UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA E ELETRÔNICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

Efficient Methods for Massive Random Access

Gustavo Kasper Facenda

Orientador: Prof. Danilo Silva, Ph.D.

Florianópolis, June 12, 2019.

GUSTAVO KASPER FACENDA

EFFICIENT METHODS FOR MASSIVE RANDOM ACCESS

Dissertação submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina para a obtenção do Grau de Mestre em Engenharia Elétrica
Orientador: Prof. Danilo Silva, Ph.D.

FLORIANÓPOLIS
2019

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Facenda, Gustavo
Efficient Methods for Massive Random Access / Gustavo Facenda ; orientador, Danilo Silva, 2019.
xxiv + 62 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico, Programa de Pós Graduação em Engenharia Elétrica, Florianópolis, 2019.

Inclui referências.

1. Engenharia Elétrica. 2. Random Access. 3. Massive Multiple Access. 4. Scheduling. 5. Finite Blocklength. I. Silva, Danilo. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia Elétrica. III. Título.

Gustavo Kasper Facenda

EFFICIENT METHODS FOR MASSIVE RANDOM ACCESS

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia Elétrica, área de concentração Comunicações e Processamento de Sinais, e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Santa Catarina.

Florianópolis, 23 de Maio de 2019.

Prof. Bartolomeu Ferreira Uchôa Filho, Ph.D.
Coordenador do Programa de Pós-Graduação em Engenharia Elétrica

Prof. Danilo Silva, Ph.D.
Orientador
Universidade Federal de Santa Catarina

Banca examinadora

Prof. Bartolomeu Ferreira Uchôa Filho, Ph.D.
Universidade Federal De Santa Catarina

Prof. Richard Demo Souza, Dr.
Universidade Federal de Santa Catarina

Prof. Rodrigo Caiado de Lamare, Ph.D.
Pontifícia Universidade Católica do Rio de Janeiro

Prof. Bartolomeu Ferreira Uchôa Filho, Ph.D.
Coordenador do Programa de
Pós-Graduação
em Engenharia Elétrica - UFSC

Dedico esta dissertação a todos que me ajudaram.

Agradecimentos

Gostaria de agradecer ao meu orientador Danilo Silva pelas muitas e longas, porém sempre produtivas, reuniões e discussões sobre o trabalho, assim como pelo ótimo trabalho de orientação de forma geral.

Aos meus pais, pelo apoio constante que me permitiu focar na minha educação e realizar este trabalho.

E aos meus amigos, que não serão nomeados para eu não esquecer ninguém, por estarem aqui sempre que precisei de algum tipo de ajuda, seja acadêmica ou pessoal.

What kind of mad scientist worries about not getting enough vegetables? (Rintaro Okabe from Steins;Gate)

RESUMO

Este trabalho investiga o canal de acesso aleatório com um número massivo de usuários sob ruído gaussiano, com foco em pacotes pequenos. Para esse problema, esquemas coordenados são considerados ineficientes devido à necessidade de grande quantidade de *feedbacks* e ao uso de técnicas de requisição de acesso ineficientes. O trabalho revisa a teoria necessária e o método sem coordenação apresentado por Ordentlich e Polyanskiy. Em sequência, apresenta-se um modelo que permite a comparação justa entre métodos com e sem coordenação, considerando usos de canal e energia por bit. Após isso, um método coordenado é proposto. O esquema proposto usa o método apresentado por Ordentlich e Polyanskiy (método OP) para transmitir pequenos índices de coordenação, realizando a requisição de acesso, permitindo uma requisição eficiente e pouco *feedback*. Também é apresentado um aprimoramento ao método OP para decodificar índices iguais, melhorando significativamente a eficiência do método para mensagens muito pequenas. Um método de otimização é apresentado e, usando os parâmetros projetados com este método, os resultados do método proposto são comparados ao estado-da-arte. Resultados em simulação mostram que, se for permitida uma quantidade pequena de *feedback*, o método proposto requer uma energia por *bit* menor que a dos métodos sem coordenação existentes. Neste trabalho, também é investigada a alcançabilidade do método proposto e mostra-se que, no regime de interesse, as probabilidades de erro são alcançáveis com pouca ou nenhuma perda de energia. Finalmente, o desempenho do método proposto é verificado utilizando códigos existentes e observa-se que, apesar dos resultados práticos serem comparáveis a outros métodos, há uma melhoria significativa a ser feita nos códigos para este problema específico.

Palavras-chave: Canal de múltiplo acesso, Acesso aleatório, Coordenação.

RESUMO ESTENDIDO

Introdução

Recentemente, devido ao interesse em comunicações de máquinas, em especial à Internet das Coisas, e ao crescimento do número de aparelhos conectados, uma área de pesquisa voltada a um canal com um número massivo de usuários que transmitem pequenos pacotes infrequentemente surgiu.

Devido à natureza do problema, métodos de comunicação coordenada são considerados ineficientes, devido a técnicas de requisição de acesso ineficientes e necessidade de grande quantidade de *feedback*, que levam a latência e consumo de energia elevados.

Polyanskiy propôs um modelo sem coordenação para este problema e, com este modelo, obteve limites teóricos para comunicação. Ao comparar as técnicas utilizadas atualmente ao limite teórico, observou-se que o desempenho destas técnicas é pobre. Desde então, vários trabalhos buscam técnicas mais eficientes de transmissão sem coordenação, buscando se aproximar do limite teórico demonstrado por Polyanskiy.

O primeiro destes trabalhos foi feito por Ordentlich e Polyanskiy, que apresentam uma variação do *slotted* ALOHA chamada T-*slotted* ALOHA, em que uma "colisão" é definida por mais de T usuários transmitindo no mesmo *timeslot*. Também é apresentada uma forma prática de implementar o método. Recentemente, o resultado mais próximo foi apresentado por Amalladinne et al. analisando o problema por uma perspectiva de *compressive sensing*.

O objetivo deste trabalho é investigar o uso de técnicas com coordenação para este problema. Para isto, um modelo que permite comparar técnicas coordenadas de técnicas descoordenadas de forma justa é proposto, considerando número de usos de canal e energia consumida. Em sequência, propõe-se um método coordenado, que utiliza o método sem coordenação de Ordentlich e Polyanskiy para coordenar os usuários. O desempenho do método proposto é comparado ao desempenho dos métodos sem coordenação. Observa-se que, se for permitida uma pequena quantidade de *feedback*, é possível obter uma redução significativa na energia consumida, comparado ao esquema proposto por

Amalladinne.

Adicionalmente, uma melhoria para o método de Ordentlich e Polyanskiy é apresentada, que permite decodificar mensagens replicadas num mesmo *timeslot*. Finalmente, também é analisado o desempenho do método proposto utilizando códigos práticos atuais, como LDPC.

Contribuições

Modelo Proposto

O modelo proposto consiste de três fases. A primeira fase—realizada sem coordenação—é usada para sinalizar atividade e utiliza N_1 usos de canal. Usando a informação decodificada na primeira fase, a estação base gera um sinal de *feedback* e transmite num canal de *broadcast*. Esta segunda fase utiliza N_f usos de canal.

Em sequência, os usuários, utilizando a informação obtida através do *feedback*, transmitem um sinal contendo os dados. O número de usos de canal nesta etapa é N_2.

Denota-se por $N = N_1 + N_2 + N_f$ o número total de usos de canal para realizar a transmissão. O modelo sem coordenação é um caso particular onde $N_2 = N_f = 0$ e a mensagem é transmitida na primeira fase.

Para comparação, a energia total consumida, normalizada pelo número de usuários ativos, é utilizada como métrica de comparação.

Método Proposto

O método proposto consiste de, inicialmente, em vez de transmitir uma mensagem de k bits, utilizar o método Ordentlich e Polyanskiy (OP) para transmitir pequenos índices de coordenação, em que o tamanho destes índices é um parâmetro de projeto. O método OP consiste de dividir os usos de canal em V sub-blocos e cada usuário escolhe, aleatoriamente, um sub-bloco para transmitir. O índice, em conjunto com o sub-bloco escolhido pelo usuário, permite identificar os usuários ativos.

Em cada sub-bloco, a estação base estima uma lista de índices. Com esta informação, a estação base gera uma sequência de *feedback* contendo quantos índices foram decodificados no sub-bloco e quais são os índices. Então, esta sequência é codificada utilizando um código para canal AWGN e transmitida para todos os usuários. A estação base realiza o alocamento de recursos baseado nesta sequência. A forma de alocação é pré-determinada e conhecida pelos usuários.

Finalmente, os usuários que decodificaram com sucesso seu próprio índice e sub-bloco na sequência de *feedback* transmitem utilizando o recurso alocado pela estação base.

No trabalho, os possíveis eventos de erro do método proposto são apresentados. A análise é feita considerando um erro por usuário.

Resultados

Comparação com o estado-da-arte

O método proposto é comparado ao estado-da-arte e outros métodos práticos recentemente apresentados. Comparado aos métodos existentes, o método proposto apresenta uma melhoria significativa no uso de energia por usuário.

Alcançabilidade da taxa para o canal bi-AWGN mod-2

Considerando o regime de interesse obtido nos resultados anteriores, foram projetados códigos lineares simples para cada valor de K_a (número de usuários ativos) e a probabilidade de erro para o canal bi-AWGN mod-2 utilizando uma decodificação aproximada de máxima verossimilhança (ML) foi simulada. As taxas de erro projetadas são alcançáveis para todos os valores de K_a exceto 300, em que um aumento de $0,04$ dB (de $6,89$ dB para $6,93$ dB) na energia consumida é necessário para obter o erro desejado.

Desempenho com códigos LDPC

O valor $K_a = 100$ foi utilizado como um estudo de caso. Foram utilizados códigos LDPC regulares, com grau 3 nos nós de variáveis, projetados utilizando PEG (progressive edge-growth). Verificou-se experimentalmente a folga necessária e então os parâmetros foram otimizados novamente considerando a folga. Para obter a probabilidade de erro desejada, foi necessário um aumento de 1.3 dB (de 2.8 para 4.1 dB) na energia consumida.

Conclusões

O interesse no canal de acesso aleatório com número massivo de usuários cresceu consideravelmente com o aumento do número de dispositivos conectados. Limites teóricos foram demonstrados e, recentemente, métodos práticos foram apresentados. No entanto, estes métodos exigem códigos e algoritmos complexos.

Neste trabalho, um novo esquema para este problema foi apresentado. O esquema permite coordenação de usuários, melhorando o consumo de energia ao custo de transmitir uma curta sequência de *feedback*.

O método proposto tem a vantagem de ser possível utilizar códigos práticos atuais e ainda obter resultados comparáveis aos outros métodos. Além disso, resultados utilizando teoria de informação mostram que o desempenho pode ser ainda melhorado utilizando códigos especificamente projetados para o problema de pequenos pacotes, que é também um tema importante de pesquisa atualmente.

Palavras-chave: Canal de múltiplo acesso, Acesso aleatório, Coordenação.

ABSTRACT

This work investigates the massive random access Gaussian channel with a focus on small payloads. For this problem, grant-based schemes have been regarded as inefficient due to the necessity of large feedbacks and the use of inefficient scheduling request methods. The necessary theory and the grantless method presented by Ordentlich and Polyanskiy is briefly revised. Then, a model that allows fair comparison between grantless and grant-based methods is presented, taking into account energy spent and number of channel uses. In the sequence, we propose a novel grant-based scheme. The scheme uses Ordentlich and Polyanskiy's method to transmit small coordination indices in order to perform the scheduling request, which allows both the request from the users to be efficient and the feedback to be small. The proposed method also contains an improvement to the Ordentlich and Polyanskiy's scheme, allowing it to handle collisions of the same message, significantly improving the method for very small messages. An optimization framework is presented and, using the parameters designed with this framework, the performance of the proposed method is compared to the state-of-art. Simulation results show that, if a short feedback is allowed, the proposed method requires lower energy per bit than existing practical grantless methods. The achievability of the method is also investigated and it is shown that, in the regime of interest, the probabilities of error can be achieved with small or no energy losses. Finally, the performance using off-the-shelf codes is investigated and it can be seen that, while the proposed method results are comparable to that of other methods, there is a significant improvement to be made in code design for this specific problem.

Keywords: Multiple-access channel, Random access, Grant-based, Scheduling.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

In recent years, interest in Machine-type Communication (MTC) has increased, mostly due to the growing trend of the Internet of Things. Due to the growing number of connected devices, within MTC, a special case of interest is that when the number of devices is massively large and they transmit infrequently a small amount of information, called mMTC (massive MTC). As a particular application, one of the goals of 5G is to allow mMTC with little or no human intervention [1].

While the problem is similar to the multiple access channel problem, which is well studied in information theory [2], the burst nature of this new problem leads to some important differences between them. The small blocklengths involved make asymptotic analysis less meaningful, only a fraction of the users is active, thus the access is random, but the total number of users is massively large.

Parts of this problem are already studied in different areas of research, mainly divided between information theory, network theory and coding theory (see [3] and references therein). However, since the approaches and models are different in each area, results are not comparable between them.

Aiming to unify these areas and derive fundamental results for the problem, Polyanskiy introduces a novel model for the massive random access channel in [3]. In his work, as well as in other works in the area [4, 5], it is argued that grant-based schemes which coordinate the users — so they can use coordinated data transmission methods such as TDMA, FDMA, orthogonal CDMA, CFMA [6] and rate-splitting [7] — are ineffective due to high costs in both user energy and latency as a consequence of the massive amount of users. For that reason, Polyanskiy focuses the model and analysis on grantless schemes.

In Polyanskiy's model, only a fraction K_a of the total K_{tot} users are active at any given time and the receiver decodes only a list of K_a messages regardless of the identity of the active users. This is justified by the presence of some kind of identity, such as an IP address, in the header of the payload. This contrasts to the usual information theoretic approach of the T-out-of-N channel in [8, 9, 10, 11] (see [3] and references therein), which requires identifying the transmitting user as well as their messages. Polyanskiy's model restricts the users to use the same (possibly randomized) codebook and the error probability is defined per user. This approach allows us to let $K_{\text{tot}} \to \infty$ and models the random access problem well.

Using this model, Polyanskiy presents a coding theorem, which provides an achievability bound for the random access channel under finite-blocklength. Then, Polyanskiy compares the results of the main proposed solutions for this problem, which are "treat interference as noise" (TIN) and slotted-ALOHA, to the achievable rates and shows that these schemes are still far from the bound, in particular for a large number of active users.

