets power unbalance, as well as proposing packets power distributions to further enhance the performance.
- Study of MARSALA with other coding schemes than DVB-RCS2, such as 3GPP and CCSDS.

6.2 System model

In this section, we recall the same system assumptions as presented in Section 4.2 of Chapter 4. We consider a frame structure over one frequency carrier for the return link RA channel. The frame of length T_F is divided into N_s timeslots. The transmission scheme for each terminal is the same as defined in CRDSA. N_u is the total number of users sharing the RA transmission channel within a frame duration. Each user sends N_b replicas of the same packet on randomly chosen timeslots. We suppose that only one packet (with its replicas) per user is sent on the duration of one frame. Before transmission, each packet is encoded with a turbo code of rate R and modulated with QPSK. Finally, signalling fields for the purpose of channel estimation are added. We consider a scenario with fixed terminals communicating with a geostationary satellite. Therefore, we can assume a stationary channel model with the following parameters: **(1)** carrier frequency and timing offsets randomly distributed among different users, but constant among the replicas of a same user over the duration of one frame, **(2)** constant replicas amplitude over the frame duration and **(3)** random phase shifts among the replicas of a same packet. The phase shift is considered constant over the duration of one timeslot and phase noise fluctuations on a timeslot duration are neglected.

We recall the processing scheme at the receiver side: first, the frame is stored, then each timeslot is processed in order to recover the received packets. In a first step, the receiver attempts to recover the packets by using CRDSA. Each recovered packet is removed from the frame (after channel estimation) along with its replicas localised with the decoded pointers. The next step is to re-scan the frame and repeat the process of packets recovery with SIC until all the packets are decoded successfully. However, if CRDSA is blocked, i.e., the frame is scanned iteratively with no additional packets recovered due to strong collisions, MARSALA is applied. As described in Chapters 4 and 5, MARSALA chooses one reference timeslot TS_{ref} and correlates its signal with the other timeslots of the frame. The highest correlation peaks are used to identify the replicas of the packets present on TS_{ref}. Then, the timing offsets and phase shifts between the replicas of a same packet are corrected, and coherent replicas combination is performed. With the resulting SNIR and depending on the encoding scheme used, the packet has a higher successful decoding probability. Thus, if the packet is decoded successfully thanks to this SNIR gain, it will be removed from the frame. Then, the CRDSA scheme can be applied again in the next iteration. It is worth clarifying that, although frequency, timing and phase shifts between replicas were taken into account in Chapter 5, the fluctuating phase noise was not.

6.3 MARSALA scheme with MRC

Previously, MARSALA with Equal Gain Combining (EGC) was presented. In other words, the signals containing replicas of a same packet were added together in order to achieve a higher equivalent SNIR. In this section, we propose to add Maximum Ratio Combining (MRC) [84, 85] to MARSALA in order to effectively use the information from all the received packet replicas. As a matter of fact, MRC has been widely used in diversity reception communication systems such as Multiple-Input Multiple-Output (MIMO) and Multiple-Input Single-Output (MISO). Given that MARSALA is also a diversity-based transmission/reception method, we propose two MRC techniques for MARSALA: MRC based on packet SNIR knowledge and MRC based on total received power per timeslot.

In order to describe the MRC scheme for MARSALA, let us recall the signal received on the reference timeslot TS_{ref} as shown below:

$$r_1(t) = y(t)e^{j(2\pi\Delta f_1 t+\phi_1)} + n_1(t) + \zeta_1(t) \quad , \tag{6.1}$$

where y is the signal corresponding to the first replica of a given packet on TS_{ref}, n_1 and ζ_1 denote respectively the noise and interference terms on TS_{ref}. Δf_1, ϕ_1 are respectively the frequency offset and phase shift relative to the first replica. As mentioned in the previous section, we suppose that the frequency offset is the same for all replicas, however the phase shift varies randomly from one replica to another. Supposing that the k^{th} replicas of y are localised with correlation procedures, we express the signal containing the k^{th} replica as:

$$r_k(t) = y(t-\Delta\tau_k)e^{j(2\pi\Delta f_1 t+\phi_k)} + n_k(t) + \zeta_k(t) \quad , \tag{6.2}$$

with ϕ_k denoting the phase shift of the k^{th} replica and $\Delta\tau_k$ representing the timing offset between the replicas. n_k, ζ_k are the noise and interference terms on the timeslot containing the k^{th} replica. For sake of simplicity, we suppose that timing offsets and phase shifts between the replicas are perfectly compensated. Then, each signal r_k (with $k \in [1, N_b]$) is multiplied by a weighting factor α_k. The resulting combined signal is shown below:

$$r_{sum}(t) = \left(\sum_{k=1}^{N_b} \alpha_k\right) y(t) e^{j2\pi\Delta f_1 t + \phi_1} + \sum_{k=1}^{N_b} \alpha_k \left(n_k(t) + \zeta_k(t)\right) \quad . \tag{6.3}$$

Thus, the equivalent SNIR obtained with MARSALA can be expressed as shown below:

$$SNIR_{eq} = \frac{\left(\sum_{k=1}^{N_b} \alpha_k\right)^2 P_y}{\sum_{k=1}^{N_b} \alpha_k^2 (N_k + I_k)} = \frac{\left(\sum_{k=1}^{N_b} \alpha_k\right)^2}{\sum_{k=1}^{N_b} \alpha_k^2 SNIR_k^{-1}} \quad , \tag{6.4}$$

where P_y is the power of y and $N + I$ is the noise plus interference power term. If EGC is applied, $\alpha_k = 1$, otherwise α_k is chosen according to the MRC techniques described in the following.

6.3.1 MRC based on packet SNIR knowledge

According to Equation (6.4), the optimal value of $SNIR_{eq}$ is obtained when

$$\frac{d\left(SNIR_{eq}\right)}{d\alpha_k} = 0 \quad . \tag{6.5}$$

Thus, the optimal value for α_k is $\alpha_k = SNIR_k$. In this case, $SNIR_{eq}$ would be equal to:

$$SNIR_{eq,max} = \sum_{k=1}^{N_b} SNIR_k \quad . \tag{6.6}$$

Therefore, obtaining the maximum gain of MRC requires the knowledge of the received $SNIR$ for each replica on each timeslot. A number of SNIR estimation techniques as well as the impact of estimation errors on MRC, can be found in the literature [84, 85]. It has been demonstrated in [85] that data-aided channel estimation only slightly degrades MRC performance. Therefore, in this paper, as a first contribution to analyse the impact of MRC on MARSALA, we will consider perfect SNIR knowledge per timeslot.

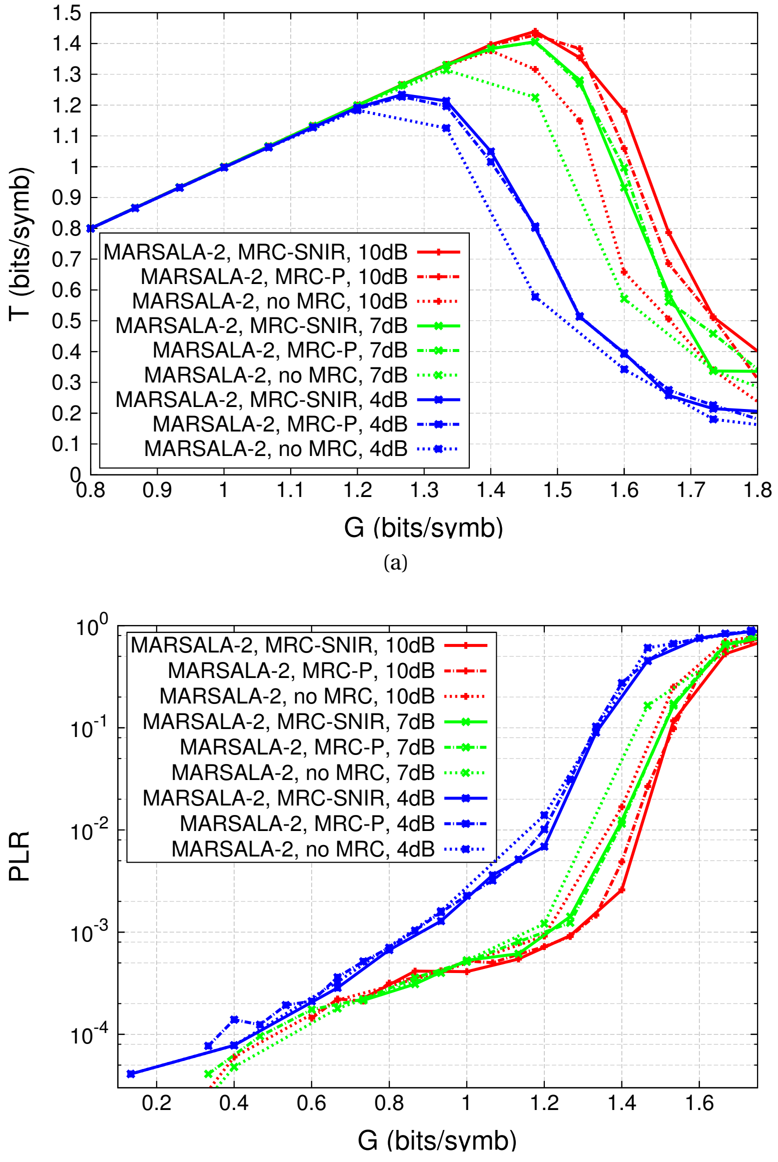
6.3.2 MRC based on received power per timeslot

In case we do not have proper SNIR estimation for each replica on each timeslot, we propose to use MRC based on received power per timeslot. This technique requires the condition that all the replicas of a given packet are transmitted with equal power, however the packets sent by distinct users can have different power levels. We also consider that on a frame duration, the attenuation is constant, so each set of packet replicas is received at the same power level. The concept of MRC based on received power per timeslot is the following: given that replicas of a same packet are equi-powered, then we can deduce the interference level on each timeslot by measuring P_k, the total received power on each timeslot. In other words, among the N_b signals on the timeslots containing replicas of a given packet, the signal having the highest received power contains the highest interference level. Therefore, once the interference level is known for each packet replica with the power measurement method, we can use this criteria to choose the MRC weighting factor as $\alpha_k = (P_k)^{-1}$.

6.3.3 Simulations results

Following the two MRC techniques described above, we evaluate the performance gain of MARSALA with MRC compared to EGC, in terms of throughput and PLR. In the simulations, CRDSA and MARSALA are combined according to the system model described in Section 6.2. For MARSALA, the impact of imperfect replicas combination on the performance is taken into account, following the model defined in Chapter 5. Also, the effect of residual channel estimation errors caused by imperfect interference cancellation [65, 66] is taken into consideration. In order to only evaluate the MRC gain metric, we consider that all the packets are equi-powered. QPSK modulation with DVB-RCS2 turbo code for linear modulation of rate 1/3 is used. Figure 6.1 and Figure 6.2 show the normalised throughput and PLR in function of the normalised load on the frame, obtained respectively with MARSALA-2 and MARSALA-3, for several values of E_s/N_0 and for a payload length equal to 456 symbols. The normalised load (G) and the normalised throughput (T) are expressed in bits per symbol. PLR(G) is the probability that a packet is not decoded for a given G and a given SNIR. Three schemes are compared in both figures 6.1 and 6.2: MARSALA without MRC, MARSALA with MRC based on packets SNIR (MRC-SNIR) and MARSALA with MRC based on received power per timeslot (MRC-P).

We can observe that the performance of MARSALA with MRC based on SNIR knowledge and MARSALA with MRC based on received power per timeslot, is nearly the same. Therefore, the complexity of SNIR estimation procedures in very low SNIR regimes, can be replaced with less complex operations of received power measurement per timeslot. Based on the results in Figure 6.1 and Figure 6.2, let us discuss the impact of MRC techniques on the performance of MARSALA. On the one hand, we notice that the throughput of MARSALA-2 is increased by around 5% at $E_s/N_0 = 4$, 7 and 10 dB. Yet, the PLR plots do not show an important enhancement for MARSALA-2. On the other hand, we can observe that the throughput of MARSALA-3 is increased by 5% at $E_s/N_0 = 4$ dB and 10% at $E_s/N_0 = 7$ and 10 dB, reaching a throughput up to 1.47 bits/symbol. In addition, we can notice that the throughput obtained with MARSALA-3 and MRC at 4 dB is approximately equal to the throughput obtained at 10 dB without MRC. The PLR performance above 10^{-5} does not present a floor for $E_s/N_0 = 7$ and 10 dB, but a floor starts to appear between 10^{-5} and 10^{-4} when $E_s/N_0 = 4$ dB. Given the

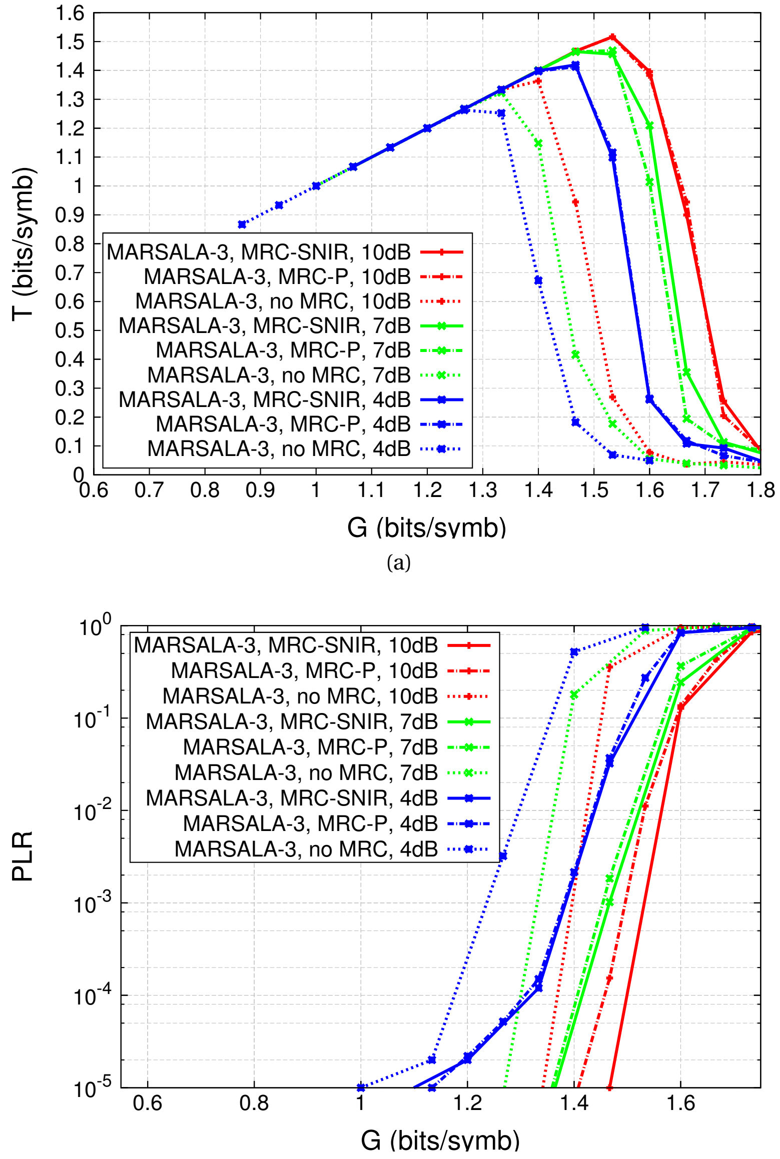


(a)

(b)

Figure 6.1 – Comparison between MARSALA without MRC, with MRC based on packet SNIR (MRC-SNIR) and with MRC based on received power per timeslot (MRC-P). $N_b = 2$ replicas. QPSK modulation, DVB-RCS2 turbo code $R = 1/3$, $E_s/N_0 = 4$, 7 and 10 dB. Equi-Powered packets. Payload length = 456 symbols. (a) Throughput. (b) PLR.

performance gain achieved, MRC will be taken into account with MARSALA for the simulations scenarios in the rest of this chapter.



(a)

(b)

Figure 6.2 – Comparison between MARSALA without MRC, with MRC based on packet SNIR (MRC-SNIR) and with MRC based on received power per timeslot (MRC-P). $N_b = 3$ replicas. QPSK modulation, DVB-RCS2 turbo code $R = 1/3$, $E_s/N_0 = 4$, 7 and $s10$ dB. Equi-Powered packets. Payload length = 456 symbols. (a) Throughput. (b) PLR.

6.4 MARSALA with Packets Power Unbalance

It has been shown in [27, 38] that received packets power unbalance between different users has a positive impact on the performance of CRDSA. In fact, power unbalance enables CRDSA to decode packets in collision thanks to the capture effect [39]. In other words, the strongest packets are decoded first with a higher successful decoding probability, then they are removed with SIC iterations. Thus, the weaker packets are less interfered and have a higher successful decoding probability as well.

In this section, we apply the same concept to MARSALA, given that it is also a diversity based RA method with SIC. First, in order to give a comparison with CRDSA, we study the same packets power distribution as described in [27, 38]: lognormal distribution. Then, we analyse three other probability distribution functions in order to further enhance the performance: half normal distribution, reversed half normal distribution and uniform distribution.