Since then, research has been seeking practical methods that approach the theoretical limits. In the following section, we briefly review the recent research in the area.

1.1 Literature Review

In [12], Ordentlich and Polyanskiy present a scheme that outperforms previously practical methods for a sufficiently large ($K_a > 150$) number of active users. In this scheme, the total number of channel uses is split in V timeslots, and each user chooses randomly one timeslot to transmit. The number of users in each timeslot is limited to T, and, if more than T users choose to transmit in the same timeslot, an error occurs in that timeslot. The channel coding is done using a linear code for a modulo-2^ℓ AWGN channel and the separation of the users' messages is done using a code for the binary adder channel (BAC).

In [4], Vem et al. note two major drawbacks in the method presented in [12]. First, if more than T users transmit in a timeslot, all the messages are simply lost. Second, the reduction of the AWGN channel to a modulo-2^ℓ AWGN channel implies rate loss. For that reason, the method in [4] proposes a method that allows serial interference cancellation, allowing to recover messages from timeslots that had more than T users using information from other timeslots, as well as make full use of the AWGN channel. These improvements significantly decrease the required energy.

Recently, in [13], Amalladinne et al. approach the problem through a compressive sensing (CS) point of view. In their work, a feasible compressive sensing algorithm is presented, tailored for the massive random access channel. As they note, naive application of CS in the problem would require sensing matrices with 2^k columns, where $k \approx 100$ is the number of bits in the message. For that reason, they split the message in smaller sub-blocks of significantly smaller lengths, which allow smaller sensing matrices. However, this approach introduces a new challenge, which is the pairing of the sub-blocks belonging to the same message. To overcome this challenge, parity bits are introduced to the message, allowing the method to recover the messages. To the best of our knowledge, the results in [13] are the closest to the theoretical achievable rates.

1.2 Contributions

In this work, we take a different approach to improve the method in [12]. Motivated by the simplicity and good performance of coordinated orthogonal methods, we propose a practical grant-based scheme which does not assume any a priori coordination. We show that, contrary to what has been commonly assumed, the benefits of coordination can outweigh its costs, even for a massive number of users. Specifically, our scheme is able to transmit using the same number of channel uses (including the necessary feedback), while spending less energy (including base station energy) than that of [13]. Our approach for the scheduling request is to identify the transmitting users using two pieces of information: small indices randomly chosen by the users, transmitted efficiently using the scheme in [12]; and the timeslots in which each user has chosen to transmit their index. This approach significantly reduces the overhead of the scheduling request and feedback, therefore making the coordination viable.

Additionally, we propose an Index Collision Resolution method for the OP scheme, which reduces the error probability in the scenario where two or more users transmit the same message. While this improvement may not be significant when using the OP scheme to transmit *messages* (of about 100 bits), it becomes significant in our method, which uses the OP scheme to transmit *indices* of lengths as short as 5 bits.

The main contributions of this work are summarized as follows:

- We present a model that includes both Polyanskiy's model and certain grant-based schemes;
- We improve the OP scheme in order to handle the scenario where two or more users transmit the same message;
- We present a grant-based scheme that improves the state-of-the-art results for the massive random access Gaussian channel;

The contributions of this work have been submitted in a shortened conference paper to the XXXVII Brazilian Symposium on Telecommunications and Signal Processing and as a full paper to IEEE Transactions on Wireless Communications.

1.3 Organization

This thesis is organized as follows. In Chapter 2, we review fundamental concepts that are helpful in our method, as well as a more detailed revision of the scheme presented in [12]. In Chapter 3, we present our method and derive bounds for the probabilities of error. In Chapter 4, we present our optimization method and our results, comparing them to the state-of-art. We also study the achievability of our results and the performance using off-the-shelf LDPC codes. Finally, we conclude this thesis in Chapter 5.

Chapter 2

Preliminaries

In this chapter, we present known finite blocklength information theory results and present the grantless model for the random access problem. Finally, we review the grantless method presented in [12].

2.1 Finite Blocklength Information Theory

Finite blocklength information theory studies the fundamental limits of communication in a limited number of channel uses, opposed to capacity results which are obtained with the number of channel uses going to infinity.

In this section, we present the finite blocklength results for the two channels of interest in our work—the Additive White Gaussian Noise (AWGN) channel and the binary AWGN (bi-AWGN) mod 2 channel.

Before we study the specific channels of interest, let us introduce the concept of information density. Given a channel described by an output y with a probability density function (p.d.f.) $p(y)$, input x with p.d.f. $p(x)$ and joint distribution $p(x,y)$, the information density is given by

$$i(x;y) = \log_2\left(\frac{p(x,y)}{p(x)p(y)}\right). \tag{2.1}$$

Note that, by taking the expected value of $i(x;y)$, we obtain the known mutual information expression

$$I(X;Y) = \mathbf{E}[i(X;Y)] = \int_{y\in\mathcal{Y}}\int_{x\in\mathcal{X}} p(x,y)\log_2\left(\frac{p(x,y)}{p(x)p(y)}\right)dxdy. \tag{2.2}$$

However, in finite blocklength theory, another important measurement is the variance of the information density. In particular, for a p.d.f. $p^*(x)$ that achieves the capacity $C = \max_{p^*(x)} I(X;Y)$ for some channel, the variance of the information density computed with that p.d.f. is called channel dispersion

$$\mathsf{V} = \mathbf{E}[i(X;Y)^2] - I(X;Y)^2. \tag{2.3}$$

In particular, Polyanskiy et al. [14] show that a useful approximation for the achievable rates is given by

$$R_{\text{normal}} \approx C(P) - \sqrt{\frac{\mathsf{V}(P)}{n_c}}Q^{-1}(\epsilon) \tag{2.4}$$

where $C(P)$ is the capacity of the channel for a power P, $\mathsf{V}(P)$ is the channel dispersion, n_c is the code length and Q is the Q-function and ϵ is the desired probability of error.

We say that a rate R is achievable with probability of error ϵ if a code exists such that its rate is equal to or larger than R and its probability of error is equal to or smaller than ϵ.

In the following subsections, we study particular results for the channels of interest.

2.1.1 Additive White Gaussian Noise Channel

Consider the AWGN channel with input $\mathbf{x} \in \mathbb{R}^{n_c}$ and output $\mathbf{y} \in \mathbb{R}^{n_c}$ described by

$$\mathbf{y} = \mathbf{x} + \mathbf{z} \tag{2.5}$$

where $\mathbf{z} \sim \mathcal{N}(0, \mathbf{I})$ and $\mathbf{x}$ is subjected to power constraint $\|\mathbf{x}\|^2 \leq n_c P$, where n_c is the length of $\mathbf{x}$.

For this channel, the tightest bound on the maximum achievable rate is provided by Shannon in [15]. However, Shannon studies asymptotic behaviors of the expressions, and, for small values of n_c, the expressions provided by Shannon are not easily computable. While works such as [16] study Shannon's results numerically, other works, such as [17], provide looser bounds which are easier to compute. The normal approximation for the AWGN channel [14] is given by

$$R_{\text{AWGN}} \approx C_{\text{AWGN}}(P) - \sqrt{\frac{\mathsf{V}_{\text{AWGN}}(P)}{n_c}} Q^{-1}(\epsilon). \tag{2.6}$$

where $C_{\text{AWGN}}(P) = \frac{1}{2}\log_2(1+P)$, $\mathsf{V}_{\text{AWGN}}(P) = \frac{P}{2}\frac{P+2}{(P+1)^2}\log_2^2 e$. In particular, in [14] (Theorem 54), it is shown that, for the AWGN channel, the rate is asymptotically given by

$$R_{\text{AWGN}} = C_{\text{AWGN}}(P) - \sqrt{\frac{\mathsf{V}_{\text{AWGN}}(P)}{n_c}} Q^{-1}(\epsilon) + \frac{O(\log(n_c))}{n_c} \tag{2.7}$$

and that (2.6) is a pessimistic approximation for the achievability, i.e., the rate is achievable.

2.1.2 Binary Additive White Gaussian Noise mod 2 Channel

Consider the bi-AWGN mod 2 channel with input $\mathbf{c} \in \{0,1\}^{n_c}$ and output $\mathbf{y} \in [0,2)^{n_c}$, described by

$$\mathbf{y} = (\mathbf{c} + \mathbf{z}) \bmod 2 \tag{2.8}$$

where $\mathbf{z} \sim \mathcal{N}(0, \frac{1}{4P}\mathbf{I})$.

To the best of our knowledge, no achievability results have been derived for the binary AWGN mod 2 channel, including for the normal approximation (2.4), i.e., the normal approximation not necessarily is achievable. However, it is useful for a closed-form analysis, thus, we use

$$R_{\text{mod2}} \approx C(P) - \sqrt{\frac{\mathsf{V}(P)}{n_c}} Q^{-1}(\epsilon) \tag{2.9}$$

where $C(P)$ and $\mathsf{V}(P)$ are given in Appendix B, and in Chapter 4, we analyze the achievability of this approximation.

2.2 Random Access Channel

In this section, we introduce the model proposed by Polyanskiy in [3].

2.2.1 Grantless Model

Let $K_{\text{tot}} \to \infty$ be the total number of users in the network. At the beginning of a session, K_a of these users are active and wish to transmit k bits of information using the same channel. Let $w_i \in \{1, 2, \ldots, 2^k\}$ be the message of the i-th user, where $i \in \{1, \ldots, K_a\}$, for simplicity. We assume w_i is uniform and independent across the users.

The channel is a Multiple Access Channel described as

$$\mathbf{y} = \sum_{i=1}^{K_a} \mathbf{x}_i + \mathbf{z} \tag{2.10}$$

where $\mathbf{y} \in \mathbb{R}^{N_1}$ is the received signal, $\mathbf{x}_i \in \mathbb{R}^{N_1}$ is the transmitted signal by the i-th user and $\mathbf{z} \sim \mathcal{N}\left(0, \frac{N_0}{2}\mathbf{I}\right)$, with $\frac{N_0}{2} = 1$. Each transmitted signal $\mathbf{x}_i$ is power-constrained in expectation, i.e., $\mathbf{E}[\|\mathbf{x}_i\|^2] \le N_1 P_1$.

Note that the model assumes the same gain (at the receiver) for every user. This can be achieved assuming a static channel that satisfies reciprocity. The users estimate the channel through a periodic transmitted (by the base station) pilot and invert the channel before the transmission [1].

The receiver produces a list of estimated messages denoted by $\mathcal{L} \subseteq \{1, 2, \ldots, 2^k\}$. The probability of error, as in [3], is defined per user and regardless of the order of the messages. More precisely, the probability of error is defined as

$$\epsilon = \frac{1}{K_a} \sum_{i=1}^{K_a} \Pr[w_i \notin \mathcal{L}]. \tag{2.11}$$

2.3 Ordentlich-Polyanskiy Scheme

In this section, we review the scheme proposed by Ordentlich and Polyanskiy in [12]. In their work, they present a scheme which is a special case of the T-fold ALOHA, referred to here as OP scheme. Each block of length N is split in V sub-blocks and each user chooses randomly one of the sub-blocks to transmit. Unlike ALOHA, where a collision happens if two or more users transmit in the same sub-block, in OP scheme an error only happens if more than T users transmit in the same sub-block. The scheme is reviewed in the sequence.

Each active user i encodes its message w_i using a code for the Binary Adder Channel (BAC), generating the codeword $\mathbf{c}_{\mathrm{BAC},i} \in \{0,1\}^{k_c}$. Observing a modulo-2 sum of T or less codewords, this code must be able to successfully recover each codeword. The particular code used is discussed later, in Section 2.3.1.

Then, this codeword is encoded using a binary linear code $\mathcal{C}$ with rate $R_c = \frac{k_c}{n_c}$ and length n_c which is good for the bi-AWGN mod 2 channel, generating $\mathbf{c}_i = \mathbf{c}_{\mathrm{BAC},i}\mathbf{G} \bmod 2$, where $\mathbf{G} \in \{0,1\}^{k_c \times n_c}$ is the generator matrix of the linear code. For analysis, this code is assumed to achieve (2.9). Finally, the user maps the resulting binary codeword into a real signal $\mathbf{x}_i = 2\sqrt{PV}\left(\mathbf{c}_i - \frac{1}{2}\right)$. For convenience, since $\mathbf{x}_i$ depends only on the message w, we also denote $\mathbf{x}(w)$ as the transmitted signal constructed from the message w.

Let $\mathcal{A}_j$ be a subset of $\{1, 2, \ldots, K_a\}$ that represents the users that transmitted in the jth timeslot. For each timeslot, the receiver receives

$$\mathbf{y}_j = \sum_{i \in \mathcal{A}_j} \mathbf{x}_i + \mathbf{z}_j. \tag{2.12}$$

Let $\hat{t}_j$ be a receiver's estimate of $t_j = |\mathcal{A}_j|$. The receiver computes

$$\mathbf{y}_{\mathrm{CoF},j} = \left[\frac{1}{2\sqrt{VP}}\mathbf{y}_j + \frac{\hat{t}_j}{2}\right] \bmod 2 \tag{2.13}$$

where the modulo 2 reduction is into the interval $[0,\ 2)$ and is taken componentwise.

If $\hat{t}_j = t_j$, then

$$\mathbf{y}_{\mathrm{CoF},j} = \left[\sum_{i \in \mathcal{A}_j} \mathbf{c}_i + \tilde{\mathbf{z}}_j\right] \bmod 2 \tag{2.14}$$

where $\tilde{\mathbf{z}}_j = \frac{\mathbf{z}_j}{2\sqrt{PV}}$. Note that, since the code $\mathcal{C}$ is linear, the sum of codewords $\tilde{\mathbf{c}}_j = \sum_{i \in \mathcal{A}_j} \mathbf{c}_i \bmod 2$ also belongs to the code $\mathcal{C}$. Thus, the effective channel is

$$\mathbf{y}_{\mathrm{CoF},j} = (\tilde{\mathbf{c}}_j + \tilde{\mathbf{z}}_j) \bmod 2. \tag{2.15}$$

The codeword $\tilde{\mathbf{c}}_j$ is decoded and, since $\tilde{\mathbf{c}}_j = \left(\sum_{i\in\mathcal{A}_j} \mathbf{c}_{\mathrm{BAC},i}\right) \mathbf{G} \bmod 2$, if the decoding is successful, we recover

$$\mathbf{y}_{\mathrm{BAC},j} = \sum_{i\in\mathcal{A}_j} \mathbf{c}_{\mathrm{BAC},i} \bmod 2. \tag{2.16}$$

Finally, the receiver decodes $\mathbf{y}_{\mathrm{BAC},j}$, and, if the decoding is successful, it generates a list of codewords $\mathbf{c}_{\mathrm{BAC},i}$, $i \in \mathcal{A}_j$. These codewords are mapped into a list $\mathcal{L}(\hat{t}_j)$ of estimated messages w_i.

On the other hand, if $\hat{t}_j \neq t_j$, the computation of $\mathbf{y}_{\mathrm{CoF},j}$ results in

$$\mathbf{y}_{\mathrm{CoF},j} = \left[\sum_{i\in\mathcal{A}_j} c_i + (\hat{t}_j - t_j) + \tilde{\mathbf{z}}_j\right] \bmod 2. \tag{2.17}$$

which will likely cause an error in the decoding of the linear code if $(\hat{t}_j - t_j) \bmod 2 \neq 0$. If an undetected error occurs, a wrong $\hat{\mathbf{y}}_{\mathrm{BAC},j} \neq \mathbf{y}_{\mathrm{BAC},j}$ is returned, which will be then decoded by the BAC code, generating a list $\mathcal{L}(\hat{t}_j)$ of estimated messages.

For any $\hat{t}_j$, if a detected error occurs in any decoding step, the output is set to $\mathcal{L}(\hat{t}_j) = \emptyset$ and an error is flagged.

Let this decoding procedure, which depends on $\mathbf{y}_j$ and the estimated $\hat{t}_j$, be denoted by a function $\mathcal{L}(\hat{t}_j) = \Phi(\mathbf{y}_j, \hat{t}_j)$. The OP scheme computes $\mathcal{L}(\hat{t}_j) = \Phi(\mathbf{y}_j, \hat{t}_j)$ for $0 \leq \hat{t}_j \leq T$, generating $T+1$ lists $\mathcal{L}(\hat{t}_j)$. For each list, the base station can regenerate an estimate of the transmitted signals $\mathbf{x}(w)$ based on $\mathcal{L}(\hat{t}_j)$ and subtract them from $\mathbf{y}_j$, generating $\hat{\mathbf{z}}_j(\hat{t}_j) = \mathbf{y}_j - \sum_{w\in\mathcal{L}(\hat{t}_j)} \mathbf{x}(w)$. For $\hat{t}_j = t_j$ and if no error has occurred in any step of the decoding, this yields $\hat{\mathbf{z}}_j = \mathbf{z}_j$. Otherwise, it results in $\hat{\mathbf{z}}_j = \mathbf{z}_j + \sum \mathbf{x}$, where $\sum \mathbf{x}$ is some unknown sum of transmitted signals. Therefore, as argued in [12], the base station can easily choose the $\hat{t}_j^*$ which yields the best agreement with $\mathbf{y}_j$ and set $\mathcal{L}_j = \mathcal{L}(\hat{t}_j^*)$. If no list yields a good enough agreement, the decoder returns $\mathcal{L}_j = \emptyset$, which happens if there is an error in the decoding of the linear code for $\hat{t}_j = t_j$, if more than T users transmitted in the jth timeslot or if more than one user transmitted the same message w in the jth timeslot.

It is important to note that the OP scheme does not handle message collisions, i.e., more than one user transmitting the same message. The reasoning is that the probability of message collisions is negligible, as the number of possible messages is extremely large when $k \approx 100$.

Finally, although we only review the modulo-2 AWGN channel, the OP scheme is extended in [12] to multi-level codes, thus the effective channel is a modulo-2^ℓ AWGN channel. In order to do that in a multiple access channel, the authors add a preamble in the message, which is used to align the messages in each level and recover the complete message.

2.3.1 BAC Code

Let $\mathbf{H} \in \mathbb{F}_2^{mT\times(2^m-1)}$, where $m \in \mathbb{Z}$, be a parity-check matrix of a binary, narrow-sense, primitive BCH code of length 2^m-1 and designed minimum distance $2T+1$ [9]. We construct the code for the BAC using the columns of $\mathbf{H}$ as codewords. This BAC code has length $k_c = mT$ and cardinality $|\mathcal{C}_{\text{BAC}}| = 2^m - 1$, we set $k = \log_2(2^m - 1)$ in order to transmit k bits. For $m \gg 1$, in particular $m \approx 100$, we have $k \approx m$.

To see that this code is able to successfully decode the BAC channel output, recall that, since the BCH code is able to correct any T or fewer errors, all modulo-2 sums of T or less distinct columns of $\mathbf{H}$ are distinct.

Both the encoding and decoding of this code can be done with low complexity through the implementation described in [12].

2.3.2 Drawbacks

Although the OP scheme achieves good results compared to previous known methods, some drawbacks can be observed. First, the reduction to a modulo-2^ℓ channel implies in some loss of capacity. Second, adding a preamble to the message in order to use multi-level codes causes overhead. Finally, using the BAC code allows us to recover at most T users, but there is a non-negligible probability that strictly $t_j < T$ users transmit in the timeslot j. For example, for $K_a = 100$ and the parameters described in [12] results, the probability of fewer than $T = 5$ users transmitting in a timeslot is more than 95%. This means the BAC code is operating at a rate lower than the channel capacity at most of the time.

Our proposed scheme handles most of these drawbacks by transmitting only a small preamble using the OP scheme, and then transmitting the message in a coordinated MAC, as described in the following chapter.

Chapter 3

Contributions

In this chapter, we first present the proposed grant-based model that allows us to compare our method to grantless methods. We then present the proposed grant-based method for this model. We use the OP scheme, reviewed in Section 2.3, in the scheduling request phase of our method. We also improve the OP scheme with an index collision resolution method that allows the base station to decode the received codeword even if the same index has been transmitted by more than one user in the same timeslot.

3.1 Proposed Model

In our model, the first phase, described in Section 2.2, is used for signaling activity. Using the information decoded in $\mathbf{y}$, the base station generates a feedback signal $\mathbf{x}^{[f]}$ and transmits it in a broadcast channel to all users. Finally, using the information decoded from $\mathbf{y}^{[f]}$, the users transmit in a multiple access channel, as in the first phase.

More precisely, in our model, the transmission is split in three phases: Scheduling Request, Resource Distribution and Data Transmission. These three phases compose a session. The model for the Scheduling Request phase is the same as Polyanskiy's transmission. The Resource Distribution phase, which occurs in the downlink, is modeled as a broadcast transmission where the channel gain is the same for all users, i.e., each user i receives a signal

$$\mathbf{y}_i^{[f]} = \mathbf{x}^{[f]} + \mathbf{z}_i^{[f]} \tag{3.1}$$

where $\mathbf{x}^{[f]} \in \mathbb{R}^{N_f}$ is subject to $\mathbf{E}\left[\|\mathbf{x}^{[f]}\|^2\right] \leq N_f P_f$ and depends on $\mathbf{y}$, and $\mathbf{z}^{[f]} \sim \mathcal{N}(0, \mathbf{I})$.

The Data Transmission phase is modeled, again, as a multiple access channel, i.e.,

$$\mathbf{y}^{[2]} = \sum_{i=1}^{K_a} \mathbf{x}_i^{[2]} + \mathbf{z}^{[2]} \tag{3.2}$$

where $\mathbf{x}_i^{[2]} \in \mathbb{R}^{N_2}$ is subject to $\mathbf{E}\left[\|\mathbf{x}_i^{[2]}\|^2\right] \leq N_2 P_2$ and $\mathbf{z}^{[2]} \sim \mathcal{N}(0, \mathbf{I})$. However, this phase differs from (2.10) in that each user i has side information about $\mathbf{x}^{[f]}$ provided by $\mathbf{y}_i^{[f]}$. Finally, the receiver produces a list $\mathcal{L}$ of estimated messages based on both $\mathbf{y}$ and $\mathbf{y}^{[2]}$. The error probability is defined exactly as in (2.11).

We denote $N = N_1 + N_2 + N_f$. This is the total number of channel uses each user spends on this transmission. Note that the grantless model proposed by Polyanskiy is a particular case where $N_2 = N_f = 0$ and the data are transmitted in the first phase. It is clear to see that, if we wish to maintain the session length N, we require higher rates in the data transmission phase in order to transmit the same data.

This model is similar to the random access procedure described in, e.g., [1], where the users transmit a random preamble to the base station, receive a random access response and then transmit their data in separate channels. However, unlike traditional grant-based schemes, our model allows interference between users in the scheduling request phase, which is handled through random access transmission methods, and allows (coordinated) non-orthogonal multiple access in the data transmission phase.

3.1.1 Metrics of comparison

Following [3], [12] and [4], we are interested in the problem where, given some target probability of error ϵ and number of channel uses N, we wish to minimize the energy required for the K_a users to send k bits of information. For comparison, similar to [12], we define

$$\frac{E_b}{N_0} = \frac{P_1 N_1 + P_2 N_2 + P_f N_f / K_a}{2k} \tag{3.3}$$

which is proportional to the total energy spent in a session[1], normalized by the number of active users. In the particular case $N_f = N_2 = 0$, this is the same definition used in the previous works. Note that the energy in the downlink transmission $P_f N_f$ is used once to transmit to all users, therefore, the per-user energy in that phase is $P_f N_f / K_a$. [2]

3.2 Proposed Method

In this section, we describe the three phases of a session in our transmission method. In the Scheduling Request phase, instead of using the OP scheme to transmit k bits of payload, we use it to transmit a significantly smaller random identification preamble u with length $m \ll k$. Note, however, that this identification is not made across the K_{tot} users, but across only the K_a active users.

Additionally, we use the timeslot[3] where each user transmitted as part of their identification, decreasing significantly the number of indices required to differentiate the K_a users. Also, note that the identification preamble is random, therefore the user symmetry of the problem is preserved.

In the Resource Distribution phase, the base station transmits a simple feedback to all users, which consists of the recovered indices concatenated, ordered by the timeslot where each index was recovered. Based on the index and its position, the users are able to identify which resources are allocated to them.

[1] More precisely, (3.3) is proportional to an upper bound on the total energy spent in a session. Since errors may occur in the first two phases, impairing coordination, the total energy spent depends on the number of users that actually transmitted in the data transmission phase.

[2] More generally, we want to minimize both the energy spent by the users and by the base station. However, in order to compare it with other methods, we need to define a utility function, weighing the energy used by the base station against the energy used by the users. In our understanding, the energy spent by the base station should be less significant than the energy spent by the users. Therefore, a weight coefficient of "1" may be considered a conservative choice.

[3] We usually use timeslots to refer to sub-blocks. However, more precisely, the channel uses can be divided in any type of resources, e.g., frequency slots.

In the third phase, the users transmit information using some coordinated scheme. In principle, we can use any coordinated orthogonal or non-orthogonal multiple access scheme for this phase. In this work, we use an orthogonal scheme, which achieves the symmetric capacity of the MAC and presents good results under finite blocklength, in particular for $K_a \leq 100$, as can be seen in Fig. 1 from [3]. Even for a higher number of active users, it presents a better result than other known methods. Additionally, since each user is transmitting alone in this stage, known point-to-point AWGN codes can be used, and the known finite blocklength results from [14] can be used for analysis.

In the following subsections, the three phases are described in detail.

3.2.1 Scheduling Request Phase

In this stage, instead of transmitting the message w_i, each user randomly picks an index $u_i \in \{1, 2, \ldots, n_p\}$, where $n_p = 2^m - 1$ is a design parameter, and transmits it using the OP scheme. We denote n_p as the length of the BCH code and $m_p = mT$ as the length of its parity matrix columns, therefore also the length of the BAC code. For the linear code, we denote $n_{c,1}$ as the length of the code. One important distinction from the original OP scheme is that $n_p \ll 2^k$, thus the probability of two users transmitting the same index is not negligible. Because of that, we need to introduce an index collision resolution (ICR) method for the OP scheme, which is explained in detail in Section 3.2.5. The total number of channel uses in this stage is given by $N_1 = n_{c,1}V$, where V is the number of timeslots and $n_{c,1}$ is the number of channel uses in each timeslot.

The identification of the users can be represented in a matrix form with dimension $V \times n_p$. For example, a user choosing the index 3 (among $n_p = 7$), and transmitting in the timeslot 1 (among $V = 5$) can be represented with the following matrix.

$$
\begin{array}{cc}
 & \overbrace{\hphantom{0\ 0\ 1\ 0\ 0\ 0\ 0}}^{n_p} \\
V\left\{\begin{array}{c}\\ \\ \\ \\ \\ \end{array}\right. & \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}
\end{array}
$$

Furthermore, the received identification matrix can be seen as the sum of the user identification matrices, and the decoding of each timeslot as the estimation of each row of the matrix. Consider $K_a = 10$, then an example of the received identification matrix is

$$
\begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}
$$

3.2.2 Resource Distribution Phase

After the first phase is complete, we allocate resources to the users based in the estimation of the identification matrix. The pattern of allocation is known to the users. Then, we wish to broadcast the estimation of the identification matrix. Since the matrix is sparse, the receiver generates a feedback sequence $(|\mathcal{L}_j|, \{u \in \mathcal{L}_j\})$, $j = 1, \ldots, V$, i.e., we inform the weight of each row and the non-zero columns of each row. This sequence is broadcast to the users. With that information, each transmitter knows exactly what indices were transmitted (as long as the indices were recovered), and in which timeslots. Then, each user is able to identify its own position in the matrix and transmit using the resource respective to its position. If a user receives a zero in its position, this user does not transmit in the data transmission phase.

For example, consider $V = 5$, $n_p = 3$ and $K_a = 5$. An example of estimated identification matrix is

$$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

and the respective feedback sequence is $(1, \{3\}, 2, \{1, 2\}, 0, \{\}, 2, \{1, 3\}, 0, \{\})$. Now, consider that the pattern of resource allocation is simply ordering, row by row, as the indices appear. Then, the user that transmitted the index 2 in the timeslot 2, highlighted in blue, will receive the third resource to transmit. Since the user knows its own position $(2, 2)$ and received the estimated matrix (assuming the user is able to decode), the user knows which resource is allocated to the data transmission.

This feedback sequence has length $\leq \lceil \log_2(T+1) \rceil + t_j m$ per timeslot and a total feedback length $\leq V \lceil \log_2(T+1) \rceil + K_a m$.