6.4.1 Lognormal packets power distribution

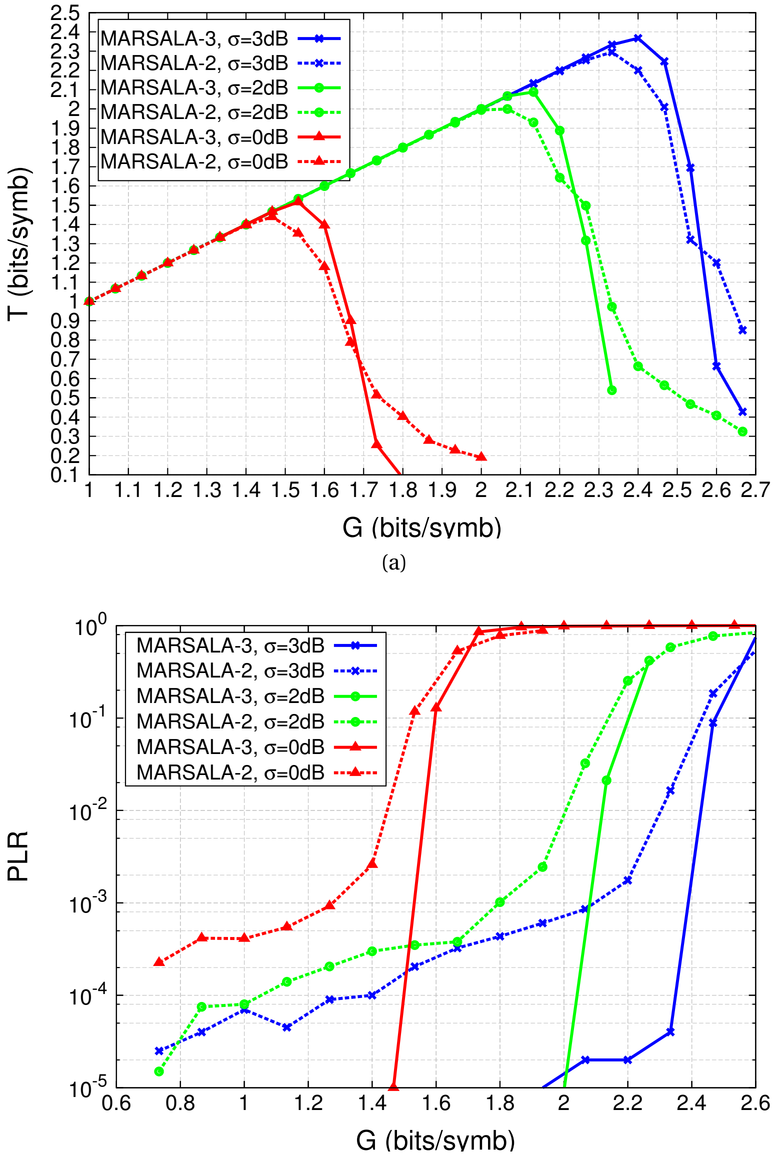
In realistic channel conditions, power unbalance among different transmitters is practically unavoidable. As a matter of fact, the terminal EIRP may randomly vary around a certain value and the path losses experienced by each user may be different depending on the area of coverage. It has been shown in [46] that in mobile communications channels, packets power approximately follows a truncated lognormal distribution of parameters $\mu = 0$ dB and σ varying between 2 and 3 dB, depending on the channel characteristics. Nevertheless the replicas corresponding to a same packet can still be considered equi-powered over the duration of one frame.

Figure 6.3 shows the throughput and PLR of MARSALA-2 and MARSALA-3 with MRC and received packets power following a truncated lognormal distribution. We suppose that all the users transmit their packets at an $E_s/N_0 = 10$ dB and use a QPSK modulation and a DVB-RCS2 turbo code of rate 1/3. The payload length is equal to 456 symbols per packet. At the receiver side, the received packets are attenuated following a lognormal Probability Density Function (PDF) of parameters $\mu = 0$ dB and $\sigma = 0$, 2 or 3 dB ($\sigma = 0$ dB indicate the equi-powered packets case).

As expected, we can observe that the performance enhancement with $\sigma = 2$ and 3 dB is significant compared to the equi-powered packets case. With $\sigma = 2$ dB, the maximum throughput is increased by around 40% in MARSALA-2 and MARSALA-3, with respect to the equi-powered case. Moreover, with $\sigma = 3$ dB, the maximum throughput is increased by around 65%. We notice a PLR floor at 10^{-5} for $\sigma = 3$ dB, due to the increased probability to receive packets at lower power and hence, with a higher PER.

6.4.2 Proposed packets power distributions for MARSALA

Given the important positive impact of packets power unbalance on the performance of RA methods employing SIC, recent research has proposed to apply power control techniques on the terminals and to derive an optimal PDF for the packets transmission power. Viterbi in [86] showed that the theoretical optimal packets power distribution for a synchronous CDMA scheme with SIC is exponentially distributed with the user index. The receiver starts by decoding the packet with the highest power, removes it then decodes the next packet and so on.



(a)

(b)

Figure 6.3 – Comparison between MARSALA-2 and MARSALA-3 with equi-powered packets and lognormally distributed packets power. QPSK modulation, DVB-RCS2 turbo code $R = 1/3$, $E_s/N_0 = 10$ dB. Payload length = 456 symbols. MRC applied at the receiver side. (a) Throughput. (b) PLR.

In [26], another optimal packets power distribution has been derived for a scheme based on spread spectrum Aloha with iterative interference cancellation. The authors have found that a Probability Density Function (PDF) uniform in dB is the optimal packets power distribution to achieve a maximum throughput with their scheme. Also, as stated in [27], ongoing work in the European Space Agency (ESA) is studying the optimal packets power distribution for CRDSA. As for MARSALA, an analytical study shall be done in the future. However, in this work, we consider three PDFs for packets power shown in Figure 6.4 and listed as follows:

- **The uniform distribution (in dB)**, with E_s/N_0 varying in the interval $[4,16]$ dB.
- **The half normal distribution (in dB)**, with parameters $\mu = 4$ and $\sigma = 7$ and E_s/N_0 varying in $[4,16]$ dB. We pick the value $\sigma = 7$ in order to have a large variance and to approach the uniform distribution, but still have lower probabilities at the rightmost edges of the PDF.
- **The reversed half normal distribution (in dB)**, with $\mu = 4$ and $\sigma = 7$.

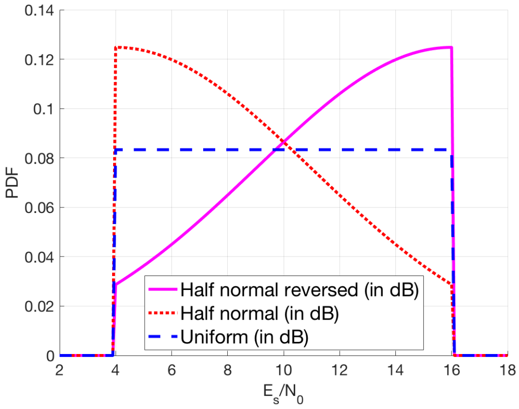


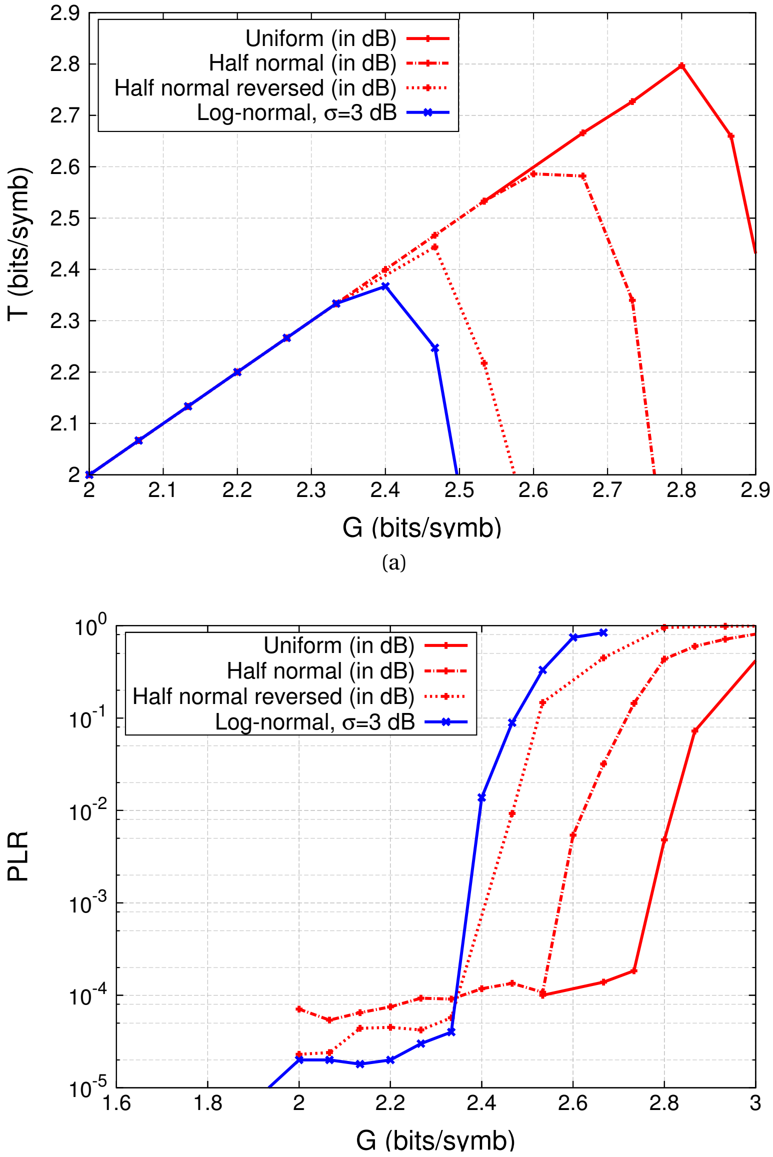
Figure 6.4 – PDFs proposed for packets power in MARSALA-3: uniform, half normal and reversed half normal (in dB).

Figure 6.5 shows the performance of MARSALA-3 with the packets power distributions described above compared to the equi-powered packets case as well as the lognormal distribution with parameters $\mu = 0$ dB, $\sigma = 3$ dB and $E_s/N_0 = 10$ dB. Based on the results obtained, Table 6.1 summarises the maximum throughput values achieved for a target PLR of 10^{-4}, as well as the percentage of gain compared to the equi-powered packets case.

We can notice that, among the three proposed PDFs, the packets power following the uniform distribution (in dB) , presents a maximum gain (77%) compared to the equi-powered packets case. At the same time, we observe that the half normal distribution shows a 72% throughput gain. The advantage of using this distribution is that a high probability of users is given a low E_s/N_0 ratio (between 4 dB and 6 dB).

PDF	Equi-powered $E_s/N_0 = 10$ dB	Logn $\sigma = 2$ dB	Logn $\sigma = 3$ dB	Half Normal	Uniform	Half Normal Reversed
T	1.47	2	2.27	2.53	2.6	2.27
Gain	-	36%	54%	72%	77%	54%

Table 6.1 – Comparison of the maximum throughput of MARSALA-3 (T in bits/symbol) and the performance gain compared to equi-powered packets, achieved at a PLR around 10^{-4}, with various packets power distributions and MRC.



(a)

(b)

Figure 6.5 – MARSALA-3 performance with various proposed packets power distributions. QPSK modulation, DVB-RCS2 turbo code $R = 1/3$ and $E_s/N_0 = 10$ dB. (a) Throughput. (b) PLR.

6.5 Performance of MARSALA with various coding schemes

Previously in this chapter, the performance of MARSALA with MRC in terms of throughput and PLR has been evaluated only with the DVB-RCS2 turbo code for linear modulation. However, it has been shown in previous research [27] that the turbo code performance for PER values between 0.9 and 1, has a direct impact on the performance of RA schemes using SIC. In fact, due to the high risk of collisions on the RA channel, the received packets usually have low SNIR and high PER values. Therefore, the error correction capability of the FEC code in such high PER regions is important for the throughput performance. This can be observed in [28], by comparing the performance of CRDSA with the DVB-RCS2 turbo code and the 3GPP code, especially with unbalanced packets power. It can be concluded that the choice of the 3GPP code for CRDSA instead of the DVB-RCS2 code, enables to recover more packets that have experienced collisions. Consequently, using the 3GPP code can trigger more SIC iterations and achieve a better performance. Seen that MARSALA is also a RA method that uses SIC, the choice of the coding scheme will also affect its performance. Therefore, in this section we compare the throughput and PLR of MARSALA obtained with three different encoding schemes, all using the QPSK modulation:

- **DVB-RCS 2 turbo code for linear modulation [10]**, which is a 16-states double binary Circular Recursive Systematic Convolutional (CRSC) code. In the simulations, we consider the reference waveform id-3 as defined in the DVB-RCS2 standard (payload burst length = 456 symbols, with QPSK modulation and code rate $R = 1/3$).
- **3GPP TS turbo code [25]** for Evolved Universal Terrestrial Radio Access (E-UTRA), with a payload burst length equal to 225 symbols.
- **Consultative Committee for Space Data Systems (CCSDS) turbo code [12]**, of rate 1/3 constructed from information block lengths of 456 bits. The CCSDS turbo code used in the simulations is provided by the Coded Modulation Library (CML) [87].

Figure 6.6 shows the theoretical PER curve for each of the 3 coding schemes presented above. We can remark that for the PER region [0.3, 1], the 3GPP turbo code performs better than both the DVB-RCS2 and CCSDS turbo codes.

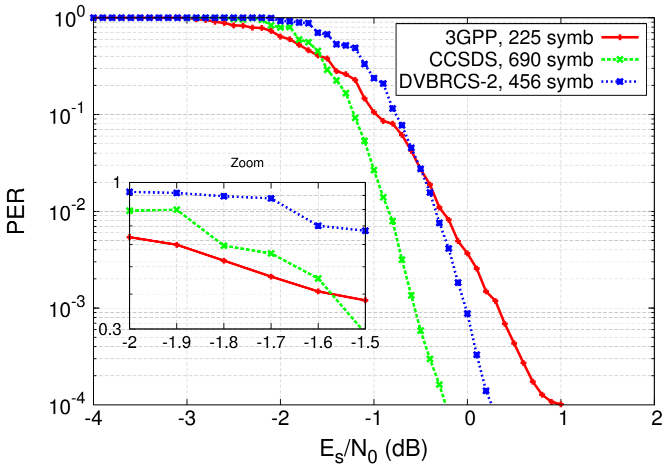
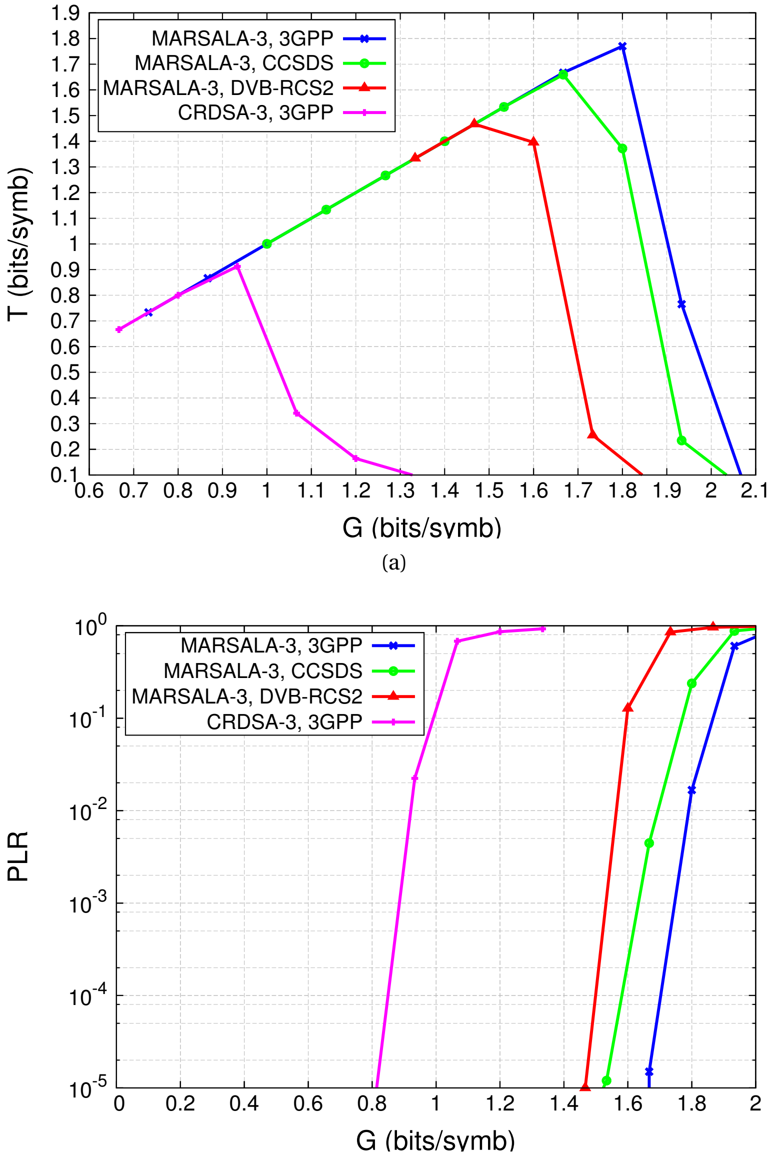


Figure 6.6 – PER vs. E_s/N_0 for 3GPP, DVB-RCS 2 and CCSDS turbo codes. QPSK modulation with code rate $R = 1/3$.

Figure 6.7 shows the impact of the coding schemes on the throughput and PLR of MARSALA-3 with equi-powered packets in real channel conditions. As expected, MARSALA-3 with 3GPP and CCSDS achieves a better performance than DVB-RCS2, and the best results are obtained with 3GPP. Compared to the DVB-RCS2 coding scheme, and for a target PLR equal to 10^{-5}, the throughput is 10% higher with CCSDS and 20% higher with 3GPP. Compared to CRDSA with 3GPP, MARSALA can almost double the throughput for the same target PLR.