3.2.3 Data Transmission Phase

In this phase, each user i that has successfully received its index in the resource distribution phase encodes its message w_i using a code for the point-to-point AWGN channel and transmits it with power $P_2 K_a$ within their respective resource. Although we do not restrain the method to any particular code, in practice, good codes for the AWGN channel under small blocklength such as those in [18] can be used.

In this phase, the number of channel uses is given by $N_2 = n_{c,2} K_a$, where $n_{c,2}$ is the length of the channel code.[4]

3.2.4 Error Analysis

In this section, we present an upper bound on the probability of error. Recall that the probability of error is defined per user. For the first

[4] An obvious improvement to this is to allocate $K_2 \leq K_a$ resources, where K_2 is the number of indices that have been successfully received in the scheduling request phase. In this case, the users transmit with power $P_2 K_2$ and the code has length $n_{c,2} = N_2 / K_2$. However, since K_2 is a random variable, this approach complicates both the analysis and the practical implementation, thus it will not be considered in this paper.

and second phases, we make the error analysis given that the user $i \in \{1, 2, \dots, K_a\}$ transmitted the index $u_i \in \{1, 2, \dots, n_p\}$ in the timeslot $j \in \{1, 2, \dots, V\}$. For the third phase, we make the error analysis given that the user $i \in \{1, 2, \dots, K_a\}$ transmitted the message $w_i \in \{1, 2, \dots, 2^k\}$ using the resource $j \in \{1, 2, \dots, K_a\}$. By symmetry, the resulting probability of error equals the average probability of error per user.

In the first stage, four types of error can occur. First, if other T or more users choose to transmit in the timeslot j, the error event E_1 occurs, with probability

$$\epsilon_1 \triangleq 1 - \sum_{t=0}^{T-1} \binom{K_a - 1}{t} \left(\frac{1}{V}\right)^t \left(1 - \frac{1}{V}\right)^{K_a - 1 - t}. \tag{3.4}$$

This probability of error can be derived from a binomial probability distribution originated from the sum of $K_a - 1$ Bernoulli random variables with probability $1/V$. More details are provided in Appendix A.

Second, if the receiver is unable to decode $\mathbf{y}_{\mathrm{CoF},j}$, the error event E_2 occurs, with probability upper bounded by

$$\epsilon_2 \triangleq Q\left(\left(C(P_1 V) - \frac{m_p}{n_{c,1}}\right)\sqrt{\frac{n_{c,1}}{\mathsf{V}(P_1 V)}}\right). \tag{3.5}$$

This follows from the fact that, as in the OP scheme, we have a bi-AWGN mod 2 channel and the information we wish to transmit through this channel consists of m_p bits. The first stage uses power P_1, and since each user transmits only once in V timeslots, they transmit with power $P_1 V$ during that timeslot. This probability of error can be easily derived from (2.9).

A third type of error occurs when two or more users pick the same index and transmit it in the timeslot j. In the original OP scheme, this would lead to an error in the timeslot j, i.e., an error for every user that transmitted in that timeslot, therefore the per-user probability of error would be upper-bounded by $T(T-1)/2n_p$, as in [12]. However, with the index collision resolution method presented in Section 3.2.5, we show that this event only produces an error for the users that picked the same index. More precisely, given that E_1 and E_2 have not occurred, the error event E_3 occurs if and only if any of the other users $i' \in \mathcal{A}_j, i' \neq i$

pick the index u_i, which occurs with probability

$$1 - \left(1 - \frac{1}{Vn_p}\right)^{(K_a-1)} < \frac{K_a - 1}{Vn_p} \triangleq \epsilon_3. \tag{3.6}$$

This approach significantly reduces the probability of error, which allows us to use a smaller n_p, reducing the user identification phase overhead. Again, this is derived from a binomial distribution and more details are given in Appendix A.

In the resource distribution phase, if the user i is unable to decode the feedback correctly, the error event E_f occurs, with probability upper bounded by

$$\epsilon_f \triangleq Q\left(\left(C_{\text{AWGN}}(P_f) - \frac{k_f}{N_f}\right)\frac{\sqrt{N_f}}{\sqrt{\mathsf{V}_{\text{AWGN}}(P_f)}}\right) \tag{3.7}$$

where $k_f = V\lceil \log_2(T+1)\rceil + K_a m$. This expression, as well as the one which will follow for the data transmission phase, can be easily derived from (2.6).

In the data transmission phase, two errors can occur. First, if the receiver is unable to decode the signal transmitted in the resource j, the error event E_4 happens. For this analysis we assume that (2.6) holds, thus the probability of error for the AWGN channel code is upper bounded by

$$\epsilon_4 \triangleq Q\left(\left(C_{\text{AWGN}}(P_2K_a) - \frac{k}{n_{c,2}}\right)\frac{\sqrt{n_{c,2}}}{\sqrt{\mathsf{V}_{\text{AWGN}}(P_2K_a)}}\right). \tag{3.8}$$

Second, if some user i' is subject to the error E_f and it transmits using the resource j which is allocated to the user i, the user i' causes an error to the user i. We denote this event by $E_{f,2}$ with probability upper bounded by

$$1 - \left(1 - \frac{\epsilon_f}{K_a}\right)^{K_a-1} \leq \frac{K_a - 1}{K_a}\epsilon_f \leq \epsilon_f \triangleq \epsilon_{f,2} \tag{3.9}$$

where $\frac{\epsilon_f}{K_a}$ is a loose upper bound on probability that a user i' is subject to E_f and transmits using the resource j. Note that, since the feedback has a distinguishable pattern, incorrect decoding of the feedback can usually be detected by the users. However, for simplicity, we do not consider this possibility in this upper bound.

Finally, the probability of error per user can be upper bounded by $\epsilon \leq \epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4 + \epsilon_f + \epsilon_{f,2}$.

3.2.5 Index Collision Resolution

In this section, we show that, if the error events E_1 and E_2 do not occur, i.e. if no more than T users transmitted in the same timeslot and the linear code is able to decode the linear codeword correctly, we are always able to recover all the indices that were transmitted, even if one or more indices u were transmitted more than once in the timeslot j.

Similar to the OP scheme, we run our decoding procedure for $0 \leq \hat{t}_j \leq T$. For each $\hat{t}_j$, our method returns either an error or a sequence of sets $\mathcal{L}^{(1)}(\hat{t}_j), \mathcal{L}^{(2)}(\hat{t}_j), \ldots, \mathcal{L}^{(\tau(\hat{t}_j))}(\hat{t}_j)$, where $\tau(\hat{t}_j)$ is the number of lists and $\mathcal{L}^{(\ell)}(\hat{t}_j) \subseteq \{1, 2, \ldots, n_p\}$ for all $\ell \geq 1$. Then, for each $\hat{t}_j$ that did not return an error, as in the OP scheme, we compute

$$\hat{\mathbf{z}}_j(\hat{t}_j) = \mathbf{y}_j - \sum_{\ell=1}^{\tau(\hat{t}_j)} 2^{(\ell-1)} \sum_{u \in \mathcal{L}^{(\ell)}(\hat{t}_j)} \mathbf{x}(u) \tag{3.10}$$

which allows us to find the $\hat{t}_j^*$ which gives the best agreement with $\mathbf{y}_j$, if any. Finally, we output

$$\mathcal{L}_j = \mathcal{L}^{(1)}(\hat{t}_j^*) \setminus \cup_{\ell=2}^{\tau(\hat{t}_j^*)} \mathcal{L}^{(\ell)}(\hat{t}_j^*). \tag{3.11}$$

As will be clear later, this output list consists of all the indices in the timeslot j that have been transmitted by a single user, i.e., indices for which E_3 has not occurred.

We now describe how the method computes its output for a given $\hat{t}_j$. We first compute $\mathcal{L}^{(1)}(\hat{t}_j) = \Phi(\mathbf{y}_j^{(1)}, \hat{t}_j^{(1)})$, where $\mathbf{y}_j^{(1)} = \mathbf{y}_j$ and $\hat{t}_j^{(1)} = \hat{t}_j$, exactly as described in the OP scheme.

For $\ell \geq 1$, after computing $\mathcal{L}^{(\ell)}(\hat{t}_j) = \Phi(\mathbf{y}_j^{(\ell)}, \hat{t}_j^{(\ell)})$, if $|\mathcal{L}^{(\ell)}(\hat{t}_j)| = \hat{t}_j^{(\ell)}$, we return the sequence $\mathcal{L}^{(1)}(\hat{t}_j), \mathcal{L}^{(2)}(\hat{t}_j), \ldots, \mathcal{L}^{(\ell)}(\hat{t}_j)$, setting $\tau(\hat{t}_j) = \ell$.

If $|\mathcal{L}^{(\ell)}(\hat{t}_j)| > \hat{t}_j^{(\ell)}$, we return an error. If $|\mathcal{L}^{(\ell)}(\hat{t}_j)| < \hat{t}_j^{(\ell)}$, we compute

$$\hat{t}_j^{(\ell+1)} = \frac{\hat{t}_j^{(\ell)} - |\mathcal{L}^{(\ell)}(\hat{t}_j)|}{2} \tag{3.12}$$

and return an error if $\hat{t}_j^{(\ell+1)} \notin \mathbb{Z}$. Otherwise, we compute

$$\mathbf{y}_j^{(\ell+1)} = \frac{\mathbf{y}_j^{(\ell)} - \sum_{u \in \mathcal{L}^{(\ell)}(\hat{t}_j)} \mathbf{x}(u)}{2} \tag{3.13}$$

and $\mathcal{L}^{(\ell+1)}(\hat{t}_j) = \Phi(\mathbf{y}^{(\ell+1)}, \hat{t}_j^{(\ell+1)})$.

It is easy to see that this procedure halts in at most $\tau(\hat{t}_j) \leq \log_2(\hat{t}_j) + 1$ iterations, since $\hat{t}_j^{(\ell+1)} \leq \hat{t}_j^{(\ell)}/2$ and $\hat{t}_j^{(\tau(\hat{t}_j))} \geq 1$.

Before we show that our method works, let us introduce some notation. We denote by $\mathcal{A}_j^{(1)}(u) \subseteq \mathcal{A}_j$ the set of users that transmitted the index u in the timeslot j, i.e., if $i \in \mathcal{A}_j^{(1)}(u)$, then $\mathbf{x}_i = \mathbf{x}(u)$. Additionally, for $\ell \geq 1$, let $\mathcal{B}_j^{(\ell)}(u) \subseteq \mathcal{A}_j^{(\ell)}(u)$ be some subset such that

$$|\mathcal{B}_j(u)| = 2\left\lfloor \frac{|\mathcal{A}_j(u)|}{2} \right\rfloor$$

let $\mathcal{A}_j^{(\ell+1)}(u) \subseteq \mathcal{B}_j^{(\ell)}(u)$ be some subset such that

$$|\mathcal{A}_j^{(\ell+1)}(u)| = \frac{|\mathcal{B}_j^{(\ell)}(u)|}{2}$$

and let

$$\mathcal{B}_j^{(\ell)} \triangleq \cup_{u \in \{1,2,\ldots,n_p\}} \mathcal{B}_j^{(\ell)}(u) \tag{3.14}$$

$$\mathcal{A}_j^{(\ell+1)} \triangleq \cup_{u \in \{1,2,\ldots,n_p\}} \mathcal{A}_j^{(\ell+1)}(u). \tag{3.15}$$

Finally, we define $\mathcal{U}(\mathcal{S}) \triangleq \{u_i, i \in \mathcal{S}\}$, where $\mathcal{S} \subseteq \{1, 2, \ldots, K_a\}$ is some set of users.

For example, if the users $\mathcal{A}_j^{(1)} = \{2, 3, 9, 12\}$ transmitted the corresponding indices $\{7, 7, 7, 4\}$ in the j-th timeslot, then $\mathcal{A}_j^{(1)}(7) = \{2, 3, 9\}$ and $\mathcal{A}_j^{(1)}(4) = \{12\}$. Some of the possible subsets are $\mathcal{B}_j^{(1)}(7) = \mathcal{B}_j^{(1)} = \{2, 3\}$ and $\mathcal{A}_j^{(2)}(7) = \mathcal{A}_j^{(2)} = \{2\}$. We also have $\mathcal{U}\left(\mathcal{A}_j^{(1)}\right) = \{7, 4\}$ and

$\mathcal{U}(\mathcal{A}_j^{(2)}) = \mathcal{U}(\mathcal{B}_j^{(1)}) = \{7\}$.

Note that, by construction, $|\mathcal{A}_j^{(\ell)}(u) \setminus \mathcal{B}_j^{(\ell)}(u)|$ is either 0 or 1, therefore, $\left|\mathcal{U}\left(\mathcal{A}_j^{(\ell)} \setminus \mathcal{B}_j^{(\ell)}\right)\right| = |\mathcal{A}_j^{(\ell)} \setminus \mathcal{B}_j^{(\ell)}|$ for all $\ell \geq 1$. It is also easy to see that $\mathcal{U}(\mathcal{A}_j^{(\ell+1)}) = \mathcal{U}(\mathcal{B}_j^{(\ell)})$. This will be useful later.

To simplify notation, let $\tau = \tau(t_j)$ for the remainder of this subsection.

Lemma 1 *If E_1 and E_2 do not occur and $\hat{t}_j = t_j$, then, for all $\ell \geq 1$*

(i) $\hat{t}^{(\ell)} = |\mathcal{A}_j^{(\ell)}|$.

(ii) $\mathbf{y}_j^{(\ell)} = \sum_{i \in \mathcal{A}_j^{(\ell)}} \mathbf{x}_i + \mathbf{z}_j / 2^{\ell-1}$.

(iii) The decoding of the ℓth iteration is successful.

(iv) $\mathcal{L}^{(\ell)}(\hat{t}_j) = \mathcal{U}\left(\mathcal{A}_j^{(\ell)} \setminus \mathcal{B}_j^{(\ell)}\right)$.

Moreover, $\mathcal{B}_j^{(\tau)} = \emptyset$, which implies $\mathcal{L}^{(\tau)}(\hat{t}_j) = \mathcal{U}\left(\mathcal{A}_j^{(\tau)}\right)$.

Proof: First, we prove the main statement for $\ell = 1$. Part (i) follows from the assumption $\hat{t}_j = t_j$, since $\hat{t}_j^{(1)} = \hat{t}_j$ and $t_j = |\mathcal{A}_j| = |\mathcal{A}_j^{(1)}|$ by definition. Part (ii) also follows directly from definition. Part (iii) follows from the assumptions.