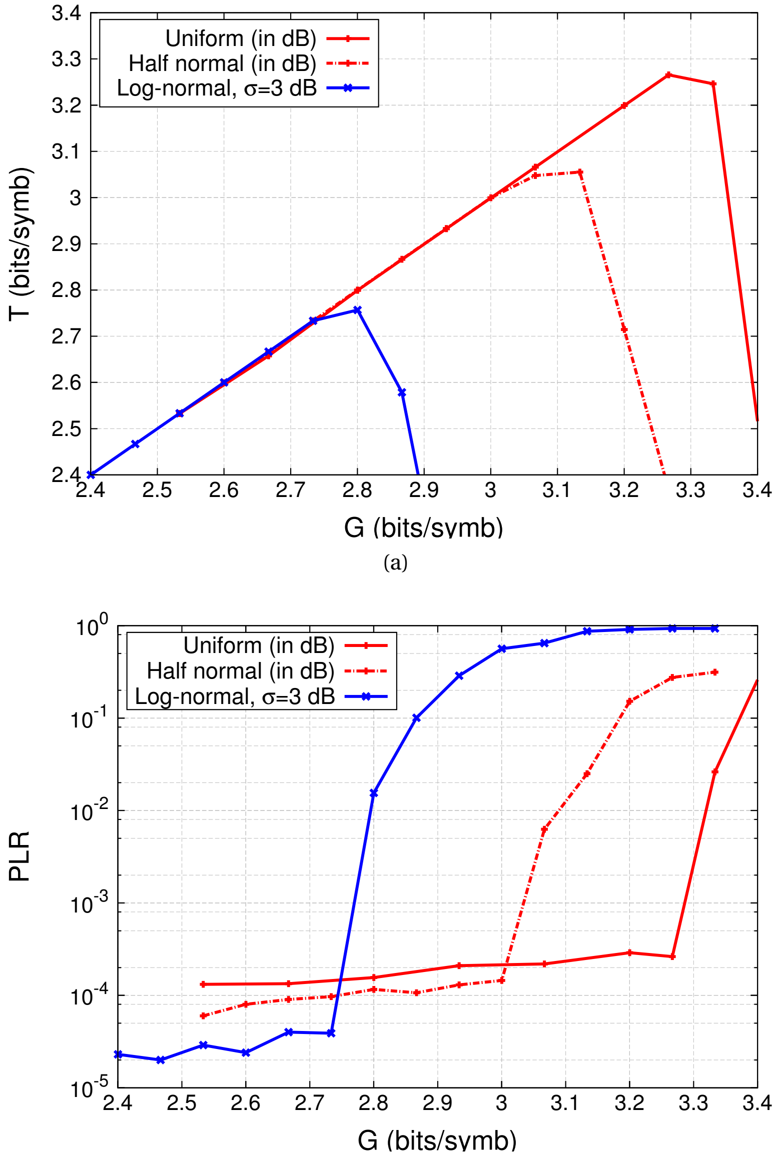
(a)

(b)

Figure 6.7 – Comparison between MARSALA-3 with DVB-RCS2, 3GPP and CCSDS turbo codes. Reference curve: CRDSA-3 with 3GPP. QPSK modulation, code rate $R = 1/3$. Equi-powered packets in real channel conditions with MRC. (a) Throughput. (b) PLR.

6.5.1 An optimal configuration for MARSALA

To conclude, we evaluate via simulations the gain obtained by combining the 3GPP coding scheme and packets power unbalance. The results obtained with MARSALA-3 in terms of throughput and PLR are shown in Figure6.8. Several packets power distributions are evaluated: **(1)** the lognormal distribution with parameters $\mu = 0$ dB, $\sigma = 3$ dB and $E_s/N_0 = 10$ dB. **(2)** The half normal distribution with parameters $\mu = 4$ and $\sigma = 7$ (in the logarithmic scale). **(3)** The uniform distribution with $E_s/N_0 \in [4, 16]$ dB.



(a)

(b)

Figure 6.8 – Performance comparison of MARSALA-3 with 3GPP turbo code and three packets power distributions: lognormal, half normal & uniform (in dB). QPSK modulation, code rate $R = 1/3$. MRC applied. (a) Throughput. (b) PLR.

We can observe that with 3GPP, the maximal throughput for a target PLR of 10^{-4} can reach up to 3.33 bits/symbol with the uniform distribution, 3 bits/symbol with the half-normal distribution and 2.7 bits/symbol with the log-normal distribution (σ = 3 dB). However, the PLR floor caused by low E_s/N_0 values is slightly lower with the log-normal and the half normal distribution, than the uniform distribution. Compared to the 3GPP equi-powered case, the maximal throughput is almost doubled.

6.6 Conclusion

MARSALA is a diversity based RA method employing SIC, therefore its performance is highly affected by the following design metrics: the method used for replicas combination, the packets power unbalance and the coding scheme behaviour. In this chapter, we proposed to add MRC to MARSALA. Two MRC techniques were described and they both showed a performance gain of around 10% in terms of throughput and PLR. We also concluded that, similarly to CRDSA, packets power unbalance enables MARSALA to achieve a better performance. In addition, we evaluated MARSALA with various coding schemes, and the maximum throughput was obtained with 3GPP turbo code and a payload length of 225 symbols. In conclusion, combining MARSALA with MRC, packets power unbalance and 3GPP turbo code can present an optimal design for an optimal performance, by maximising the throughput and minimising the PLR.

7

Conclusion & Future Work

> “When you reach the end of what you should know,
> you will be at the beginning of what you should sense.”
>
> — Kahlil Gibran

Despite the well-conducted research on the topic of RA methods in satellite communications, the performance achieved on the RA channel in terms of throughput and packet losses can still be enhanced. This thesis permits to fill some particular gaps in this domain. It presents solutions to the problem of channel estimation for multiple packets in collision. Furthermore, it advocates an alternative decoding technique for CRDSA when the packets experience high collisions. MARSALA, this new decoding scheme for CRDSA, is the major contribution of this thesis. In this chapter, we conclude by discussing how this thesis contributed to achieve its initial goals and by enumerating several remaining challenges. Finally, we provide some directions for future work.

7.1 A major contribution: MARSALA to break the deadlock on a RA channel

In a system that uses the CRDSA scheme, a deadlock can occur when all the packets on the frame are strongly interfered and no more packet replicas can be resolved. In this case the proposed MARSALA scheme enables to break this deadlock by performing additional signal-level operations such as cross-correlation and replicas combination. With the SIC process also applied at the receiver side, we conclude that MARSALA enables to decrease the interference level on the frame iteratively and permits to reach a lower load regime where CRDSA can resolve collisions again.

- **MARSALA for replicas localisation before decoding:**

Given that the frame is divided into timeslots, MARSALA computes the correlation between the signal of one timeslot and the remaining signals on the other timeslots of the frame. Amplitude peaks are obtained whenever two signals contain a same replica. Thus, replicas of a same packet can be localised even before decoding any signalling information. We conclude that we can always find a way to use the redundancy transmitted by a given user in order to successfully receive the useful information. In MARSALA, we simply benefit from the common information contained in signals received on separate timeslots, in order to localise the replicas of a same packet.

- **MARSALA for replicas combination:**

MARSALA shows how replicas combination can boost the SNIR of interfered replicas and increase their probability for successful decoding. In real channel conditions, this combination step requires prior estimation and adjustment of timing offsets and phase shifts between the replicas of a same packet. We present a method to estimate and compensate the timing and phase differences between packet replicas. Our analytical study for the impact of imperfect replicas combination as well as the simulation results show a slight degradation in the performance compared to the perfect channel knowledge case. However, even with the synchronisation errors taken into account, the throughput and the PLR results still show substantial gains compared to the results of CRDSA without MARSALA.

Another important observation to make, is the robustness of MARSALA at low E_s/N_0 values. For example, with MRC, QPSK modulation and DVB-RCS2 turbo code of rate 1/3, we could still achieve a PLR $< 10^{-4}$ with $E_s/N_0 = 4$ dB, even when the channel load is equal to 1.2 bits/symbol. Therefore, we can conclude that MARSALA is well suited for communication scenarios with low power transmissions. This result motivates to further evaluate MARSALA in scenarios of IoT and M2M applications where the terminals are low-cost and require very lower power transmissions.

In addition, MARSALA shows that the achieved throughput can reach up to 2.6 bits/symbol with a lognormal packets power distribution ($\sigma = 3$ dB, $E_s/N_0 = 10$ dB), a 3GPP turbo code of rate 1/3 and a payload length of 225 symbols. We conclude that even in presence of power fluctuations, more significant performance gains can be obtained when MARSALA is applied.

With MARSALA, we can maintain a relatively low PLR ($< 10^{-4}$) for much higher channel loads. Although no analytical studies for end-to-end delays or system stability have been done, we can predict some conclusions to be tested in the future. With the high throughput and low PLR achieved at high channel loads, the average packet delays are also expected to remain lower. Moreover, we can expect that system will stay in the stability region for higher values of channel loads.

7.2 Future work

7.2.1 Remaining Challenges

In this dissertation, we presented MARSALA RA scheme and evaluated its performance in real channel conditions. In particular, we took into account the effect of timing offsets and phase shifts resulting from phase and carrier frequency variations between replicas on separate timeslots. However, another parameter to take into consideration in future studies is phase noise. Phase noise can be represented as a stochastic process of short-term frequency variations. Therefore, it can have an impact on the accuracy of replicas localisation using correlation. In fact, phase fluctuations depend on the phase noise mask and the symbol rate; lower symbol rate induces higher phase noise. For future work, it would be interesting to show for a specific phase noise mask (for example the DVB-RCS2 case), what is the minimum symbol rate at which the constant phase assumption holds, and the way to cope with the opposite case.

Another remaining challenge is to treat the issues related to MARSALA practical implementations aspects. Also, an interesting future work is finding an optimal packets power distribution in order to achieve an optimal throughput for a predefined target PLR.

Furthermore, future work should also provide a performance evaluation of MARSALA in terms of end-to-end packet delays and in terms of its stability in schemes involving packets retransmissions. It is also important to evaluate the complexity induced by MARSALA at the receiver side in terms of number of additional correlation operations required for replicas localisation and combining.

7.2.2 Perspectives

Throughout this thesis, we are inspired to open many perspectives for future work. We discuss in the following several ideas of RA schemes based on MARSALA that deserve to be studied in the future. A first concept worth investigation is Irregular MARSALA. Similarly to IRSA, an irregular number of packet replicas can be transmitted on the frame. At the receiver side, MARSALA can be applied to recover packets that have not been resolved with IRSA. In this case, an optimal distribution for the number of replicas transmitted by each user in MARSALA can be derived.

A second open perspective is MARSALA with irregular coding rates for different users. Given that MARSALA resolves strong collisions and performs successive interference cancellation, packets that are encoded with a lower coding rate can be decoded first (they are also the packets with the highest interference level on the frame). Then, after interference cancellation, the remaining packets are the packets encoded with a higher coding rate. Given that those packets become less interfered, then they can be successfully decoded even if their code rate is higher. Consequently, the throughput for some users can be further enhanced.

A third suggestion for future work is Asynchronous MARSALA. In this case MARSALA shall be applied jointly with ACRDA. The concept of virtual frames and sliding window frame processing can be kept. MARSALA can intervene whenever ACRDA is blocked and no additional packets are resolved. A substantial challenge in this case, is the ability to detect the beginning of each packet on the frame in order to compute its correlation with the rest of the frame. Once the replicas of a given packet are localised, MARSALA can combine the replicas in a similar way to the synchronous scheme.

7.3 Final remarks

This thesis finds a way to exploit the redundancy transmitted by each user on the RA channel. By applying signal processing techniques such as time domain correlations and diversity based techniques such as MRC, the throughput can be significantly enhanced. The MARSALA RA scheme resolves packets that CRDSA was unable to recover, and it also applies SIC in order to release the remaining packets on the frame from collisions. MARSALA is one of many recent RA methods proposed for satellite communications. It can be added to a long list of RA schemes such as CRDSA, IRSA, MuSCA, CSA, ECRA, E-SSA, ACRDA and many others. Although all these methods have the same main goal of enhancing throughput and reducing the PLR, each of them differ in many aspects. And each of these methods is adequate for a specific system. For example, while E-SSA is for asynchronous systems that use spread spectrum techniques, CRDSA and MARSALA are more suited for the DVB-RCS2 systems. A dilemma also exists around the choice of synchronous or asynchronous techniques. In fact, the synchronous RA methods are more practical in terms of packet detection because we can always search for replicas in the boundaries of one timeslot. However, they induce more end-to-end delays compared to the asynchronous methods and they require signalling transmissions to ensure terminals synchronisation. Finally, we can conclude that research in the domain of RA methods for satellite communications has significantly advanced during the last years and it is going on the right track for system implementation. Of course, practical challenges still exist but they open questions for future work and motivation for further optimisation.

7.4 List of publications

1. K. Zidane, J. Lacan, M. L. Boucheret and C. Poulliat, “Improved channel estimation for interference cancellation in random access methods for satellite communications,” 2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), Livorno, 2014, pp. 73-77.
2. K. Zidane, J. Lacan, M. L. Boucheret, C. Poulliat, M. Gineste, D. Roques, C. Bes and A. Deramecourt, “Effect of Residual Channel Estimation Errors in Random Access Methods for Satellite Communications,” 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), Glasgow, 2015, pp. 1-5.
3. H. C. Bui, K. Zidane, J. Lacan and M. L. Boucheret, “A Multi-Replica Decoding Technique for Contention Resolution Diversity Slotted Aloha,” 2015 IEEE 82nd Vehicular Technology Conference (VTC Fall), Boston, MA, 2015, pp. 1-5.
4. K. Zidane, J. Lacan, M. Gineste, C. Bes and C. Bui, “Enhancement of MARSALA Random Access with Coding Schemes, Power Distributions and Maximum Ratio Combining,” 2016 8th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), Palma de Mallorca, 2016.
5. K. Zidane, J. Lacan, M. Gineste, C. Bes and C. Bui, “Estimation of Timing Offsets and Phase Shifts Between Packet Replicas in MARSALA Random Access,” 2016 IEEE Global Communications Conference: Wireless Communications, Washington, DC, USA, 2016.

8

Résumé de la Thèse en Français

8.1 Introduction

8.1.1 Objectifs & motivations

De nos jours, le réseau Internet est en expansion rapide et continue et joue un rôle primordial dans plusieurs secteurs tels que l'éducation, la santé, la finance, les communications sociales et plein d'autres. Pourtant, presque la moitié de la population terrestre ne profitait pas d'une couverture Internet en 2013 [2]. Cela est dû à la mauvaise situation sociale et économique dans certains pays, mais aussi à l'emplacement géographique de certaines zones rurales avec de pauvres densités en population. Dans ce contexte, les satellites géostationnaires (GEO) et les satellites en orbite basse (LEO) et moyenne (MEO) peuvent assurer une couverture terrestre globale et permettent un accès Internet aux zones blanches et déprivées.

Le problème principal dans ce type de réseaux est le long délai de communication dû principalement aux délais de propagation du signal ainsi que d'autres types de délais d'accès au système. En effet, plusieurs études ont déjà montré que de longs délais de communication affectent négativement sur l'expérience utilisateur, et peuvent entraîner à des pertes financières importantes. Pour ces raisons, les chercheurs et industriels s'intéressent à trouver des solutions pour réduire les délais de communications par satellite, que ce soit en GEO, LEO ou MEO. Dans le cadre de cette thèse, nous allons nous placer dans un contexte de communications par satellite GEO. Un second problème lié aux communications par satellite est de pouvoir supporter des réseaux à grande échelle surtout avec l'apparition de nouveaux services d'Internet par satellite et de téléphone par satellite. Dans le but de proposer une solution à tels problèmes, nous allons nous intéresser à un sujet bien étudié de nos jours dans le contexte de communications par satellite : les méthodes d'accès aléatoire.

Pourquoi étudier les méthodes d'accès aléatoires pour les communications par satellite?

En effet, les protocoles qui permettent d'organiser l'accès des terminaux terrestres sur le lien retour (c'est à dire, le lien allant du terminal terrestre au satellite ou à la passerelle), sont nommés protocoles d'accès (en anglais, *Media Access Control* - MAC). En utilisant ces protocoles, les terminaux peuvent accéder au lien retour soit en mode dédié (en anglais, *Demand Assignment Multiple Access* - DAMA), soit en mode aléatoire (en anglais, *Random Access* - RA). En DAMA, des procédures d'allocation de ressources sont effectuées au début de chaque communication et chaque utilisateur est alloué une ou plusieurs bandes de fréquences ainsi qu'un ou plusieurs créneaux temporels. Au contraire, en mode aléatoire, les utilisateurs n'ont pas besoin d'allocations de ressources. L'avantage de l'accès aléatoire se manifeste alors par une réduction du délai de communication mais avec un risque plus important de collisions de paquets. D'autre part, les méthodes d'accès aléatoires sont surtout bien adaptées aux types de traffic Internet sporadique avec de longs moments de silence et de courtes transmissions de données, tel que le traffic HTTP. Par conséquent, l'utilisation des méthodes d'accès aléatoire en combinaison avec DAMA sur le lien retour serait intéressante.

Problème principal des méthodes d'accès aléatoire

Il est évident que dans un mode d'accès aléatoire, le risque de collisions entre les paquets transmis sur le lien retour devienne plus élevé. Notamment, les méthodes d'accès aléatoire telles que Aloha [14] et Slotted Aloha (SA) [15] offrent des performances limitées à cause des collisions. Néanmoins, afin de résoudre les collisions, les nouvelles méthodes d'accès aléatoire se basent sur des techniques de suppression successive d'interférences (en anglais, *Successive Interference Cancellation* - SIC) et de transmissions de la redondance. Ces méthodes peuvent être synchrones (temporellement slottées), parmi lesquelles nous pouvons citer Contention Resolution Diversity Slotted Aloha (CRDSA) [16] et Multi-Slot Coded Aloha (MuSCA) [21]. Elles peuvent aussi être asynchrones (non slottées), telles que Asynchronous Contention Resolution Diversity Aloha (ACRDA) [50] et Enhanced Spread Spectrum Aloha (E-SSA) [45].

CRDSA est une technique d'accès aléatoire incluse dans le standard DVB-RCS2 [10] pour satellites GEO. Dans CRDSA, chaque utilisateur transmet deux ou plusieurs répliques du même paquet en ajoutant un champ de signalisation à chaque paquet, dans lequel il précise l'emplacement des autres répliques sur la trame (une trame est l'ensemble des slots temporels partagés par les utilisateurs sur le lien retour). Au récepteur, la trame est scannée et chaque paquet décodé avec succès est reconstruit et supprimé avec ces répliques correspondantes. Cependant, si toutes les répliques des paquets sont fortement interférées, CRDSA n'est pas capable de résoudre les collisions et les paquets sont perdus.