In order to prove (iv), we write the received signal as

$$\mathbf{y}_j^{(1)} = \mathbf{y}_j = \sum_{i \in \mathcal{B}_j^{(1)}} \mathbf{x}_i + \sum_{i \in \mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}} \mathbf{x}_i + \mathbf{z}_j. \tag{3.16}$$

Since $\hat{t}_j^{(1)} = t_j$, the computation of (2.13) yields

$$\mathbf{y}_{\mathrm{CoF},j}^{(1)} = \left[\sum_{i \in \mathcal{B}_j^{(1)}} \mathbf{c}_i + \sum_{i \in \mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}} \mathbf{c}_i + \tilde{\mathbf{z}}_j\right] \bmod 2 \tag{3.17}$$

$$= \left[\sum_{i \in \mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}} \mathbf{c}_i + \tilde{\mathbf{z}}_j\right] \bmod 2 \tag{3.18}$$

where (3.18) follows from $|\mathcal{B}_j^{(1)}(u)|$ being even for all u by construction, which implies that the sum $\sum_{i\in\mathcal{B}_j^{(1)}} \mathbf{c}_i$ vanishes.

Since the linear code is assumed to be able to decode this sum of codewords and $|\mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}| \leq |\mathcal{A}_j| \leq T$, the BAC code is also able to successfully decode all the indices transmitted by the users in $\mathcal{A}_j^{(1)}\setminus\mathcal{B}_j^{(1)}$. Therefore, $\mathcal{L}^{(1)}(\hat{t}_j) = \mathcal{U}\left(\mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}\right)$.

We now show that, if the lemma holds for ℓ, it also holds for $\ell+1$.

Since the lemma holds for ℓ, i.e., $\hat{t}_j^{(\ell)} = |\mathcal{A}_j^{(\ell)}|$ and $|\mathcal{L}^{(\ell)}(\hat{t}_j)| = |\mathcal{A}_j^{(\ell)} \setminus \mathcal{B}_j^{(\ell)}|$, then we have $\hat{t}_j^{(\ell+1)} = \frac{|\mathcal{A}_j^{(\ell)}|-|\mathcal{A}_j^{(\ell)}\setminus\mathcal{B}_j^{(\ell)}|}{2} = \frac{|\mathcal{B}_j^{(\ell)}|}{2} = |\mathcal{A}_j^{(\ell+1)}|$, which completes the proof of (i).

The computation of the $(\ell+1)$th iteration is given by

$$
\begin{aligned}
\mathbf{y}_j^{(\ell+1)} &= \frac{\mathbf{y}_j^{(\ell)} - \sum_{u\in\mathcal{L}_j^{(\ell)}} \mathbf{x}(u)}{2} && (3.19)\\
&= \frac{\mathbf{y}_j^{(\ell)} - \sum_{i\in\mathcal{A}_j^{(\ell)}\setminus\mathcal{B}_j^{(\ell)}} \mathbf{x}_i}{2} && (3.20)\\
&= \frac{\sum_{i\in\mathcal{B}_j^{(\ell)}} \mathbf{x}_i + \mathbf{z}_j/2^{\ell-1}}{2} && (3.21)\\
&= \sum_{u\in\{1,2,\ldots,n_p\}} \sum_{i\in\mathcal{B}_j^{(\ell)}(u)} \frac{\mathbf{x}_i}{2} + \frac{\mathbf{z}_j}{2^\ell} && (3.22)\\
&= \sum_{u\in\{1,2,\ldots,n_p\}} |\mathcal{B}_j^{(\ell)}(u)|\frac{\mathbf{x}(u)}{2} + \frac{\mathbf{z}_j}{2^\ell} && (3.23)\\
&= \sum_{u\in\{1,2,\ldots,n_p\}} |\mathcal{A}_j^{(\ell+1)}(u)|\mathbf{x}(u) + \frac{\mathbf{z}_j}{2^\ell} && (3.24)\\
&= \sum_{u\in\{1,2,\ldots,n_p\}} \sum_{i\in\mathcal{A}_j^{(\ell+1)}(u)} \mathbf{x}_i + \frac{\mathbf{z}_j}{2^\ell} && (3.25)\\
&= \sum_{i\in\mathcal{A}_j^{(\ell+1)}} \mathbf{x}_i + \frac{\mathbf{z}_j}{2^\ell} && (3.26)
\end{aligned}
$$

where (3.19) follows from definition (3.13); (3.20) and (3.21) follow from hypothesis; and the remaining follows from the construction of the sets. This completes the proof of (ii).

In order to prove (iii), we rewrite the signal as

$$\mathbf{y}_j^{(\ell+1)} = \sum_{i\in\mathcal{B}_j^{(\ell+1)}} \mathbf{x}_i + \sum_{i\in\mathcal{A}_j^{(\ell+1)}\setminus\mathcal{B}_j^{(\ell+1)}} \mathbf{x}_i + \mathbf{z}_j^{(\ell+1)} \tag{3.27}$$

since $\mathcal{B}_j^{(\ell+1)}$ is a subset of $\mathcal{A}_j^{(\ell+1)}$.

Since $\hat{t}_j^{(\ell+1)} = |\mathcal{A}_j^{(\ell+1)}|$, the dither is successfully corrected in the $(\ell+1)$th iteration, thus the computation of (2.13) for this iteration yields

$$\mathbf{y}_{\mathrm{CoF},j}^{(\ell+1)} = \left[\sum_{i\in\mathcal{A}_j^{(\ell+1)}\setminus\mathcal{B}_j^{(\ell+1)}} \mathbf{c}_i + \frac{\tilde{\mathbf{z}}_j}{2^\ell}\right] \bmod 2. \tag{3.28}$$

Since we are able to decode $\mathbf{y}_{\mathrm{CoF},j}^{(\ell)}$, which is subject to noise $\frac{\tilde{\mathbf{z}}_j}{2^{\ell-1}}$, we should be able to decode $\mathbf{y}_{\mathrm{CoF},j}^{(\ell+1)}$, as the variance of the noise $\frac{\tilde{\mathbf{z}}_j}{2^\ell}$ is reduced by a factor of 4, thus the decoding of the linear code is successful. Additionally, since $|\mathcal{A}_j| = t_j \leq T$ and $\mathcal{B}_j^{(\ell)} \subseteq \mathcal{A}_j$, then $|\mathcal{B}_j^{(\ell)}|/2 < T$, thus the decoding of the BAC code is successful, completing the proof of (iii). This immediately implies that we have $\mathcal{L}^{(\ell+1)}(\hat{t}_j) = \mathcal{U}\left(\mathcal{A}_j^{(\ell+1)} \setminus \mathcal{B}_j^{(\ell+1)}\right)$, completing the proof of the main statement.

Note that, under the conditions of the lemma, the method does not return an error, therefore it ends with $|\mathcal{A}_j^{(\tau)} \setminus \mathcal{B}_j^{(\tau)}| = |\mathcal{L}^{(\tau)}| = \hat{t}_j^{(\tau)} = |\mathcal{A}_j^{(\tau)}|$, which implies $\mathcal{B}_j^{(\tau)} = \emptyset$. It immediately follows that $\mathcal{L}^{(\tau)} = \mathcal{U}\left(\mathcal{A}_j^{(\tau)} \setminus \mathcal{B}_j^{(\tau)}\right) = \mathcal{U}\left(\mathcal{A}_j^{(\tau)}\right)$. ■

Corollary 1 *If E_1 and E_2 do not occur and $\hat{t}_j = t_j$, then $\mathcal{U}\left(\mathcal{A}_j^{(\ell)}\right) = \cup_{\ell'=\ell}^{\tau} \mathcal{L}^{(\ell')}(\hat{t}_j)$.*

Proof: First, recall that, by construction, $\mathcal{U}\left(\mathcal{A}_j^{(\ell)}\right) = \mathcal{U}\left(\mathcal{B}_j^{(\ell-1))}\right)$. Thus, we have

$$\begin{aligned}
\mathcal{U}\left(\mathcal{A}_j^{(\ell-1)}\right) &= \mathcal{U}\left(\mathcal{B}_j^{(\ell-1)}\right) \cup \mathcal{U}\left(\mathcal{A}_j^{(\ell-1)} \setminus \mathcal{B}_j^{(\ell-1)}\right) && (3.29)\\
&= \mathcal{U}\left(\mathcal{A}_j^{(\ell)}\right) \cup \mathcal{U}\left(\mathcal{A}_j^{(\ell-1)} \setminus \mathcal{B}_j^{(\ell-1)}\right) && (3.30)
\end{aligned}$$

for all $\ell \geq 2$. Applying Lemma 1 to (3.30) with $\ell = \tau$ yields

$$\mathcal{U}\left(\mathcal{A}_j^{(\tau-1)}\right) = \mathcal{U}\left(\mathcal{A}_j^{(\tau)}\right) \cup \mathcal{U}\left(\mathcal{A}_j^{(\tau-1)} \setminus \mathcal{B}_j^{(\tau-1)}\right) \tag{3.31}$$

$$= \mathcal{L}^{(\tau)}(\hat{t}_j) \cup \mathcal{L}^{(\tau-1)}(\hat{t}_j). \tag{3.32}$$

We can then solve (3.30) recursively, for example, for $\ell = \tau - 1$, we have

$$\mathcal{U}\left(\mathcal{A}_j^{(\tau-2)}\right) = \mathcal{U}\left(\mathcal{A}_j^{(\tau-1)}\right) \cup \mathcal{U}\left(\mathcal{A}_j^{(\tau-2)} \setminus \mathcal{B}_j^{(\tau-2)}\right) \tag{3.33}$$

$$= \mathcal{L}^{(\tau)}(\hat{t}_j) \cup \mathcal{L}^{(\tau-1)}(\hat{t}_j) \cup \mathcal{L}^{(\tau-2)}(\hat{t}_j). \tag{3.34}$$

Generally, solving the recursion yields

$$\mathcal{U}\left(\mathcal{A}_j^{(\ell)}\right) = \cup_{\ell'=\ell}^{\tau} \mathcal{L}^{(\ell')}(\hat{t}_j). \tag{3.35}$$

■

Lemma 2 *If E_1 and E_2 do not occur and $\hat{t}_j = t_j$, then $\hat{\mathbf{z}}_j(\hat{t}_j) = \mathbf{z}_j$.*

Proof: Let

$$\mathbf{y}_j^{(\tau+1)} \triangleq \frac{\mathbf{y}_j^{(\tau)} - \sum_{u \in \mathcal{L}_j^{(\tau)}} \mathbf{x}(u)}{2} \tag{3.36}$$

$$= \frac{\mathbf{y}_j^{(\tau)} - \sum_{i \in \mathcal{A}_j^{(\tau)}} \mathbf{x}_i}{2} \tag{3.37}$$

$$= \frac{\mathbf{z}_j}{2^\tau}. \tag{3.38}$$

where the equalities follow immediately from Lemma 1.

Additionally, solving the recursion in $\mathbf{y}_j^{(\tau)}$ using (3.13), we also have

$$\mathbf{y}_j^{(\tau+1)} = \frac{\mathbf{y}_j}{2^\tau} - \sum_{\ell=1}^{\tau} \frac{1}{2^{\tau+1-\ell}} \sum_{u \in \mathcal{L}_j^{(\tau)}} \mathbf{x}(u) \tag{3.39}$$

$$= \frac{\hat{\mathbf{z}}_j(\hat{t}_j)}{2^\tau} \tag{3.40}$$

where (3.40) follows from the definition in (3.10).

Comparing both equations, we have $\hat{\mathbf{z}}_j(\hat{t}_j) = \mathbf{z}_j$. ■

Theorem 1 *If E_1 and E_2 do not occur, with negligible probability of error the list $\mathcal{L}_j$ contains all the indices that were transmitted only once in the timeslot j.*

Proof: Since the method computes an output for $0 \leq \hat{t}_j \leq T$ and $t_j \leq T$, eventually we have $\hat{t}_j = t_j$. From Lemma 2, this will result in $\hat{\mathbf{z}}_j(\hat{t}_j) = \mathbf{z}_j$. Therefore, $\hat{t}_j = t_j$ will, with negligible probability of error, yield the best agreement with $\mathbf{y}_j$, as discussed in Chapter 2.3, and it suffices to consider this case.