Une autre méthode d'accès aléatoire synchrone est MuSCA. Dans MuSCA, chaque utilisateur encode son paquet avec un code correcteur d'erreur (en anglais, *Forward Error Correction* - FEC code), puis divise le mot de code obtenu en plusieurs fragments. Chaque fragment contient un champ de signalisation pour localiser ses autres fragments correspondants sur la trame. Les champs de signalisation dans MuSCA sont encodés séparément et nécessitent un taux de codage trés bas, afin de pouvoir les dćoder même en présence de fortes interférences.

Notre objectif principal

Notre objectif principal dans cette thèse est d'améliorer les performances des méthodes d'accès aléatoires sans ajouter de la redondance à la signalisation par rapport à CRDSA.

8.1.2 Contributions

Les contributions majeures de cette thèse se résument dans la liste ci-dessous :

- **Estimation de canal pour la suppression des interférences :** Afin de réaliser la suppression d'interférences, il est important d'effectuer une estimation de canal pour chaque paquet à supprimer. Pour cela, nous proposons une technique d'estimation de canal pour les paquets en collision, combinant l'algorithme Expectation-Maximisation (EM) avec l'initialisation par autocorrélation et l'estimation basée sur les symboles pilotes distribués. Nous évaluons ainsi les performances de cette méthode en ajoutant une boucle de décodage associé à l'estimation.
- **MARSALA, une nouvelle technique de décodage pour CRDSA :** Au cas où CRDSA est incapable de résoudre les collisions, nous proposons une nouvelle technique pour décoder les paquets appelée MARSALA (en anglais, *Multi-Replica Decoding using Correlation Based Localisation*). Dans MARSALA, les répliques appartenant à un même paquet sont localisées par corrélation et ensuite combinées afin d'obtenir un rapport Signal sur Bruit plus Interférerences (en anglais, *Signal to Noise plus Interference Ratio* - SNIR) plus élevé. Nous pouvons lister trois contributions principales ayant pour objectif d'évaluer les performances de MARSALA :
  - Présentation d'une méthode de synchronisation pour estimer et corriger les différences en phase et en temps entre les répliques d'un même paquet reçues sur des slots temporels distincts.
  - Définition d'un modèlle analytique permettant d'évaluer le niveau de dégradation induite par la combinaison imparfaite des répliques.
  - Optimisation de MARSALA en utilisant du Maximum Ratio Combining (MRC), des puissances inégales et des schémas de codage différents.

8.2 Estimation de canal pour les nouvelles méthodes d'accès aléatoire

Dans les nouvelles méthodes d'accès aléatoire pour les communications par satellite, le canal doit être estimé avec précison, afin de ne pas laisser des erreurs résiduelles qui affecteront sur le décodage des paquets restants après la suppréssion des interférences. Plusieurs techniques d'estimation de canal existent déjà dans la littérature mais nécessitent encore des améliorations surtout dans des régimes à fortes collisions. Dans cette section nous allons détailler notre schéma d'estimation de canal proposé.

8.2.1 Définition du problème

Comme dans tout système de communication, l'estimation de canal au niveau du récepteur est une étape primordiale pour obtenir une démodulation et un décodage réussis. Dans les nouvelles méthodes d'accès aléatoire basée sur le SIC, l'estimation de canal est faite sur les paquets même si ces derniers sont en collision. D'autre part, l'estimation de canal pour chaque paquet interférent doit être précise afin de laisser le minimum d'erreurs résiduelles une fois que l'interférence est supprimé.

Par conséquent, le problème à traiter dans ce chapitre est l'impact des erreurs résiduelles d'estimation de canal sur les méthodes d'accès aléatoire synchrones avec SIC. Pour illustrer ce problème, nous allons considérer l'exemple de la figure Figure 8.1 montrant 5 slots temporels d'une trame partagée entre 3 utilisateurs utilisant un mode d'accès aléatoire synchrone (par exemple, CRDSA ou MuSCA). Si la méthode d'accès utilisée est CRDSA, les notations (a) et (b) correspondent aux répliques d'un même paquet, alors que dans le cas de MuSCA, (a) et (b) correspondent aux fragments d'un même paquet. Nous supposons que le récepteur décode le paquet (2b) avec succès sur le slot 4 et dois supprimer ce paquet et sa copie (2a) de la trame. Pour cela, le récepteur doit estimer le canal de l'utilisateur 2 sur les slots 1 et 4 avant la suppression des paquets. Dans le cas où l'estimation de canal est parfaite, le paquet (2a) est supprimé complètement du slot 1 et paquet (1a) se retrouve sans collisions (Figure 8.1b). Par contre, comme l'estimation de canal n'est jamais parfaite, le paquet (2a) laisse des erreurs résiduelles après sa suppression, ce qui affectera sur le décodage du paquet (1a) (Figure 8.1c).

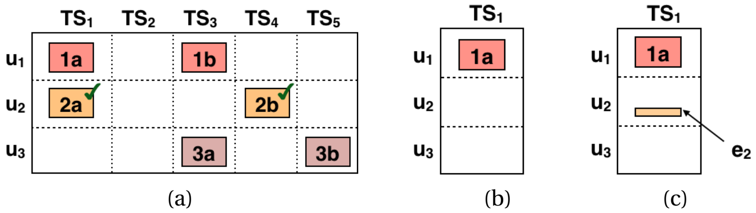


Figure 8.1 – Exemple de suppression parfaite et imparfaite de l'interférence sur une trame. $N_u = 3$ utilisateurs (u) and $N_s = 5$ slots temporels (TS).

8.2.2 Hypothèses système

Dans cette section, nous supposons les hypothèses système définies ci-après. Nous considérons un slot temporel (TS) d'une trame reçue sur la liaison montante d'un système de communication par satellite, avec des terminaux terrestres fixes. TS contient K paquets en collision transmis par K utilisateurs différents. Le coefficient canal h_k pour chaque utilisateur k est défini dans l'equation 8.1 ci-dessous.

$$h_k(i) = A_k e^{j(2\pi\Delta f_k i T_s + \phi_k)} \quad , \tag{8.1}$$

avec:

- A_k étant l'amplitude du canal supposée constante sur la durée d'une trame;
- Δf_k étant le décalage fréquentiel du paquet k modélisé par une variable aléatoire uniforme sur $[0, \Delta f_{max}]$ avec Δf_{max} égal à 1% du débit symbole $1/T_s$;
- ϕ_k étant le décalage en phase pour chaque paquet k et variant aléatoirement d'un slot à l'autre. ϕ_k peut être modélisé par une variable aléatoire uniforme entre $[0, 2\pi]$.

Le signal y reçu sur chaque slot temporel peut être défini comme suit :

$$y(i) = \sum_{k=1}^{K} \underbrace{h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT)}_{s_k(i)} + w(i) \quad , \tag{8.2}$$

avec:

- T et $T_s = \frac{T}{Q}$ étant respectivement la période symbole et la période de sur-échantillonnage;
- $i = 0, 1, \cdots, LQ-1$ et $n = 0, 1, \cdots, L$ étant les indexes des échantillons et des sur-échantillons symboles, et L étant la longueur totale du paquet en symboles;
- $x_k(n)$ correspondant au $n^{\text{ème}}$ symbole transmis par l'utilisateur k;
- g dénotant la fonction du filtre de mise en forme modélisé par une racine de cosinus surélevé;
- w étant le processus de bruit blanc additif (en anglais, *Additive White Gaussian Noise* - AWGN) complexe, de moyenne $\mu = 0$ et variance $\sigma_w^2 = N_0$.

Chaque paquet contient un préambule (désigné par $\boldsymbol{x_{pre}^k}$) et un postambule (désigné par $\boldsymbol{x_{post}^k}$) constistués à partir de séquences de symboles connus au niveau du récepteur et utilisés pour l'estimation de canal. Ces séquences sont uniques et orthogonales et modulés en BPSK (en anglais, *Binary Phase Shift Keying*). Nous supposons que le récepteur est capable de détecter les utilisateurs présents sur chaque slot en utilisant des techniques connues de détection multi-utilisateurs [67]. Suite à leur décodage préalable sur d'autres slots temporels, le récepteur connaît les séquences des symboles x_2, x_3, $\cdots$ et x_K et n'a pas encore décodé le paquet correspondant au signal s_1. Ainsi, le récepteur peut supprimer les symboles d'interférence x_2, x_3, $\cdots$ et x_K. L'objectif est d'être capable de démoduler et de décoder le signal s_1, en présence des résidus des erreurs d'estimation de canal laissés par les autres signaux après annulation des interférences. Pour cela, le récepteur doit estimer les coefficients de canal $\widehat{\boldsymbol{h_1}}$, $\widehat{\boldsymbol{h_2}}$, $\widehat{\boldsymbol{h_3}}$, $\cdots$, et $\widehat{\boldsymbol{h_K}}$, puis supprimer les signaux interférents de y, afin d'obtenir le signal discret $\widehat{\boldsymbol{s_1}}$ exprimé comme suit :

$$\widehat{s_1}(i) = h_1(i)\sum_{n=0}^{L-1} x_1(n)g(iT_s - nT) + \sum_{k=2}^{K}\left[\left(h_k(i) - \widehat{h}_k(i)\right)\sum_{n=0}^{L-1} x_k(n)g(iT_s - nT)\right] + w(i) \quad (8.3)$$

Après suppression des interférences et en présence des erreurs d'estimation résiduelles, le signal estimé $\widehat{\boldsymbol{s_1}}$ passe à travers un filtre adapté puis il est échantillonné à la période symbole T et les symboles estimés obtenus sont exprimés tels que :

$$\widehat{s_1}(n) = h_1(n)x_1(n) + \sum_{k=2}^{K}\left[\left(h_k(n) - \widehat{h}_k(n)\right)x_k(n)\right] + w(n) \quad . \quad (8.4)$$

$\widehat{\boldsymbol{s_1}}$ est démodulé puis décodé et supprimé de la trame. Dans ce qui suit, nous proposons un algorithme d'estimation de canal qui permet de réduire l'impact des erreurs d'estimation de canal résiduelles $\left(\boldsymbol{h_k} - \widehat{\boldsymbol{h_k}}\right)$ et d'améliorer ainsi la probabilité de décodage réussie pour les symboles estimés de l'utilisateur 1.

8.2.3 Algorithme proposé d'estimation de canal

Afin d'estimer conjointement les canaux de plusieurs paquets en collision, nous choisissons d'utiliser l'algorithme Expectation-Maximisation (EM). C'est un algorithme itératif constitué de deux étapes principales : Expectation et Maximisation. Dans la littérature [52], l'estimation de canal utilisant l'algorithme EM est appliquée seulement sur le préambule du signal et les paramètres d'initialisation de cet algorithme sont choisis aléatoirement. Afin de suivre des variations des paramètres du canal au début et à la fin de chaque paquet, nous proposons d'appliquer l'algorithme EM non seulement sur les symboles du préambule mais aussi sur les séquence du postambule. Dans la suite, nous utilisons l'expression 'symboles de signalisation' pour désigner les séquences de préambule et postambule d'un certain paquet. Nous utiliserons la notation $\boldsymbol{x_{tr}^k}$ pour indiquer la séquences de signalisation de chaque utilisateur k. La longueur totale d'une séquence de signalisation est égale à $L_{tr} = L_{pre} + L_{post}$, avec L_{pre} et L_{post} étant les longueurs respectives des séquences de préambule et de postambule.

Dans le cas de K paquets en collision sur un slot, les equations EM à chaque itération m sont exprimées ci-dessous :

Etape Expectation : pour $k = 1, ..., K$

$$p_k^{(m)}(n) = x_{tr}^k(n)\widehat{h}_k(n)^{(m-1)} + \beta_k\left[y_{tr}(n) - \sum_{l=1}^{K} x_{tr}^l(n)\widehat{h}_l(n)^{(m-1)}\right] \quad . \quad (8.5)$$

Etape Maximisation : pour $k = 1, ..., K$

$$\left\{\widehat{A}_k, \widehat{\Delta f}_k, \widehat{\phi}_k\right\} = \underset{A', \Delta f', \phi'}{\text{argmin}} \sum_{n\in\Theta}\left|x_{tr}^k(n)p_k^{(m)}(n) - A'e^{j(2\pi\Delta f' nT + \phi')}\right|^2 \quad , \quad (8.6)$$

avec Θ étant l'ensemble des indexes des preambules et postambules, $\Theta = \{0, \cdots, L_{pre}-1, L_{pre}+L_{pay}, \cdots, L_{pre}+L_{pay}+L_{post}-1\}$. Le vecteur $\boldsymbol{y_{tr}}$ désigne les échantillons de $\boldsymbol{y}$ correspondants aux séquences de signalisation.

EM avec initialisation par autocorrélation

Afin d'améliorer l'estimation EM, nous proposons d'initialiser les paramètres de canal pour la première itération en utilisant l'autocorrélation avec les symboles connus de signalisation. En effet, on a démontré dans des recherches antérieures [55] que l'initialisation aléatoire des paramètres de l'algorithme EM pourrait conduire à des résultats incohérents. Inspirés de cette étude, nous allons utiliser l'estimation par autocorrélation pour initialiser les paramètres EM pour l'estimation du canal. Pour la première itéeation, nous effectuons l'estimation de l'amplitude $\widehat{A_k}^{(0)}$ pour l'utilisateur k comme suit :

$$\widehat{A_k}^{(0)} = \frac{\left|\boldsymbol{y_{pre}} \times \left(\boldsymbol{x_{pre}^k}\right)^T\right|}{L_{pre}} \quad , \tag{8.7}$$

Pour l'estimation initiale de phase $\widehat{\phi_k}^{(0)}$ nous procédons comme suit :

$$\widehat{\phi_k}^{(0)} = \angle\left(\boldsymbol{y_{pre}} \times \left(\boldsymbol{x_{pre}^k}\right)^T\right) \quad , \tag{8.8}$$

avec $(\times)$ étant l'opérateur de multiplication vectorielle, $(\cdot)^T$ dénotant la transposée d'un vecteur et $(\angle)$ étant l'angle d'une valeur complexe.

Premiers résultats

Nous évaluons les performances de l'algorithme d'estimation de canal proposé : EM sur préambule et postambule avec initialisation par autocorrélation, en calculant le taux d'erreurs paquets (en anglais Packet Error Rate - PER). Pour cela nous effectuons des simulations en considérant deux paquets interférés sur un même slot. Nous estimons les paramètres de canal pour les deux utilisateurs à la fois, puis nous supprimons l'un des paquets et nous essayons de démoduler et décoder l'autre en présence des erreurs d'estimation résiduelles du premier. Les résultats de PER en fonction de E_s/N_0 sont affichés dans la figure 8.2.

Nous observons un gain de performance avec l'initialisation par autocorrélation et nous remarquons que la dégradation du PER est essentiellement induite par les erreurs d'estimation de décalage fréquentiel. Pour cela, nous proposons de raffiner l'estimation du décalage fréquentiel en utilisant des séquences de symboles pilotes distribuées à l'intérieur des paquets.

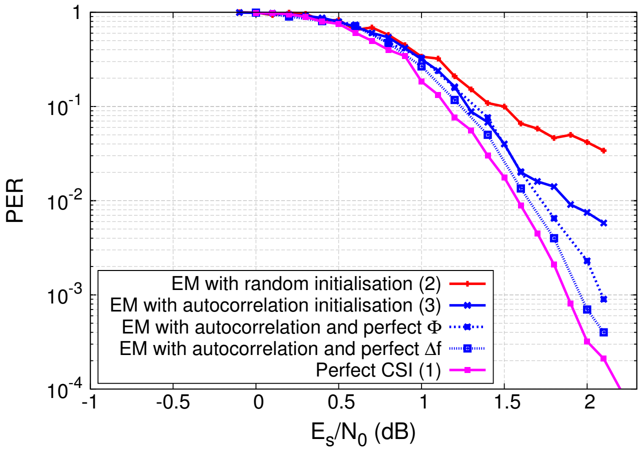


Figure 8.2 – PER vs. E_s/N_0 après suppression d'un paquet interférent et utilisant différents algorithmes d'estimation de canal. QPSK, turbo code CCSDS de taux 1/2 et $L = 460$ symboles.

Estimation de canal avec symboles pilotes

Etant donné que le décalage fréquentiel fait varier la phase du signal linéairement avec le temps, les séquences de préambule et de postambule des différents utilisateurs peuvent perdre leurs propriétés d'orthogonalité. Pour résoudre ce problème, nous proposons d'appliquer l'algorithme EM sur des séquences de symboles pilotes plus courtes, distribuées à l'intérieur des paquets (en anglais, *Pilot Symbol Assisted Modulation* - PSAM[23, 69]). La structure d'un paquet avec PSAM est illustrée dans la figure 8.3 ci-dessous.

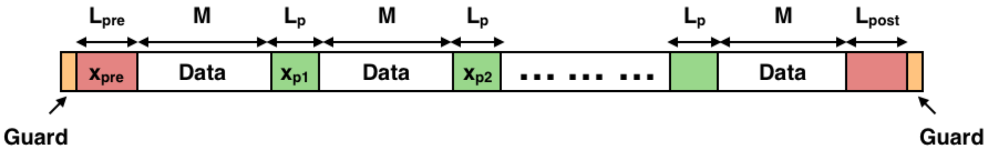


Figure 8.3 – Structure d'un paquet avec préambule, blocs pilotes, postambule et intervalles de guarde.

Les résultats de simulations avec PSAM sont illustrés dans la figure 8.4. Nous considérons des tailles de préambule, blocs pilotes et postambule égales à 40, 12 et 12 symboles au lieu de 80 symboles de préambule et 48 symboles de postambule dans le cas sans PSAM. Le gain de performance avec PSAM est observé par rapport à une structure de paquet sans PSAM, et le palier de PER disparaît autor de 10^{-4}.

Prise en compte du décalage temporel

Dans ce qui suit, nous allons prendre en compte le décalage temporel existant entre les paquets de différents utilisateurs en collision sur un même slot. L'objectif est d'évaluer les performance

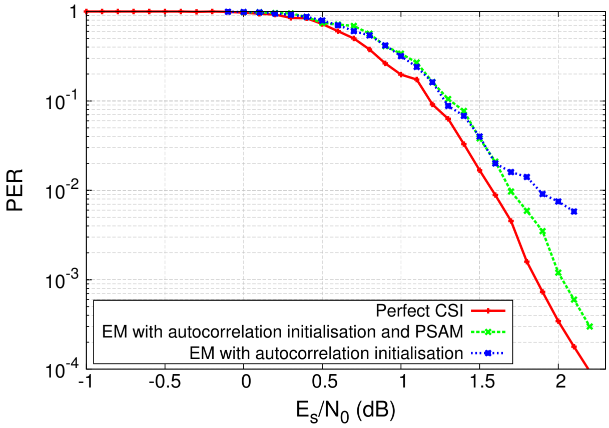


Figure 8.4 – PER vs. E_s/N_0 après suppression d'un paquet interférent avec et sans PSAM. QPSK, turbo code CCSDS de taux 1/2 et $L = 460$ symboles.

de l'algorithme EM au cas où les signaux en collision ne sont pas alignés temporellement et les séquences de signalisation ne sont pas orthogonales entre les différents utilisateurs. En d'autres termes, le signal $\boldsymbol{y}$ reçu sur un slot quelconque serait exprimé comme suit :

$$y(i) = \sum_{k=1}^{K} \underbrace{h_k(i) \sum_{n=0}^{L-1} x_k(n) g(iT_s - nT - \tau_k T)}_{s_k(i)} + w(i) \quad , \tag{8.9}$$

avec τ_k étant le décalage temporel relatif au signal $\boldsymbol{s_k}$ transmis par l'utilisateur k. Nous allons supposer que τ_k est une variable aléatoire uniforme discrète dans $\left[0, \frac{1}{Q}, \frac{2}{Q}, \cdots, \frac{Q-1}{Q}\right]$. Pour estimer la valeur intiale du décalage temporel $\widehat{\tau_k}^{(0)}$, nous utilisons l'equation ci-dessous :

$$\widehat{\tau_k}^{(0)} = \underset{\tau}{\text{argmax}} \left| \sum_{i=0}^{QL_{pre}-1} y_{pre}(i) \sum_{n=0}^{L-1} \left(x_{pre}^{k}(i) g(iT_s - nT - \tau T) \right)^{*} \right|^2 \quad . \tag{8.10}$$

Ensuite, pour chaque itération $m > 0$:

$$\widehat{\tau_k}^{(m)} = \underset{\tau}{\text{argmax}} \left| \sum_{i=0}^{QL_{tr}-1} y_{tr}(i) e^{-j2\pi \widehat{\Delta f_k}^{(m-1)} iT_s} \sum_{n=0}^{L-1} \left(x_{tr}^{k}(i) g(iT_s - nT - \tau T) \right)^{*} \right|^2 \quad . \tag{8.11}$$

Cette estimation du décalage temporel est intégrée à l'intérieur de l'algorithme EM, et répétée itérativement jusqu'à la convergence de l'algorithme.

Estimation associée au décodage

Étant donné que l'estimation de canal basée sur EM repose sur la connaissance des symboles de signalisation pour chaque utilisateur, cette estimation peut être améliorée par l'introduction de séquences de signalisation plus longues. Dans ce cas, le problème est une plus grande charge de signalisation sur le paquet. Pour cette raison, nous proposons une autre solution inspirée de la littérature [24, 70, 71, 72] et basée sur l'estimation associée au décodage (en anglais, Joint Estimation and Decoding - JED). Notre contribution se résume par l'intégration du JED avec l'algorithme EM afin d'optimiser les opérations d'annulation d'interférences dans les nouvelles méthodes d'accès aléatoire. Les étapes de l'algorithme d'estimation de canal proposé avec JED sont listées ci-dessous :

1. Dans la première itération de JED, les paramètres du canal sont estimés en utilisant l'algorithme EM (avec 3 itérations) avec initialisation par autocorrélation et PSAM;
2. En se servant des valeurs de paramètres canal estimées, les signaux interférents sont reconstruits et supprimés du slot temporel en question. En pratique, la première estimation n'est pas parfaite et des erreurs résiduelles sont ajoutées au signal restant;
3. Même en présence d'erreurs résiduelles d'estimation de canal, le signal d'intérêt est envoyé au décodeur;
4. Si le décodage n'est pas réussi, les bits décodés erronnés sont retransmis au dispositif d'estimation de canal dans la prochaine itération;
5. À cette étape, le dispositif d'estimation de canal utilise les séquences de symboles reconstruites à partir des bits décodés au lieu des symboles de signalisation. Les valeurs de paramètres de canal estimées sont utilisées pour la démodulation et le décodage du signal utile;
6. Le recepteur répète les étapes 2-5 en utilisant pas seulement les symboles de signalisation, mais aussi les symboles reconstruits à partir des bits décodés, afin de rendre les résultats d'estimation de canal plus précis.

8.2.4 Résultats avec PSAM

Les résultats de PER obtenus avec PSAM sont illustrés dans les figures 8.5 et 8.6. Nous pouvons observer une dégradation de 0.3 dB en présence de décalages temporels entre les utilisateurs même en appliquant l'algorithme d'estimation EM avec JED. Par contre, sans décalages temporels, la dégradation ne dépasse pas 0.1 dB même avec 4 paquets interférents.

8.2.5 Conclusion

Dans cette section, nous avons étudié l'impact des erreurs résiduelles d'estimation de canal sur l'annulation des interférences dans les nouvelles méthodes d'accès aléatoire. La technique d'estimation de canal proposée combine l'algorithme EM avec initialisation par autocorrélation et l'utilisation du PSAM. Nous avons également proposé d'appliquer une estimation combinée au décodage afin d'améliorer la performance du SIC. Grâce à l'algorithme proposé, nous avons pu obtenir une perte de performances relativement faible. De plus, nous avons pu remarquer que les décalages temporels entre différents paquets causent des pertes supplémentaires en termes de PER. Nous pouvons en conclure que la performance de l'algorithme d'estimation proposé n'est pas optimale dans ce cas.

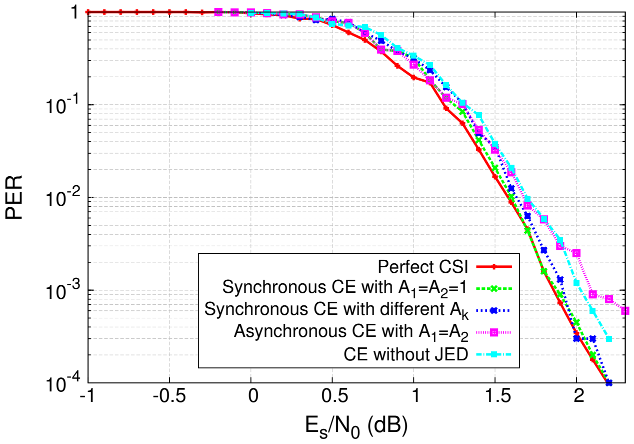


Figure 8.5 – PER vs. E_s/N_0 avec estimation de canal et suppression d'interferences dans le cas d'un seul paquet interférent, avec et sans décalages temporels. CCSDS turbo code avec taux 1/2 et QPSK. $L = 460$ symbols.

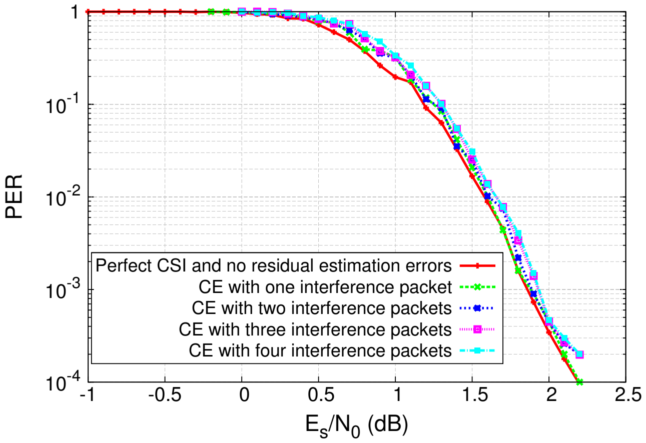


Figure 8.6 – PER vs. E_s/N_0 avec estimation de canal et suppression d'interferences dans le cas de plusieurs paquets interférens. CCSDS turbo code avec taux 1/2 et QPSK. $L = 460$ symbols.

8.3 Multi-replicA decoding using corRelation baSed locALisAtion

Dans la section précédente, nous avons évalué les problèmes liés aux erreurs résiduelles d'estimation de canal pour les nouvelles méthodes d'accès aléatoire. En se basant sur les résultats obtenus, nous proposons une nouvelle technique de réception des paquets sur un canal d'accès aléatoire, qui permet d'améliorer le débit et de réduire le PER. Cette méthode est appelée *Multi-replicA decoding using corRelation baSed locALisAtion* (MARSALA). Elle utilise le même schéma de transmission de répliques de paquets que CRDSA. Cependant, MARSALA présente une nouvelle technique de décodage pour CRDSA qui permet de localiser toutes les répliques d'un paquet en utilisant des corrélations temporelles. Ensuite, les répliques d'un même paquet sont combinées afin de décoder les données reçues.

8.3.1 Hypothèses

La figure 8.7 montre le modèle du système considéré. Nous considérons un nombre N_u de terminaux terrestres fixes partageant une seule bande de fréquence sur le canal d'accès aléatoire d'une liaison montante. Chaque trame a une durée T_F et contient N_s slots. Chaque terminal transmet un nombre N_b de répliques d'un même paquet sur des slots temporels quelconques. Nous supposons que chaque terminal doit attendre le début de la trame suivante afin d'envoyer un autre paquet. De plus, nous supposons qu'il n'y a pas de lien direct entre les terminaux. Le modèle de canal considéré est un canal AWGN. La densité spectrale de puissance du bruit notée N_0 est constante sur une durée T_F, et les paquets sont tous transmis et reçus à la même puissance.

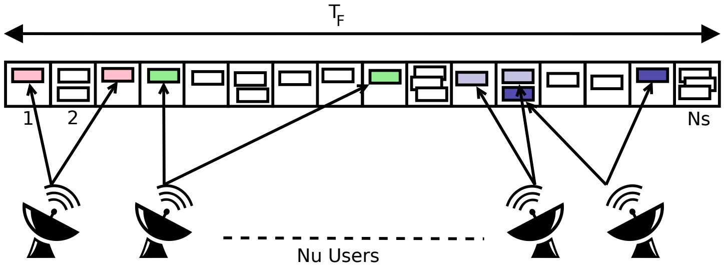


Figure 8.7 – Modèle du système: une trame avec N_s slots temporels sur une liaison montante partagée entre N_u terminaux terrestres, chacun transmettant $N_b = 2$ répliques par paquet.

Chaque paquet est encodé avec un turbocode de taux R et modulé avec une modulation de l'ordre M. Un préambule, un postambule et des symboles pilotes sont ajoutés au mot de code obtenu, permettant l'estimation de canal. Des champs de signalisation servants à localiser les répliques sont également ajoutés à chaque paquet. La longueur totale d'un paquet est égale à L symboles. La séquence de symboles constituant chaque paquet entre dans un filtre de mise en forme modélisé par une racine de cosinus surélevé. Au niveau du récepteur, le signal est suréchantillonné par un facteur Q, et il est entré dans un filtre adapté afin d'atténuer le bruit de manière optimale avant d'être échantillonné à la période symbole. Nous considérons un système combinant CRDSA et MARSALA.

Supposons qu'au niveau du récepteur nous recevons sur des slots séparés N_b signaux, r_1 et r_i (avec $i \in [2, N_b]$) correspondants à N_b répliques d'un même paquet. Le slot de référence TS_{ref} contient le signal r_1 correspondant à la première réplique d'un paquet quelconque plus des paquets interférents. Donc, r_1 peut être exprimé comme suit :

$$r_1(t) = y(t+\tau_1)e^{j(\phi_1+2\pi\Delta f_1 t)} + n_1(t) + \zeta_1(t) \quad , \tag{8.12}$$

avec y étant le signal utile correspondant à la première réplique d'un paquet quelconque. τ_1, ϕ_1, et Δf_1 notent respectivement le décalage temporel, le déphasage, et le décalage fréquentiel du signal y sur TS_{ref}. n_1 est le terme d'AWGN et ζ_1 représente la somme des interférences sur

TS_{ref}. Le signal y est détaillé comme suit :

$$y(t) = \sum_{k=0}^{L-1} a_k h(t - kT) \quad , \tag{8.13}$$

avec a_k étant le $k^{ème}$ symbole transmis, et h étant la fonction du filtre de mise en forme. Le signal r_i reçu sur les slots autres que TS_{ref} et contenant la $i^{ème}$ réplique de y, est exprimé comme montré ci-dessous :

$$r_i(t) = y(t - N_i T + \tau_i) e^{j(\phi_i + 2\pi\Delta f_1 t)} + n_i(t) + \zeta_i(t) \quad , \tag{8.14}$$

avec N_i étant le nombre de symboles séparant la première réplique du paquet utile sur TS_{ref} de sa $i^{ème}$ réplique. τ_i et ϕ_i sont respectivement le décalage temporel et le déphasage du signal y sur le slot contenant la $i^{ème}$ réplique. n_i et ζ_i représentent respectivement, le terme d'AWGN et la somme des interférences sur le slot contenant la $i^{ème}$ réplique.

8.3.2 Étapes détaillées de MARSALA

Localisation des répliques

Pour localiser les répliques d'un paquet donné, le récepteur calcule l'inter-corrélation temporelle entre r_1 et les signaux reçus sur chaque slot temporel. Les pics de corrélation les plus hauts indiquent les emplacements des répliques d'un paquet présent sur TS_{ref}. Le signal r_i décrit dans l'équation (8.15), peut être exprimé en fonction de r_1, n_1, ζ_1, n_i et ζ_i comme indiqué ci-dessous :

$$r_i(t) = r_1(t - N_i T + \Delta\tau_{i,1}) e^{j\Delta\phi_{i,1}} + n_{tot}(t) \quad , \tag{8.15}$$

avec $N_i T - \Delta\tau_{i,1} = N_i T_s - (\tau_i - \tau_1)$ étant le décalage temporel entre un paquet sur TS_{ref} et sa $i^{ème}$ réplique. $\Delta\phi_{i,1} = \phi_i - \phi_1 + 2\pi\Delta f_1 (N_i T + \ \Delta\tau_{i,1})$ est le déphasage entre les deux répliques. n_{tot} est un signal englobant le bruit AWGN et les signaux d'interférence exprimé comme suit :

$$n_{tot}(t) = [n_i(t) + \zeta_i(t)] - [n_1(t - N_i T + \Delta\tau_{i,1}) + \zeta_1(t - N_i T + \Delta\tau_{i,1})] e^{j\Delta\phi_{i,1}} \quad . \tag{8.16}$$

Nous calculons la fonction d'inter-corrélation entre r_i et r_1 notée $R_{r_i,r_1}(\tau)$ comme suit :

$$R_{r_i,r_1}(\tau) = \int_0^{T_{slot}} r_i(t) r_1^*(t - \tau) dt = R_{r_1}\left(\tau - (N_i T - \Delta\tau_{i,1})\right) e^{j\Delta\phi_{i,1}} + R_{n_{tot},r_1}(\tau) \quad , \tag{8.17}$$

avec R_{r_1} étant la fonction d'autocorrélation du signal r_1 et R_{n_{tot},r_1} étant la fonction d'inter-corrélation entre n_{tot} et r_1. $(\cdot)^*$ représente la valeur conjuguée. Nous pouvons conclure que l'amplitude de $R_{r_i,r_1}(\tau)$ atteint sa valeur maximale lorsque $\tau = \tau_{max,i} = N_i T - (\tau_i - \tau_1)$. Cette valeur représente le décalage temporel entre la première et la $i^{ème}$ réplique d'un même paquet. Au cas où le nombre de répliques $N_b > 2$, une première corrélation est effectuée pour détecter la première réplique, ensuite le signal sur TS_{ref} et sa première réplique sont combinées et d'autres opérations de corrélation entre le signal combiné et le reste des signaux sur la trame permettent de localiser les autres répliques.

Estimation et correction des décalages temporels et des déphasages entre les répliques

Étant donné que les répliques d'un même paquet sont reçues sur des slots temporels distincts, elles ont des décalages temporels et des déphasage différents. Par conséquent, avant de procéder à la combinaison de répliques, le récepteur doit estimer et corriger le décalage temporel $\tau_{max_i} = N_i T - (\tau_i - \tau_1)$ et le déphasage $\Delta\phi_{i,1} = \phi_i - \phi_1 + 2\pi\Delta f_1(N_i T + \ \Delta\tau_{i,1})$ entre les répliques reçues. Vu que τ_{max_i} et $\Delta\phi_{i,1}$ sont censés rester constants pendant la durée d'un paquet, ils peuvent être calculés en utilisant la fonction d'inter-corrélation entre les deux signaux r_1 et r_i.