Since E_1 and E_2 are assumed to not occur and we are considering the case $\hat{t}_j = t_j$, then, from Corollary 1,

$$\mathcal{U}\left(\mathcal{A}_j^{(\ell)}\right) = \cup_{\ell'=\ell}^{\mathcal{T}} \mathcal{L}^{(\ell')}(\hat{t}_j). \tag{3.41}$$

Finally, it is easy to see that $\mathcal{U}\left(\mathcal{A}_j^{(2)}\right)$ contains all the indices that collided, i.e., indices that were transmitted more than once. Note that $\mathcal{L}^{(1)}(\hat{t}_j) = \mathcal{U}\left(\mathcal{A}_j^{(1)} \setminus \mathcal{B}_j^{(1)}\right)$ might contain indices that collided as well, in case of an odd number of colliding users. Therefore, the list of all indices that were transmitted only once is given by

$$\mathcal{U}\left(\mathcal{A}_j^{(1)}\right) \setminus \mathcal{U}\left(\mathcal{A}_j^{(2)}\right) = \left(\cup_{\ell=1}^{\mathcal{T}} \mathcal{L}^{(\ell)}(\hat{t}_j)\right) \setminus \cup_{\ell=2}^{\mathcal{T}} \mathcal{L}^{(\ell)}(\hat{t}_j) \tag{3.42}$$

$$= \mathcal{L}^{(1)}(\hat{t}_j) \setminus \cup_{\ell=2}^{\mathcal{T}} \mathcal{L}^{(\ell)}(\hat{t}_j). \tag{3.43}$$

Therefore, the output $\mathcal{L}_j$ given in (3.11) contains all the indices that have not collided. ■

Example 1 *Consider the scenario where the users $\mathcal{A}_j = \{1, 2, 5, 6, 7, 8, 9\}$ transmitted the indices $\{11, 11, 23, 10, 23, 23, 23\}$ in the timeslot j and $T \geq 7$. Note that $\mathcal{B}_j = \{1, 2, 5, 7, 8, 9\}$, as only the user 6 did not collide.*

The received signal can then be written as

$$\mathbf{y}_j^{(1)} = \sum_{i \in \{1,2,5,7,8,9\}} \mathbf{x}_i + \mathbf{x}_6 + \mathbf{z}_j = 2\mathbf{x}_1 + 4\mathbf{x}_5 + \mathbf{x}_6 + \mathbf{z}_j \tag{3.44}$$

Assuming $\hat{t}_j = 7$, the computation of (2.13) is given by

$$\mathbf{y}_{CoF,j}^{(1)} = [\mathbf{c}_6 + \tilde{\mathbf{z}}_j] \bmod 2. \tag{3.45}$$

Assuming we are able to successfully decode the linear code, then we are able to correctly recover $\mathcal{L}^{(1)}(\hat{t}_j) = \{u_6\} = \{10\}$*, since there is no collision in this subset and* $1 < T$*. We can then reconstruct* $\mathbf{x}_6 = \mathbf{x}(10)$*, subtract it from* $\mathbf{y}_j$ *and divide the result by two, yielding*

$$\mathbf{y}_j^{(2)} = \frac{\left(\mathbf{y}_j^{(1)} - \mathbf{x}_6\right)}{2} = \mathbf{x}_1 + 2\mathbf{x}_5 + \frac{\mathbf{z}_j}{2}. \tag{3.46}$$

We can then repeat the decoding algorithm for $t_j^{(2)} = \frac{7-1}{2} = 3$ *and recover* $\mathcal{L}^{(2)}(\hat{t}_j) = \{u_1\} = \{u_2\} = \{11\}$*. We then reconstruct* $\mathbf{x}_1 = \mathbf{x}(11)$ *and compute*

$$\mathbf{y}_j^{(3)} = \frac{\left(\mathbf{y}_j^{(2)} - \mathbf{x}_1\right)}{2} \tag{3.47}$$

$$= \mathbf{x}_5 + \frac{\mathbf{z}_j}{4}. \tag{3.48}$$

Repeating the algorithm with $\hat{t}_j^{(3)} = 1$*, we are able to recover* $\mathcal{L}^{(3)}(\hat{t}_j) = \{u_5\} = \{23\}$*. Now,* $|\{23\}| = 1$*, thus the method returns all the computed lists.*

Finally, we compute

$$\hat{\mathbf{z}}_j = \mathbf{y}_j - 1 \cdot \mathbf{x}(10) - 2 \cdot \mathbf{x}(11) - 4\mathbf{x}(23). \tag{3.49}$$

Recall that $\mathbf{x}(u_i) = \mathbf{x}_i$*, e.g.* $\mathbf{x}(11) = \mathbf{x}_1$*. Therefore, we have* $\hat{\mathbf{z}}_j = \mathbf{z}_j$*, and we are able to verify, with negligible probability of error, that this is pure Gaussian noise, thus* $\hat{t}_j = t_j$ *and the decoding is correct.* ■

Chapter 4

Optimization and Results

In this chapter, we present our optimization method, which allows us to efficiently design the parameters of our scheme. We then compare our results to the state-of-art using the same parameters as other works in the area. We also study the achievability of the rate presented in (2.9) using real codes with maximum likelihood decoding. Then, we present simulation results with LDPC codes to verify the performance of our method with practical codes. Finally, we analyze the effects of parameters such as packet length and error probability in the performance of our method.

4.1 Optimization Method

Given k and K_a, our scheme depends on the parameters T, m, $n_{c,1}$, $n_{c,2}$, V, N_f, P_1, P_2 and P_f. We wish to choose these parameters in order to minimize E_b/N_0, subject to constraints on ϵ and N. In order to simplify the optimization, we use[1] $P_1 = P_2 = P_f/K_a = P$. In this case, since N and k are fixed, minimizing $E_b/N_0 = \frac{PN}{2k}$ is equivalent to minimizing P.

This optimization is hard in general since all variables, except P, are integers. Thus, we follow a heuristic approach. Given a fixed $T \in \mathbb{Z}$, we

[1]For the scenarios considered here, this simplification is found experimentally to have negligible impact on performance.

relax the integer constraints on the remaining variables, finding their optimal values over $\mathbb{R}$. This is possible since all the error expressions are computable with real variables (except ϵ_1 with T). The resulting value of P gives a lower bound on the achievable P. Then, for each variable in the sequence m, n_{c_1}, $n_{c,2}$, V, its optimal relaxed value is rounded to $\mathbb{Z}$ and kept fixed, followed by a new run of the relaxed optimization for the remaining variables. We use rounding to the nearest integer, except in the case of m, where we compare floor and ceiling, choosing the one that yields the smallest P. After these variables are fixed, we choose $N_f = N - N_1 - N_2$ and compute the required power P to achieve the target error ϵ. This process gives an achievable P for the initially chosen T. Finally, this process is repeated for $T = 1, \ldots, T_{\max}$ to find the smallest P. More details of the implementation can be found in Appendix C.

While this approach is not guaranteed to be optimal, it has shown experimentally to yield values of P very close to the relaxed lower bound (ratio below 0.06 dB for all cases tested).

4.2 Results

This section presents the results of our scheme using the optimization method described.

4.2.1 State-of-art comparison

First, as in [12, 4, 13], we use $k = 100$, $N = 30000$ and $\epsilon = 0.05$, which allows a direct comparison. The results are presented in Fig 4.1 and compared to [12, 4, 13]. We also plot the random coding bound and the orthogonal MA bound [3], where perfect coordination is assumed a priori. Note that, since we use orthogonal methods in the data transmission, the latter provides a lower bound to our method. As can be seen, even without ICR, our method outperforms that in [13] for a moderate number of users ($K_a \leq 150$), but its performance comparatively worsens as K_a grows. With ICR, our method outperforms [13] significantly for all the values of K_a tested. Note that the use of ICR presents higher benefits as K_a grows, which is easy to see since the number of users transmitting in the same timeslot increases, increasing the probability that two or more choose the same index.

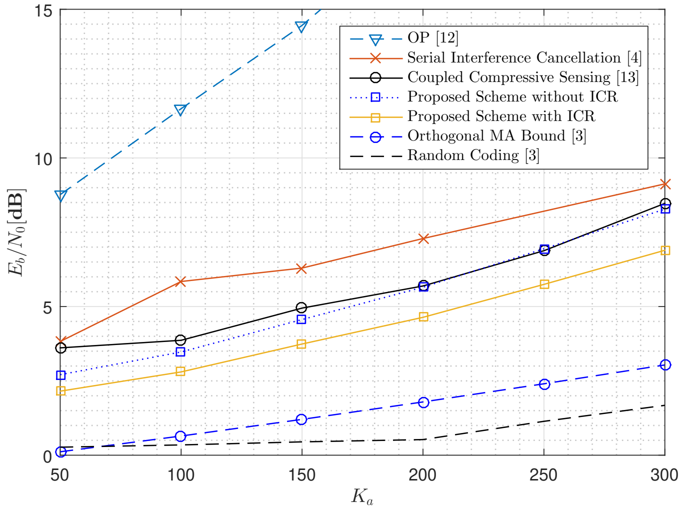

Figure 4.1: Comparison between the E_b/N_0 required for $k = 100$ bits, $N = 30000$ channel uses, $\epsilon = 0.05$.

Table 4.1 presents the parameters that achieve minimum E_b/N_0 with ICR. A few practical considerations can be made concerning these parameters. First, note that m and T are generally small, thus maximum likelihood (ML) decoding of the linear code in the scheduling request may be feasible, especially since it is done in the base station. Alternatively, ordered statistics decoding [19] may be used to achieve near-ML performance. Also, N_f is large enough so off-the-shelf codes for the AWGN channel may be used. Additionally, note that the optimal ϵ_f is small and the upper bound $\epsilon_{f,2}$ is loose, therefore, user collision in the data transmission should be negligible using our method.

4.2.2 Probability of Error for the Binary AWGN mod 2 Channel

Using the optimal parameters from Table 4.1, we designed simple linear codes for each K_a and simulated the error probability under the bi-AWGN mod 2 channel described in (2.8) using an approximated maximum likelihood decoder which is described in Appendix D. The results are presented in Table 4.2. Note that for $K_a \geq 250$, the rate is

Table 4.1: Optimized parameters with IRC

K_a	50	100	150	200	250	300
T	2	2	3	3	3	4
m	3	3	4	4	4	5
$n_{c,1}$	13	8	17	14	13	22
$n_{c,2}$	424	220	144	108	87	71
V	404	632	391	480	517	326
N_f	3548	2944	1753	1680	1529	1528
$\epsilon_1(\%)$	0.666	1.097	0.684	0.863	1.292	1.416
$\epsilon_2(\%)$	0.716	0.309	0.472	0.180	0.002	0.045
$\epsilon_3(\%)$	1.732	2.237	2.540	2.763	3.210	2.958
$\epsilon_f(\%)$	0.0588	0.0345	0.0176	0.00840	0.0113	0.0315
$\epsilon_4(\%)$	1.823	1.319	1.281	1.179	0.479	0.545

not achieved. However, note that, for $K_a = 250$, the total error ϵ is still smaller than the target 5%, therefore our results are achievable even if ϵ_2 is not. In other words, if we set the target error to $\epsilon_2 = 0.005\%$, the designed parameters and required energy do not change.

Finally, in order to obtain achievable results for $K_a = 300$, we redesign the parameters using $\epsilon_2' = \epsilon_2 + 0.001$, i.e., designing with a gap in the error probability. This increases the required E_b/N_0 from 6.89 dB to 6.93 dB, i.e., only a 0.04 dB loss, and yields an achievable result, which is presented in the Table 4.2 in the last row.

Additional simulation results are presented in Appendix E, as well as a list of the parity check matrices used. One particularly interesting result is that, for all the codes considered, relatively high probabilities of error are easily achieved, while the information theory approximation is clearly optimistic for small probabilities of error. However, when the error is small, its contribution in the sum decreases as well, therefore, a small gap in the target ϵ should be sufficient.

4.2.3 Performance Comparison Using LDPC Codes

Using $K_a = 100$ as a case study, we are interested in the performance of off-the-shelf LDPC codes in our problem. We use regular LDPC codes

Table 4.2: Probability of error in the bi-AWGN mod 2 channel

K_a	Code Error (%)	Target Error (%)
50	0.655	0.716
100	0.167	0.309
150	0.179	0.473
200	0.181	0.181
250	0.005	0.002
300	0.131	0.045
300	0.120	$0.133^{(*)}$

constructed using progressive edge-growth [20] and a variable-node degree of 3. We have verified experimentally that, in order to achieve the required target ϵ_4, a gap of approximately 1.4 dB is required, and in order to achieve the required target ϵ_f, a gap of approximately 2.2 dB is required. Therefore, we re-optimize the parameters including these gaps in the equations, i.e., we modify the equations to

$$\epsilon'_f \triangleq Q\left(\left(C_{\text{AWGN}}(P'_f) - \frac{k_f}{N_f}\right)\frac{\sqrt{N_f}}{\sqrt{\mathsf{V}_{\text{AWGN}}(P'_f)}}\right) \tag{4.1}$$

$$P'_f = \frac{PK_a}{10^{2.2/10}} \tag{4.2}$$

for the feedback code and

$$\epsilon'_4 \triangleq Q\left(\left(C_{\text{AWGN}}(P'_2K_a) - \frac{k}{n_{c,2}}\right)\frac{\sqrt{n_{c,2}}}{\sqrt{\mathsf{V}_{\text{AWGN}}(P'_2K_a)}}\right) \tag{4.3}$$

$$P'_2 = \frac{P}{10^{1.4/10}} \tag{4.4}$$

for the data transmission code. Note that the required energy is still computed as $\frac{PN}{2k}$. This increases the required E_b/N_0 to 4.10 dB, an increase of about 1.3 dB compared to the theoretic result.

Afterwards, we verify if the new design is achievable. Table 4.3 presents the error probabilities[2], comparing the target probability and

[2] For the probability of error in in the feedback ϵ_f, no errors were encountered

Table 4.3: Comparison between theoretical and practical probabilities of error of the codes

	Practical Code Error	Target Error
ϵ_2	0.00018	0.0014
ϵ_4	0.0143	0.0140
ϵ_f	$< 10^{-4}$	0.00015
ϵ_c	< 0.01458	0.01555

the simulated code probability with the new parameters, where ϵ_c is the sum of probabilities of error of the codes. Note that ϵ_4 is still not achievable due to changes in the optimal parameters, but it is compensated by the significantly smaller error probability of ϵ_2. Therefore, the method achieves the desired probability of error. Recall that this probability is still upper bounded due to the union bound.

4.2.4 Variation of parameters

In this subsection, we analyze the effects of varying input parameters. First, we verify the effects of changing K_a while maintaining fixed the ratio

$$\rho = \frac{kK_a}{N} \tag{4.5}$$

i.e., the system spectral efficiency. Then, we investigate the effects of varying ρ through variation of N while maintaining k. Finally, we investigate the effects of changing ϵ. For the remaining of the subsection, the fixed values are $k = 100$, $N = 30000$, $K_a = 200$ and $\epsilon = 0.05$ when these parameters are not specified. For all the following results, we use the value of power of the relaxed optimization in order to simplify the problem. For the scenarios that we have tested the full (integer) optimization presented in this chapter, the difference is negligible.

Variation of K_a while fixing ρ

We use a fixed $\rho = \frac{kKa}{N} = 1/3$ and vary K_a and N while maintaining k fixed, varying K_a from 100 to 500. It is interesting to note that, in this scenario, the energy required for transmission does not depend on K_a.

in 10^7 iterations, therefore we consider it to be significantly smaller than 10^{-4}.

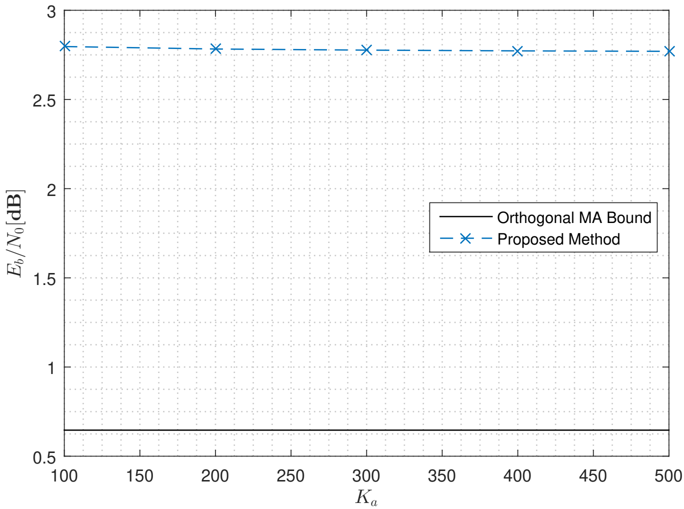

Figure 4.2: Comparison between the E_b/N_0 required in function of K_a for $\rho = 1/3$, $k = 100$.