Le décalage temporel entre les répliques peut être estimé par la valeur de τ qui maximise l'amplitude de $R_{i,1}$ dans l'equation (8.17). Néanmoins, étant donné que le signal reçu est échantillonné avec un facteur de sur-échantillonnage Q, donc si τ_{max_i} n'est pas un multiple entier de $\frac{T}{Q}$, le décalage temporel estimé contiendrait une erreur de synchronisation err_i. Cette erreur de synchronisation peut être modélisée par une variable aléatoire uniformément répartie sur $[-T/2Q, T/2Q]$.

$$\hat{\tau}_{max_i} = \underset{\tau}{\operatorname{argmax}} \left| R_{r_i,r_1}(\tau) \right| = N_i T_s - (\tau_i - \tau_1) + err_i \quad . \tag{8.18}$$

La valeur de la fonction d'inter-corrélation $R_{r_i,r_1}(\tau)$ pour $\tau = \hat{\tau}_{max_i}$ est :

$$\begin{aligned} R_{r_i,r_1}(\hat{\tau}_{max_i}) =& R_{r_1}(err_i)\, e^{j\Delta\phi_{i,1}} + R_{n_{tot},r_1}(\hat{\tau}_{max_i}) \\ \simeq& LP(a_k) \left[\frac{H(err_i)}{T} \right] e^{j(\Delta f_1 err_i + \Delta\phi_{i,1})} + R_{tot} \quad , \end{aligned} \tag{8.19}$$

avec $H(t)$ et $h(t)$ étant respectivement les fonctions de cosinus sur-élevé et sa racine carrée. $P(a_k)$ est la puissance des symboles a_k, supposée normalisée pour tout $k \in [0, L-1]$. la somme des termes d'inter-corrélation notée R_{tot} est détaillée ci-dessous :

$$R_{tot} = R_{y,(n_1+\zeta_1)}(\hat{\tau}_{max_i}) + R_{(n_1+\zeta_1),y}(\hat{\tau}_{max_i}) + R_{(n_1+\zeta_1),(n_1+\zeta_1)}(\hat{\tau}_{max_i}) + R_{n_{tot},r_1}(\hat{\tau}_{max_i}) \quad . \tag{8.20}$$

Ainsi, l'estimation du déphasage $\widehat{\Delta\phi}_{i,1}$ est calculé comme étant l'angle de $R_{r_i,r_1}(\hat{\tau}_{max_i})$. Comme le montre l'équation (8.21), $\widehat{\Delta\phi}_{i,1}$ contient une erreur de phase ϕ_{err_i} qui dépend de $\Delta f_1 err_i$, $H(err_i)$ et les fonctions d'inter-corrélation entre le bruit, les interférences et le signal utile.

$$\widehat{\Delta\phi}_i = \operatorname{angle}\left(R_{r_i,r_1}(\hat{\tau}_{max_i}) \right) = \Delta\phi_i + \phi_{err_i} \quad . \tag{8.21}$$

Le signal r_i obtenu après correction du décalage temporel et du déphasage et suite au suréchantillonnage, est :

$$r_i(n) = y\left(n\frac{T}{Q} + \tau_1 + err_i\right) e^{j\left(\phi_1 + \phi_{err_i} + 2\pi\Delta f_1 n\frac{T}{Q}\right)} + \left[n_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right) + \zeta_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right)\right] e^{-j\widehat{\Delta\phi}_{i,1}} \quad . \tag{8.22}$$

Nous exprimons aussi le signal r_1 après suréchantillonnage d'un facteur Q :

$$r_1(n) = y\left(n\frac{T}{Q} + \tau_1\right) e^{j\left(\phi_1 + 2\pi\Delta f_1 n\frac{T}{Q}\right)} + n_1\left(n\frac{T}{Q}\right) + \zeta_1\left(n\frac{T}{Q}\right). \tag{8.23}$$

Combinaison des répliques

Après la synchronisation en temps et en phase des répliques selon le schéma décrit ci-dessus, nous effectuons la combinaison des répliques en sommant les signaux r_1 (de l'equation (8.23)) et r_i (de l'équation (8.22)) pour $i = [2, \cdots, N_b]$. Le signal résultant r_{sum} est exprimé ci-dessous :

$$r_{sum}(n) = y_{sum}(n) e^{j\left(\phi_1 + 2\pi\Delta f_1 n\frac{T}{Q}\right)} + z_1\left(n\frac{T}{Q}\right) + \sum_{i=2}^{N_b} z_i\left(n\frac{T}{Q} + \widehat{\tau}_{max_i}\right) e^{-j\widehat{\Delta\phi}_i} \quad , \tag{8.24}$$

avec z_1 et z_i étant les termes de bruit plus interférence dans les signaux r_1 et r_i, respectivement. Le signal y_{sum} est détaillé ci-dessous :

$$y_{sum}(n) = y\left(n\frac{T}{Q} + \tau_1\right) + \sum_{i=2}^{N_b} y\left(n\frac{T}{Q} + \tau_1 + err_i\right) e^{j\phi_{err_i}} \quad . \tag{8.25}$$

Estimation de canal

Avant la démodulation et le décodage, le récepteur doit effectuer une estimation de canal pour le signal utile y_{sum}. Comme le SNIR est relativement bas et nous n'avons pas d'informations sur les utilisateurs présents sur chaque slot, nous proposons d'utiliser un algorithme d'estimation se basasnt sur les symboles de signalisation et de l'associer à une boucle itérative d'estimation et de décodage (JED). D'abord, nous estimons le décalage fréquentiel en utilisant le signal après passage dans le filtre adapté $\tilde{r}_{sum}$ et la séquence de symboles de signalisation de l'utilisateur 1, notée y_{train} :

$$\widehat{\Delta f}_1 = \underset{\Delta f}{argmax} \left| \sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum}(n) y_{train}(n-m) e^{-j2\pi\Delta f n\frac{T}{Q}} \right|^2 \quad . \tag{8.26}$$

Le décalage temporel est estimé en utilisant l'equation suivante :

$$\widehat{\tau}_1 = \underset{n}{argmax} \left| \sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum}(n) y_{train}(n-m) e^{-j2\pi\widehat{\Delta f}_1 n\frac{T}{Q}} \right|^2 \quad . \tag{8.27}$$

Le déphasage est estimé comme le montrent les equations ci-dessous :

$$n_{peak} = \underset{n}{argmax} \left| \underbrace{\sum_{m=-(QL-1)}^{(QL+1)} \tilde{r}_{sum_{\tau_1}}(n) y_{train}(n-m) e^{-j2\pi\widehat{\Delta f}_1 n \frac{T}{Q}}}_{R_{\widehat{\phi}_1}(n)} \right|^2 \qquad (8.28)$$

$$\widehat{\phi}_1 = angle\Big(R_{\widehat{\phi}_1}(n_{peak})\Big) \qquad (8.29)$$

Filtrage adapté et sous-échantillonnage

Dans cette dernière étape, le signal discret r_{sum} passe dans un filtre adapté prenant en compte le décalage temporel estimé $\widehat{\tau}_1$ et il est ensuite sous-echantillonné aux instants $k'T_s$, avec k' étant un entier variant entre 0 et $L-1$. Le signal résultant $\tilde{r}_{sum_{\tau_1}}$ est exprimé ci-dessous :

$$\tilde{r}_{sum_{\tau_1}}(k'T) = \tilde{y}_{sum_{\tau_1}}(k'T) e^{j(\phi_1 + 2\pi\Delta f_1 k'T)} + \tilde{z}_1(k'T) + \sum_{i=2}^{N_b} \tilde{z}_i(k'T + \widehat{\tau}_{max_i}) e^{-j\widehat{\Delta\phi}_{i,1}} \quad , \qquad (8.30)$$

avec $\tilde{(\cdot)}$ représentant le signal à la sortie du filtre adapté, et $\tilde{y}_{sum_{\tau_1}}$ est detaillé comme suit :

$$\tilde{y}_{sum_{\tau_1}}(k'T) = \tilde{s}\left(k'T + \underbrace{(\tau_1 - \widehat{\tau}_1)}_{err_1}\right) + \sum_{i=2}^{N_b} \tilde{s}\left(k'T + \underbrace{(\tau_1 - \widehat{\tau}_1)}_{err_1} + err_i\right) e^{j\phi_{err_i}} \quad , \qquad (8.31)$$

avec $err_1 = \tau_1 - \widehat{\tau}_1$ modélisé par une variable aléatoire uniforme sur $\left[\frac{-T}{2Q}, \frac{T}{2Q}\right]$.

8.3.3 Modèle analytique pour la combinaison des répliques avec erreurs de synchronisation

Comme les répliques combinées dans $\tilde{y}_{sum_{\tau_1}}$ n'ont pas exactement les mêmes décalages temporels et déphasages, nous proposons d'évaluer analytiquement l'impact de ces erreurs de synchronisation sur la combinaison des répliques dans MARSALA. Pour cela, nous divisons le signal $\tilde{y}_{sum_{\tau_1}}$ en deux termes : un terme utile et un terme contenant les interférences inter-symboles (ISI), comme montré ci-dessous :

$$y_{sum,des}(k'T) = a_{k'} H(err_1) + \sum_{i=2}^{N_b} a_{k'} H\underbrace{(err_1 + err_i)}_{err_{1,i}} e^{j\phi_{err_i}} \quad , \qquad (8.32)$$

avec $a_{k'}$ éatant le $k'^{ème}$ symbole dans y. $err_{1,i}$ suit une loi triangulaire sur $[-\frac{T}{Q}, \frac{T}{Q}]$. Pour le terme d'ISI, nous considérons juste les 3 premiers lobes latéraux du filtre en cosinus sur-élevé:

$$y_{sum,isi}(k'T) = \sum_{l=k'-3,l\neq k'}^{k'+3} a_l H\big((k'-l)T + err_1\big) + \sum_{i=2}^{N_b}\left(\sum_{l=k'-3,l\neq k'}^{k'+3} a_l H\big((k'-l)T + err_{1,i}\big)\right) e^{j\phi_{err_i}}. \tag{8.33}$$

Ainsi, nous pouvons écrire le SNIR équivalent moyen tel que :

$$SNIR_{eq} = \frac{E\left[P\left(y_{sum,des}\right)\right]}{E\left[P\left(y_{sum,isi}\right)\right] + (I_1 + N_0) + \sum\limits_{i=2}^{N_b}(I_i + N_0)} \quad , \tag{8.34}$$

avec $P\left(y_{sum,des}\right)$, $P\left(y_{sum,isi}\right)$, I_1 et I_i étant les puissances de $y_{sum,des}$, $y_{sum,isi}$, et des signaux d'interférence sur TS_{ref} et sur le slot contenant la $i^{ème}$ réplique, respectivement.

Puissance moyenne du signal utile

Si nous détaillons la puissance de $y_{sum,des}$, nous obtenons :

$$\begin{aligned} P(y_{sum,des}) &= \frac{1}{L}\sum_{k'=0}^{L-1}\left|y_{sum,des}(k'T_s)\right|^2 \\ &= H^2(err_1) + \sum_{i=2}^{N_b} H^2(err_{1,i})\left[cos^2(\phi_{err_i}) + sin^2(\phi_{err_i})\right] \\ &\quad + 2\sum_{i,j\neq i}^{N_b} H(err_{1,i})H(err_{1,j})cos(\phi_{err_i})cos(\phi_{err_j}) \\ + 2\sum_{i,j\neq i}^{N_b} & H(err_{1,i})H(err_{1,j})sin(\phi_{err_i})sin(\phi_{err_j}) + 2H(err_1)\sum_{i=2}^{N_b} H(err_{1,i})cos(\phi_{err_i}) \quad . \end{aligned} \tag{8.35}$$

La moyenne de $P(y_{sum,des})$ est calculée en détail dans les pages 83-84 de la version en anglais de cette thèse.

Puissance moyenne du signal d'ISI

Pour calculer la puissance moyenne du signal d'ISI, nous procédons de manière analogue à l'étude dans [82] pour les systèmes MISO coopératifs avec erreurs de synchronisation. Par conséquent, nous rapprochons la fonction de cosinus sur-élevé à une fonction linéaire par morceaux avec des pentes m_l à chaque instant $K't$. Pour considérer un pire cas, nous approximons $a_l G((k'-l)T + err) = |m_l|\frac{err}{T}$. Ainsi, la puissance moyenne du signal d'ISI peut être écrit comme suit :

$$E\left[P\left(y_{sum,isi}\right)\right] = E\left[\left|y_{sum,isi}\right|^2\right] = var\left[s_{sum,isi}\right] + E^2\left[s_{sum,isi}\right] \quad , \tag{8.36}$$

avec :

$$E\left[y_{sum,isi}\right] = \frac{\beta}{4Q} + \sum_{i=2}^{N_b} \frac{\beta}{3Q} \quad , \tag{8.37}$$

$$var\left[y_{sum,isi}\right] = \frac{\beta^2}{48Q^2} + \sum_{i=2}^{N_b} \frac{\beta^2}{18Q^2} \quad . \tag{8.38}$$

8.3.4 Résultats numériques

Dans ce qui suit, nous utilisons les résultats numériques du modèle analytique proposé ci-dessus, afin d'évaluer l'impact des erreurs de synchronisation sur les performances de MARSALA, en termes de débit et de taux d'erreurs paquets (en anglais, *Packet Loss Ratio* - PLR). Tous les paquets sont supposés reçus avec le même niveau de puissance. Chaque trame est composée de $N_s = 100$ slots. Le nombre total de terminaux est égal à λ. Les erreurs résiduelles d'estimation de canal causées par l'annulation imparfaite des interférences ont été prises en compte. Nous utilisons une modulation QPSK et le turbo code DVB-RCS2 pour modulations linéaires de taux de 1/3 (désigné par la forme d'onde id 3 dans [28]). Le scénario considéré dans nos simulations est le suivant: une fois qu'une trame est reçue, le récepteur tente de décoder un paquet en utilisant CRDSA. Si le décodage n'est pas réussi, le récepteur applique MARSALA pour localiser et combiner les répliques de ce même paquet. Afin de comparer plusieurs schémas de modulation et de codage, la charge normalisée (G) est exprimée en bits par symbole et calculée telle que

$$G = R \times \log_2(M) \times \frac{\lambda}{N_s} \quad ,$$

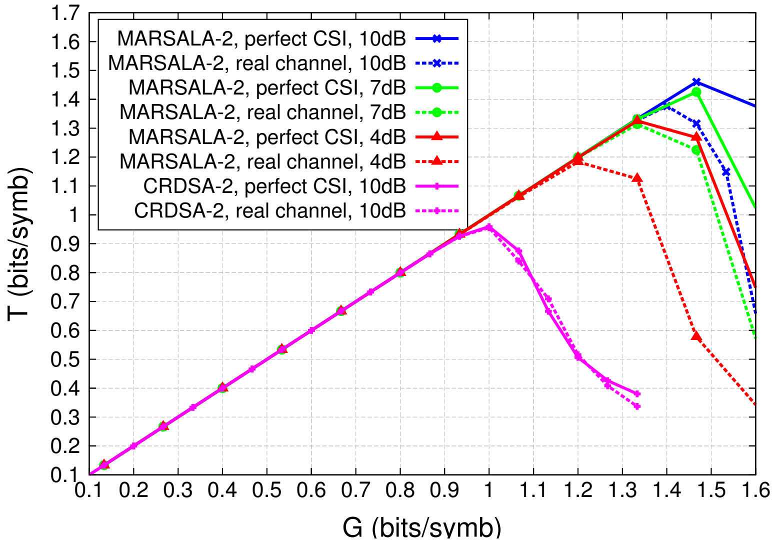
R étant le taux de code et M l'ordre de modulation. Le débit normalisé (T) est donné par:

$$T = G \times (1 - PLR(G)) \quad ,$$

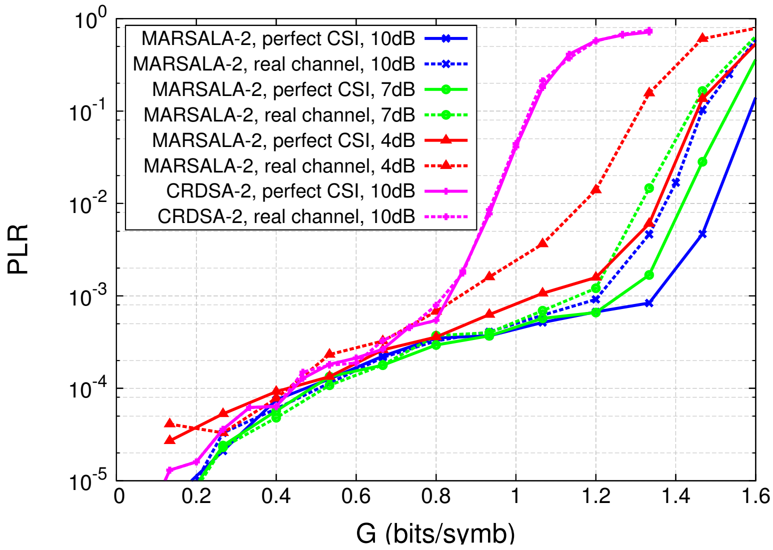
où PLR(G) est la probabilité qu'un paquet soit non décodé pour une certaine valeur G et pour un SNIR donné. Les figures 8.8 et 8.9 illustrent le débit et le PLR obtenus avec MARSALA-2 et MARSALA-3. Deux scénarios sont comparés: le cas de la connaissance parfaite de l'état du canal (CSI), ainsi que le cas de conditions de canal réel. Ces deux figures montrent également une comparaison entre les performances de MARSALA et CRDSA.

8.4 Schémas d'optimisation des performances de MARSALA

Préceédemment, nous avons pu observer de bonnes performances de MARSALA dans un scénario de paquets à même niveaux de puissance, utilisant le turbocode DVB-RCS2 pour modulations linéaires. Cependant, il a été prouvé dans des études antérieures, en particulier pour CRDSA [27, 38], que la réception des paquets avec des niveaux de puissance différents, améliore considérablement les performances du SIC. De même, il a été démontré que le



(a)

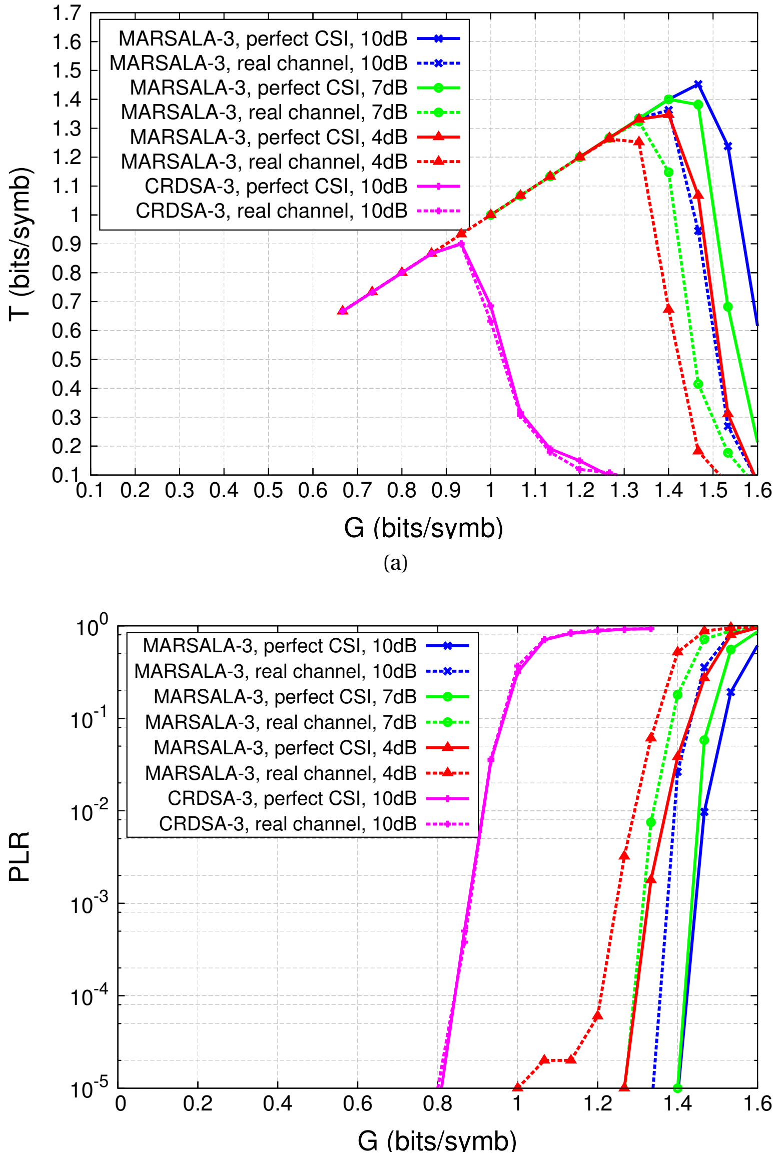


(b)

Figure 8.8 – Performance de MARSALA-2 dans des conditions de canal réel comparées à une connaissance parfaite du canal. $E_s/N_0 = 4$, 7 et 10 dB; QPSK avec turbocode DVB-RCS2 $R = 1/3$. (a) Débit. (b) PLR.

schéma de codage aurait un impact sur le déclenchement de plus d'itérations de SIC. Pour ces raisons, nous proposons trois schémas d'optimisation de MARSALA qui se résument par:

- Ajout du principe de *Maximum Ratio Combining* (MRC) [84, 85] à MARSALA.
- Évaluation de MARSALA avec des paquets reçus à des niveaux de puissance inégaux.
- Étude de MARSALA avec des schémas de codage autre que DVB-RCS2, tels que 3GPP et CCSDS.



(a)

(b)

Figure 8.9 – Performance de MARSALA-3 dans des conditions de canal réel comparées à une connaissance parfaite du canal. $E_s/N_0 = 4$, 7 et 10 dB; QPSK avec turbocode DVB-RCS2 $R = 1/3$. (a) Débit. (b) PLR.

8.4.1 MARSALA avec MRC

MRC consiste à multiplier chaque signal contenant une réplique d'un même paquet par un coefficient α_k proportionnel à son rapport SNIR. Nous appliquons aussi le MRC sur MARSALA en divisant chaque signal par sa puissance reçue. Le SNIR équivalent obtenu avec MRC est exprimé tel que :

$$SNIR_{eq} = \frac{\left(\sum_{k=1}^{N_b} \alpha_k\right)^2}{\sum_{k=1}^{N_b} \alpha_k^2 SNIR_k^{-1}} \quad , \tag{8.39}$$

Les résultats de débit et de PLR obtenus avec les deux techniques de MRC basées sur le SNIR (MRC-SNIR) ainsi que la puissance reçue (MRC-P), sont illustrés dans les figures 8.10 et 8.11.

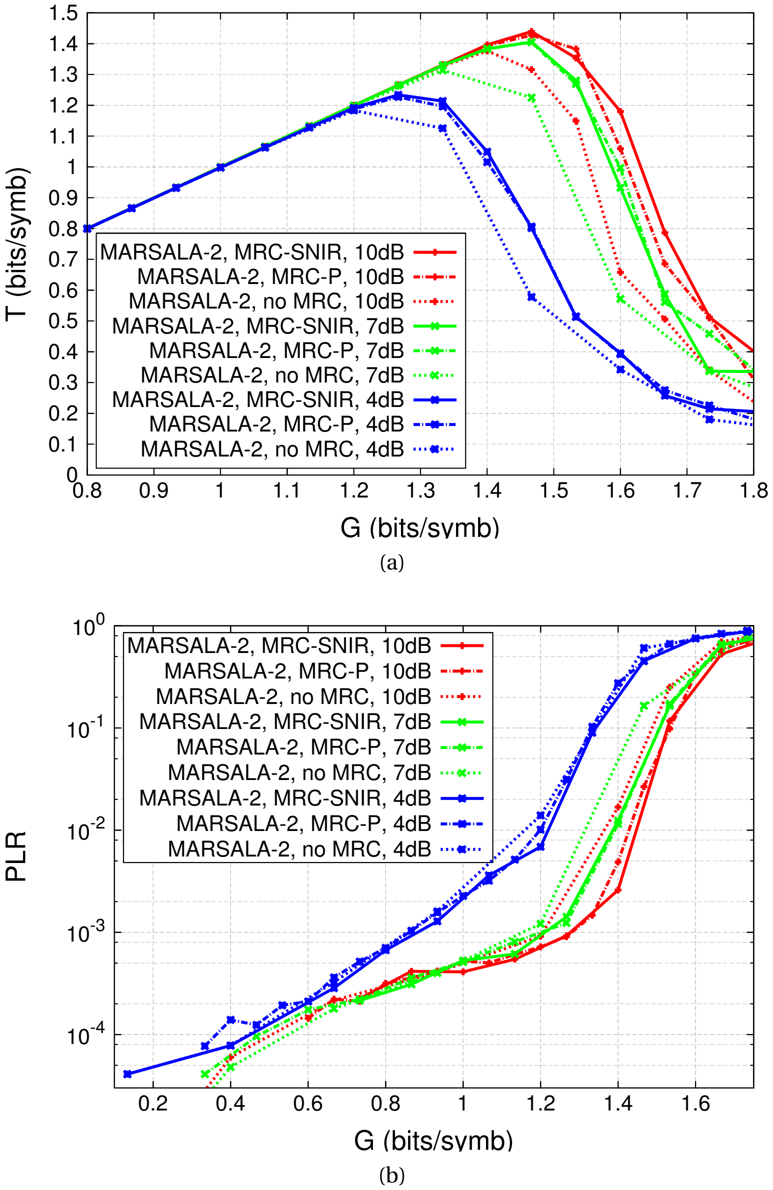
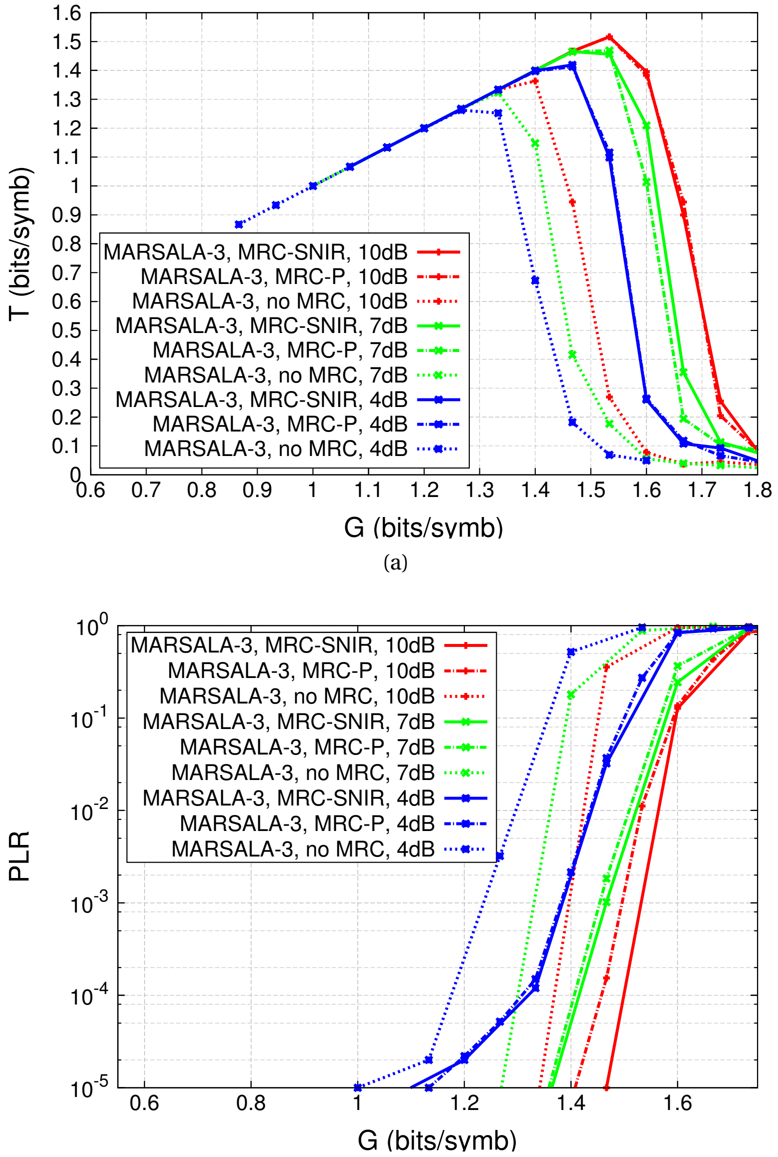


Figure 8.10 – Comparaison de MARSALA avec et sans MRC. $N_b = 2$ répliques. Modulation QPSK, turbocode DVB-RCS2 $R = 1/3$, $E_s/N_0 = 4$, 7 et 10 dB. Paquets à même puissance. $L = 456$ symboles. (a) Débit. (b) PLR.



(a)

(b)

Figure 8.11 – Comparaison de MARSALA avec et sans MRC. $N_b = 3$ répliques. Modulation QPSK, turbocode DVB-RCS2 $R = 1/3$, $E_s/N_0 = 4$, 7 et 10 dB. Paquets à même puissance. $L = 456$ symboles. (a) Débit. (b) PLR.

8.4.2 MARSALA avec des puissances inégales

Dans ce qui suit, nous évaluons les performances de MARSALA avec différentes lois de distribution de puissances des paquets reçus, listées ci-dessous :

- **La loi log-normale (en linéaire)**, avec les paramètres $\mu = 0$ et $\sigma \neq 0$ et $E_s/N_0 = 10$ dB.
- **La loi uniforme (en dB)**, avec E_s/N_0 variant dans l'intervalle $[4, 16]$ dB.
- **La loi demi-normale (en dB)**, avec les paramètres $\mu = 4$ et $\sigma = 7$ et E_s/N_0 variant sur

$[4, 16]$ dB.

- **La loi demi-normale inversée (en dB)**, avec $\mu = 4$ et $\sigma = 7$.

Les comparaisons des performances de MARSALA avec ces différentes distributions sont résumées dans la table 8.1.

Loi	Iso puissances $E_s/N_0 = 10$ dB	Logn $\sigma = 2$ dB	Logn $\sigma = 3$ dB	Demi- normale	Uniforme	Demi-normale inversée
T	1.47	2	2.27	2.53	2.6	2.27
Gain	-	36%	54%	72%	77%	54%

Table 8.1 – Comparaison du débit maximal obtenu avec MARSALA-3 (T en bits/symbol) et du gain en performance par rapport à un scénario Iso-puissance, obtenu à un PLR autour de 10^{-4}, avec différentes distributions de puissance et MRC.

8.4.3 MARSALA avec différents schémas de codage

Les performances de MARSALA sont évaluées avec 3 schémas de codage définis dans la liste ci-dessous, et les résultats des simulations en termes de débit et de PLR sont montrés dans la figure 8.12.

- **Turbocode DVB-RCS2 pour modulations linéaires [10]**, qui est un code circulaire récursif systématique convolutif (CRSC) à 16-états. Dans les simulations, nous considérons la forme d'onde id-3 définie dans la norme DVB-RCS2 (longueur de charge utile = 456 symboles, avec modulation QPSK et taux de code $R = 1/3$).
- **Turbocode 3GPP [25]**, avec une longueur de charge utile égale à 225 symboles.
- **Turbocode du *Consultative Committee for Space Data Systems* (CCSDS) [?]**, avec un taux de code $R = 1/3$ construit à partir de longueurs de bloc d'information de 456 bits.

Finalement, nous pouvons observer que les performances optimales sont obtenues en utilisant le turbocode 3GPP avec 225 symboles de charge utile. Pour cela, nous combinons ce schéma de codage avec les distributions de puissances déjà définies et nous montrons les résultats obtenus dans la figure 8.13.

8.5 Conclusion et perspectives

Dans cette thèse nous pouvons conclure les points suivants :

- Dans MARSALA, nous avons pu profiter de l'information commune contenue dans les signaux reçus sur des slots temporels distincts, afin de localiser les répliques d'un même paquet et les combiner.
- L'étude analytique que nous avons présentée pour évaluer l'impact de la combinaison

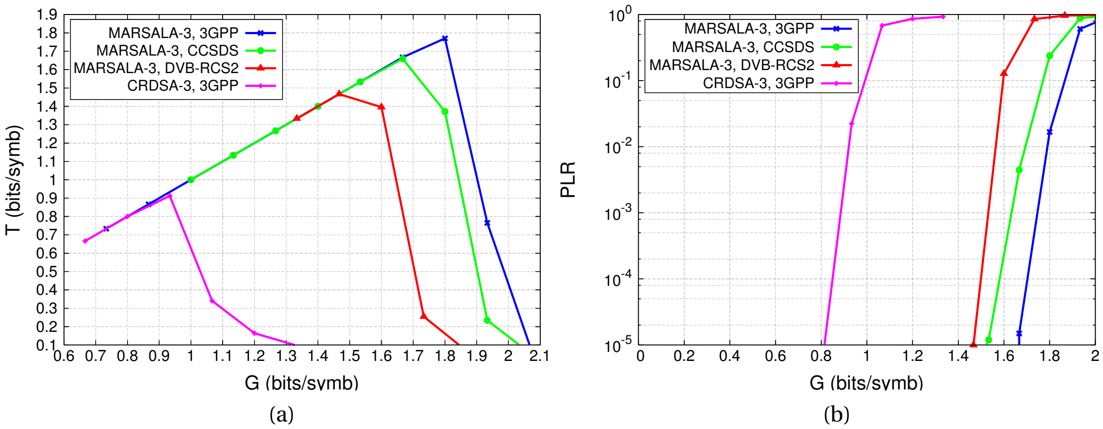


Figure 8.12 – Comparaison de MARSALA-3 avec les turbocodes DVB-RCS2, 3GPP et CCSDS. Courbe de référence: CRDSA-3 avec 3GPP. Modulation QPSK, taux de code $R = 1/3$. Paquets à iso-puissance dans des conditions de canal réel avec MRC. (a) Débit. (b) PLR.

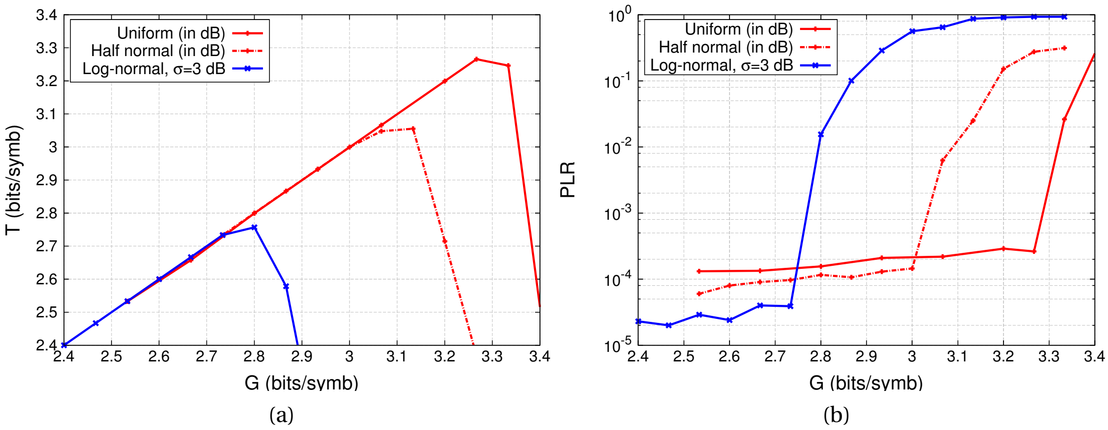


Figure 8.13 – Comparison des performances de MARSALA-3 avec le turbocode 3GPP et différentes lois de distribution de puissances. Modulation QPSK, taux de code $R = 1/3$. MRC appliqué. (a) Débit. (b) PLR.

imparfaite de répliques, ainsi que les résultats de simulations montrent une légère dégradation des performances par rapport au cas de connaissance parfaite du canal. Cependant, même avec les erreurs de synchronisation prises en compte, les gains en débit et PLR sont substantiels par rapport à CRDSA.

- MARSALA a permis d'obtenir de bonnes performances en termes de débit et de PLR même à faibles niveaux d'E_s/N_0. Par conséquent, nous pouvons conclure qu'elle serait bien adaptée pour des scénarios de communication avec des transmissions à faible puissance. Ce résultat incite à évaluer MARSALA avec des applications pour l'Internet des Objets (IdO) et les communications M2M où les terminaux sont à faible coût et nécessitent des transmissions à puissances très faibles.