This pattern is observed for all the tested values of ρ, even for values higher than 1. This is strong evidence that, at least for the results of our method, the x-axis in Figure 4.1 may be changed from K_a to its respective value of ρ with little or no change.

Variation of N

To further investigate the results in function of ρ, we use $N = \frac{kK_a}{\rho}$ and vary ρ from 0.1 to 1. We do this for for $K_a = 50$ and $K_a = 300$. As can be seen, the curves are almost the same. The small differences are due to the optimization process, because although the energy per bit is the same, the parameters are not. This further indicates that analysis can be done in function of ρ while allowing K_a and N jointly go to infinity.

The results are presented in Figure 4.3 and compared to the Orthogonal MA bound. While both curves present a linear behavior, it is clear that our method comparatively worsens as higher spectral efficiencies are required. This is mostly due to the spectral efficiency in the scheduling request phase being reduced by the channel conversion to a bi-AWGN modulo 2 channel.

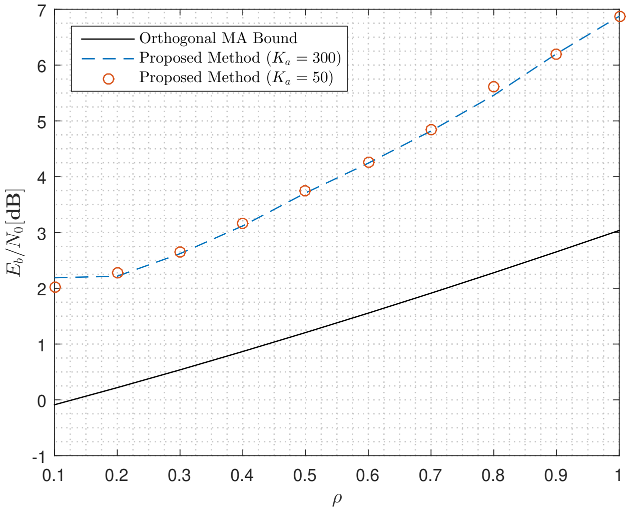

Figure 4.3: Comparison between the E_b/N_0 required in function of ρ for $k = 100$ and $\epsilon = 0.05$.

Variation of ϵ

We vary ϵ from 0.01 to 0.1. It can be easily seen that our method becomes significantly worse, compared to the orthogonal bound, with low probabilities of error, but improves as the allowed probability of error increases. This can be understood by the fact that, when we assume perfect coordination a priori, we allow the data transmission to contain all the probability of error, while in our method there are possible errors in the coordination. As the allowed probability of error decreases, we require larger coordination phases, specifically larger m, T (consequently, also $n_{c,1}$) and V, which increases the overhead and therefore amplifies the losses compared to the perfect coordinated bound.

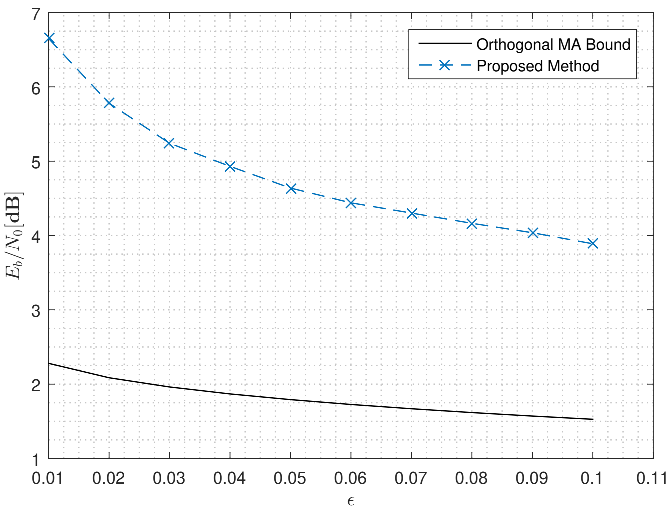

Figure 4.4: Comparison between the E_b/N_0 required in function of ϵ for $K_a = 200$, $k = 100$, $N = 30000$.

Chapter 5

Conclusion

Research in the random access channel has increased due to the growing number of connected devices and the trend of IoT. For that problem, [3] presents fundamental bounds and works such as [12, 4, 13] have proposed practical schemes. While practical schemes have approached the theoretical limits, there is still a significant gap between the finite blocklength theory and the schemes. Furthermore, most analysis still relies on information theory and implementation requires relatively complex codes and algorithms.

In our work, we presented a new scheme for the massive random access problem which allows coordinating the users for a better overall performance at the low cost of a short broadcast feedback from the base station.

We presented an information theoretic error analysis and an efficient optimization method, which allows designing the parameters of the scheme, and showed that our method outperforms the state-of-art in [13], as well as the practical schemes in [12] and [4], as long as a short feedback can be transmitted from the base station and decoded by the users.

We have briefly studied the performance of off-the-shelf codes in our scheme and it is clear that the loss, compared to our theoretical bounds, is significant. In particular, the code design for the short blocklengths involved in the data transmission phase is challenging. However, we would like to emphasize that even after this loss, our scheme provides results comparable to the other works in the area, while using known simple codes.

Limitations of our method

In our work, we use Ordentlich and Polyanskiy's scheme [12] in the scheduling request phase. This is an interesting scheme because it allows us to use both the transmitted index and the timeslot as information for identifying the users. However, in their work, it is clear that the coding strategy is far from the theoretical bound of the scheme. Further investigation is required for our case, where the transmitted messages are significantly smaller, but this indicates that our method might be improved by different coding and decoding strategies in the scheduling request phase.

Additionally, we use a normal approximation for the bi-AWGN mod 2 channel. However, we show that this approximation is not generally achievable, thus, our method relies on empirical gaps in order to obtain achievable results.

Finally, Kowshik and Polyanskiy [21] recently approached the many-access channel, i.e., the multiple access channel with many users, with the same perspective as in [3], defining the probability of error *per user* and obtaining converse and achievability results for the channel. One particularly interesting result in this paper is that orthogonal methods perform poorly when high spectral efficiencies are required, which indicates that our method can be improved in the data transmission phase as well.

Future Works

As interesting future works, we suggest:

(i) Designing codes for the AWGN channel specifically tailored for the small blocklength regime, as off-the-shelf usual codes, such as the LDPC codes tested, seem to perform poorly in this scenario;

(ii) Obtaining theoretical results for the T-slotted ALOHA for very small payloads, and, if the results differ significantly from our coding and decoding strategy, improve this strategy;

(iii) Finding precise finite blocklength information achievability equations for the bi-AWGN mod 2 channel, as our results show that the current approximation is loose and not generally valid;

(iv) Analyze the gains from using the many-user channel theory in the data transmission phase;

(v) Improve the proposed model, considering a random number of active users, imperfections in synchronization and processing delays, and obtain theoretical and practical results for the improved model.

(vi) Improve the proposed model, considering a random static fading to each user. Improve the proposed method in the data transmission phase for that model and compare it to the cost of channel inversion, which is currently proposed for grantless methods.

Appendix A

Derivation of Probabilities of Error

For the derivation of (3.4), consider a Bernoulli random variable with probability $1/V$. The probability that the number of other (than the user i) users in the timeslot j is t is given by

$$\binom{K_a - 1}{t} \left(\frac{1}{V}\right)^t \left(1 - \frac{1}{V}\right)^{K_a - 1 - t}. \tag{A.1}$$

If we take the sum for $t = 0, \ldots, T-1$, this gives the probability that less than T other users transmitted in the timeslot j, which is our event of success. Therefore the probability of error is given by (3.4), i.e., 1 minus the probability os success.

For the derivation of (3.6), a similar approach is taken. The probability that a user $i' \neq i$ transmits the index u in the timeslot j is given by $1/(Vn_p)$. Therefore, the probability that a user $i' \neq i$ does **not** transmit such index in such timeslot is $1 - (\frac{1}{Vn_p})$. Since each user chooses a timeslot and index independently, the probability that all $K_a - 1$ other users choose a different index or a different timeslot is given by $\left(1 - \frac{1}{Vn_p}\right)^{K_a - 1}$, which is our probability of success. Finally, the probability of error is given by (3.6). The bound is derived from the inequality $(1+x)^y > 1 + x \cdot y$ for $x > 0$ and $y > 1$.

The probabilities of error for the channels, i.e., (3.5); (3.7) and (3.8) are derived from the normal approximations for such channels. For example, for the AWGN channel we have, as in (2.6), the equation

$$R_{\mathrm{AWGN}} \approx C_{\mathrm{AWGN}}(P) - \sqrt{\frac{\mathsf{V}_{\mathrm{AWGN}}(P)}{n_c}} Q^{-1}(\epsilon) \tag{A.2}$$

$$C_{\mathrm{AWGN}}(P) - R_{\mathrm{AWGN}} \approx \sqrt{\frac{\mathsf{V}_{\mathrm{AWGN}}(P)}{n_c}} Q^{-1}(\epsilon) \tag{A.3}$$

$$(C_{\mathrm{AWGN}}(P) - R_{\mathrm{AWGN}}) \sqrt{\frac{n_c}{\mathsf{V}_{\mathrm{AWGN}}(P)}} \approx Q^{-1}(\epsilon). \tag{A.4}$$

Finally, for example, we use $n_c = n_{c,2}$, $P = P_2 K_a$ and $R_{\mathrm{AWGN}} = \frac{k}{n_{c,2}}$, apply the Q-function to both sides and achieve (3.8).

Appendix B

Binary AWGN mod-2 Channel

Assume we use an equiprobable distribution for x, which achieves capacity for this channel. This allows us to simplify (2.1) to

$$i(x;y) = \log\left(\frac{p(y|x)}{\frac{p(y|x=0)+p(y|x=1)}{2}}\right). \tag{B.1}$$

For the computation of the expected value, we have

$$C(P) = I(X;Y) = \sum_{t\in\mathcal{X}} p(x=t) \int_{y\in\mathcal{Y}} p(y|x=t) \log\left(\frac{p(y|x=t)}{\frac{p(y|x=0)+p(y|x=1)}{2}}\right) dy \tag{B.2}$$

where $\mathcal{X} = \{0,1\}$ and $\mathcal{Y} = [0,2]$. Now, due to the symmetry of the channel, we know that the inner integral has the same result for $t=0$ and $t=1$. Therefore, we have

$$C(P) = I(X;Y) = \int_{y\in\mathcal{Y}} p(y|x=0) \log\left(\frac{p(y|x=0)}{\frac{p(y|x=0)+p(y|x=1)}{2}}\right) dy. \tag{B.3}$$

The same arguments can be made for the channel dispersion, i.e., the variance. Therefore, the information density and its statistics can be simplified to

$$
\begin{aligned}
i(\tilde{Z}) &= \log_2 \left(\frac{p_{\tilde{Z}}(\tilde{Z})}{\frac{1}{2}p_{\tilde{Z}}(\tilde{Z}) + \frac{1}{2}p_{\tilde{Z}}(\tilde{Z} - 1 \bmod 2)} \right) \\
C(P) &= \mathbf{E}[i(\tilde{Z})] \\
\mathsf{V}(P) &= \mathrm{var}[i(\tilde{Z})]
\end{aligned}
$$

where $\tilde{Z} = \mathbf{z} \bmod 2$. These equations can be found in [12] as well.

We now present how to compute the p.d.f. $p_{\tilde{Z}}(\tilde{Z})$. Let $z \sim \mathcal{N}(0, \frac{1}{4P})$. The p.d.f. of $\tilde{Z} = z \bmod 2$ is given by

$$
p_{\tilde{Z}}(\tilde{Z}) = \sum_{i=-\infty}^{\infty} e^{-\frac{1}{\sqrt{2\pi\sigma^2}}\left(\tilde{Z}-2i\right)^2} \tag{B.4}
$$

where $\sigma^2 = \frac{1}{4P}$ and $\tilde{Z} \in [0, 2)$. Generally, computation of this p.d.f. depends on σ, but most of our results involve small values of σ. Note that, due to the e^{-x^2} nature of the probability function, for small σ, the dominant terms are $i = 0$ and $i = 1$, therefore this sum may be reduced to simply

$$
p_{\tilde{Z}}(\tilde{Z}) = \sum_{i=0}^{1} e^{-\frac{1}{\sqrt{2\pi\sigma^2}}\left(\tilde{Z}-2i\right)^2} \tag{B.5}
$$

which is easily computable.

Appendix C

Optimization Method

For the implementation of our optimization method, we used the MATLAB function fmincon, which finds the optimal real parameters for a constrained optimization problem.

For the pseudo-code, consider that the function x = fmincon(L, con, {param}) returns a structure x with the parameters listed in {param} which minimizes the loss function L under constraint con. Consider that, when the parameter is set a value in the function call, e.g., x = fmincon(L, con, $\{m = 2\}$), that parameter (m) is fixed to that value (2), i.e., it is not changed in the optimization.

The loss function, as we described in the optimization method, is simply the power P. The constraints are easily computed with the equations described in our error analysis (Section 3.2.4) and the lengths described for each session, therefore, our pseudo-code focuses on the optimization method and not on the computation of the loss and constraint functions.

$i \leftarrow 1$
for $T = T_{\min}, \ldots, T_{\max}$ **do**
  x = fmincon(P, con, $\{m, n_{c,1}, n_{c,2}, V, P\}$)
  $m_1 = \lfloor \mathsf{x}.m \rfloor$
  $m_2 = \lceil \mathsf{x}.m \rceil$
  x_1 = fmincon(P, con, $\{m = m_1, n_{c,1}, n_{c,2}, V, P\}$)
  x_2 = fmincon(P, con, $\{m = m_2, n_{c,1}, n_{c,2}, V, P\}$)

if $\mathsf{x}_1.P < \mathsf{x}_2.P$ **then**
$m(i) \leftarrow m_1$
else
$m(i) \leftarrow m_2$
end if
x = fmincon(P, con, $\{m = m(i), n_{c,1}, n_{c,2}, V, P\}$)
$n_{c,1}(i) \leftarrow$ round(x.$n_{c,1}$)
x = fmincon(P, con, $\{m = m(i), n_{c,1} = n_{c,1}(i), n_{c,2}, V, P\}$)
$n_{c,2}(i) \leftarrow$ round(x.$n_{c,2}$)
x = fmincon(P, con, $\{m = m(i), n_{c,1} = n_{c,1}(i), n_{c,2} = n_{c,2}(i), V, P\}$)

$V(i) \leftarrow$ round(x.V)
x = fmincon(P, con, $\{m = m(i), n_{c,1} = n_{c,1}(i), n_{c,2} = n_{c,2}(i), V = V(i), P\}$)
$P(i) \leftarrow$ x.P
$i \leftarrow i + 1$
end for

After this, we simply choose the value of T which provided the smallest power $P(i)$.

Appendix D

Maximum Likelihood Approximation for the bi-AWGN mod 2 channel

We wish to find [22]

$$\mathbf{c}^* = \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmax}} \sum_{i=1}^{n} c_i \mathrm{LLR}(y_i) \tag{D.1}$$

where n is the length of the codeword $\mathbf{c}$ and the log-likelihood is defined as

$$\mathrm{LLR}(y) = \frac{p(y|b=1)}{p(y|b=0)} \tag{D.2}$$

where b is the transmitted bit in the codeword $\mathbf{x}$ that generates y.

For simplicity, let us change the definition of the operation mod 2 to return a real value in the interval $[-1, 1)$ instead of $[0, 2)$. Recall that the bi-AWGN mod 2 channel is described as

$$\mathbf{y} = (\mathbf{c} + \mathbf{z}) \bmod 2 \tag{D.3}$$

where $\mathbf{z} \sim \mathcal{N}(0, \frac{1}{4P}\mathbf{I})$.

For this channel, the LLR can be written as

$$\mathrm{LLR}(y) = \log\left(\frac{\sum_{i\in\mathbb{Z}} e^{-\frac{1}{\sqrt{2\pi\sigma^2}}(y-2i-1)^2}}{\sum_{i\in\mathbb{Z}} e^{-\frac{1}{\sqrt{2\pi\sigma^2}}(y-2i)^2}}\right) \tag{D.4}$$

where $\sigma^2 = \frac{1}{4P}$. If P is high enough (not extremely small), we can approximate these sums as

$$\mathrm{LLR}(y) = \log\left(\frac{e^{-\frac{1}{\sqrt{2\pi\sigma^2}}(y+1)^2} + e^{-\frac{1}{\sqrt{2\pi\sigma^2}}(y-1)^2}}{e^{-\frac{1}{\sqrt{2\pi\sigma^2}}(y^2)}}\right). \tag{D.5}$$

With the same approximation, the upper side of the fraction will be dominated by the left term if $y < 0$ and by the right term if $y > 0$. Simplifying the term, this approximations yields

$$\begin{aligned}
\mathrm{LLR}(y) &\approx \begin{cases} -\left[(y-1)^2 - y^2\right] = [y - 1/2], & \text{if } y > 0 \\ -\left[(y+1)^2 - y^2\right] = [-y - 1/2], & \text{if } y < 0 \end{cases} && \text{(D.6)} \\
&\approx |y| - 1/2 && \text{(D.7)}
\end{aligned}$$

Furthermore, the codeword that maximizes (D.1) also minimizes the Euclidean distance between $|\mathbf{y}|$ and $\mathbf{c}$, as can be seen from the following operations

$$\begin{aligned}
\mathbf{c}^* &= \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmax}} \sum_{i=1}^{n} |y_i|c_i - \frac{1}{2}c_i && \text{(D.8)} \\
&= \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmax}} \sum_{i=1}^{n} |y_i|c_i - \frac{1}{2}c_i^2 && \text{(D.9)} \\
&= \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmin}} \sum_{i=1}^{n} c_i^2 - 2|y_i|c_i && \text{(D.10)} \\
&= \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmin}} \sum_{i=1}^{n} c_i^2 - 2|y_i|c_i + |y_i|^2 && \text{(D.11)} \\
&= \underset{\mathbf{c}\in\mathcal{C}}{\operatorname{argmin}} \sum_{i=1}^{n} (|y_i| - c_i)^2 && \text{(D.12)}
\end{aligned}$$

where (D.9) follows from $c_i \in \{0, 1\}$, therefore $c_i = c_i^2$, and (D.11) follows from y_i being constant with $\mathbf{c}$.

Appendix E

Comparison of Real Codes to Normal Approximation Results

The following results are achieved using the maximum-likelihood approximation presented in Appendix D. We present the generator matrix **G** or the parity check matrix **H** for the code and then the curve of error, compared to the approximations provided by (3.5) for the binary-AWGN mod-2 channel. Note that the (13, 6) code does not achieve the maximum minimum distance of 4 for that code, i.e., the code is not the best code for this pair of n_c and k. However, it achieves the desired probability of error. Possibly the main observation in these simulations is that the approximation is not achievable for small probabilities of error.

$$\mathbf{G}(13,6) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} \tag{E.1}$$

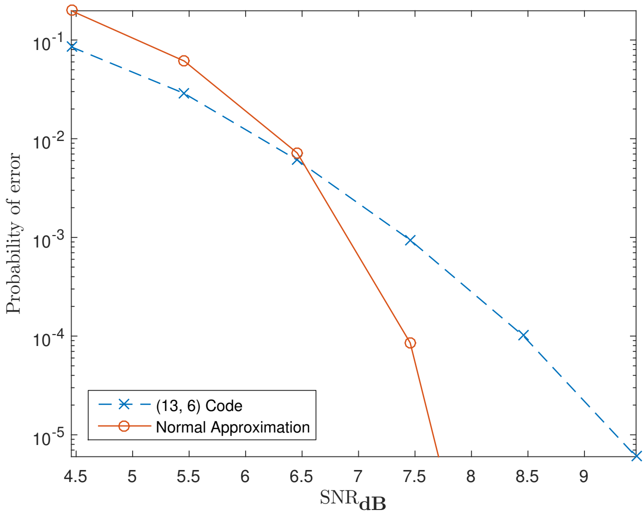

Figure E.1: Comparison between error rate of a linear (13,6) code and approximation results from finite blocklength theory.

$$\mathbf{G}(8,6) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} \tag{E.2}$$

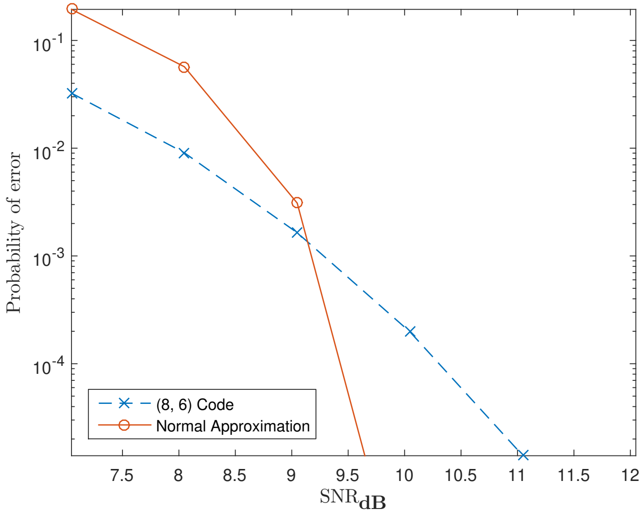

Figure E.2: Comparison between error rate of a linear (8,6) code and approximation results from finite blocklength theory.

$$\mathbf{H}(17,12) = \begin{pmatrix} 0&0&0&1&1&1&0&1&1&1&1&1&1&0&0&0&0 \\ 0&1&1&1&1&1&1&0&1&1&1&1&0&1&0&0&0 \\ 1&1&1&0&0&1&1&1&0&1&1&1&0&0&1&0&0 \\ 1&0&1&0&1&0&1&1&1&0&1&1&0&0&0&1&0 \\ 1&1&0&1&0&0&1&1&1&1&0&1&0&0&0&0&1 \end{pmatrix} \tag{E.3}$$

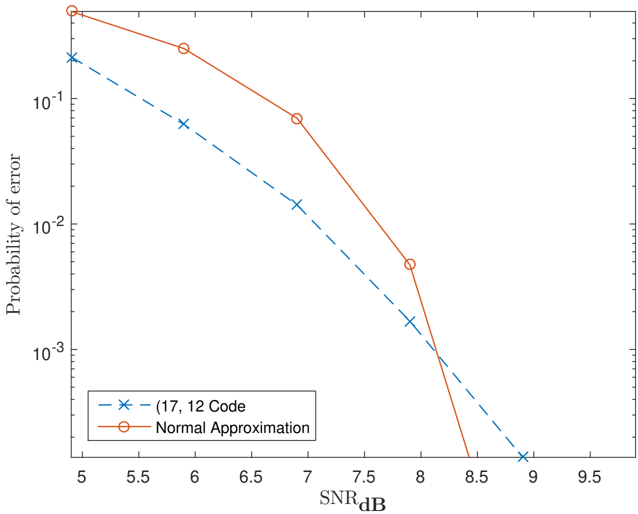

Figure E.3: Comparison between error rate of a linear (17,12) code and approximation results from finite blocklength theory.

$$\mathbf{H}(14,12) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \tag{E.4}$$

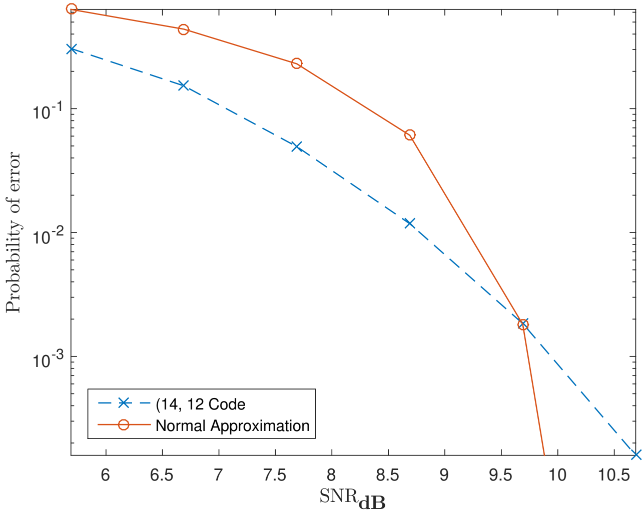

Figure E.4: Comparison between error rate of a linear (14,12) code and approximation results from finite blocklength theory.

Referências bibliográficas

[1] M. Shirvanimoghaddam, M. Dohler, and S. J. Johnson, "Massive non-orthogonal multiple access for cellular IoT: Potentials and limitations," *IEEE Communications Magazine*, vol. 55, no. 9, pp. 55–61, Sept 2017.

[2] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. New York, NY, USA: Wiley-Interscience, 2006.

[3] Y. Polyanskiy, "A perspective on massive random-access," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 2523–2527.

[4] A. Vem, K. R. Narayanan, J. Cheng, and J. Chamberland, "A user-independent serial interference cancellation based coding scheme for the unsourced random access Gaussian channel," in *2017 IEEE Information Theory Workshop (ITW)*, Nov 2017, pp. 121–125.

[5] C. Bockelmann, N. Pratas, H. Nikopour, K. Au, T. Svensson, C. Stefanovic, P. Popovski, and A. Dekorsy, "Massive machine-type communications in 5g: physical and mac-layer solutions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 59–65, Sep. 2016.

[6] J. Zhu and M. Gastpar, "Gaussian multiple access via compute-and-forward," *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 2678–2695, May 2017.

[7] B. Rimoldi and R. Urbanke, "A rate-splitting approach to the Gaussian multiple-access channel," *IEEE Transactions on Information Theory*, vol. 42, no. 2, pp. 364–375, March 1996.

[8] P. Mathys, "A class of codes for a T active users out of N multiple-access communication system," *IEEE Transactions on Information Theory*, vol. 36, no. 6, pp. 1206–1219, Nov 1990.

[9] I. Bar-David, E. Plotnik, and R. Rom, "Forward collision resolution-a technique for random multiple-access to the adder channel," *IEEE Transactions on Information Theory*, vol. 39, no. 5, pp. 1671–1675, Sept 1993.

[10] X. Chen, T. Chen, and D. Guo, "Capacity of Gaussian many-access channels," *IEEE Transactions on Information Theory*, vol. 63, no. 6, pp. 3516–3539, June 2017.

[11] J. Goseling, C. Stefanović, and P. Popovski, "Sign-compute-resolve for tree splitting random access," *IEEE Transactions on Information Theory*, vol. 64, no. 7, pp. 5261–5276, July 2018.

[12] O. Ordentlich and Y. Polyanskiy, "Low complexity schemes for the random access Gaussian channel," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 2528–2532.

[13] V. K. Amalladinne, A. Vem, D. K. Soma, K. R. Narayanan, and J. Chamberland, "A coupled compressive sensing scheme for unsourced multiple access," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 6628–6632.

[14] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Transactions on Information Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[15] C. E. Shannon, "Probability of error for optimal codes in a Gaussian channel," *The Bell System Technical Journal*, vol. 38, no. 3, pp. 611–656, May 1959.

[16] D. Slepian, "Bounds on communication," *The Bell System Technical Journal*, vol. 42, no. 3, pp. 681–707, May 1963.

[17] R. G. Gallager, *Information Theory and Reliable Communication.* New York, NY, USA: John Wiley & Sons, Inc., 1968.

[18] G. Liva, L. Gaudio, and T. Ninacs, "Code design for short blocks: A survey," in *Proc. EuCNC*, Athens, Greece, Jun. 2016.

[19] J. V. Wonterghem, A. Alloum, J. J. Boutros, and M. Moeneclaey, "On short-length error-correcting codes for 5g-nr," *Ad Hoc Networks*, vol. 79, pp. 53 – 62, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1570870518303147

[20] Xiao-Yu Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Transactions on Information Theory*, vol. 51, no. 1, pp. 386–398, Jan 2005.

[21] S. S. Kowshik and Y. Polyanskiy, "Fundamental limits of many-user MAC with finite payloads and fading," *arXiv e-prints*, p. arXiv:1901.06732, Jan 2019.

[22] L. Szczecinski and A. Alvarado, *Bit-Interleaved Coded Modulation: Fundamentals, Analysis and Design.* John Wiley & Sons, Inc., 2014.