- Le système est plus stable avec MARSALA, comme nous pouvons obtenir un PLR faible pour des charges élevées sur la trame.

Les futurs travaux de cette thèse que nous pouvons citer sont les suivants:

- Dans MARSALA, un paramètre à prendre en considération dans les études futures est le bruit de phase. Ce bruit peut être représenté sous la forme d'un processus stochastique avec des variations de fréquence à court terme. Par conséquent, il peut avoir un impact sur la précision de la localisation des répliques en utilisant la corrélation.
- Il pourrait s'avérer intéressant pour les travaux futurs de trouver analytiquement une distribution de puissance de paquets optimale afin d'atteindre un débit maximal pour un PLR cible.
- Les travaux futurs devraient également fournir une évaluation des performances de MARSALA en termes de délais de paquets de bout en bout et en termes de sa stabilité dans les régimes impliquant des retransmissions de paquets.
- Il est également important d'évaluer la complexité induite par MARSALA au niveau du récepteur en termes de nombre d'opérations de corrélation nécessaires à la localisation des répliques et à leur combinaison.

Quelques perspectives ouvertes sont :

- La proposition d'une version irrégulière de MARSALA, par analogie à la version irrégulière de CRDSA, *Irregular Repitition Slotted Aloha* (IRSA) [18], où un nombre irrégulier de répliques peut être transmis par chaque utilisateur sur la trame.
- Une version de MARSALA avec des taux de codages irréguliers.
- Une suggestion pour appliquer MARSALA en mode asynchrone.

BIBLIOGRAPHY

[1] H. C. Bui. *Méthodes d'accès basées sur le codage réseau couche physique*. PhD thesis, Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), University of Toulouse, 2012.

[2] R. A. Ferdman. 4.4 billion people around the world still don't have Internet. Here's where they live. *The Washigton Post*, Oct 2014.

[3] K. Finly. Internet by Satellite Is a Space Race With No Winners. *Wired*, Dec 2015.

[4] Forrester Consulting. eCommerce Web Site Performance Today: An Updated Look At Consumer Reaction To A Poor Online Shopping Experience. *A commissioned study conducted on behalf of Akamai Technologies, Inc.*, August 2009.

[5] Aberdeen Group, a Harte-Hanks company. The Performance of Web Applications: Customers are Won or Lost in One Second. Technical report, Aberdeen, Nov 2008.

[6] DVB Document ETSI EN 300 421, Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for 11/12 GHz satellite services, 1997.

[7] DVB Document ETSI EN 300 421 V1.1.2, Digital Video Broadcasting (DVB); Implementation guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 1: DVB-S2, 2015.

[8] DVB Document ETSI EN 300 421 V1.1.2, Digital Video Broadcasting (DVB); Implementation guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 2: S2 Extensions (DVB-S2X), 2015.

[9] DVB Document ETSI EN 301 790, Digital Video Broadcasting (DVB);Interaction channel for satellite distribution systems, 2003.

[10] DVB Document ETSI A155-1, Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (RCS2); Part2, 2011.

[11] Satellite Earth Stations and Systems; Air Interface for S-band Mobile Interactive Multimedia (S-MIM); V 1.1.1, 2011.

[12] CCSDS. TM Synchronization and channel coding: Recommended Standard, CCSDS 131.0-B-2; Blue Book. 2011.

[13] R. De Gaudenzi and O. del Rio Herrero. Advances in Random Access Protocols for Satellite Networks. In *International Workshop on Satellite and Space Communications, 2009.*, pages 331–336, Sept 2009.

[14] N. Abramson. The aloha system: Another alternative for computer communications. In *Proceedings of the November 17-19, 1970, Fall Joint Computer Conference*, AFIPS '70 (Fall), pages 281–285, New York, NY, USA, 1970. ACM.

[15] L. G. Roberts. ALOHA packet system with and without slots and capture. *ACM, SIGCOMM Computer Communication Review*, 5(February):28–42, 1975.

[16] E. Casini, R. D. Gaudenzi, and O. D. R. Herrero. Contention resolution diversity slotted ALOHA (CRDSA) : An enhanced random access scheme for satellite access packet networks. *IEEE Transactions on Wireless Communications*, 2007.

[17] L. Choudhury Gagan and S. Rappaport Stephen. Diversity aloha–a random access scheme for satellite communications. *Communications, IEEE Transactions on*, 31(3):450–457, Mar 1983.

[18] G. Liva. Graph-based analysis and optimization of contention resolution diversity slotted aloha. *Communications, IEEE Transactions on*, 59(2):477–487, February 2011.

[19] E. Paolini, G. Liva and M. Chiani. High Throughput Random Access via Codes on Graphs: Coded Slotted ALOHA. In *IEEE International Conference on Communications (ICC) 2011*, pages 1–6, June 2011.

[20] E. Paolini, G. Liva and M. Chiani. Graph-Based Random Access for the Collision Channel without Feedback: Capacity Bound. In *IEEE Global Telecommunications Conference (GLOBECOM 2011)*, pages 1–5, Dec 2011.

[21] H.-C Bui, J. Lacan, and M.-L. Boucheret. An enhanced multiple random access scheme for satellite communications. In *Wireless Telecommunications Symposium (WTS), 2012*, pages 1–6, April 2012.

[22] M. Feder and E. Weinstein. Parameter estimation of superimposed signals using the EM algorithm. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(4):477–489, Apr 1988.

[23] J. M. Palmer. *Real-Time Carrier Frequency Estimation using Disjoint Pilot Symbol Blocks.* PhD thesis, Brigham Young University, 2009.

[24] K. H. Chang and C. N. Georghiades. Iterative Joint Sequence and Channel Estimation for Fast Time-varying Intersymbol Interference Channels. In *IEEE International Conference on Communications, 1995. ICC '95 Seattle, 'Gateway to Globalization'*, volume 1, pages 357–361, Jun 1995.

[25] Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation, 3GPP TS 36.211 v8.9.0, 2009.

[26] F. Collard and R. De Gaudenzi. On the Optimum Packet Power Distribution for Spread Aloha Packet Detectors With Iterative Successive Interference Cancelation. *IEEE Transactions on Wireless Communications*, 13(12):6783–6794, Dec 2014.

[27] R. De Gaudenzi. A high-performance mac protocol for consumer broadband satellite systems. *IET Conference Proceedings*, pages 512–512(1).

[28] Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Guidelines for Implementation and Use of LLS: EN 301 545-2, 2012.

[29] C. Terman H. Balakrishnan and G. Verghese. Sharing a Channel: Media Access (MAC) Protocols. MIT 6.02 Lecture Notes, November 2012.

[30] J. Allen and K. Solomon. A time division multiple access synchronization technique, October 22 1974. US Patent 3,843,843.

[31] J. Knuutila A. Leppisaari K. Malmivirta J. Oksala J. Hämäläinen, P. Järvinen and A. Salminen. Time division multiple access radio systems, June 17 1998. US Patent 6967943 B1.

[32] I.S. Hwang, S.Y. Yoon, S.H. Sung, J. Cho, H. Huh, and K.H. Roh. Apparatus and method for assigning subchannel in a mobile communication system using orthogonal frequency division multiple access scheme, February 3 2005. US Patent App. 10/901,738.

[33] F. Ozluturk, A. Jacques, G. Lomp, and J. Kowalski. Code division multiple access communication system, September 17 1998. WO Patent App. PCT/US1998/004,716.

[34] Digital Video Broadcasting (DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2);Part 1: Overview and System Level specification, 2013.

[35] P. M. Feldman. *An Overview and Comparison of Demand Assignment Multiple Access (DAMA) Concepts for Satellite Communications Networks*. RAND, Santa Monica, CA: RAND Corporation, 1996.

[36] T. Le-Ngoc and I. M. Jahangir. Performance Analysis of CFDAMA-PB Protocol for Packet Satellite Communications. *IEEE Transactions on Communications*, 46(9):1206–1214, Sep 1998.

[37] A. Munari and M. Heindlmaier and G. Liva and M. Berioli. The Throughput of Slotted Aloha with Diversity. In *51st Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 698–706, Oct 2013.

[38] O. del Rio Herrero and R. De Gaudenzi. Generalized Analytical Framework for the Performance Assessment of Slotted Random Access Protocols. *IEEE Transactions on Wireless Communications*, 13(2):809–821, February 2014.

[39] K. Whitehouse, A. Woo, F. Jiang, J. Polastre and D. Culler. Exploiting the Capture Effect for Collision Detection and Recovery. In 45-52, editor, *The Second IEEE Workshop on Embedded Networked Sensors, 2005. EmNetS-II.*, May 2005.

[40] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke. Design of capacity-approaching irregular low-density parity-check codes. *IEEE Trans. Inf. Theor.*, 47(2):619–637, September 2006.

[41] R. G. Gallager. Low-Density Parity-Check Codes. Cambridge, MA: M.I.T. Press, 1963.

[42] G. Liva. Method for contention resolution in time-hopping or frequency-hopping, 2011. US Patent App. 12/903,246.

[43] M. Chiani, G. Liva, and E. Paolini. The marriage between random access and codes on graphs: Coded slotted aloha. In *Satellite Telecommunications (ESTEL), 2012 IEEE First AESS European Conference on*, pages 1–6, Oct 2012.

[44] H. C. Bui and J. Lacan and M. L. Boucheret. Multi-Slot Coded ALOHA with Irregular Degree Distribution. In *2012 IEEE First AESS European Conference on Satellite Telecommunications (ESTEL)*, pages 1–6, Oct 2012.

[45] O. del Rio Herrero and R. De Gaudenzi. A high efficiency scheme for quasi-real-time satellite mobile messaging systems. In *2008 10th International Workshop on Signal Processing for Space Communications*, pages 1–9, Oct 2008.

[46] O. Del Rio Herrero and R. De Gaudenzi. High Efficiency Satellite Multiple Access Scheme for Machine-to-Machine Communications. *IEEE Transactions on Aerospace and Electronic Systems*, 48(4):2961–2989, October 2012.

[47] N. Abramson. Spread aloha CDMA data communications, July 16 1996. US Patent 5,537,397.

[48] F. Clazzer and C. Kissling. Enhanced Contention Resolution Aloha - ECRA. In *Proceedings of 2013 9th International ITG Conference on Systems, Communication and Coding (SCC)*, pages 1–6, Jan 2013.

[49] C. Kissling. Performance Enhancements for Asynchronous Random Access Protocols over Satellite. In *2011 IEEE International Conference on Communications (ICC)*, pages 1–6, June 2011.

[50] R. De Gaudenzi and O. del Río Herrero and G. Acar and E. Garrido Barrabés. Asynchronous Contention Resolution Diversity ALOHA: Making CRDSA Truly Asynchronous. *IEEE Transactions on Wireless Communications*, 13(11):6193–6206, Nov 2014.

[51] E. Casini, O. Herrero, R. De Gaudenzi, D. Delaruelle, and J.P.G.J. Choffray. Method of packet mode digital communication over a transmission channel shared by a plurality of users, August 3 2006. US Patent App. 11/342,980.

[52] G. Cocco, N. Alagha, C. Ibars, and S. Cioni. Network-coded diversity protocol for collision recovery in slotted aloha networks. *International Journal of Satellite Communications and Networking*, 32(3):225–241, 2014.

[53] Demrsrsa, AP and Lamb, NM and Rubin, DB. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the royal statistical society. Series B (methodological)*, 39(1):1–38, 1977.

[54] S. Liew S. Zhang and P. Lam. Hot Topic: Physical-Layer Network Coding. *Proc. ACM Mobicom'06*, pages 358–365, 2006.

[55] S. Dicintio. Comparing Approaches to Initializing the Expectation-Maximization Algorithm. Master of Science Thesis, 2012.

[56] M. Luise and R. Reggiannini. Carrier frequency recovery in all-digital modems for burst-mode transmissions. *IEEE Transactions on Communications*, 43(2/3/4):1169–1178, Feb 1995.

[57] J. J. van de Beek, M. Sandell, and P. O. Borjesson. ML Estimation of Time and Frequency Offset in OFDM Systems. *IEEE Transactions on Signal Processing*, 45(7):1800–1805, 1997.

[58] J. J. van de Beek and M. Sandell and M. Isaksson and P. Ola Borjesson. Low-complex Frame Synchronization in OFDM Systems. In *Fourth IEEE International Conference on Universal Personal Communications. 1995. Record.*, pages 982–986, Nov 1995.

[59] T. M. Schmidl and D. C. Cox. Robust Frequency and Timing Synchronization for OFDM. *IEEE Transactions on Communications*, 45(12):1613–1621, Dec 1997.

[60] M. Oerder and H. Meyr. Digital Filter and Square Timing Recovery. *IEEE Transactions on Communications*, 36(5):605–612, May 1988.

[61] M. Meyers and L. Franks. Joint Carrier Phase and Symbol Timing Recovery for PAM Systems. *IEEE Transactions on Communications*, 28(8):1121–1129, Aug 1980.

[62] Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[63] Van Trees, Harry L and Bell, Kristine L. *Detection Estimation and Modulation Theory, pt. I*. Wiley, 2013.

[64] Alexander Ly, Josine Verhagen, Raoul Grasman, and Eric-Jan Wagenmakers. A Tutorial on Fisher Information. Online.

[65] K. Zidane and J. Lacan and M. L. Boucheret and C. Poulliat. Improved Channel Estimation for Interference Cancellation in Random Access Methods for Satellite Communications. In *2014 7th Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, pages 73–77, Sept 2014.

[66] K. Zidane and J. Lacan and M. L. Boucheret and C. Poulliat and M. Gineste and D. Roques and C. Bes and A. Deramecourt. Effect of Residual Channel Estimation Errors in Random Access Methods for Satellite Communications. In *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, pages 1–5, May 2015.

[67] B. Tacca. *Implementation of Iterative Multiuser Joint Detection Techniques in a DVB-RCS Satellite Scenario*. PhD thesis, University of Rome, 2009.

[68] B. Sklar. *Digital communications*, volume 2. Prentice-Hall NJ, 2001.

[69] J. K. Cavers. An analysis of Pilot Symbol Assisted Modulation for Rayleigh Fading Channels. *IEEE Transactions on Vehicular Technology*, 40(4):686–693, Nov 1991.

[70] M. Sandell and C. Luschi and P. Strauch and R. Yan. Iterative Channel Estimation using Soft Decision Feedback. In *Global Telecommunications Conference, 1998. GlobeCom 1998. The Bridge to Global Integration. IEEE*, volume 6, pages 3728–3733, 1998.

[71] M. C. Valenti and B.D. Woerner. Iterative Channel Estimation and Decoding of Pilot Symbol Assisted Turbo Codes over Flat-Fading Channels. *IEEE Journal on Selected Areas in Communications*, 19(9):1697–1705, Sep 2001.

[72] T. Wang and S. C. Liew. Joint Estimation and Decoding in Physical-Layer Network Coding Systems. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5104–5108, May 2013.

[73] Ali Arshad Nasir and Hani Mehrpouyan and Salman Durrani and Steven D. Blostein and Rodney A. Kennedy. Timing and Carrier Synchronization with Channel Estimation in AF Two-Way Relaying Networks. *CoRR*, abs/1309.6690, 2013.

[74] H. C. Bui, K. Zidane, J. Lacan and M. L. Boucheret. A Multi-Replica Decoding Technique for Contention Resolution Diversity Slotted Aloha. In *Vehicular Technology Conference (VTC Fall), 2015 IEEE 82nd*, pages 1–5, Sept 2015.

[75] J. Lacan, H. C. Bui and M-L. Boucheret. Reception de donnees par paquets a travers un canal de transmission a acces multiple, 2013. Brevet d'invention. BFF130051GDE.

[76] M. Oerder and H. Meyr. Digital filter and square timing recovery. *Communications, IEEE Transactions on*, 36(5):605–612, May 1988.

[77] A. Viterbi. Nonlinear estimation of psk-modulated carrier phase with application to burst digital transmission. *Information Theory, IEEE Transactions on*, 29(4):543–551, Jul 1983.

[78] K. Zidane, J. Lacan, M. Gineste, C. Bes, A. Deramecourt and M. Dervin. Estimation of Timing Offsets and Phase Shifts Between Packet Replicas in MARSALA Random Access. In *2016 IEEE Global Communications Conference*, Dec 2016.

[79] M. Simon, J. Omura, R. Scholtz and B. Levitt. *Spread spectrum communications handbook*, volume 2. Citeseer.

[80] A.J. Viterbi. *CDMA: principles of spread spectrum communication.* Addison Wesley Longman Publishing Co., Inc., 1995.

[81] M. Chiani. Noncoherent Frame Synchronization. *IEEE Transactions on Communications*, 58(5):1536–1545, May 2010.

[82] S. Jagannathan, H. Aghajan, and A. Goldsmith. The effect of time synchronization errors on the performance of cooperative miso systems. In *Global Telecommunications Conference Workshops, 2004. GlobeCom Workshops 2004. IEEE*, pages 102–107, Nov 2004.

[83] K. Zidane, J. Lacan, M. Gineste, C. Bes and C. Bui. Enhancement of MARSALA Random Access with Coding Schemes, Power Distributions and Maximum Ratio Combining. In *2016 8th Advanced Satellite Multimedia Systems Conference and the 14th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, Sept 2016.

[84] T.K.Y. Lo. Maximum ratio transmission. In *Communications, 1999. ICC '99. 1999 IEEE International Conference on*, volume 2, pages 1310–1314 vol.2, 1999.

[85] S.C. Jasper, M.A. Birchler, and N.C. Oros. Diversity reception communication system with maximum ratio combining method, September 1996. US Patent 5,553,102.

[86] A. J. Viterbi. Very low rate convolution codes for maximum theoretical performance of spread-spectrum multiple-access channels. *IEEE Journal on Selected Areas in Communications*, 8(4):641–649, May 1990.

[87] M. Valenti. Coded Modulation Library. http://www.iterativesolutions.com/Matlab.htm. [Online].